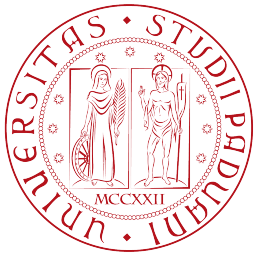


UNIVERSITY OF PADUA

PH.D. SCHOOL IN BRAIN, MIND AND COMPUTER SCIENCE
Curriculum in Computer Science for Societal Challenges and Innovation
XXXVIII Cycle

Leveraging AI Approaches for Robust Wireless Security: From Physical to Network Layer

---

Candidate:
Shuo WANG

Supervisor:
Prof. Mauro CONTI
University of Padua

Co-Supervisor:
Prof. Anna SPAGNOLLI
University of Padua

Acknowledgements

Pursuing my PhD has been both a challenging and rewarding journey, and I am deeply grateful to all those who have supported me along the way.

I would like to express my sincere gratitude to my supervisor, Prof. Mauro Conti, for his invaluable guidance, patience, and encouragement throughout my doctoral studies. I am also thankful to my co-supervisors, Prof. Anna Spagnolli and Dr. Alessandro Brighente, for their constructive feedback and continuous support.

I am fortunate to have been part of SPRITZ, where I benefited from an energetic research environment and the friendship of many colleagues. I also wish to thank my collaborators and co-authors, whose insights and cooperation have enriched my work. Beyond academia, I owe heartfelt thanks to my family and friends for their constant encouragement and understanding. This thesis reflects not only my own efforts but also the guidance, support, and kindness I have received from many people, to whom I extend my deepest gratitude.

Abstract

Wireless communication has become the backbone of modern information infrastructure, connecting billions of devices and supporting a wide range of services, including mobile broadband, industrial automation, autonomous systems, and the Internet of Things (IoT). The evolution of technologies from 5G to emerging 6G, alongside Wi-Fi, Bluetooth, and low-power IoT protocols, has transformed global connectivity but also introduced significant security risks. At the physical layer, the broadcast nature of the medium exposes systems to eavesdropping, interference, and sophisticated jamming. Radio frequency fingerprint identification (RFFI) offers lightweight authentication but struggles with receiver heterogeneity and open-set recognition, where unseen or rogue devices may appear. At the network layer, growing traffic volumes, limited labeled data, and heterogeneous deployment environments complicate anomaly detection, threatening confidentiality, integrity, and availability.

This thesis addresses the challenges of wireless security in a holistic manner by designing novel methods tailored to the vulnerabilities of different layers. At the **physical layer**, we develop *Capodoglio*, which models adaptive multi-armed bandit (MAB) jammers and proposes three defense strategies, namely Speed-Up, Helper-Node, and Mirror-MAB, to stabilize packet delivery under adversarial scenarios. *DOA-CGAN* introduces a conditional generative adversarial network (GAN) that suppresses structured interference in the covariance matrix of received signals, enabling classical direction-of-arrival (DOA) estimators to recover accuracy under jamming. For authentication, *Triad-GAN* learns receiver-invariant and transmitter-discriminative embeddings, improving the robustness of RFFI in multi-receiver scenarios. To extend to open-set recognition, *OS-TriadGAN* integrates label smoothing, temperature scaling, and an F1-optimized threshold, ensuring reliable rejection of unknown devices while preserving closed-set accuracy. At the **network layer**, *Catch-All* leverages graph neural networks (GNNs) and topological features to detect anomalies in Wi-Fi traffic, with robustness validated against perturbations and adversarial manipulations. Complementarily, *GANSec* introduces a classification-oriented conditional GAN that synthesizes wireless time-series data, strengthening supervised anomaly detection under domain shifts and data scarcity.

Overall, this thesis contributes a portfolio of resilient and realistic methods that reinforce wireless security at both signal and traffic levels. At the physical layer, we advance defense strategies against adaptive jamming, introduce a generative approach that operates in the covariance matrix to remove interference, and establish receiver-invariant authentication extended to open-set scenarios. At the network layer, we develop graph-based anomaly detection and GAN-based augmentation tailored to various distribution challenges. Collectively, these studies underscore the value of generative, adversarial, and calibration-aware techniques in guiding the design of resilient and trustworthy wireless networks.

Contents

Abstract v

1 Introduction 1
1.1 Background and Motivation . . . 1
1.2 Publications . . . 4

I Physical Layer Security 7

2 Adaptive Jamming Defenses (Capodoglio) 9
2.1 Introduction . . . 9
2.2 Related Work . . . 11
2.2.1 Channel Hopping with Reinforcement Learning . . . 11
2.2.2 Wi-Fi Jamming and Anti-Jamming in Practice . . . 11
2.2.3 Learning based Jamming and Mitigation . . . 12
2.3 System Overview . . . 12
2.3.1 Network Model . . . 12
2.3.2 Threat Model . . . 13
2.3.3 Defense Strategy . . . 13
2.4 MAB Framework . . . 14
2.5 The MAB-based Attacks . . . 16
2.6 Defense Strategy Against MAB . . . 17
2.6.1 Speed-Up . . . 17
2.6.2 Helper-Node . . . 18
2.6.3 Mirror-MAB . . . 18
2.7 Performance Evaluation . . . 19
2.7.1 Evaluation Settings . . . 20
2.7.2 Attacks Performance . . . 20
2.7.3 Performance of Our Proposed Defense Approaches . . . 21
2.8 Conclusion . . . 23

3 Generative DOA Estimation (DOA-CGAN) 25
3.1 Introduction . . . 25
3.2 Related Work . . . 27
3.2.1 Classical DOA Estimation and Anti–Interference Techniques . . . 27
3.2.2 Machine and Deep Learning for DOA . . . 28
3.3 Primer on FMCW Radars . . . 28
3.4 Threat Model . . . 29
3.5 System Model . . . 30
3.6 Learning Networks for DOA Estimation . . . 31
3.6.1 Intuition Behind Our Learning-Based Approach . . . 31
3.6.2 Data Preprocessing . . . 32
3.6.3 Adversarial Model . . . 34

3.6.4 Optimization of Objective Function . . . 36
3.6.5 DOA-CGAN Framework . . . 37
3.7 Numerical Results and Discussion . . . 37
3.7.1 Estimation Performance with Several Neural Models . . . 37
3.7.2 DOA Estimates . . . 38
3.7.3 Estimation Error versus SNR and Snapshot . . . 39
3.8 Conclusion . . . 40

4 Multi-Receiver RFFI (Triad-GAN) 41
4.1 Introduction . . . 41
4.2 Related Work . . . 43
4.2.1 RFFI Methods and Applications . . . 43
4.2.2 Generative Adversarial Networks Applications . . . 43
4.3 System Model and Problem Statement . . . 44
4.3.1 System Model . . . 44
4.3.2 Problem Statement . . . 45
4.4 Proposed Method . . . 46
4.4.1 System Architecture . . . 46
4.4.2 Optimization Objectives . . . 47
4.4.3 Network Architecture . . . 48
4.5 Experiments and Evaluation . . . 49
4.5.1 Datasets and Experiment Setup . . . 49
4.5.2 Experimental Results and Analysis on LoRa . . . 49
4.5.3 Experimental Results and Analysis on WiSig . . . 54
4.6 Conclusion . . . 57

5 Open-Set Extension for RFFI (OS-TriadGAN) 59
5.1 Introduction . . . 59
5.2 Related Work . . . 61
5.2.1 From Closed-Set RFFI to Open-Set Recognition . . . 61
5.2.2 Learning Paradigms for Open-Set RFFI . . . 62
5.3 Open-Set RFFI Scenario and Problem Formulation . . . 62
5.3.1 Open-Set RFFI System Model . . . 62
5.3.2 Problem Formulation . . . 64
5.4 Proposed Method . . . 65
5.4.1 Structure of OS-TriadGAN . . . 65
5.4.2 Adversarial Feature Learning with Supervised Classification . . . 66
5.4.3 Threshold Calibration and Testing Application . . . 68
5.5 Experiments and Analysis . . . 69
5.5.1 Dataset Description . . . 69
5.5.2 Closed-Set Classification Accuracy . . . 70
5.5.3 Open-Set and Unknown Device Detection . . . 71
5.6 Conclusion . . . 74

II Network Layer Security 75

6 Graph Anomaly Detection (Catch-All) 77
6.1 Introduction . . . 77
6.2 Background and Related Work . . . 78
6.2.1 Graph Neural Network in Wireless Networks . . . 78

6.2.2 Adversarial Attacks to GNN . . . 79
6.3 System and Threat Model . . . 80
6.3.1 Network Model . . . 80
6.3.2 Threat Model . . . 81
6.4 Our Proposed GNN Anomaly Detector . . . 81
6.4.1 Network Graph Representation . . . 81
6.4.2 Framework . . . 82
6.4.3 Graph-based Dataset Construction . . . 82
6.4.4 The GNN-based Anomaly Detection . . . 83
6.5 Numerical Results . . . 84
6.5.1 Dataset Construction . . . 84
6.5.2 Experimental Results Analysis . . . 87
6.5.3 Adversarial Attack Robustness Evaluation . . . 89
6.6 Conclusion . . . 92

7 GAN-based Data Augmentation (GANSec) 93
7.1 Introduction . . . 93
7.2 Motivation . . . 95
7.2.1 The Prohibitive Cost of Wireless Data Acquisition . . . 95
7.2.2 Data Quality Limitations and the Naivety of Traditional Augmentation . . . 95
7.2.3 Enhancing Model Generalization and Robustness . . . 96
7.3 Related Work . . . 96
7.3.1 Anomaly Detection in Wireless Security . . . 96
7.3.2 Data Augmentation Techniques for Wireless Anomaly Detection 97
7.3.3 Challenges and Limitations in GAN-based Augmentation . . . 97
7.4 The GANSec Framework . . . 98
7.4.1 GANSec Architecture and Training Mechanisms . . . 99
7.4.2 Tailoring GANSec for Wireless Data . . . 101
7.4.3 Synthetic Data Generation and Usage Strategy . . . 102
7.5 Evaluation . . . 103
7.5.1 Experimental Setup . . . 103
7.5.2 Synthetic Data Quality Analysis . . . 105
7.5.3 Anomaly Detection Performance: Cross-Scenario Evaluation . . . 107
7.6 Discussion . . . 108
7.7 Conclusion . . . 109

Conclusion and Future Directions 113

8 Conclusion and Future Directions 113
8.1 Conclusion . . . 113
8.2 Future Directions . . . 114

List of Figures

1.1 Overview of thesis contributions across the **physical** and **network** layers of the wireless stack. In addition to these core studies, we also explored physical-layer location privacy through *Confundus* as a complementary direction. . . . 4

2.1 System Model. . . . 13
2.2 MAB with Thompson Sampling Framework. . . . 16
2.3 Workflow for Helper-Node. . . . 18
2.4 Countermeasure: Mirror-MAB. . . . 19
2.5 PDR for different attack strategies. . . . 20
2.6 PDR for Speed-Up parameters. . . . 21
2.7 PDR for Helper-Node. . . . 22
2.8 PDR for Mirror-MAB. . . . 22
2.9 PDR for different strategies. . . . 23

3.1 System model of an FMCW radar. . . . 29
3.2 System model. . . . 30
3.3 Comparison of GAN and CGAN configurations. . . . 32
3.4 The schema of CGAN. . . . 34
3.5 The configuration of DOA-CGAN. . . . 37
3.6 Spatial spectrum of DOA-CGAN. . . . 39
3.7 The estimation error in terms of varied JNR. . . . 40

4.1 Multi-Receiver RFFI system. . . . 44
4.2 Overall workflow of the Triad-GAN framework. . . . 46
4.3 The network architecture of each component. . . . 48
4.4 LoRa classification accuracy comparison. . . . 50
4.5 LoRa classification accuracy under receiver drift. . . . 51
4.6 LoRa classification accuracy under receiver change. . . . 52
4.7 LoRa classification accuracy under RX number. . . . 53
4.8 t-SNE visualization of LoRa signal features. . . . 54
4.9 WiSig classification accuracy comparison. . . . 54
4.10 WiSig classification accuracy under receiver drift. . . . 55
4.11 WiSig classification accuracy under receiver change. . . . 55
4.12 WiSig classification accuracy under RX number. . . . 56
4.13 t-SNE visualization of WiSig signal features. . . . 57

5.1 Open-Set RFFI system. . . . 63
5.2 Overall workflow of the OS-extension Triad-GAN framework. . . . 65
5.3 Closed-set classification result of OS-TriadGAN (accuracy = 99.74%). . . . 72

6.1 The framework of data collecting and model training. . . . 80
6.2 NS3-simulation introduction. . . . 85
6.3 PCAP conversion. . . . 86

6.4 Embedding distribution. 90

7.1 GAN-based Data Augmentation for Anomaly Detection in Wireless Security: Workflow Overview. 99
7.2 Architecture of the GANSec framework. The diagram shows distinct paths and objectives for the Embedded Conditional (EC-Discriminator) and Classification Oriented (CO-Discriminator) training mechanisms. 100
7.3 Experimental Setup for Collecting Datasets A and B. 104
7.4 Distribution Analysis on Synthetic Dataset (FID). 105
7.5 Distribution Analysis on Synthetic Dataset (MMD). 106
7.6 Impact of the augmentation ratio (relative size of synthetic data used for training) on the accuracy achieved on the unseen Dataset B. 108

List of Tables

2.1 Simulation parameters. 20

3.1 Structural Settings of DOA-CGAN (Generative Model G). 37
3.2 Structural Settings of DOA-CGAN (Discriminative Model D). 38
3.3 The DOA estimation RMSE of different models and running time. . . . 38

5.1 Closed-set classification accuracy comparison among representative RFFI methods. 71
5.2 Closed-set classification accuracy on the WiSig `SingleDay` subset. Best results in **bold**. 71
5.3 Comparison of open-set detection performance on the LoRa dataset. . 73
5.4 Comparison of Open-Set RFFI Results on the SingleDay Dataset. All thresholds τ are reported in the $p_{\max}$ domain, obtained from the MSP threshold τ_{F1} via the conversion $\tau = 1 - \tau_{F1}$. 73

6.1 Precision of different models under Balanced and Imbalanced datasets. 87
6.2 Recall of different models under Balanced and Imbalanced datasets. . 87
6.3 F1-score of different models under Balanced and Imbalanced datasets. 87
6.4 Accuracy of different models under Balanced and Imbalanced datasets. 87
6.5 Precision under feature perturbation and topology attack. 90
6.6 Recall under feature perturbation and topology attack. 90
6.7 F1-score under feature perturbation and topology attack. 90
6.8 Accuracy under feature perturbation and topology attack. 90

Chapter 1

Introduction

1.1 Background and Motivation

Over the past decades, wireless communication has evolved from domain-specific applications in the military, maritime, and broadcasting into the backbone of global connectivity. The current wireless landscape is characterized by a heterogeneous mix of standards, encompassing cellular technologies from 3G to 5G and the emerging 6G, alongside local connectivity protocols such as Wi-Fi and Bluetooth, and low-power Internet of Things (IoT) frameworks including LoRa and Zigbee. This rapid evolution has expanded both the capacity and the coverage of communication infrastructures while fundamentally transforming the way individuals, communities, and industries engage with digital services [6, 43, 54, 90]. Wireless systems are no longer confined to basic voice and data transmission but instead support applications such as autonomous vehicles, remote healthcare, industrial automation, and large-scale sensor networks. Simultaneously, the proliferation of smart devices and the rise of edge–cloud computing ecosystems have accentuated system-level demands for scalable connectivity, characterized by data rates, ultra-low latency, and stringent reliability guarantees. Emerging paradigms such as massive machine-type communication (mMTC), ultra-reliable low-latency communication (URLLC), and integrated sensing and communication (ISAC) further illustrate the expanding role of wireless systems in shaping future societies. The widespread adoption of these technologies not only reflects their versatility in supporting diverse services and performance requirements but also reinforces the necessity of the established layered design in wireless networks, which provides the structural foundation for handling complexity and ensuring reliability. The heterogeneity of standards and the multiplicity of use cases demand clear functional separation: the *physical layer* ensures reliable signal transmission through spectrum utilization and channel management, while the *network layer* manages routing, traffic optimization, anomaly detection, and resource allocation. Together, these layers constitute an interdependent ecosystem that ensures the reliability, security, and overall trustworthiness of contemporary wireless communication systems.

As wireless connectivity becomes indispensable, the risks associated with openness, heterogeneity, and decentralization have become more pressing. At the *physical layer*, the broadcast nature of radio signals makes them inherently vulnerable to eavesdropping, interference, and malicious jamming [196, 9, 20, 109, 139]. Adaptive adversaries can exploit knowledge of frequency hopping or modulation patterns, leading to denial-of-service in critical applications. Physical-layer authentication based on radio frequency fingerprint identification (RFFI) offers a lightweight alternative to conventional cryptographic schemes [40, 69, 82, 152, 188, 211], yet it remains limited in its ability to generalize across receivers and to provide robust

identification of unauthorized or unknown devices. At the *network layer*, anomaly detection faces challenges from massive traffic volumes, distribution shifts across deployment scenarios, and the scarcity of labeled data [33, 32, 2, 133]. Such limitations allow adversaries to hide within legitimate traffic, performing stealthy attacks such as spoofing or botnet-based denial-of-service. Without effective countermeasures, such vulnerabilities undermine the confidentiality, integrity, and availability of wireless services, thereby jeopardizing the reliability of infrastructures that support daily social and economic activities.

To confront these challenges, researchers have proposed a broad range of solutions across different layers of the wireless stack. Reinforcement learning has been explored to enhance spectrum agility and develop adaptive anti-jamming strategies [67, 105, 108, 36, 198]. Deep learning methods have been applied to direction-of-arrival (DOA) estimation, anomaly detection, and device authentication [53, 215, 100]. Generative models and graph-based approaches have shown promise in tackling data scarcity and structural heterogeneity [110, 210, 37, 86]. Nevertheless, important research gaps remain, underscoring the need for further exploration and tailored solutions. Adaptive jammers can still track reinforcement learning policies, degrading performance over time [198]. Deep DOA estimators perform well under benign conditions but collapse in the presence of structured interference [53, 100]. Network anomaly detectors remain highly sensitive to domain shifts, leading to poor generalization [142, 202, 66]. RFFI systems achieve high accuracy in laboratory settings but fail to scale to multi-receiver deployments or open-set conditions [152, 78, 156]. These gaps reveal the urgent need for methods that are robust to adversaries, aware of domain shifts, and capable of reliable operation under realistic wireless environments.

Building on these observations, this thesis addresses wireless security challenges in a holistic manner. Instead of concentrating on a single layer of the communication stack, we develop solutions that span both the physical and network layers. By aligning each method with the specific vulnerabilities and requirements of its respective layer, we provide both theoretical insights and practical mechanisms that strengthen the robustness, adaptability, and trustworthiness of modern wireless systems. Addressing security at these two layers is particularly crucial: the physical layer represents the first line of defense where attacks directly impair signal integrity, while the network layer governs traffic-level behaviors where stealthy or large-scale threats may emerge. At the **physical layer**, our work covers defenses against adaptive jamming, generative interference removal for DOA estimation, and receiver-invariant authentication through RFFI, further extended to open-set recognition. At the **network layer**, we develop graph-based anomaly detection and conditional generative adversarial network (GAN) augmentation to address traffic imbalance, distribution shifts, and data scarcity. This thesis makes several key contributions to the field of wireless security research. The contributions can be grouped into two categories, corresponding to the physical and network layers of the wireless stack.

1. **Physical Layer Contributions.** At the physical layer, we focus on defending against signal-level attacks, enhancing interference resilience, and improving device authentication. The contributions are as follows:

    - **Capodoglio (Chapter 2).** We formally model adaptive jammers as multi-armed bandit (MAB) learners operating in both online and offline settings. Based on this model, we design three countermeasures, namely *Speed-Up*, *Helper-Node*, and *Mirror-MAB*. Each of these strategies mitigates

jamming from a different perspective, collectively stabilizing packet delivery ratios while clarifying the trade-offs between communication stability and defense overhead.

- **DOA-CGAN (Chapter 3).** We reformulate the problem of DOA estimation under jamming as one of generative interference removal in the covariance matrix. To address this challenge, we develop a conditional GAN with an unsupervised generator and a supervised discriminator. This design enables the network to learn how to "evaporate" structured interference, thereby restoring lawful covariance patterns and allowing classical estimators such as multiple signal classification (MUSIC) to regain accuracy under heavy jamming. The method reduces estimation error compared to learning-based baselines while maintaining the interpretability of traditional signal processing.
- **Triad-GAN (Chapter 4).** We design a feature-level triadic adversarial framework that combines an extractor, a generator, and a discriminator into a unified system. The interaction among these components encourages the learning of receiver-invariant yet transmitter-discriminative embeddings, which are crucial for authentication in multi-receiver RFFI. Experiments on LoRa and WiSig datasets confirm that this approach substantially improves classification accuracy in the presence of receiver drift, changes in deployment, and scaling to larger numbers of devices.
- **OS-TriadGAN (Chapter 5).** Building on Triad-GAN, we propose an open-set extension that explicitly incorporates calibration into the authentication pipeline. This framework integrates label smoothing, temperature scaling, and a global F1-optimal maximum softmax probability (MSP) threshold, thereby enabling reliable rejection of previously unseen or unauthorized devices. The approach maintains closed-set accuracy while introducing principled mechanisms for handling open-set recognition, and its algebraic mapping supports reproducibility and deployment across different environments.

2. **Network Layer Contributions.** At the network layer, we focus on traffic-level security, addressing anomaly detection and data scarcity in supervised learning. The contributions are summarized below:

   - **Catch-All (Chapter 6).** We introduce a graph-based anomaly detection framework that abstracts Wi-Fi traffic into graph structures and employs graph neural networks (GNNs) trained with perturbation-aware strategies. This framework captures both feature-level and structural dependencies in traffic data, allowing it to detect irregular patterns with improved precision–recall balance under imbalanced datasets. Furthermore, we evaluate its robustness against distribution imbalance and perturbations, demonstrating its effectiveness in dynamic and adversarial environments.
   - **GANSec (Chapter 7).** We propose a classification-oriented conditional GAN framework designed to generate realistic wireless time-series data for supervised anomaly detection. By augmenting datasets with synthetic but high-quality samples, this approach alleviates the challenges of domain shift and limited labeled data. Sequence-aware classifiers such as LSTMs and CNNs show significant performance gains when trained with

this augmentation. Moreover, we demonstrate that models trained entirely on synthetic data can generalize effectively to real-world scenarios, highlighting the potential of generative augmentation in wireless security.

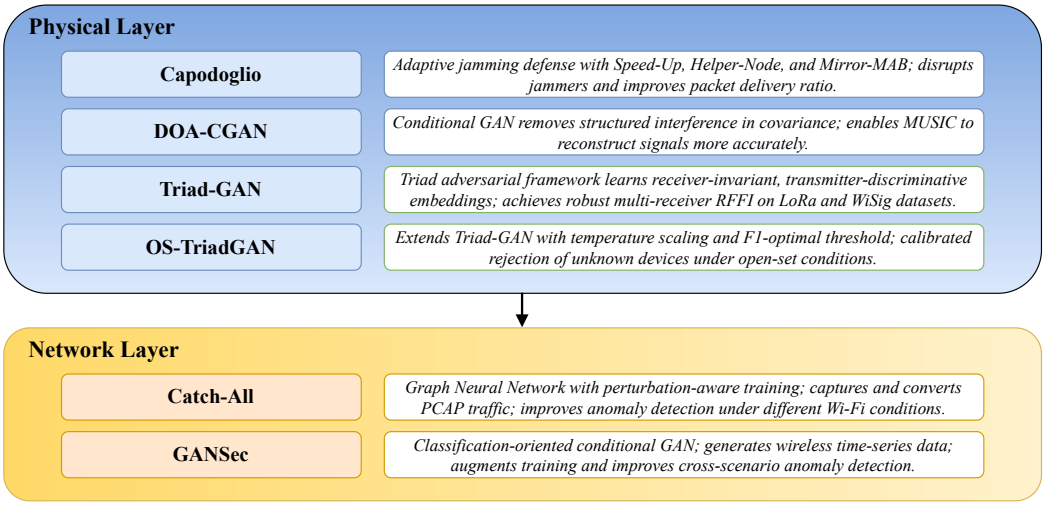


FIGURE 1.1: Overview of thesis contributions across the **physical** and **network** layers of the wireless stack. In addition to these core studies, we also explored physical-layer location privacy through *Confundus* as a complementary direction.

The overall structure of this thesis is illustrated in Figure 1.1. Part I addresses physical-layer security, covering *Capodoglio* (adaptive jamming defenses, Chapter 2), *DOA-CGAN* (generative interference removal, Chapter 3), and *Triad-GAN* with its open-set extension *OS-TriadGAN* (receiver-invariant and calibration-aware authentication, Chapters 4 and 5). Part II addresses network-layer security, including *Catch-All* (graph-based anomaly detection, Chapter 6) and *GANSec* (GAN-based augmentation for distribution shifts, Chapter 7). Collectively, these studies advance a systematic, cross-layer approach to wireless security. In parallel to the main line of this thesis, we conducted a complementary study on physical-layer location privacy, *Confundus: Mitigating Hostile Wireless Source Localization*, which employs fake multipath injection to mislead TDOA-based localization. Although not included as part of the thesis contributions, this work highlights privacy as an additional dimension of physical-layer security. Finally, Chapter 8 concludes the thesis by synthesizing the findings and outlining promising avenues for future research.

1.2 Publications

This section summarizes manuscripts produced during my Ph.D. period and published or currently submitted to peer-reviewed journals and conferences. All manuscripts are listed in chronological order of acceptance and submission.

Journal Publication

1. Xu, S., Brighente, A., Conti, M., Chen, B., & Wang, S. (2024). Evaporative Angle: A Machine Learning Approach-Assisted DOA Estimation under Jamming. *IEEE Wireless Communications Letters, 13(10), 2922–2926*. IEEE. **Published**. **(JCR IF 5.5)**. This paper is discussed in Chapter 3.

2. Liu, G., Zhao, Z., Li, C., Wang, S., Brighente, A., & Conti, M. Step into Balance: A Consistency-Aware and Loose Homophily Guided Generative Method for Class-Imbalanced Graphs. *IEEE Transactions on Knowledge and Data Engineering*. IEEE. Submitted.

3. Chen, H., Conti, M., Giannetsos, T., Wang, S., & Liang, K. FHEW/TFHE in the General Cyclotomic Rings, Revisited. *Designs, Codes and Cryptography*. Springer. Submitted.

4. Mao, J., Fu, J., Spolaor, R., Chen, Q., Zhang, L., Xiong, Z., Conti, M., & Wang, S. PreyLoop: Towards Long-Term Privacy Protection and Security Defense in Federated Learning. *IEEE Transactions on Dependable and Secure Computing*. Submitted.

Conference or Workshop Publication

1. Xu, J., Wang, S., Loscri, V., Brighente, A., Rouvoy, R., & Conti, M. (2025). GANSec: Enhancing Supervised Wireless Anomaly Detection Robustness through Tailored Conditional GAN Augmentation. *30th European Symposium on Research in Computer Security (ESORICS 2025)*. IEEE. **Accepted**. This paper is discussed in Chapter 7.

2. Xu, S., Wang, S., Sciancalepore, S., Brighente, A., & Conti, M. (2025). Confundus: Mitigating Hostile Wireless Source Localization. *IEEE Conference on Communications and Network Security (CNS 2025)*. **Accepted**. IEEE.

3. Wang, S., Brighente, A., Conti, M., Liu, G., & Zhao, Z. Catch-All GNN: A Graph Neural Network Approach for Anomaly Detection in WiFi Networks. *24th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2025)*. IEEE. Submitted. This paper is discussed in Chapter 6.

4. Wang, S., Brighente, A., Loscri, V., Zhang, J., & Conti, M. Capodoglio: Tackling Multi-Armed Bandit Jamming Attacks. *24th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom 2025)*. IEEE. Submitted. This paper is discussed in Chapter 2.

5. Wang, S., Zhang, J., Wang, L., Brighente, A., & Conti, M. Triad-GAN: Feature-Level Generative Adversarial Network for Multi-Receiver Radio Frequency Fingerprint Identification. *40th AAAI Conference on Artificial Intelligence (AAAI 2026)*. AAAI. Submitted. This paper is discussed in Chapter 4.

Part I

Physical Layer Security

Chapter 2

Adaptive Jamming Defenses (Capodoglio)

Jamming attacks represent a critical threat to wireless networks by exploiting the openness of the wireless medium, leading to denial-of-service, excessive interference, and network overloading. Recent advances have enabled more sophisticated forms of disruption, such as protocol-aware and learning-driven jamming, where adversaries selectively transmit upon detecting legitimate activity. This selective behavior not only conserves attacker resources but also complicates detection. While channel hopping is widely adopted as a countermeasure, recent studies have shown that intelligent jammers employing multi-armed bandit (MAB) strategies can effectively learn and predict hopping patterns, thereby sustaining persistent interference and severely degrading network performance. In this paper, we formalize jamming attacks modeled as MAB problems and introduce two refined variants under online and offline learning paradigms. To defend against these advanced adversaries, we propose three strategies: (i) *Speed-Up*, which increases the frequency of channel-hopping decisions to disrupt the jammer's predictions, (ii) *Helper-Node*, which introduces decoy transmissions to mislead the attacker's learning process, and (iii) *Mirror-MAB*, a defender-side bandit algorithm that anticipates and avoids highly jammed channels. Extensive evaluations demonstrate that the proposed defenses achieve the packet delivery ratio (PDR) above 90% under sustained jamming conditions, while preserving the adaptability and robustness of dynamic channel hopping.

2.1 Introduction

Guaranteeing security in the wireless communication, such as Wi-Fi protocols based on the IEEE 802.11 family of standards, is challenging due to the vast number of interconnected devices and their inherently distributed nature [72, 62]. Typically, wireless networks are widely used in information acquisition, command transmission, drone patrol, construction site communication, and emergency communication. The wireless network's infrastructure relies on wireless connections, whose openness inherently exposes communication to security and privacy threats [212, 195]. Among these classical threats, *jamming* is particularly harmful, leading to denial-of-service, overloading the network. Unlike higher-layer threats, jamming operates at the physical layer, which will disrupt the communications directly. A promising countermeasure against jamming is *channel hopping*, which dynamically switches communication channels to mitigate interference [60, 55].

Various defensive strategies have been explored, including reputation-based threat detection and isolation mechanisms [160], decentralized Multi-Armed Bandit (MAB) frameworks [44], and coded rendezvous schemes such as SPADE [146]. Nevertheless, existing solutions continue to face limitations, such as reactive-only responses, added communication overhead, or incomplete protection against smart attackers [22, 114]. In [117, 219], the algorithm employs Q-learning to finalize the selection of the communication channel. These strategies vary from basic sequential or random approaches to sophisticated game theory-based and machine learning-driven techniques [166, 129, 174, 67, 187]. Recently, Reinforcement Learning-based methods have attracted significant attention due to their adaptability. Specifically, MAB algorithms have demonstrated substantial efficacy in dynamically managing noisy wireless environments [172, 158]. For instance, the authors in [67, 187] utilized quantized Signal to Interference plus Noise Ratio values as input states for Reinforcement Learning frameworks. More adaptive approaches, such as those proposed by Liu *et al.* [113], consider spectrum waterfalls, thus removing the dependency on prior knowledge of the jamming pattern. Furthermore, multi-agent Reinforcement Learning-based strategies have been proposed to address both jamming and mutual interference [179, 218]. Nonetheless, most of these methods overlook the power limitations inherent to wireless communication networks' infrastructure, with only a few studies explicitly addressing this critical issue [191]. Advancements in inexpensive and accessible hardware have enabled attackers to also adopt increasingly intelligent and adaptive strategies. Recent studies [22, 21] demonstrated that an advanced MAB-based jammer, namely FOLPETTI, can efficiently disrupt communications despite sophisticated defensive channel hopping techniques (random or deep Reinforcement Learning-based).

In this paper, we analyze and develop the MAB-based attacks, and meanwhile address the limitations by explicitly focusing on defense strategies against the MAB-based jamming. We begin by describing the underlying MAB approach and differentiating between two MAB-based attack implementations (i.e., online and offline attacks), clarifying the design for each attack. Subsequently, we propose three novel defense strategies: *Speed-Up*, *Helper-Node*, and *Mirror-MAB*. Each strategy is detailed in terms of operational principles, highlighting its respective advantages and drawbacks. Our simulations demonstrate that these proposed countermeasures significantly improve network resilience, boosting the PDR even under sophisticated MAB-based attacks. The main contributions of this paper are summarized as follows:

- We extend the analysis of the MAB attacker model, develop and evaluate the online and offline attack strategies against basic channel hopping methods, to provide the baseline for different countermeasures.
- We propose and thoroughly analyze three novel defense strategies (*Speed-Up*, *Helper-Node*, and *Mirror-MAB*), discussing their operational characteristics and effects for the MAB-based attacks.
- We evaluate the proposed strategies by the metrics of packet delivery ratio (PDR), and test various settings or parameters of each strategy, demonstrating their effectiveness against the MAB-based jamming algorithm we used in the paper.

The project focuses on exploring mitigation for MAB-based attacks. Capodoglio, aka the "sperm whale", represents a mammal that is deep-diving (simulating the

exploration of jammer) and highly adaptable (using a variety of strategies for defense). This metaphor highlights the framework's ability to operate at varying levels of complexity while maintaining a single, coherent, and effective outcome.

The rest of the paper is structured as follows. Section 2.2 surveys related work on jamming, anti-jamming, and learning-based methods. Section 2.3 introduces the system overview used for our work, while Section 2.4 provides an overview of the MAB framework as a core methodology of attacks or defense strategies. Section 2.5 discusses the online and offline MAB-driven attacks. Next, Section 2.6 elaborates on our proposed defense mechanisms. Section 2.7 presents the simulation setup and results, validating the effectiveness of the proposed solutions. Finally, Section 2.8 concludes the paper and outlines potential directions for future research.

2.2 Related Work

2.2.1 Channel Hopping with Reinforcement Learning

Channel hopping countermeasures range from simple sequential policies to adaptive strategies driven by learning. We focus on approaches where devices select the next channel using feedback from the network. Multi-armed bandit algorithms have proved effective in noisy wireless settings [172]. In [67], a reinforcement learning framework decides whether a user should leave a heavily jammed network and associate with another access point, or instead switch channels. A related idea in [187] uses reinforcement learning to choose between combating the jammer and leaving the network. Both studies discretize the signal-to-interference-plus-noise ratio as the learning state, which limits applicability in real-world scenarios. To avoid an unbounded state space of signal to interference plus noise ratio values, [113] proposes to use spectrum waterfall snapshots as states, which removes the need to know the jamming pattern and extends naturally to intelligent and dynamic jammers. Multi-agent reinforcement learning has also been investigated to address both the attacker and the mutual interference among legitimate nodes [179, 218]. Most of these works, however, do not consider the power constraints typical of Internet of Things devices. [191] formulates a Markov decision process and reports power consumption under sweep, random, and sensing-based jammers, but does not cover intelligent adversaries.

2.2.2 Wi-Fi Jamming and Anti-Jamming in Practice

Physical layer jamming is a long-standing threat to IEEE 802.11 networks since modest power interferers can reduce throughput or even deny service by corrupting control and data frames. Early defenses emphasized reactive spectrum agility and channel blacklisting, where a victim moves to a less congested channel once packet loss or energy rises [60, 55]. To control false alarms, practical detectors often combine packet delivery-based indicators such as PDR with energy or RSSI heuristics, which keep overhead low and avoid PER/PDR ambiguity.

Beyond agility, protocol, and network-level defenses have been explored. Reputation-based isolation can quarantine suspected transmitters at higher layers, but coordination overhead and slow reaction to agile attackers limit effectiveness [160]. Rendezvous coding and coordinated hopping, such as SPADE, increase rendezvous probability under jamming, yet require pre-shared logic and introduce synchronization and control costs that may reduce throughput under benign conditions [146]. Measurement studies show that inexpensive software-defined radio

platforms enable adaptive jammers that remain effective even against randomized hopping, exposing the limits of purely reactive defenses [22, 21]. System surveys also note that dense and heterogeneous Wi-Fi and Internet of Things deployments increase vulnerability due to overlapping spectrum use and fragmented management domains [72, 62, 212, 195].

2.2.3 Learning based Jamming and Mitigation

Recent work has shifted toward data-driven algorithms for both attacks and defenses. On the attacker side, context-free multi-armed bandits and deep reinforcement learning methods allow jammers to learn channel preferences that maximize disruption without assuming a known hopping policy. Quantized signal to interference plus noise ratio or PDR can serve as immediate rewards, which supports online adaptation with minimal prior knowledge [67, 187]. Thompson sampling is attractive because it balances exploration and exploitation with closed-form Beta Bernoulli updates and performs well under non-stationary wireless returns [172]. Variants that ingest spectrum waterfall snapshots further reduce reliance on engineered features [113]. Multi-agent reinforcement learning has been studied to co-adapt across multiple interferers and victims [179, 218]. Lightweight designs target embedded constraints in Wi-Fi access points and Internet of Things nodes [158], and a few studies include power budgets and energy-aware behavior [191]. Empirical evaluations consistently show that learning based jammers can track randomized or deterministic hopping with limited observations [22, 21].

On the defender side, learning appears in three complementary roles. First, learning to detect uses supervised or semi-supervised models to flag jamming by fusing physical and medium access indicators, which improves sensitivity over simple thresholds while controlling false positives in dynamic traffic [166, 129, 174]. Second, learning to evade uses reinforcement learning agents to drive channel, rate, and power adaptation directly to maximize long-run packet delivery under adversarial feedback [117, 219]. Third, learning the adversary instantiates a mirror multi-armed bandit or reinforcement learning process on the defender that anticipates jammer preferences and proactively avoids vulnerable channels. Compared with static agility, these methods react faster to regime changes and reduce dwell time on compromised channels, but they must avoid oscillations and keep training overhead small to preserve throughput.

2.3 System Overview

In this section, we describe the system model considered in this paper, including the network model, threat model, and defense strategy.

2.3.1 Network Model

In this paper, the network model is a wireless network system, which includes an Access Point (AP) and three stations, as shown in Figure 2.1. Specifically, the AP in our network model plays the critical role in running the defense strategy and reacting to the environment state changes. The wireless network system is used to test the performance of our proposed defense strategies.

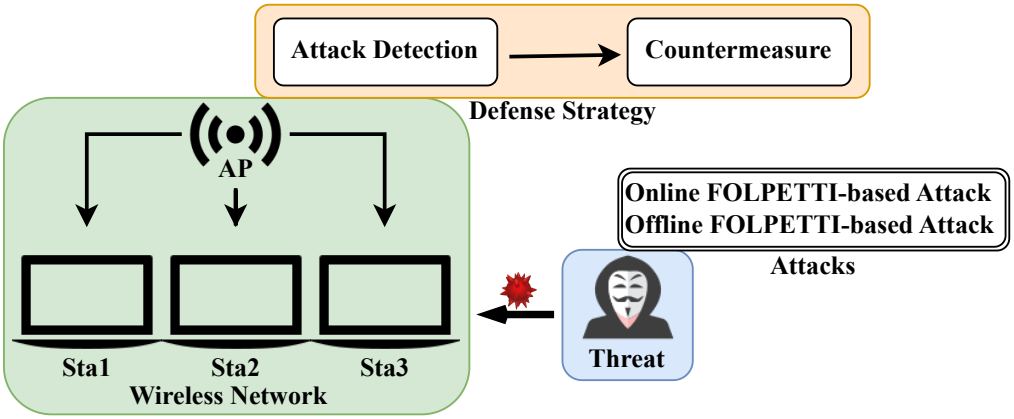


FIGURE 2.1: System Model.

2.3.2 Threat Model

As shown in Figure 2.1, the threat model in our paper is the jammer, which aims to attack the target frequency channel under the guidance of the MAB-based attacks. In the field of wireless security, jammers typically emit continuous noise, mitigating the quality of communication at the expense of energy efficiency. Specifically, the jammer in our paper is a kind of intelligent jammer, which employs machine-learning techniques to infer channel hopping patterns and target them adaptively. Also shown in Figure 2.1, the jammer runs online or offline MAB-based attacks to track the channel switching of the target wireless network and keep the jamming persistent and successful. The MAB-based attacks will be described in Section 2.5.

2.3.3 Defense Strategy

As shown in Figure 2.1, the wireless network, specifically the AP, runs the defense strategies for avoiding compromise from attackers. In Section 2.6, we proposed three countermeasures and tested them in Section 2.7. Despite the countermeasures having different specific implementations, they are still based on the following two essential steps:

Attack Detection

In the wireless network, the AP runs a detection method to be aware of the outside attacks. To develop the detection method, we use the PDR as the performance metric for the detection method. This type of metric is often used to identify jamming attacks [197, 3] and can be computed as

$$\text{PDR} = \frac{\sum \text{Number of PSD}}{\sum \text{Number of PT}}, \tag{2.1}$$

where PSD is the number of packets successfully received at the destination, whereas PT represents the overall number of packets transmitted by the source.

We implement an effective detector based on the network status and PDR metric, which runs in the AP as a fundamental component. In particular, it computes the average PDR in the latest rounds and thus defines a detection threshold δ. If the AP detects a PDR below δ for consecutive intervals, then it raises an alarm that the current channel is under attack. The choice of PDR as a detection approach is mainly

based on the consideration that it allows the detection of jamming attacks without increasing the computational overhead.

Countermeasure

Once an attack is detected, the AP activates one of the following defense strategies:

- **Speed-Up** uses the random channel hopping algorithm, with an increasing frequency of channel hopping, to make the Reinforcement Learning-based attacker hard to follow the network's current channel.
- **Helper-Node** introduces a third-party device to mislead the jammer into losing the channel change tracking. The random channel hopping algorithm runs for channel switching.
- **Mirror-MAB** runs in the wireless network and predicts the vulnerable channels, updating the parameters alongside the system's running. The AP decides the next channel from the available channels, except for channels that are probably being attacked.

These defense strategies, i.e., Speed-Up, Helper-Node, and Mirror-MAB, will be described in Section 2.6.

2.4 MAB Framework

In this study, both the jammer and the legitimate AP can be formulated within the multi-armed bandit (MAB) framework, each adapting its channel-selection strategy to its own objectives. Exploiting the adaptive nature of MAB algorithms, both agents continuously update their choices, thereby influencing network robustness under adversarial interference.

To situate this model, we briefly recall the reinforcement learning (RL) formulation. A generic RL environment can be described by the tuple $\langle S, A, R, P \rangle$, where S is the state space, A the action space, R the reward distribution, and P the state transition dynamics. We retain this notation to contrast with RL; in *our* bandit setting, however, the reward depends only on the selected action and is thus *state-free*. This simplification aligns with the multi-armed bandit abstraction, where each available frequency channel corresponds to an arm, and the jammer sequentially selects arms in order to maximize its long-term reward.

Formally, the MAB provides a framework for sequential decision-making under uncertainty. At each round t, the agent selects an action (i.e., pulls an arm), observes an immediate reward r_t, and updates its policy to maximize the cumulative reward $\sum_t r_t$, which is equivalent to minimizing cumulative regret. The key distinction from general RL lies in the absence of state transitions: in MAB problems, the reward r_t depends only on the chosen action a_t, not on any external state or past observation. This makes MAB a natural abstraction for frequency-hopping systems under adaptive jamming.

In this paper, the MAB-based attacks, or Mirror-MAB, can be modeled via the MAB algorithm. The main objective of the algorithm is to identify a policy π associating an action with each state tuple (i.e., $\pi \colon S \rightarrow A$). We use the Thompson sampling algorithm as our policy to model the problems, inspired by [172]. Thompson sampling is one of the most effective algorithms for solving contextual MAB problems. In Thompson sampling, there are K arms (i.e., actions) and k indicates the

arm index, the Beta(α, β) distribution for each action corresponds to one arm with parameters α_k and β_k [145, 1]. We denote by μ_k the event of success $(r_t(a_t) = 1)$ when selecting action a at round t. For each arm k, the probability of success is given by:

$$p(\mu_k) = \frac{\Gamma(\alpha_k + \beta_k)}{\Gamma(\alpha_k)\,\Gamma(\beta_k)} (\mu_k)^{\alpha_k - 1} \left(1 - \mu_k\right)^{\beta_k - 1}, \tag{2.2}$$

where $\Gamma(\cdot)$ is the gamma function [74]. Take these priors to be *beta* distributed with parameters $\alpha \in (\alpha_1, ..., \alpha_K)$ and $\beta \in (\beta_1, ..., \beta_K)$, where K is the available actions number in the model. If the MAB algorithm applies arm m at round t and returns a reward r_t, the *Beta* parameters for that channel are updated by Bayes' rule, taking advantage of the conjugacy properties:

$$(\alpha_k, \beta_k) \leftarrow \begin{cases} (\alpha_k, \beta_k) & \text{if } m \neq k, \\ (\alpha_k + r_t,\ \beta_k + 1 - r_t) & \text{if } m = k, \end{cases} \tag{2.3}$$

where $k \in K$ is the arm indicator in the paper. Hence, the MAB algorithm with Thompson sampling forms the core methodology in our project. A beta distribution with parameters (α_k, β_k) has mean $\alpha_k/(\alpha_k + \beta_k)$, and the distribution becomes more concentrated as $\alpha_k + \beta_k$ grows.

The MAB with Thompson sampling flowchart is shown in Figure 2.2. The *Beta* distribution ensures that arms with higher observed success rates gradually receive larger α_k relative to β_k, increasing their probability of being selected, while still allowing occasional exploration of less-sampled arms. By updating the *Beta* parameters in proportion to observed successes and failures, Thompson sampling naturally skews future samples toward arms with higher empirical performance, while preserving randomness for continued exploration.

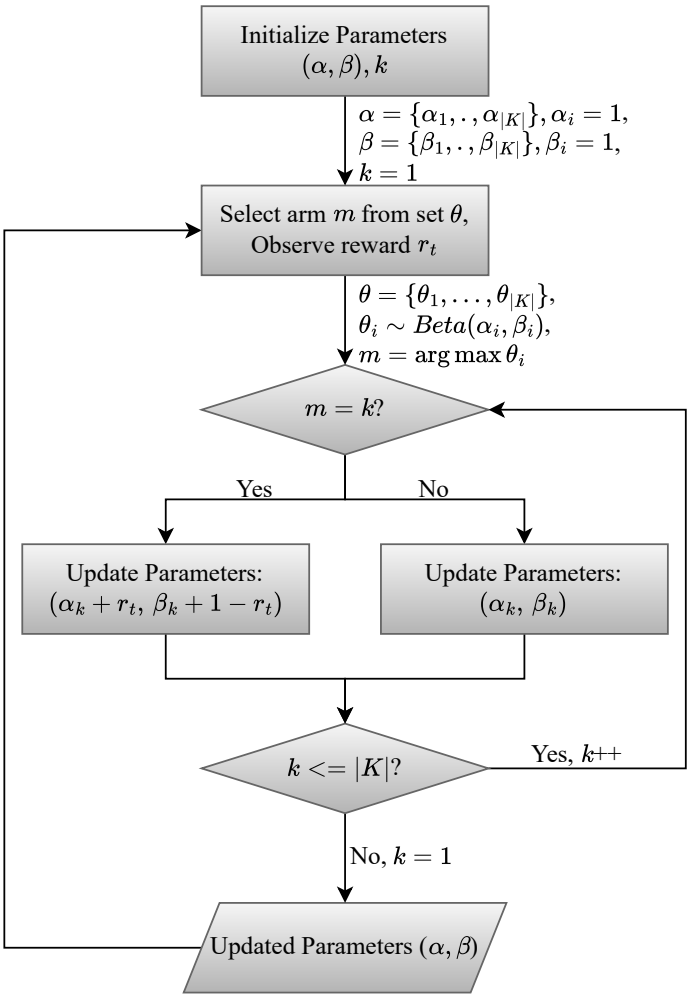


FIGURE 2.2: MAB with Thompson Sampling Framework.

2.5 The MAB-based Attacks

In this section, we provide a detailed description of the proposed attacks. E. Bout *et al.* [22] have validated the effectiveness of MAB, showing its application to a jamming attack. Typically, an efficient attack against channel hopping has two fundamental requirements: (i) it should not depend on specific assumptions about the victim's hopping pattern, and (ii) it needs to be continuous in time.

In this paper, we developed two MAB-based attacks, i.e., online and offline, based on different considerations [111]. The main difference between the two approaches is the frequency of channel updates and the reward obtained in each round. Based on the approach proposed in [22], we implemented the online MAB-based attack and modified the original model to develop the offline MAB-based attacks. The following are the explanations of these attacks.

- **Online Attack:** In [22], the authors proposed the original MAB model. We develop the online MAB-based attack on the original model, where the channel selection and rewards computation are performed on a per-round basis. This means that, for each round, if the attack selects the arm a_t and successful attacks on arm k (i.e., $a_t = k$), then $\alpha_k = \alpha_k + 1$; if not, $\beta_k = \beta_k + 1$.

Algorithm 1 Offline MAB-based Algorithm.

Require: channel index: k, total transmissions: c_k, successful jam: j_k, Batch Round: N, *Beta* distribution parameters: (α, β);

for `all` k **do**
  $c_k = 0, j_k = 0, \alpha_k = 1, \beta_k = 1$
end for
while True **do**
  if $c_k \mod N == 0$ **then** ▷ Update every N rounds;
    for `all` k **do**
      Update parameters: $(\alpha_k + j_k, \beta_k + (c_k - j_k))$
      Sample Probability: $\theta_k \sim \beta(\alpha_k, \beta_k)$
    end for
    Select next channel: $m = \arg\max_k \theta_k$ ▷ MAB selection;
  end if
  c_m++;
  JAM(); ▷ Attempt transmission on channel m;
  if Successful transmission **then**
    j_m++;
  end if
end while

- **Offline Attack:** An alternative approach is offline learning, i.e., we assume that rewards and channel updates are computed on batch rounds. Specifically, let us assume that the offline MAB-based attack updates its choice and parameter every N rounds. Then, the parameters are updated based on the number of successes and failures of the attack. In particular, assume the attacker pulls arm a_t for the current round t. By denoting as r_{succ} the number of successes for arm $a_t = k$ and as r_{fail} the number of failures for arm $a_t \neq k$ at a batch round N. Then, at the end of the batch, $\alpha_k = \alpha_k + r_{succ}$, and $\beta_k = \beta_k + r_{fail}$. In this setting, the attacker proceeds as described in Algorithm 1.

2.6 Defense Strategy Against MAB

In this section, we describe the proposed approaches to defend against MAB-based attacks. We propose a *Speed-Up* strategy in Section 2.6.1, a *Helper-Node* strategy in Section 2.6.2, and a *Mirror-MAB* strategy in Section 2.6.3.

2.6.1 Speed-Up

In wireless networks, the radio channel is ordinarily held for long intervals, such as minutes to hours, thereby the attacker has enough time to analyze the channel status. Thus, in real-world wireless networks, the channel hops only when external conditions or policy apply. MAB-based attacks rely on accurate feedback to update the *Beta* parameters that govern their arm-selection rule. The Speed-Up countermeasure neutralizes this condition by controlling the legitimate transmitter (AP and stations) to hop more aggressively, thereby invalidating the jammer's accumulated knowledge.

The defense strategy, i.e., Speed-Up, is that once the victim detects the attack, it increases the rate at which it performs channel hopping. In particular, the selection of a new channel is not subject to a significant decrease in the PDR, but rather to a given timing. With the Speed-Up strategy, channel changes occur periodically: after every N transmitted rounds, the transmitter switches to the next new channel according to a specific policy. Consequently, the jammer observes a non-stationary

target and must continuously re-learn the reward landscape, markedly reducing the dwell time it can spend on any single channel.

2.6.2 Helper-Node

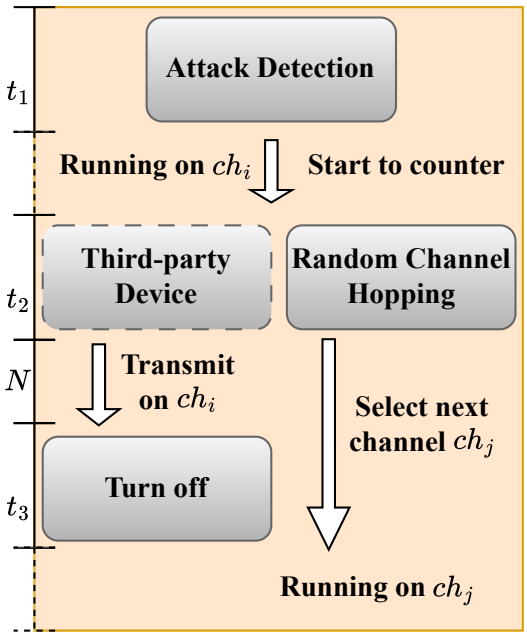


FIGURE 2.3: Workflow for Helper-Node.

Because MAB-based attacks require a certain number of steps to update their strategy and attack the targets, the victim can utilize this time, sacrificing part of its resources to defend against the MAB-based attack. To this aim, the victim can deploy an additional (third-party) node that runs a fixed hopping scheme based on the various scenarios.

As shown in Figure 2.3, let us consider the round t_1 in which the victim detects the attack on the channel with index ch_i. The Helper-Node will turn on the third-party device for transmitting noisy data on channel ch_i, while the legitimate station transmits on another channel ch_j, i.e., $ch_j \in M \setminus ch_i$, where M is the available channels list. The role of the Helper-Node is thus to poison the decision-making capabilities of MAB: by creating dummy regular traffic on channels not used by the legitimate node, the Helper-Node forces MAB to attack useless (i.e., information-less) channels. This strategy requires coordination between the Helper-Node and the legitimate transmitting device to avoid interference.

2.6.3 Mirror-MAB

In this section, we introduce the Mirror-MAB strategy (i.e., an MAB algorithm running in the device AP) to counteract the MAB-based attacks. Our objective is to safeguard Wi-Fi communications by leveraging the MAB algorithms. When the network is active, the AP executes the Mirror-MAB algorithm, continuously updating Thompson sampling parameters based on observable environment states and returned rewards.

The Mirror-MAB algorithm runs a *mock* of the jammer's behavior in parallel. More precisely, it internally simulates the attacker's jamming patterns to anticipate

which channels the attacker will attack. By modeling the jammer's state transitions, the Mirror-MAB maintains a set of parameters for each channel. These parameters are periodically updated using observed states, effectively learning how the jammer operates. The detailed steps are given below.

1. **Initialize Parameters:** At the beginning of the system, the Mirror-MAB initializes the relative parameters, e.g., $Beta(\alpha, \beta)$.
2. **Observe Reward:** Compute the reward r after a batch round.
3. **Parameters Update:** After observing the reward, the Mirror-MAB adjusts its internal parameters (i.e., α, β).
4. **Vulnerable Channels & Available Channels:** Using the updated parameters, the Mirror-MAB predicts a vulnerable channels list $B = \{ch_{b_1}, ..., ch_{|B|}\}$ most likely under attack. Meanwhile, the Mirror-MAB returns an available channels list $M = \{ch_{m_1}, ..., ch_{|M|}\}$ from the wireless network.
5. **Channel Switching:** Selecting the next channel ch, where we use the random policy to decide the new channel from the channels list $M \setminus B$.

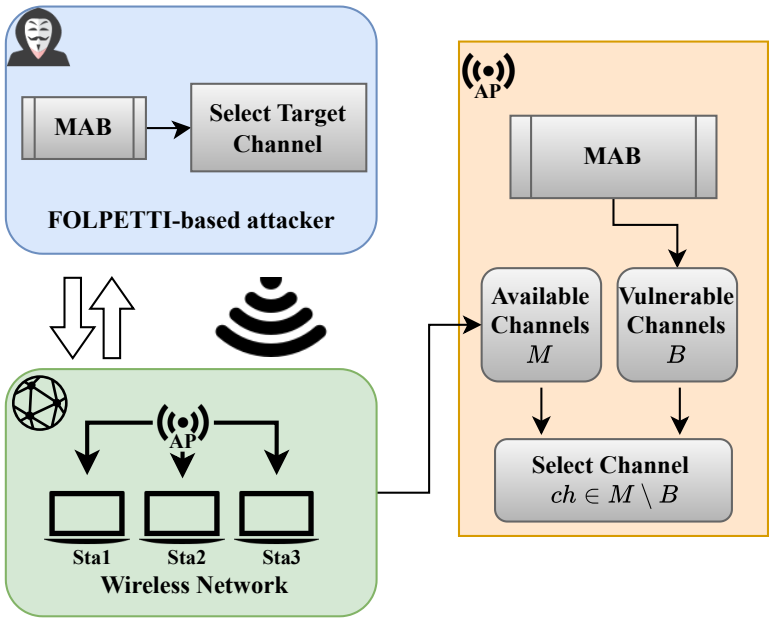


FIGURE 2.4: Countermeasure: Mirror-MAB.

As shown in Figure 2.4, the rightmost part stands for the Mirror-MAB method, where Mirror-MAB runs in the AP device. The Mirror-MAB is a trusted approach in the network, and it could collect the rewards and update the parameters according to the network status, and predict the vulnerable channels for the network.

2.7 Performance Evaluation

We describe in Section 2.7.1 the system we consider for implementing the evaluation settings. In Section 2.7.2, we describe the different MAB strategies employed by the attacker, then show the performance by running the attack method in Section 2.7.3.

2.7.1 Evaluation Settings

We consider a wireless network composed of wireless devices, specifically, the wireless network includes an AP, three legitimate stations connected via an AP, and capable of transmitting on 12 different channels. We assume that the attacker aims to jam the communication between the AP and the legitimate nodes in the network.

To follow the effects of the jamming attacks and compute the PDR, we assume that the AP constantly transmits 12 packets every round and begins its transmission at the start of the simulation ($t = 0$). After 10 seconds, the attacker starts its attack. The legitimate stations and the attacker start their communication on the same channel. Table 2.1 summarizes the simulation parameters.

TABLE 2.1: Simulation parameters.

Parameter	Setting
Simulation Time (seconds)	1800
Start of jamming (seconds)	10
Start of channel hopping (seconds)	1
Threshold Detection (%)	80

We implemented this module using a custom-built network simulator, considering only the three kinds of fundamental nodes, i.e., the legitimate stations, the AP, and the attacker (jammer). For every setting, we run the simulation one hundred times to collect the results and calculate the average performance.

2.7.2 Attacks Performance

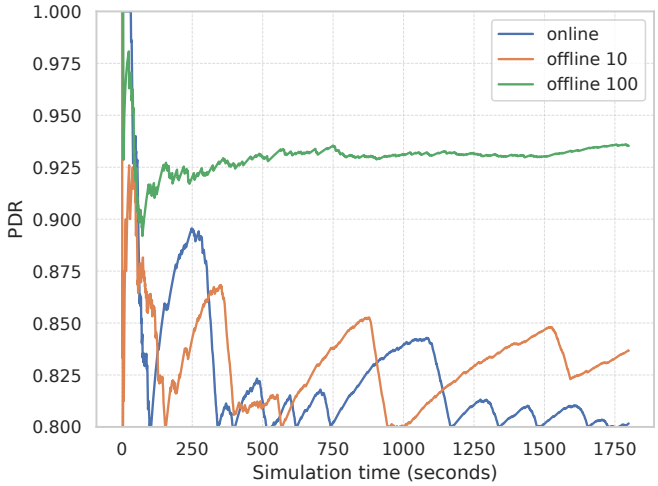


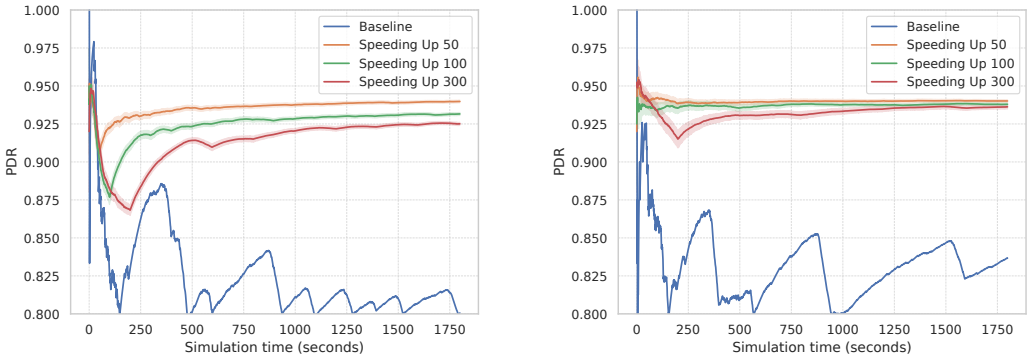
FIGURE 2.5: PDR for different attack strategies.

We develop and evaluate the two implementations of the MAB-based attacks, i.e., the online and offline attacks. The attack detection threshold is set to 80% PDR, i.e., when the victim detects a PDR below 80%, it switches to the following channel. The closer the PDR is to the detection threshold, the more the attack affects the network.

The impact of online and offline MAB-driven attacks on PDR without defense strategies is illustrated in Figure 2.5. The results show that the online attacker consistently reduces the network's PDR more significantly than the offline attacker, where 10, 100 denote the parameter update delays. Under the same evaluation settings, the online attacker successfully jams the victim's channel approximately 39.7% of the time, compared to only 20.6% for the offline attacker. For subsequent experiments, the online setting and the offline-10 configuration are used as baselines for evaluating the proposed defense strategies.

2.7.3 Performance of Our Proposed Defense Approaches

Performance of Speed-Up



(A) PDR Speed-Up for online attack.

(B) PDR Speed-Up for offline attack.

FIGURE 2.6: PDR for Speed-Up parameters.

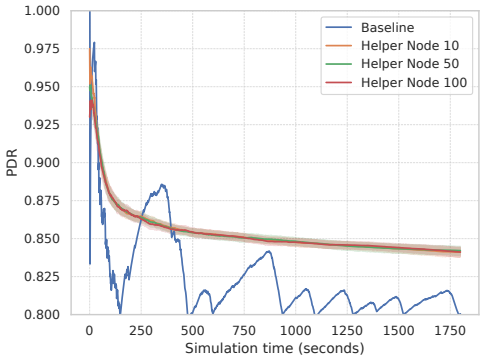
To better understand the advantage of the Speed-Up defense strategy, we initially describe its behavior and compare it with that of online and offline MAB attacks. The Speed-Up strategy envisions channel hopping occurring on a predefined time basis or detection of a drop in the PDR. In Figure 2.6, the parameter (e.g., 50) is the interval of channel change, and when the Speed-Up parameter value is set to 0, it means that no Speed-Up is deployed.

The PDR of the network under random channel hopping with different Speed-Up parameters is illustrated in Figure 2.6. In this experiment, we compare several Speed-Up configurations, namely 50, 100, 300, which represent the update intervals for each test, against a baseline case without Speed-Up. The results indicate that the Speed-Up strategy consistently improves the PDR of the network. In particular, with an update interval of 50, the network achieves a PDR of 0.93 under both online and offline attacks, whereas the baseline PDR remains below 0.9. Similarly, the settings with update intervals of 100 and 300 also outperform the baseline configuration.

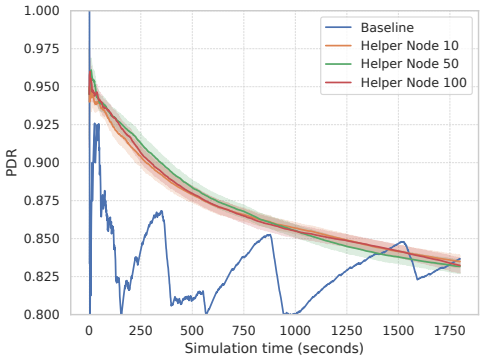
Performance of Helper-Node

In this section, we show the results of the Helper-Node strategy with different parameters. We notice that the Helper-Node defense does not depend on the specific values from Figure 2.7, as the victim just needs to move to the successive channel, regardless of how the successive channel index is selected.

As shown in Figure 2.7a and Figure 2.7b, the PDR curves are shown when considering different Helper-Node values for the online and offline attacks, respectively.



(A) PDR Helper-Node for online attack.

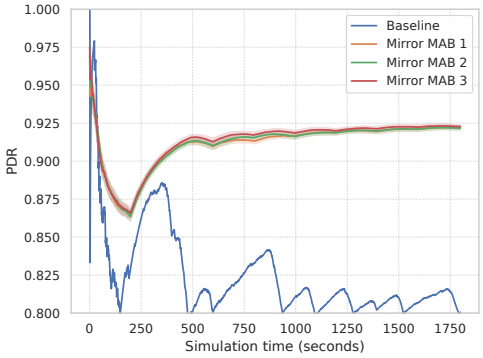


(B) PDR Helper-Node for offline attack.

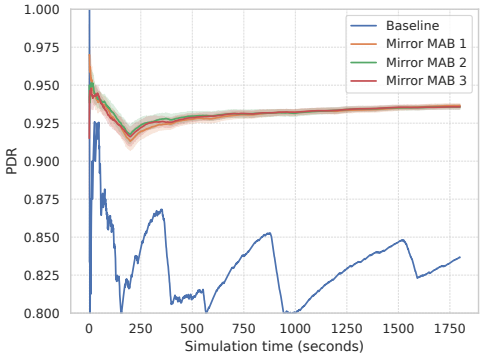
FIGURE 2.7: PDR for Helper-Node.

We see that the Helper-Node strategy is helpful in mitigating the attack effects, as the victim can improve the PDR with the defense strategy. From Figure 2.5, Figure 2.7a and Figure 2.7b, we notice that under online and offline attackers, using Helper-Node with different Helper-Node settings, e.g., $10, 50, 100$, can improve the performance of PDR, and keep the PDR around $0.825 \sim 0.85$.

Performance of Mirror-MAB



(A) PDR Mirror-MAB for online attack.



(B) PDR Mirror-MAB for offline attack.

FIGURE 2.8: PDR for Mirror-MAB.

The PDR over time under both online and offline attack scenarios is illustrated in Figure 2.8. The experiment compares the baseline approach with the proposed Mirror-MAB strategy configured with different parameter values, namely $1, 2, 3$, which denote the number of vulnerable channels predicted by the algorithm. The baseline approach exhibits significant PDR degradation, dropping below other settings before stabilizing, which reflects its inability to adapt dynamically to jamming. In contrast, the Mirror-MAB strategies maintain a higher and more stable PDR, around 0.92 and 0.93 under the online and offline attacks, respectively. For offline attacks, the initial drop is less severe due to the more predictable jamming pattern. Overall, the Mirror-MAB strategies consistently outperform the baseline by leveraging historical data for more informed channel selection. Although the offline

variant is less adaptive than the online case, it still provides a robust defense against intelligent jamming, achieving consistent improvements over the baseline.

Summary

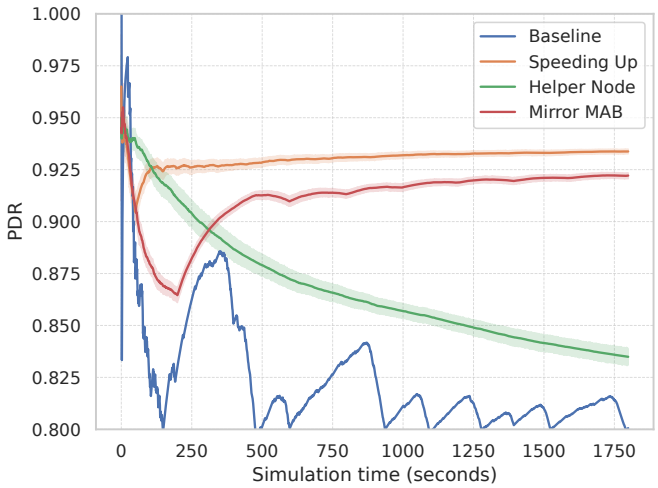


FIGURE 2.9: PDR for different strategies.

The PDR performance over time for different defense strategies against MAB attacks is illustrated in Figure 2.9. The comparison includes the *baseline*, *Speed-Up*, *Helper-Node*, and *Mirror-MAB* approaches. The horizontal axis represents the simulation time in seconds, while the vertical axis indicates the achieved PDR.

Overall, we observe that the *Speed-Up* strategy consistently surpasses the other methods in reaching higher PDR values more rapidly. Its aggressive channel-switching mechanism allows it to evade jamming attempts sooner, translating into a faster recovery of communication reliability. Meanwhile, *Helper-Node* employs additional resources in the network to counteract jamming, helping to maintain a robust PDR once the system stabilizes. The initial recovery slope for *Helper-Node* is slower than *Speed-Up*, which means a long period of development of the defense strategy to mitigate smart attacks.

By contrast, the *Mirror-MAB* approach benefits from a dynamic learning framework that updates its defense strategy based on observed rewards. Despite a slightly slower reacting phase compared with *Speed-Up*, *Mirror-MAB* proves effective in converging to a high PDR, reflecting its ability to track and counter evolving jamming tactics adaptively. In all cases, once the strategies stabilize, the victim sustains near-optimal PDR levels despite ongoing jamming. Thus, the results confirm that both *Speed-Up* and *Mirror-MAB* yield rapid improvements.

2.8 Conclusion

Jamming attacks represent a serious threat to wireless communications security. In particular, MAB-based attackers leveraging smart jamming strategies to follow channel hopping patterns can undermine the effectiveness of some defense mechanisms. Our study demonstrated that both *online* and *offline* variants of MAB-based attacks can rapidly learn the hopping behavior of typical IEEE 802.11 deployments.

To address this challenge, we proposed and evaluated three countermeasures: *Speed-Up*, *Helper-Node*, and *Mirror-MAB*, and demonstrated their effectiveness in mitigating the sustained impact of MAB-based attacks.

Although the results are encouraging, several avenues are invited for further investigation. In future works, we could implement the proposed defenses on a hardware testbed in order to assess their practicality, overhead, and robustness under real-world propagation conditions. Furthermore, we will explore the possibility of using more Reinforcement Learning algorithms as a friendly jamming mechanism to exclude possible network intruders.

Chapter 3

Generative DOA Estimation (DOA-CGAN)

Current Direction of Arrival (DOA) estimation methods are unable to provide reliable estimates when faced with jamming attacks. To address this issue, we propose Direction of Arrival Estimation via Conditional Generative Adversarial Networks (DOA-CGAN), the first generative approach to remove the jamming component from the received signal covariance matrix. We first design a generator aimed at evaporating jamming components from the input covariance matrix. In our model, we input the received signal covariance matrix to an unsupervised generator that filters it to generate a matrix that can be deemed legitimate by a supervised discriminator. After training, we leverage the generator as a filter able to remove the jamming component from the received signal covariance matrix and feed its output to classical DOA estimation algorithms. The generator synthesizes authentic covariance matrices that accurately represent legitimate signals. Subsequently, we enhance the discriminator by embedding tags that control specific properties of the generated data. Numerical results demonstrate that our proposed method delivers robust DOA estimation compared with other machine learning methods with a root mean squared error smaller than 0.2°.

3.1 Introduction

Jamming is a big threat to a radar system. From a jamming mechanism perspective, active intentional jamming is categorized into two types: suppression jamming and deception jamming [131]. Under conditions free from jamming, radar systems can readily perform Direction of Arrival (DOA) estimation [42]. However, with the ongoing advancements in jamming theory and technology, these developments pose significant challenges to the DOA estimation.

DOA estimation holds paramount importance in defense and military, sonar, earthquake analysis, and wireless communication [42, 104]. To estimate DOA, researchers developed many high-resolution algorithms, such as Digital Beamforming, Multiple Signal Classification (MUSIC) [79], and Maximum Likelihood [124] methods. However, these algorithms are not able to withstand the presence of a jammer, i.e., a malicious device generating interference signals. To address this challenge, the Minimum Mean Square Error (MMSE), Minimum Variance Distortionless Response (MVDR), and Linearly Constrained Minimum Variance (LCMV) are prominent adaptive beamforming techniques [102, 135] employed to mitigate interference. These methods dynamically adjust the weights of array antenna elements to enhance signal reception to enhance the Signal-to-Interference-plus-Noise Ratio (SINR) in targeted directions and suppress interference from others. Nevertheless,

these algorithms typically require a preliminary estimate or a predefined assumption regarding the direction from which the desired signal originates. To the best of our knowledge, no existing approach estimates DOA under conditions of random directional jamming signals.

In this regard, it is necessary to investigate DOA estimation in jamming scenarios without requiring additional prior information. Nowadays, many researchers have developed DOA estimation methods that leverage machine learning techniques [134, 182, 15, 118, 194, 207, 184, 185]. In [15], the authors introduce a neural network-based approach for DOA estimation, employing subarray sampling to estimate the full covariance matrix. In [118], the authors use a Deep Convolutional Network to estimate multiple 2D angles via processing discrete subregions in two distinct stages. Xu et al. propose a Deep Neural Network that learns the transformation from received signals to the extended correlation matrix and avoids relying on specific physical structures [194]. What is particularly intriguing is that in [207], the authors proposed an adversarial attack on a deep learning-based DOA estimation system. This approach specifically targets the model's vulnerabilities, thereby compromising the accuracy of DOA estimates. On the other hand, the authors in [184] develop a steering vector embedding complex-valued Conditional Generative Adversarial Networks (CGAN) aimed at generating additional radar data, therefore improving the DOA estimation accuracy. Moreover, many works focus on interference signal identification and classification [141, 116, 151]. The authors in [141] propose a novel jamming recognition network that automatically recognizes both single and convolutional compound jamming signals. And Lv et al. in [116] introduce a novel approach utilizing an ensemble deep network combined with transfer learning to accurately identify radar active deception jamming, even with limited training set.

These aforementioned data-driven methods either enhance DOA estimation in environments free from interference or focus solely on identifying and classifying jamming signals. However, these models currently do not account for DOA estimation in jamming scenarios. Notably, their effectiveness against counter-interference signals remains unexplored representing a significant gap in the applicability of these advanced computational approaches.

In this work, inspired by the machine learning-based DOA estimation, we propose DOA-CGAN, a model specifically designed to enhance robustness against interference. Our model leverages an unsupervised generator aimed at creating correlation matrices deemed as legitimate (i.e., only including legitimate signal) by a supervised discriminator. The main idea is to train the generator such that it implements a filter able to remove the jamming component from the input received signal covariance matrix. After training, we leverage the generator as a filter that we apply before running classical DOA estimation algorithms. Via numerical simulation, we show that Direction of Arrival Estimation via Conditional Generative Adversarial Networks (DOA-CGAN) is effective in mitigating jamming attacks to DOA estimation. The generator endeavors to produce increasingly realistic signals to deceive the discriminator. Concurrently, the discriminator exerts effort to differentiate between authentic signals and those that are subject to interference. This interaction fosters a continuous and dynamic adversarial process that enhances the network's ability to mitigate interference effects. Note that we explore the real-world applications of CGAN in DOA estimation, rather than focusing on data augmentation. The key contributions of this work are summarized as follows:

- We identify a critical gap in existing data-driven DOA estimation methods,

which either improve estimation in interference-free settings or focus only on jamming classification, without addressing DOA estimation under active jamming scenarios.

- We propose **DOA-CGAN**, a conditional generative adversarial framework that employs an unsupervised generator and a supervised discriminator to remove jamming components from received signal covariance matrices. This design allows classical DOA estimators (e.g., MUSIC) to regain accuracy in the presence of interference.
- We conduct extensive numerical simulations to validate the effectiveness of DOA-CGAN, demonstrating significant improvements in estimation accuracy under multiple jamming conditions, while highlighting its potential for real-world deployment.

The remainder of this paper is organized as follows. Section 3.2 reviews related work on classical DOA estimation, anti-interference strategies, and recent learning-based approaches. Section 3.3 provides a primer on FMCW radar fundamentals, followed by the description of the threat model in Section 3.4 and the system model in Section 3.5. Section 3.6 presents the proposed DOA-CGAN framework, including the motivation, data preprocessing pipeline, adversarial model design, and objective optimization. Section 3.7 reports numerical experiments under various interference conditions, where we evaluate the estimation accuracy of DOA-CGAN and benchmark its performance against alternative learning-based models. Finally, Section 3.8 concludes the paper and outlines directions for future research.

3.2 Related Work

3.2.1 Classical DOA Estimation and Anti–Interference Techniques

Direction-of-Arrival (DOA) estimation has a long history in array signal processing and remains fundamental in radar, sonar, wireless communications, and seismology. Early high–resolution subspace methods, most notably MUSIC and its variants, exploit the orthogonality between signal and noise subspaces to achieve super–resolution under mild assumptions on source number and array calibration [149, 161, 98]. Parametric approaches based on maximum likelihood and deterministic/stochastic Cramér–Rao performance analyses offer asymptotic optimality but at the cost of non–convex optimization [161].

In parallel, adaptive beamforming methods such as Capon/MVDR and its linearly constrained extension LCMV maximize signal–to–interference–plus–noise ratio (SINR) by shaping the receive beampattern through covariance–aware weight adaptation [30, 103, 102]. Robust variants address model mismatch via diagonal loading, covariance shrinkage, and constraint set inflation [103, 102]. To alleviate sample–starvation and nonstationarity, space–time adaptive processing (STAP) and reduced–dimension beamformers have been developed in radar to suppress non–point interference and clutter.

Beyond subspace and adaptive filtering, sparsity–promoting and gridless formulations further advanced the state of the art. Compressive sensing–based DOA solves a (group) sparse recovery over discretized angle grids (e.g., ℓ_1–SVD/OMP), while gridless approaches (e.g., atomic norm minimization/SPICE/ESPRIT) avoid

basis mismatch and deliver high resolution with modest snapshots [98]. These classical families, however, typically rely on accurate covariance estimation, array calibration, and mild interference conditions. Under deceptive or smart jamming, especially when the jammer contaminates the covariance structure, subspace separation and narrowband interference assumptions can be violated, degrading resolution or biasing the spectrum.

3.2.2 Machine and Deep Learning for DOA

Motivated by the limitations of purely model–based pipelines, recent years have witnessed substantial interest in data–driven DOA estimation. A first thread uses deep networks as discriminative regressors/classifiers to map sensor snapshots or second–order statistics to angles. Convolutional architectures process spatial covariance matrices or spectrogram–like representations to learn robust features and regress azimuth/elevation directly [182, 15]. Recurrent models (e.g., LSTM) and hybrid CNN–RNN backbones capture temporal context across snapshots, while residual networks (ResNet/GoogLeNet) improve optimization and depth, enabling multi–source and 2D DOA under moderate SNR and snapshot budgets [118, 194]. These methods often achieve strong accuracy in benign or mildly nonideal conditions and can amortize inference for real–time deployment.

A second line leverages generative modeling and augmentation to improve generalization and robustness. Conditional GANs and related generative models synthesize array observations or intermediate representations to fill coverage gaps and diversify training data; they have been applied to radar/RF tasks to enrich angle–bearing variability and mitigate overfitting [184]. Yet, most augmentation–centric uses of GANs keep the downstream DOA estimator unchanged and do not explicitly *remove* structured interference.

More recently, adversarial robustness and anti–jamming have become active topics. Studies have shown that deep DOA estimators can be vulnerable to carefully crafted perturbations or deceptive waveforms that steer predictions away from true bearings [207]. On the defensive side, robust training, confidence calibration, and uncertainty–aware decision layers have been explored to temper overconfidence and improve reliability under distribution shift. Nevertheless, the majority of deep approaches either assume interference-free conditions or treat jamming as a separate recognition/classification task, rather than integrating *jamming removal* with DOA estimation.

3.3 Primer on FMCW Radars

As depicted in Figure 3.1, we examine a radar transmitter (Tx) designed to emit a linear Frequency Modulated Continuous Wave (FMCW) signal. This designation stems from the signal's characteristic use of frequency-modulated emissions, known as chirps. These chirps are defined by an initial frequency f_0, a chirp duration t_c, a bandwidth B, and a chirp slope μ. Then the transmitted signal propagates through the atmosphere until it encounters an object, such as an aircraft, ship, or terrain feature. Upon hitting these objects, the radio waves are scattered in many directions, with some of the waves reflecting directly back towards the radar system. The radar's antenna (Rx) captures the reflected signals, often referred to as "echoes". The same antenna used for transmitting can also be used for receiving, or a separate antenna may be utilized, depending on the radar design.

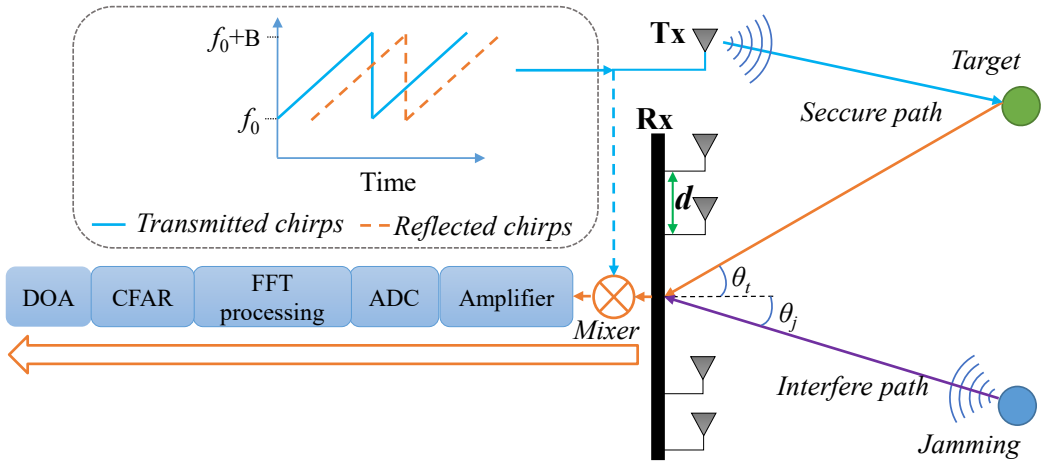


FIGURE 3.1: System model of an FMCW radar.

The received echoes, typically weak, necessitate processing to enhance their signal quality and extract pertinent target information. This processing sequence encompasses amplification to boost signal strength, noise filtering to purify the signal, and conversion into a digital format for detailed analysis. First, the signals, both transmitted and received, are directed to a quadrature mixer, producing an analog signal. This signal is then amplified and converted from analog to digital form through an Analog-to-Digital Converter (ADC) for digitization. Subsequent to digitization, the data undergoes processing with Fast Fourier Transforms (FFT), specifically Range-FFT for discerning distance information and Doppler-FFT for determining velocities, thereby generating a detailed range profile of the environment. Given that background noise often marred this range profile, the Constant False Alarm Rate (CFAR) technique is employed to isolate echoes from genuine targets amidst this noise reliably. The system is then primed to estimate the DOA of the target through the secure path.

3.4 Threat Model

However, in the military and aviation fields, adversaries may use falsified radar signals to deceive defense systems, preventing them from accurately identifying the position or trajectory of incoming targets. Similarly, by imitating legitimate radio signals, attackers can interfere with the normal operation of communication or navigation systems. As given in Figure 3.1, jamming devices process the radar signals captured by the receiver (Rx) and then retransmit them. Deceptive jamming primarily involves simulating real signals with certain modulation techniques to "create" a false target. After the deceptive jamming signal enters the enemy radar system via the receiving antenna, it is processed by the receiver. Because the power of the jamming signal is greater than that of the real target, the false target formed makes it impossible for the enemy radar to accurately and effectively obtain the parameter information of the real target, thereby affecting the detection and tracking of the real target.

To achieve this, we borrow Smeared Spectrum (SMSP) jamming technology based on Matched Signal Transform (MST), which can produce a large number of

Range False Targets (RFT) and make the radar system ineffective in measuring the information of the true target.

3.5 System Model

We consider a linear frequency-chirped waveform as a transmitting signal. The transmitted chirp signal can be analytically expressed as

$$s(t) = exp\left(j \cdot (2\pi f_0 t + \mu\pi t^2)\right), \quad 0 \leq t \leq T_c, \tag{3.1}$$

where μ denotes the frequency modulation slope and is defined by $\mu = \frac{B}{T_c}$, B represents the bandwidth of the Linear Frequency Modulation (LFM) signal, and T_c represents the signal's duration or time width. As a representative example, we consider SMSP as a jamming signal [217, 177]. This technique is characterized by

$$J_0(t) = exp\left(j \cdot (2\pi f_0 t + \mu'\pi t^2)\right), \quad 0 \leq t \leq T_j \tag{3.2}$$

where $\mu' = M\mu$ is the modulation slope of the jamming signal, $j = \sqrt{-1}$, and $T_j = T_c/M$ represents pulse width of the sub-waveform. The jamming signal model can be represented as

$$s_j(t) = J_0(t) * \sum_{m=0}^{M-1} \delta(t - mT_j), \quad 0 \leq t \leq T_j \tag{3.3}$$

where $*$ is the convolution symbol, and $\delta(\cdot)$ denotes the impulse function. As shown in Figure 3.2, the left border is the waveform of the LFM signal, the right is the one of the SMSP signal. The captured LFM signal is compressed in the time domain by a factor of M, reducing the duration of the signal to produce a sub-waveform $J_0(t)$ with a shorter time width T_j. This compressed sub-waveform is duplicated M times, generating a sequence of sub-pulses. These step-modulated sub-pulses are structured to maintain the overall time width similar to the original LFM signal. Still, each sub-waveform has only $1/M$ of the original signal's time width. This means the frequency modulation slope of the jamming signal is increased by a factor of M compared to the original LFM signal. Consequently, SMSP deception jamming possesses the capability to precisely emulate radar echo signals, rendering the radar system incapable of efficiently identifying authentic targets and discerning their attributes.

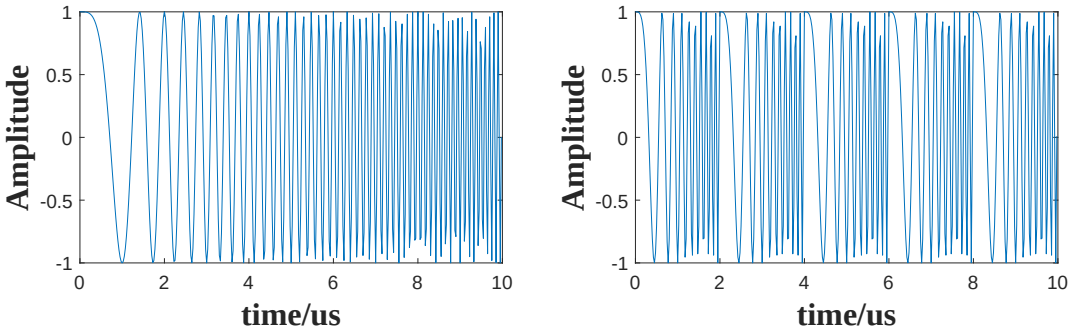


FIGURE 3.2: System model.

Note that the signal for DOA estimation is the baseband signal after the CFAR operation. Henceforth, in the ensuing chapters, unless explicitly stated otherwise,

$s(t)$ and $s_j(t)$ shall be construed as the baseband signal. In this paper, we suppose that the spoofing attack contains a counterfeit signal, which is transmitted from a single antenna, and there are We consider the signals to impinge on a phased array with N elements. The discrete received signal vector $x(t)$ at snapshot t can be model as

$$x(t) = s^S(t) + s^J(t) + n^(t), \tag{3.4}$$

where

$$s^S(t) = a(\theta_t)s(t), \quad s^J(t) = a(\theta_j)s_j(t); \tag{3.5}$$

$$s^J(t) = a(\theta_j)s_j(t), \tag{3.6}$$

in which $s^S(t)$ and $s^J(t)$ represent the authentic signal component and the jamming component, respectively, while $n(t)$ denotes the additive white Gaussian noise with zero mean and variance σ_n^2. Moreover, the variables $a(\theta_t)$ and $a(\theta_j)$ are the steering vector of the authentic target and jamming signal, respectively, which can be expressed as

$$a(\theta) = \left[1, e^{j\frac{2\pi}{\lambda}d\sin\theta}, \cdots, e^{j\frac{2\pi}{\lambda}(N-1)d\sin\theta}\right]^T, \tag{3.7}$$

where the parameter λ is the wavelength of the signal, and d denotes the distance between adjacent array elements. We compute the correlation matrix R of the received as

$$R = \mathbb{E}\left[x(t)x^H(t)\right], \tag{3.8}$$

where $\mathbb{E}[\cdot]$ denotes the statistical expectation. The superscript $(\cdot)^T$ and $(\cdot)^H$ represent the transpose and conjugate transpose, respectively. In practice, the correlation matrix R is replaced with its sample estimate

$$R = \frac{1}{K}\sum_{t=1}^{K} x(t)x^H(t), \tag{3.9}$$

where K is the number of snapshots used for its estimation. The correlation matrix of legitimate signals can be written as

$$R_s = \mathbb{E}\left[s^S(t)s^{S^H}(t)\right]. \tag{3.10}$$

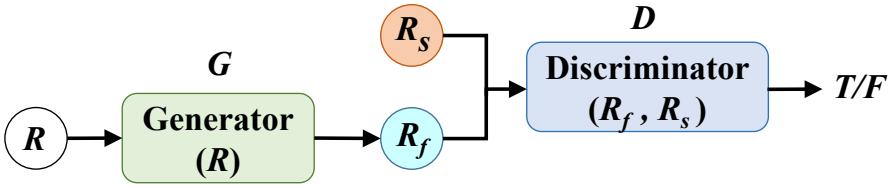
3.6 Learning Networks for DOA Estimation

3.6.1 Intuition Behind Our Learning-Based Approach

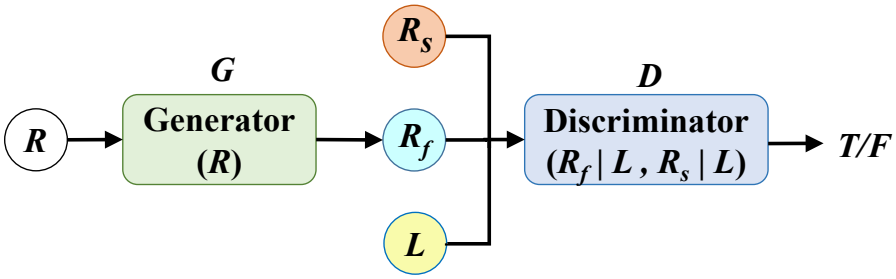
The main purpose of our work is to guarantee DOA estimation of legitimate signals withstanding the presence of a malicious jammer. To achieve our goal, it is fundamental to separate legitimate signals from illegitimate ones. To this aim, we leverage a Generative Adversarial Network (GAN) [57], wherein a generator strives to deceive a discriminator, the latter's goal being to detect whether the input signal comes or not from a jamming-free environment. The main intuition is that, at convergence, the generator will learn a filtering approach able to remove the jamming signal component from the input correlation matrix. Thus, we leverage a GAN [57] to compete the game between the generator and discriminator, fake signal from generator's output aim to fool the discriminator, which refines its ability to identify legitimate and fake signals.

As depicted in Figure 3.3a, the GAN framework includes two concurrently trained neural networks: a generative network G, and a discriminative network D. Given the autocorrelation matrix R, G endeavors to generate a synthetic instance $R_f = G(R)$ that emulates the characteristics of the legitimate correlation matrix R_s. The primary aim of training G is to refine the generative model such that the distribution of generated data closely approximates the distribution of real data. To this aim, we utilize a set of authentic samples ${R_s}^1, {R_s}^2, \cdots, {R_s}^m$. Concurrently, we train D to distinguish between samples drawn from the model-generated distribution P_{R_f} and those from the authentic data distribution P_{R_s}. This adversarial interplay incrementally enhances the verisimilitude of the samples fabricated by G, progressively narrowing the fidelity gap between synthetic and authentic samples.

Traditional GAN lack mechanisms for directing the generative process towards specific outcomes, often resulting in stochastic and unguided outputs. They are deficient in mechanisms that allow for the directed manipulation of the generative process towards producing outputs with specified characteristics or conditions, culminating in outputs that are somewhat stochastic and unguided. To mitigate this limitation, we only integrate additional conditional labels L into the generative framework of D. This approach diverges from conventional CGAN techniques [127], where conditional information is supplied to both the generator and discriminator. Nevertheless, our objective is to generate a representation of the real target without relying on conditional labels, thereby enhancing the versatility of our estimator. As illustrated in Figure 3.3b, we propose a CGAN with an unsupervised generator and a supervised discriminator tailored to our classification problem.



(A) GAN configuration.



(B) CGAN configuration.

FIGURE 3.3: Comparison of GAN and CGAN configurations.

3.6.2 Data Preprocessing

For our training set, we obtain samples from the system model in Figure 3.1. In this process, the values for the DOA are uniformly distributed across the entire field of view $[-60°, 60°]$. This means that the DOA entries are sampled from a range that covers all possible directions from which signals could arrive, ensuring a comprehensive representation of scenarios in our training data. **Complex decomposition:**

Received signals $x(t)$ and their correlation functions R_x preserve phase and amplitude information in the complex domain. By complex decomposition, we extract real-time domain signals from complex domain signals, thereby converting data into a format more amenable to intuitive analysis. Thus, we feed G with real-valued vectors, which can be written as

$$R' = [\mathrm{Re}\{\mathrm{vec}\{U \odot R_x\}\}, \mathrm{Im}\{\mathrm{vec}\{U \odot R_x\}\}]^T. \tag{3.11}$$

where $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ refer to the real and imaginary parts of a complex number. vec refers to vectorization of matrix and $\odot$ denotes the Hadamard product. U is an $N \times N$ upper triangular matrix whose elements can be defined as

$$u_{ij} = \begin{cases} 1 & \text{if } j > i, \quad i, j = 1, 2, \cdots, N, \\ 0 & \text{otherwise.} \end{cases} \tag{3.12}$$

Vector normalization: In general, real targets and interference have higher energy compared to ambient white Gaussian noise. Hence, the received signal frequently demonstrates baseline drift with respect to this noise. Vector normalization effectively mitigates such drift, thereby enhancing the precision of feature extraction and analysis. This technique particularly ensures that the range of signal values does not disproportionately affect the learning process of the model, resulting in accelerated convergence and enhanced accuracy. The normalization to unit vector for feature R' can be described mathematically as

$$R = \frac{R'}{\|R'\|}, \tag{3.13}$$

where $\|\cdot\|$ denotes the vector's L2-norm.

Data labeling: We create conditional labels L to guide D in categorizing particular directional samples. First, we build One-hot encoding L^1 to indicate the active directions from which signals are received by the sensor array. The dimensionality of the one-hot encoded vector is directly equal to the number of directions discretized within the DOA grid. Significantly, a value of 1 in a particular index within the vector indicates the presence of a signal coming from the corresponding direction, whereas a value of 0 signifies the non-presence of signals from those directions. Given the purpose of generating the clean autocorrelation matrix starting from the jammed signal one, we set our label $L = R_s$. This tag strategy is implemented to guide the output of G towards the removal of the jamming signal component while preserving the structure of the legitimate signal autocorrelation. This regulation indirectly guides the output of G, steering it from arbitrary data production toward generating data that preserves the original orientation. This method ensures that the synthesized data not only mimics the content of the target data but also adheres to specific spatial attributes. This regulation exerts an indirect influence on the output of G, redirecting it from the arbitrary production of target data towards the generation of data that maintains the original orientation. By adopting this approach, it is ensured that the synthesized data not only emulates the content of the real target data but also conforms to specific spatial attributes.

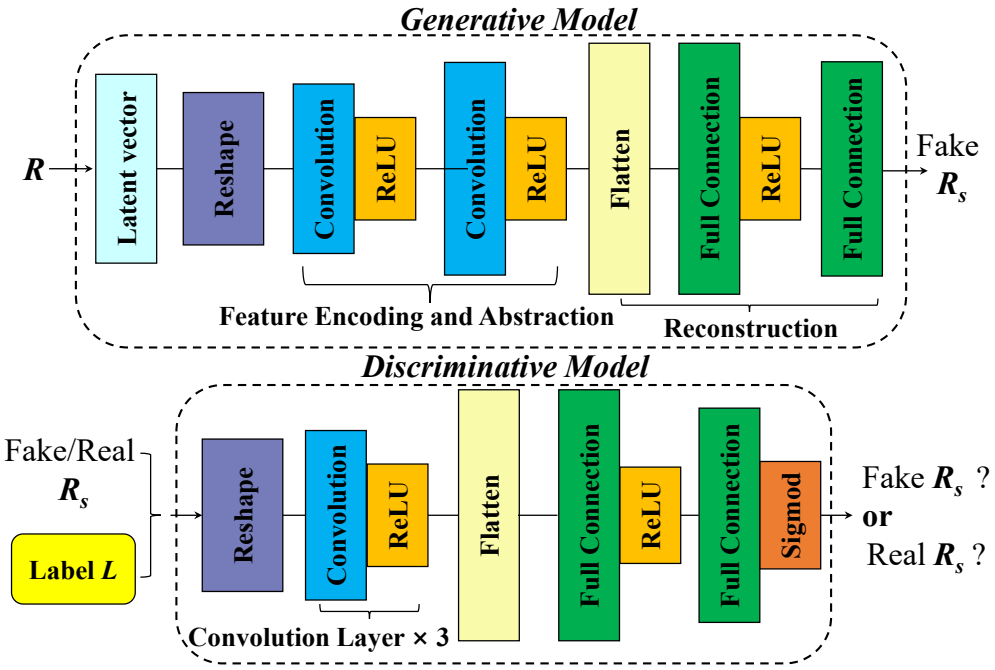


FIGURE 3.4: The schema of CGAN.

3.6.3 Adversarial Model

In the realm of computer vision, cutting-edge super-resolution and deblurring techniques increasingly leverage Convolutional Neural Network (CNN) for direct mapping from low-resolution inputs to their high-fidelity targets. Drawing inspiration from these pioneering studies, we exploit a combination of CNN layer and the Dense layer as

We exploit the generative and discriminative model in DOA-CGAN. To actualize the mapping function, we introduce an interference sensing signal R as the input. Subsequently, the generative component of the model produces an enhanced fake sensing signal, represented by $R_f = G(R)$, meticulously engineered to closely replicate the attributes of the genuine target signal. Figure 3.4 depicts the collaborative dynamics between G and D within a CGAN. The G is composed of a hierarchical architecture, integrating two convolutional layers followed by two dense layers. This configuration encapsulates distinct operational phases: *Feature Encoding and Abstraction*, and *Reconstruction*.

1) *Feature Encoding and Abstraction:* We reshape the latent tensor R into a matrix to make it compatible with the input requirements of convolutional layers. The first processing layer of G is a convolutional layer, tasked with encoding the sensing signal through various filters. Consequently, the initial convolutional segment of the model is represented as G_1

$$G_1(R) = \max(0, W_1 * R + b_1), \tag{3.14}$$

where W_1 and b_1 represent the convolutional filters and their corresponding biases tensor, respectively. The $\max(0, \cdot)$ is indicative of the Rectified Linear Unit (ReLU) activation function applied to the filter outputs. Each filter possesses a spatial dimensionality of $f_1 \times f_1$, with a stride α_1 and a total count of β_1 filters encapsulated within W_1. For this implementation, we specify $f_1 = 3$, $\alpha_1 = 1$ and $\beta_1 = 32$. Additionally, b_1 is delineated as an β_1-dimensional vector, where each element correlates

with a distinct filter. Consequently, the output of G_1 comprises $beta_1$ feature maps, each corresponding to one of the filters.

2)*High-Dimensional Feature Transformation:* In the first phase, each matrix channel is represented by a β_1 dimensional vector, where each vector element reflects the channel's response to a corresponding filter. Subsequently, the second convolutional layer succeeding G_1 transforms each β_1-dimensional tensor into a β_2-dimension. This transformation is achieved through the application of β_2 filters, each with a spatial dimension of $f_2 \times f_2$ and a stride of α_2. We formalize the operation as for representing each matrix channel, whose element denotes the channel's response on the corresponding filter. Then, the nonlinear mapping layer followed by G_1 is to map each n_1-dimensional vector to an n_2-dimensional one, which is equivalent to using n_2 filters with spatial size of 3×3 and stride $s_2 = 1$ on the generated feature maps of G_1. The operation of nonlinear mapping, denoted by G_2, can be formalized by

$$G_2(R) = \max(0, W_2 * G_1(R) + b_2), \tag{3.15}$$

where W_2 is the convolution filter and b_2 is the bias vector. The output of β_2-dimensional vector is pivotal for the subsequent approximation phase, serving as the basis for generating a high-fidelity signal. The output of an n_2- dimensional vector in this layer can be conceptually regarded as a representation of a high-quality sensing signal patch and used in the reconstruction stage.

2*) Reconstruction:* To guarantee that the output of our generative model precisely matches the target, we preprocess the input by tensor flattening, overlaying several neighboring patches. The architecture of the reconstruction layer, instrumental in refining the generative output to mirror the target dimensions accurately, can be expressed as

$$G(R) = W_4 * \max(0, W_3 * flatten(G_2(R)) + b_3) + b_4, \tag{3.16}$$

where W_3 and W_4 correspond to the weight vector with size $\beta_3 \times 1$ and $\beta_4 \times 1$, and $flatten(\cdot)$ refers to the process of transforming a multi-dimensional vector into a one-dimensional vector. The vectors b_3 and b_4 are biases.

As depicted in Figure 3.4, D employs a structure similar to G for feature analysis of input samples, albeit with differing configurations in the convolutional layers, specifically in terms of filter sizes, strides, and channel numbers, designated as f_1', f_2', f_3', α_1', α_2', α_3', β_1', β_2', β_3', respectively. Two fully connected layers are sequentially integrated to assess the importance of these extracted features for classification purposes, with the neuron counts in these layers labeled as β_4', β_5'. Here, we use the label L and the generated results R_s as the inputs. Then, the D will extract the inputs and process the hidden vectors. Finally, we get the confidence of each generated result by using the sigmod function. The architecture terminates with a sigmoid function that normalizes the output, facilitating the computation of loss in the discriminator model. For Discriminator D, we employ a similar structure of G to analyze the features of input samples, but the filter sizes, stride, and numbers of these three convolutional layers are different, denoted by f_1', f_2', f_3', s_1', s_2', s_3', n_1', n_2', n_3', respectively. After that, two fully connected layers are hierarchically connected to determine the weights of these extracted structural features for the classification task. The number of neurons within these two fully connected layers are denoted n_4' and n_5', respectively. Finally, a sigmoid function of conditional GAN or a softmax function of one-hot GAN are connected to the tail of model D, to normalize the network output and then to be used to calculate the loss of the discriminator model. Within the range

$(0,1)$ of and obtain the probability of judging an input sample from model D or R_s. Here, the discriminator model in Figure 3.4 is a standard conditional GAN model that uses the label L and the generated results R_s as the inputs.

3.6.4 Optimization of Objective Function

Consider a conditional label L to mark the sample data category. We need to divide it into a supervised D and an unsupervised G. For the conditional D, it needs to be designed as a true-or-false binary-classification network and often outputs its discriminative probability through the *sigmod* layer

$$P_d(x|L) = \frac{1}{1+e^{-x}} \tag{3.17}$$

The model's training process aims to optimize the objective function. For the supervised discriminator, we utilize Binary Cross Entropy to quantify the label sample loss $Loss_D$. This involves calculating the Kullback-Leibler divergence between the actual category distribution $P(x)$ and the predicted category distribution $Q(x)$, facilitating the precise classification of labeled authentic samples. Accordingly, the discriminator D operates as a conventional binary classifier, with its loss function being formally expressed as follows that is, calculate the Kullback-Leibler divergence of the true category distribution $P(x)$ and the predicted category distribution $Q(x)$, so as to ensure that the label the real sample is judged correctly. The D at this time is actually a standard Binary Classifier, and its loss function is expressed as According to CGAN, G aims to produce data indistinguishable from real data, while D aims to accurately classify real data from generated data. The objective function for a basic CGAN setup is represented as

$$\begin{aligned} &\min_G \max_D V(D,G) \\ &= \mathbb{E}\left[\log D(R_s|L)\right] + \mathbb{E}\left[\log(1 - D(G(R)|L))\right]. \end{aligned} \tag{3.18}$$

where $\mathbb{E}\left[\log D(R_s|L)\right]$ over real signals R_s aims to maximize the probability that D correctly identifies real signals as real. Instead, $\mathbb{E}\left[\log(1 - D(G(R)|L))\right]$ over interfered inputs R aims to maximize the probability that D correctly identifies signals generated by G as fake. Notably, only D takes conditional L into account while G has no additional information. G's success without direct conditional inputs suggests an effective indirect learning strategy through D's feedback. This approach yields a versatile signal generation conducive to accurate DOA estimation across various scenarios. The first term represents the expected log probability that D correctly identifies real data R_s. And the second term correctly identifies generated data $G(R)$ as fake. where $D(R_s|L)$ and $D(G(R)|L))$ are the discriminator's probability estimates for real data R_s and generated date being real, both conditioned on L. Logarithmic terms measure the success of the discriminator in classifying real and generated data correctly.

The Generator aims to produce data that the discriminator will classify as real. The generator loss can be generally represented as

$$L_G = -\mathbb{E}\left[\log D(G(R)|L)\right] \tag{3.19}$$

By minimizing this loss, the generator improves its ability to create data that looks real. The objective function of a two-player minimax game would be as

3.6.5 DOA-CGAN Framework

We present our proposed DOA-CGAN framework as shown in Figure 3.5. In the training phase, we use CGAN network to learn how to evaporate the jamming component. Then in the testing phase, we leverage G to retrieve the covariance matrix of the legitimate signal for DOA estimation.

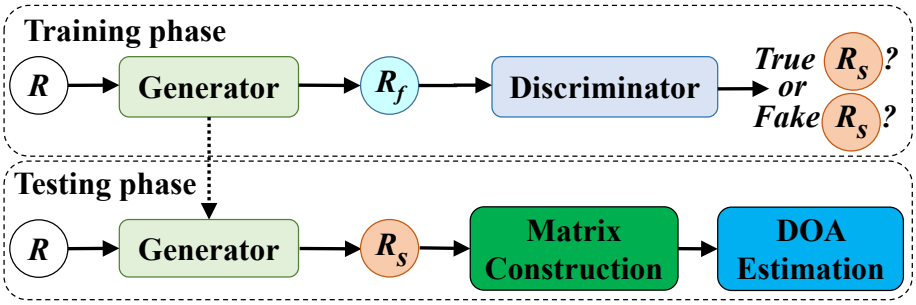


FIGURE 3.5: The configuration of DOA-CGAN.

3.7 Numerical Results and Discussion

In this section, we present the simulation results to validate the effectiveness of the proposed DOA-CGAN-based MUSIC estimation method [79, 149]. We utilize Python 3.9.12 with Pytorch 2.0.0+cu117, and NVIDIA GeForce RTX 3060 for implementing and training the GAN architectures. Table 3.1 and Table 3.2 provides the parameters selected for DOA-CGAN training. Furthermore, we conduct the DOA jamming model and data analysis on Matlab 2021A. We consider a Uniform Linear Array (ULA) with $N = 8$ and $d = 0.5\lambda$. Section 3.7.1 gives the DOA estimation Root Mean Square Error (RMSE) versus varied machine learning models.

TABLE 3.1: Structural Settings of DOA-CGAN (Generative Model G).

Layers	Parameters	Output Shapes
Input	—	$1 \times 7 \times 8$
Conv-1 + ReLU	$f_1 = 3, \alpha_1 = 1, \beta_1 = 32$	$32 \times 7 \times 8$
Conv-2 + ReLU	$f_1 = 3, \alpha_1 = 1, \beta_1 = 64$	$64 \times 7 \times 8$
Flatten	—	1×3584
FC-1 + ReLU	$\beta_3 = 128$	1×128
FC-2	$\beta_4 = 56$	1×56

3.7.1 Estimation Performance with Several Neural Models

We consider a legitimate signal and a jamming signal impinging on a radar. The Jamming signal originates from $[-30°, 30°]$, the authentic signal is randomly located at $\{-28°, -22°, -17°, -3°, 7°, 13°, 18°, 23°\}$ with random distribution sampling of $\pm 0.5°$. And the snapshot number is 50, Signal-to-Noise Ratio (SNR) and Jamming-to-Noise Ratio (JNR) are 10dB and 15dB, respectively. Moreover, $\text{JNR} = 20\log_{10}(\frac{A_j}{A_n})$, where A_j and A_j are the amplitude of jamming and noise, respectively. Finally, a total of 520000 measurement vectors are collected in the sample training dataset, and 52000 sample for the test dataset. The experiment is based on the Adam optimizer

TABLE 3.2: Structural Settings of DOA-CGAN (Discriminative Model D).

Layers	Parameters	Output Shapes
Input	—	$2 \times 7 \times 8$
Conv-1 + ReLU	$f_1' = 3, \alpha_1' = 1, \beta_1' = 32$	$32 \times 7 \times 8$
Conv-2 + ReLU	$f_2' = 3, \alpha_2' = 1, \beta_2' = 64$	$64 \times 7 \times 8$
Conv-3 + ReLU	$f_3' = 3, \alpha_3' = 1, \beta_3' = 32$	$32 \times 7 \times 8$
Flatten	—	1×1792
FC-1 + ReLU	$\beta_4' = 128$	1×128
FC-2 + Sigmod	$\beta_5' = 1$	1×1

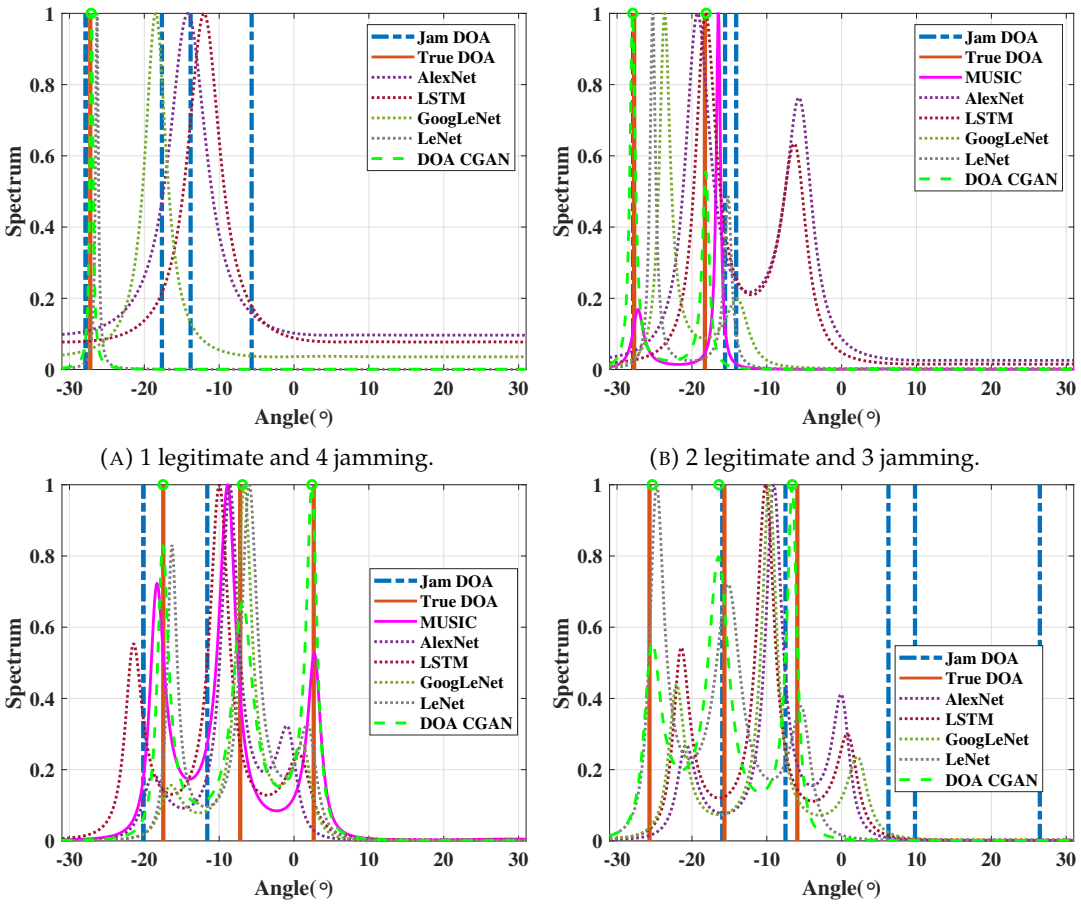
and Binary Cross-Entropy Loss, and the mini-batch size is 500 for 10 training epochs. Moreover, the initial learning rate is 2×10^{-4}. We evaluate the training performance of several state-of-the-art neural network architectures, including AlexNet, LSTM, GoogLeNet, ResNet, and LeNet [5, 13]. The comparative analysis focused on the RMSE RMSE $= \sqrt{\frac{1}{PQ}\sum_{p=1}^{P}\sum_{q=1}^{Q}(\hat{\theta}_{p,q} - \theta_{p,q})^2}$, where $\hat{\theta}_{p,q}$ are the estimates of $\theta_{p,q}$. P is the number of sources, and $Q = 500$ is the number of Monte Carlo experiments. We report the results for each model in Table 3.3. Our results maximum RMSE is 0.1366°, with the minimum not exceeding the Cramer-Rao Bound, indicating that the DOA-CGAN model we proposed exhibits superior effectiveness, particularly in scenarios involving DOA estimation under conditions of counter-interference. In the bottom two rows of Table 3.3, the running time of the proposed model is appropriate and acceptable compared to other models.

TABLE 3.3: The DOA estimation RMSE of different models and running time.

	AlexNet	LSTM	GoogLeNet	ResNet	LeNet	**DOA-CGAN**	CRB
-28°	0.3336°	0.2878°	0.2359°	0.2585°	0.1974°	**0.1366**$^\circ$	0.1014°
-22°	0.3013°	0.3401°	0.2709°	0.1989°	0.1973°	**0.1175**$^\circ$	0.0966°
-17°	0.4004°	0.2991°	0.2756°	0.2491°	0.1391°	**0.1261**$^\circ$	0.0936°
-3°	0.2864°	0.2785°	0.2661°	0.2863°	0.1973°	**0.1111**$^\circ$	0.0897°
7°	0.3266°	0.2659°	0.2595°	0.2672°	0.1769°	**0.1080**$^\circ$	0.0902°
13°	0.4677°	0.3142°	0.2363°	0.2274°	0.1560°	**0.1121**$^\circ$	0.0919°
18°	0.3186°	0.3294°	0.2467°	0.2018°	0.1749°	**0.1249**$^\circ$	0.0942°
23°	0.4779°	0.2953°	0.2363°	0.2440°	0.1504°	**0.1130**$^\circ$	0.0973°
Train (s)	1055.50	36.88	221.01	140.17	34.55	106.83	—
Test (s)	2.53	0.21	0.56	0.40	0.16	0.22	—

3.7.2 DOA Estimates

We now consider a scenario featuring a variable number of jamming sources. We fix the number of snapshots to 100, while the SNR of legitimate and jamming signals are maintained at 10dB and 15dB, respectively. We have K_1 legitimate signal and K_2 jamming signal drawn randomly from $[-30^\circ, 30^\circ]$, where $1 \leq K_1 \leq 3, 1 \leq K_2 \leq 5$. Therefore, there are 3^5 combinations, each accompanied by 12000 training sets and 1200 test sets. Figure 3.6 shows spectra for various numbers of legitimate sources and jammers. the results of DOA estimation under the influence of 1, 3, 4, and 5 jamming sources. Note that the MUSIC method cannot estimate more signals than $N - 1$. Thus, the MUSIC spectrum is not included in Figure3.6d. The outcomes demonstrate that our proposed model retains its ability to accurately estimate the



(A) 1 legitimate and 4 jamming.

(B) 2 legitimate and 3 jamming.

(C) 3 legitimate and 2 jamming.

(D) 3 legitimate and 5 jamming.

FIGURE 3.6: Spatial spectrum of DOA-CGAN.

DOA of the authentic signal, withstanding the complexity introduced by multiple jamming attacks.

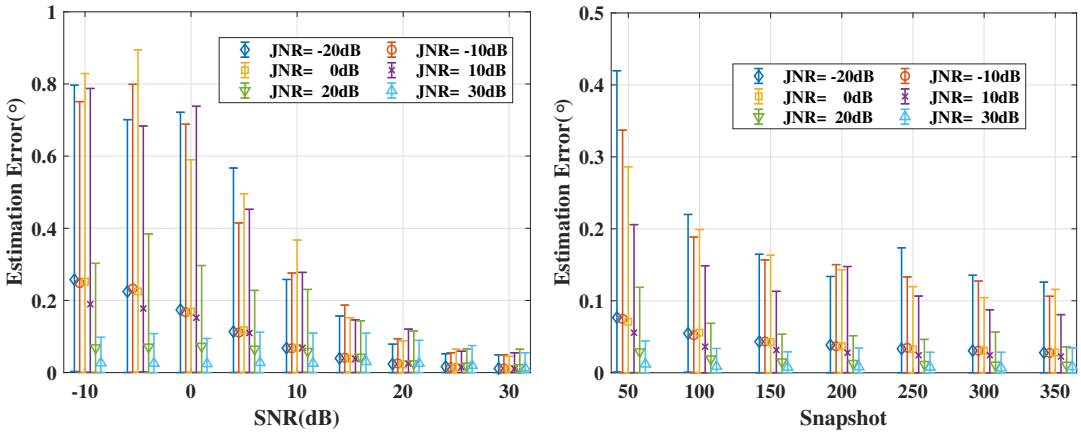
Such resilience to multifaceted interference significantly illustrates the model's suitability for deployment in signal-rich environments where discerning genuine signals from a milieu of contaminants is paramount.

3.7.3 Estimation Error versus SNR and Snapshot

We further investigate the DOA estimation performance versus SNR varied from -10dB to 30dB with step 5dB using 50 independent snapshots. Here, we set a jammer location randomly in $[-20.5^\circ : -19.5^\circ]$, whereas a legitimate signal from $[9.5^\circ : 10.5^\circ]$. For the jamming signal, we explore the effects of different JNR varied from -20dB to 30dB on DOA estimates. Figure 3.7a displays the DOA estimation error using error bars, with segments representing maximum and minimum error values. A node at the midpoint denotes the average error value. We can observe that the estimation error decreases as the SNR increases. Enhancements of JNR improve the interference elimination capabilities of our DOA-CGAN, as discussed in [177, 12, 11]. However, when the JNR falls below 0dB, the estimation error remains consistent, indicating a boundary of error under conditions of low JNR.

Finally, we conduct a comparative analysis of the impact of varying JNR on the number of snapshots utilized. In this analysis, we vary the number of snapshots from 50 to 350, keeping the SNR of legitimate signals constant at 0 dB. Note that the

snapshot number of legitimate users and jammer are the same. and the SNR of the legitimate signals is maintained at a constant 0dB, while the JNR is incrementally adjusted from 8dB to 18dB in steps of 2dB. Similarly, as depicted in Figure 3.7b, the proposed DOA estimation using CGAN demonstrates a distinct advantage in accurately estimating sources in the presence of jamming. Note that the accuracy of DOA estimation improves with increasing numbers of snapshots and with higher JNR.



(A) Estimation error versus SNR. (B) Estimation error versus snapshot number.

FIGURE 3.7: The estimation error in terms of varied JNR.

3.8 Conclusion

Diverging from GAN-based augmentation techniques [184], we leverage generators to mitigate interfering signals in angle estimation. These are particularly well-suited for scenarios involving interference signals, where their inherent adversarial mechanism promotes a dynamic game between the generator and the discriminator within the network. The generator strives to create increasingly realistic signals to deceive the discriminator, while the discriminator works hard to distinguish between genuine signals and those that are interfered with. Through this process, DOA-CGAN effectively learns how to recognize and eliminate the effects of interference signals.

Chapter 4

Multi-Receiver RFFI (Triad-GAN)

Radio frequency fingerprint identification (RFFI) is a promising authentication method for IoT networks. Typically, the RFFI model is trained on collected signals from trusted receivers to identify device-specific features. However, when this model is deployed in multi-receiver scenarios, its performance often degrades significantly if it encounters previously unseen receivers. Motivated by this challenge, we propose **Triad-GAN**, a novel feature-level generative adversarial network (GAN) framework designed to achieve robust device authentication in a multi-receiver scenario. Unlike standard GAN-based approaches, our proposed architecture explicitly incorporates a feature extractor that interacts with a feature-level generator and discriminator within a unified triadic adversarial game. This unique design facilitates the extraction of receiver-invariant and highly discriminative features directly in the feature space, substantially improving model generalization across unseen receivers and enabling effective performance in diverse wireless protocol environments. We extensively evaluate Triad-GAN on multiple RFFI benchmarks, including publicly available datasets from long-range (LoRa) and Wi-Fi protocols. Experimental results demonstrate that our method significantly outperforms existing state-of-the-art approaches, achieving authentication accuracy improvements exceeding 15% on the RFFI LoRa dataset and 16% on the WiSig dataset. We further highlight the broader applicability of our feature-level adversarial approach as a promising method for secure and generalizable representation learning across various research domains.

4.1 Introduction

The proliferation of Internet of Things (IoT) technologies has fundamentally transformed modern life, embedding wireless devices into almost every aspect of our daily routines. As billions of interconnected IoT devices continue to be deployed, ensuring the security and privacy of wireless communications is becoming increasingly critical. Robust device authentication remains essential to maintain the integrity and trustworthiness of wireless networks [193]. While traditional authentication relies on cryptographic schemes, these methods are often vulnerable to key leakage, replay attacks, and protocol weaknesses, particularly in dynamic or resource-constrained IoT scenarios [73]. Consequently, there is a pressing need for authentication mechanisms that are robust and generalizable across diverse wireless environments.

Radio frequency fingerprint identification (RFFI) has recently emerged as a promising physical-layer solution for device authentication. It leverages subtle, hardware-induced imperfections that are present in the transmitted signals of each device [208]. These so-called “hardware fingerprints” are inherently unique, even

among devices of the same model and manufacturer, and manifest as subtle differences in waveform, amplitude, phase, and spectral characteristics. Since such fingerprints are difficult to clone or forge, RFFI is widely regarded as tamper-resistant and protocol-agnostic. This approach has attracted increasing attention in both academia and industry across various wireless protocols [107]. Historically, RFFI systems relied on handcrafted features and conventional classifiers, such as support vector machine (SVM) [206] and k-nearest neighbors (KNN) [29]. With the rise of deep learning, the field has shifted to end-to-end models, such as convolutional neural networks (CNNs) [154, 96], recurrent neural networks (RNNs) [85], and transformers [69], which automatically extract discriminative features from collected signals. Typically, a deep learning-based RFFI system consists of signal acquisition, preprocessing, feature extraction, and device authentication, where signals from wireless devices are processed by neural networks for automated and robust device identification [152].

Despite these advances, deploying RFFI in multi-receiver scenarios remains a significant challenge. The received radio frequency (RF) signal characteristics are influenced by both the transmitter and the receiver hardware, which can lead to pronounced domain shifts when training and testing data are collected from different receivers. In this context, domain shift refers to changes in signal characteristics caused by variations in receiver hardware, resulting in mismatched data distributions between the training and deployment phases. This so-called "receiver domain shift" can significantly degrade the performance of models trained with data from only a single receiver, creating a substantial obstacle to the large-scale application of RFFI. Solving this problem requires new methods capable of disentangling transmitter-specific features from receiver-dependent distortions, thereby enabling reliable device identification across diverse hardware platforms and operational conditions.

Generative Adversarial Networks (GANs) have shown considerable potential for tackling domain adaptation and generalization challenges across a range of fields, including image synthesis, style transfer, and data augmentation. In wireless signal processing, GAN-based methods have demonstrated success in generating diverse, high-fidelity samples and in supporting representation learning for multi-domain scenarios [200, 56, 68]. By adversarially aligning feature distributions across different domains, GANs provide a principled way to extract domain-invariant and highly discriminative features. While most existing RFFI studies leverage adversarial training inspired by GANs [213, 47], they seldom utilize the generative capabilities of GANs to directly tackle receiver domain shift.

Drawing inspiration from these advances, this work introduces **Triad-GAN**, a novel feature-level generative adversarial network that is specifically designed to address the receiver domain shift problem in multi-receiver RFFI. Our model incorporates a triadic adversarial structure, consisting of a feature extractor, a generator, and a discriminator, to promote the learning of receiver-invariant and transmitter-discriminative features. This approach enables robust device authentication across a wide variety of receivers and communication protocols. The primary contributions of this work are as follows:

- We propose Triad-GAN, a feature-level generative adversarial network framework to extract receiver-invariant representations and to achieve robust transmitter classification in RFFI systems. Notably, the Triad-GAN is capable of being directly applied to entirely unseen receivers without requiring any data for fine-tuning.

- The effectiveness of the proposed model is validated through extensive experiments on long-range (LoRa) datasets in multi-receiver settings. Compared with existing methods, Triad-GAN achieves up to 35% higher accuracy under receiver drift and over 15% improvement in receiver change scenarios.
- Additional experiments on large-scale Wi-Fi datasets further confirm the generality and applicability of Triad-GAN across different wireless protocols and receiver configurations. In these settings, Triad-GAN achieves up to 84.4% accuracy, outperforming prior methods by 16.3% to 40.8% in receiver change classification tasks.

The remainder of this chapter is organized as follows. Section 4.2 reviews related work on RFFI, adversarial representation learning, and domain adaptation. Section 4.3 formalizes the multi-receiver setting and states the problem. Section 4.4 presents the proposed *Triad-GAN* framework, detailing the system architecture and workflow, followed by the optimization objectives and network designs. Section 4.5 reports extensive experiments on LoRa and WiSig, including receiver drift and receiver change evaluations, ablations on receiver diversity, and t-SNE analyses of learned features. Finally, Section 4.6 discusses implications and limitations.

4.2 Related Work

In this section, we describe the background in the two research areas: (a) the development and practical challenges of RFFI, with a particular focus on multi-receiver scenarios; and (b) advances in GANs and domain adaptation, which offer strong generalization capabilities and serve as the foundation for our method.

4.2.1 RFFI Methods and Applications

Early approaches to RFFI relied on handcrafted features and classical machine learning methods, such as SVM and KNN, to distinguish wireless devices based on transmitter imperfections [140, 163]. While effective in controlled environments, these techniques struggled with generalization due to their sensitivity to channel and receiver variations [209, 7]. The introduction of deep learning shifted RFFI towards automatic feature extraction, yielding improved transmitter identification accuracy and robustness [123, 148, 136]. However, most early deep models focused on single-receiver or channel-invariant scenarios and performed poorly under receiver domain shifts. To address these challenges, recent works proposed receiver-agnostic models [25, 153, 7, 122, 155], as well as adversarial and generative approaches [56, 213, 69, 14, 47]. Despite these advancements, robust and scalable receiver-invariant RFFI remains an open problem.

4.2.2 Generative Adversarial Networks Applications

GANs [57] introduce an adversarial training framework in which a generator and a discriminator compete to produce data that is indistinguishable from real samples. Since then, GANs have become a central tool in generative modeling, achieving impressive results in many areas. Key variants, including conditional GANs (cGAN) [127], CycleGAN [220], and Wasserstein GANs (WGAN) [10], have further improved their stability and versatility. More advanced architectures, such as Triple-GAN [101] and FaceID-GAN [157], incorporate auxiliary classifiers to jointly

optimize both generation quality and discriminative performance. Beyond generation, adversarial learning has inspired a range of methods for representation learning and domain adaptation. Notably, the Gradient Reversal Layer (GRL) in Domain-Adversarial Neural Networks (DANN) [51] and Adversarial Discriminative Domain Adaptation (ADDA) [173] have enabled models to learn domain-invariant features, significantly enhancing robustness under domain shift. Multi-domain adversarial frameworks such as StarGAN [38] further enable feature disentanglement and cross-domain transfer across multiple environments. In the context of RFFI, GANs have been effectively used for data augmentation to improve the performance of RFFI [221]. These advances highlight the versatility of GANs, not only for data generation but also for generalizable, domain-invariant representation learning in complex, real-world scenarios.

4.3 System Model and Problem Statement

This section introduces the overall system setting and formulates the transmitter identification problem in multi-receiver scenarios.

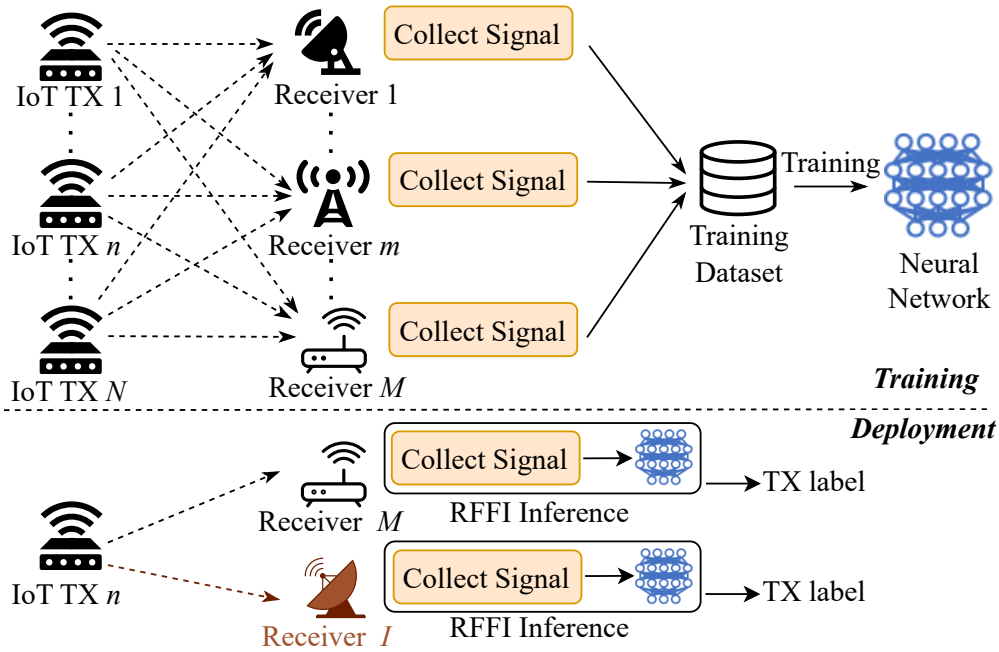


FIGURE 4.1: Multi-Receiver RFFI system.

4.3.1 System Model

A multi-receiver RFFI system comprises a wireless network with N IoT transmitters and M receivers, as illustrated in Figure 4.1. The system workflow consists of two primary stages: training and deployment.

During the *training* stage, RF signals are collected by the receivers from all participating transmitters under various channel and hardware conditions. These collected signals are then organized into a comprehensive dataset, which serves as input for

training a deep neural network-based classifier. The process is formalized by defining the received signal at time t as

$$y(t) = f_{\text{rx}}\left(h * f_{\text{tx}}(x(t))\right) + n(t), \tag{4.1}$$

where $x(t)$ is the original transmitted signal, $f_{\text{tx}}(\cdot)$ and $f_{\text{rx}}(\cdot)$ denote transmitter- and receiver-induced distortions, h is the channel impulse response, $n(t)$ is additive noise, and $*$ denotes the convolution operation. Each sample is annotated with its corresponding transmitter and receiver labels, forming the RFFI dataset as follows:

$$\mathcal{D}_{\text{train}} = \left\{(y_k, \mathbf{tx}_k, \mathbf{rx}_k)\right\}_{k=1}^{K_{\text{train}}}, \tag{4.2}$$

where y_k is the k-th received signal, $\mathbf{tx}_k \in \{1, 2, \ldots, N\}$ is the numerical identifier of the corresponding transmitter, and $\mathbf{rx}_k \in \{1, 2, \ldots, M\}$ is the numerical identifier of the corresponding receiver.

In the *deployment* stage, the trained neural network is used to authenticate new incoming signals. Each received signal is processed by the neural network to predict the transmitter identity, enabling device authentication in real-time. As shown in Figure 4.1, this multi-receiver RFFI system is expected to operate reliably even when deployed with a previously unseen receiver. After the training stage, where a neural network is learned from signals collected by Receivers 1 to M using a set of N transmitters, the trained model is deployed across multiple receivers for real-time RFFI inference. During the *deployment* phase, this trained neural network is implemented not only on the original receivers but also on new, previously unseen receivers, such as Receiver I, and is used to perform RFFI inference on locally collected signals. Throughout this work, all transmitters present in the deployment phase are included in the training dataset. The final authentication decision is made by verifying whether the transmitter label is in the training dataset.

4.3.2 Problem Statement

From a signal modeling perspective, the received signal $y(t)$ is affected by both receiver-specific distortions f_{rx} and the channel impulse response h. Although channel effects can impact signal propagation, their influence can be mitigated by collecting data in controlled environments. Therefore, this study focuses primarily on receiver-induced variability, which remains the principal challenge for achieving robust RFFI in practical multi-receiver deployments.

In multi-receiver scenarios, receiver-dependent factors such as hardware nonlinearities, front-end imperfections, and analog-digital conversion artifacts introduce complex variations in the observed signals. These effects are especially pronounced in large-scale IoT networks, distributed sensor systems, and cooperative wireless monitoring, where a single transmission may be simultaneously captured by multiple heterogeneous receivers [27]. Moreover, two dynamic phenomena exacerbate the problem:

- **Receiver drift:** Gradual changes in receiver characteristics over time, such as temperature effects or hardware aging, can lead to temporal shifts in the received signal distribution even for the same physical receiver.
- **Receiver change:** The introduction of entirely new receiver hardware in the deployment phase, which was not present during training, results in a significant domain shift because the receiver-induced distortions can differ substantially from those observed in the training set.

These factors collectively present a significant challenge for RFFI systems: the need to develop robust authentication methods that can effectively disentangle transmitter-intrinsic features from both channel- and receiver-induced artifacts. Addressing receiver-induced variability—including both drift and change—remains critical for achieving practical, scalable, and reliable authentication in modern multi-receiver wireless environments.

4.4 Proposed Method

This section details the overall design of the proposed Triad-GAN framework for multi-receiver RFFI. We first introduce the system architecture and workflow, then formalize the associated loss functions of Triad-GAN.

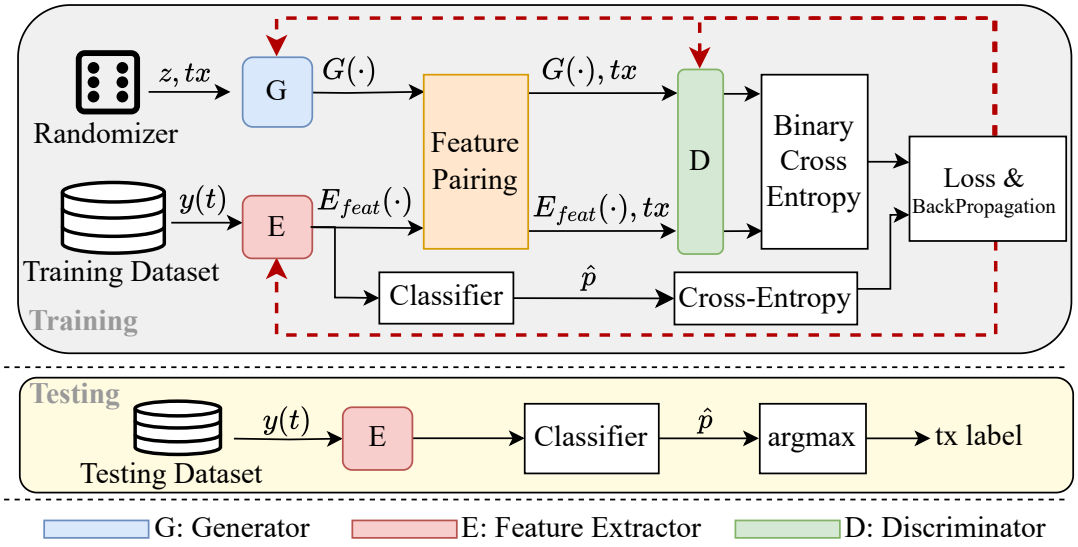


FIGURE 4.2: Overall workflow of the Triad-GAN framework.

4.4.1 System Architecture

Multi-receiver RFFI requires trained models to extract receiver-invariant features, and GANs have demonstrated strong capabilities for learning domain-invariant representations. Motivated by this, we propose **Triad-GAN**, which builds upon and optimizes the traditional GAN structure to address the unique demands of multi-receiver RFFI. The overall architecture of the proposed adversarial feature learning framework for transmitter authentication in multi-receiver scenarios is illustrated in Figure 4.2. The training process involves two primary data sources: (a) a random noise generator produces latent vectors z, which, together with transmitter labels, are input to the generator G to synthesize pseudo-feature vectors $G(z, \text{tx})$; and (b) an RF dataset provides observed signals $y(t)$, reflecting both transmitter and receiver effects.

During the *training* phase, the system uses two sources of data: (a) A **randomizer** that generates a latent noise vector z, which, together with transmitter labels tx, is provided to the generator G. The generator synthesizes pseudo-feature representations $G(z, \text{tx})$ in the feature space, emulating artificial transmitter-specific features; and (b) the **RF dataset** supplies observed signals $y(t)$, each reflecting both transmitter and receiver effects.

The **generator** $G(\cdot)$ plays a critical role in the adversarial training process. It receives as input a random latent noise vector z and a transmitter label tx, and outputs a synthesized feature vector $G(z, \text{tx})$ in the same feature space as the real feature vectors. The generator is trained to produce pseudo-feature vectors that are indistinguishable from real transmitter features, as assessed by the discriminator D.

The **feature extractor** $E(\cdot)$ processes each received signal $y(t)$ and outputs a real feature vector $E_{\text{feat}}(y(t))$ and a probability vector $\hat{p}$, where $\hat{p}$ denotes the predicted probabilities for each transmitter class. The feature vector $E_{\text{feat}}(y(t))$ is fed to the discriminator D for adversarial learning, while $\hat{p}$ is used to compute the transmitter classification loss with the ground-truth TX label.

The **feature pairing** module pairs both real features from E and synthetic features from G with their corresponding TX labels, yielding pairs $(E_{\text{feat}}(y(t)), \text{tx})$ and $(G(z, \text{tx}), \text{tx})$. The **discriminator** D then receives both $(E_{\text{feat}}(y(t)), \text{tx})$ and $(G(z, \text{tx}), \text{tx})$, conditioned on the TX label, and is trained to distinguish between real and generated feature vectors using a binary cross-entropy (BCE) loss. This adversarial process encourages the feature extractor to produce features that are invariant to receiver hardware and noise, as formalized in the signal model.

This **adversarial training framework** is explicitly constructed so that the feature extractor does not utilize receiver labels at any stage. Instead, it is guided to focus solely on learning transmitter-discriminative features, ensuring that the resulting representations are invariant to receiver-specific factors. During training, the generator synthesizes latent feature vectors conditioned solely on TX labels, while the feature extractor processes signals associated with diverse RX labels. Through adversarial optimization, the generator aims to produce feature distributions that are indistinguishable from those derived from real signals, thus confusing the discriminator. This process compels both the generator and feature extractor to align their latent feature spaces with respect to the TX label. As a result, the feature extractor learns to capture intrinsic transmitter signatures that are inherently robust to variations across different receivers.

During the *testing* phase, only the trained feature extractor E and classifier are employed. Given a new signal $y(t)$, which may be collected by either a known receiver or a previously unseen receiver in the multi-receiver scenario, the feature extractor outputs a probability vector $\hat{p}$ over all transmitter classes. The classifier then predicts the transmitter label based on $\hat{p}$.

4.4.2 Optimization Objectives

The objective of our proposed **Triad-GAN** is to facilitate the extraction of transmitter-discriminative and receiver-invariant features for robust authentication under multi-receiver scenarios. The loss functions are formulated as follows:

- **Transmitter Classification Loss:** the feature extractor $E(\cdot)$ processes received signal $y(t)$ and outputs a probability vector $\hat{p}$ over all N transmitter classes. The cross-entropy (CE) classification loss is defined as

$$\mathcal{L}_{\text{cls}}^{E} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{k=1}^{N} p_{i,k} \log\left(\hat{p}_{i,k}\right), \tag{4.3}$$

  where B is the batch size, N is the number of transmitter classes, $p_{i,k}$ is a one-hot ground-truth label for class k, and $\hat{p}_{i,k}$ denotes the predicted probability for class k of the i-th sample.

- **Generator Adversarial Loss:** the generator $G(\cdot)$ is trained to synthesizes pseudo-features. The generator adversarial loss, formulated using BCE, is

$$\mathcal{L}_{\text{adv}}^{G} = -\frac{1}{B}\sum_{i=1}^{B}\log D\left(G(z_i, tx_i), tx_i\right), \tag{4.4}$$

  where $D(\cdot, tx)$ denotes the discriminator's predicted probability that the input is a real feature conditioned on the tx label.

- **Discriminator Adversarial Loss:** The discriminator $D(\cdot)$ learns to distinguish real feature vectors $E_{\text{feat}}(y_i(t))$ from synthetic features $G(z_i, tx_i)$, both conditioned on TX labels. This objective is also formulated using the BCE loss:

$$\mathcal{L}_{\text{adv}}^{D} = -\frac{1}{B}\sum_{i=1}^{B}\Big[\log D\left(E_{\text{feat}}(y_i(t)), tx_i\right) + \log\left(1 - D\left(G(z_i, tx_i), tx_i\right)\right)\Big], \tag{4.5}$$

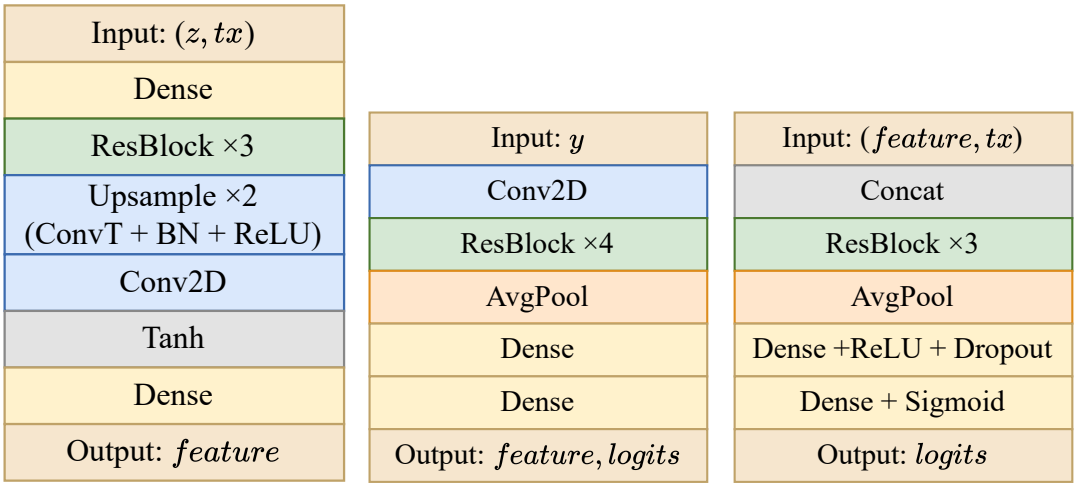
- **Overall Objective:** The total loss is a weighted sum of the above terms:

$$\mathcal{L}_{\text{total}} = \lambda_E \mathcal{L}_{\text{cls}}^{E} + \lambda_G \mathcal{L}_{\text{adv}}^{G} + \lambda_D \mathcal{L}_{\text{adv}}^{D}, \tag{4.6}$$

  where λ_E, λ_G, and λ_D are weighting hyperparameters.

We set λ_E, λ_G, and λ_D to 2.0, 0.5, and 0.5, respectively, to prioritize device authentication by emphasizing the optimization of the feature extractor. Jointly optimizing these losses allows Triad-GAN to extract robust transmitter signatures invariant to receiver diversity, thus improving authentication accuracy in practical multi-receiver scenarios.

4.4.3 Network Architecture



(A) Generator. (B) Feature Extractor. (C) Discriminator.

FIGURE 4.3: The network architecture of each component.

To realize the above optimization objectives, each component of Triad-GAN is implemented with a ResNet-based [75] architecture, drawing on its proven effectiveness in recent RFFI studies [155, 213, 214]. The detailed network structures of

the generator, feature extractor, and discriminator are shown in Figure 4.3. The network configurations are adapted to accommodate the input dimensionality and output class cardinality specific to each dataset, thereby ensuring that Triad-GAN is broadly applicable to both spectrogram-based (LoRa) and time-domain (WiSig) signal modalities.

4.5 Experiments and Evaluation

In this section, we describe the LoRa and WiSig datasets, detail the experimental setups, and comprehensively evaluate Triad-GAN across different experimental scenarios.

4.5.1 Datasets and Experiment Setup

LoRa Multi-Receiver Dataset

The LoRa dataset [155] comprises signals collected from 10 LoRa transmitters and 20 software-defined radio (SDR) receivers, specifically: RTL-1∼RTL-9, PLUTO-1, PLUTO-2, B200-1, B200-2, B200 mini-1, B200 mini-2, B210-1, B210-2, and N210-1∼N210-3. Additionally, repeated measurements were performed for RTL-6 and N210-1 over four separate days to support receiver drift analysis. The following evaluations, which are used in this section, were conducted:

- Train with the homogeneous scheme (RTL-1 to RTL-5), where the training receivers share similar hardware characteristics.
- Train with the heterogeneous scheme (RTL-1, PLUTO-1, B200-1, B200 mini-1, and B210-1), where the training receivers have different hardware characteristics.

Triad-GAN performance is compared with the baseline from [155], which we selected because it is a state-of-the-art solution for multi-receiver LoRa RFFI and closely aligns with the experimental conditions of our study.

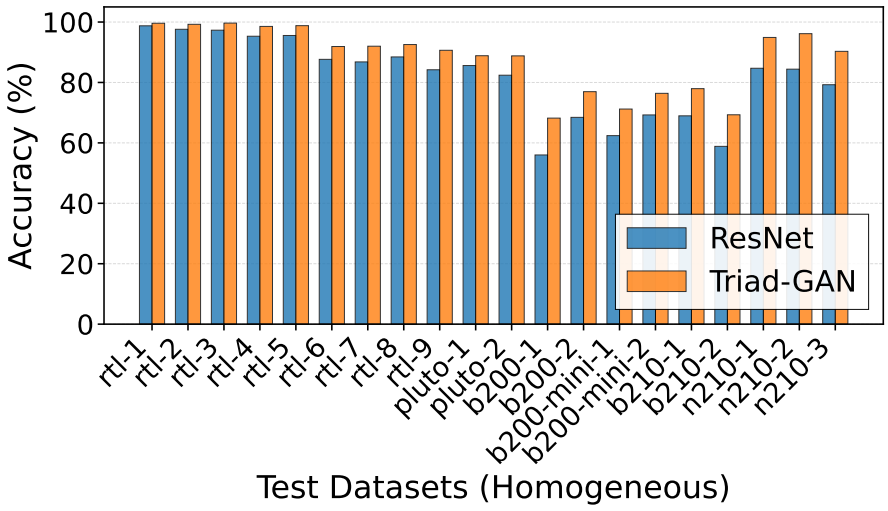
WiSig Wi-Fi Dataset

WiSig [71] is a large-scale, publicly available dataset for Wi-Fi RF fingerprinting, providing Wi-Fi RF signal datasets. Receiver locations are specified within a two-dimensional (2D) grid, identified by their horizontal and vertical coordinates. For example, a receiver positioned at coordinate $(1, 1)$ is denoted as Rx 1-1. In our work, all WiSig data are preprocessed using channel equalization as described in [213], given prior evidence that equalized signals provide superior classification performance [71].

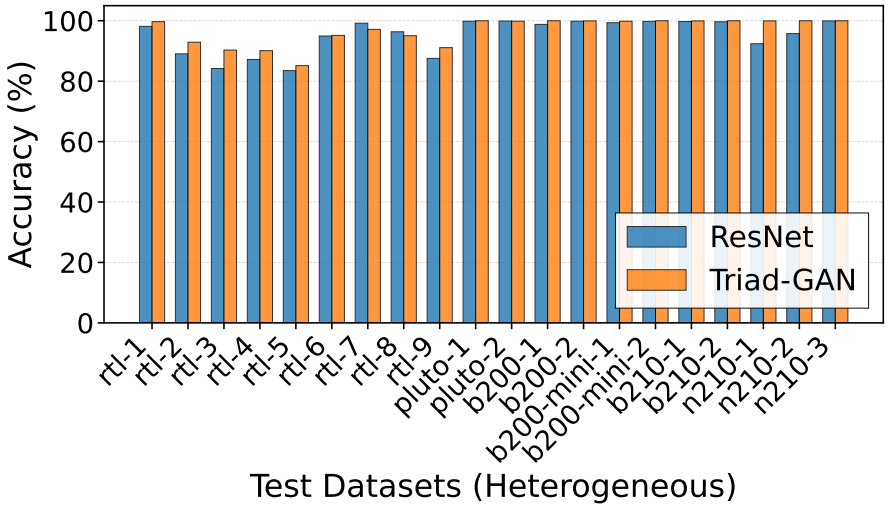
For experimental purposes, we used the WiSig subset tools [71] to construct a subset comprising 10 transmitters and 28 receivers, and also partitioned it by acquisition date.

4.5.2 Experimental Results and Analysis on LoRa

In this section, we evaluate the proposed Triad-GAN framework on both receiver drift and receiver change scenarios, as well as examining the impact of training schemes on feature distribution, as shown below.



(A) Classification accuracy (homogeneous).

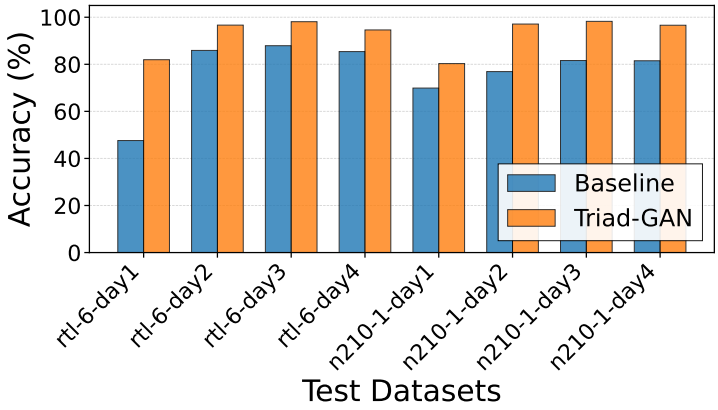


(B) Classification accuracy (heterogeneous).

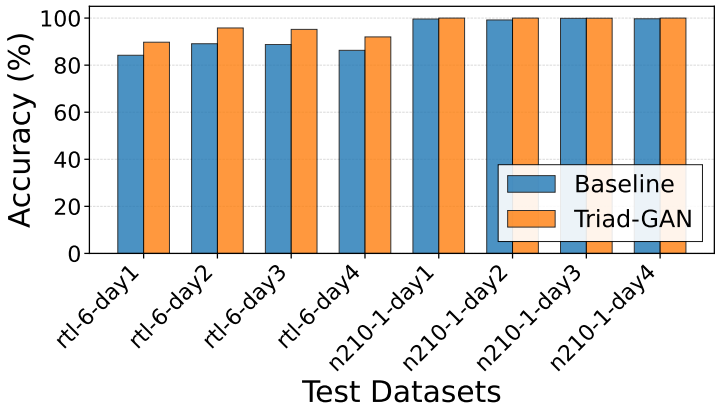
FIGURE 4.4: LoRa classification accuracy comparison.

As illustrated in Figure 4.4a, under the homogeneous training scheme, Triad-GAN achieves consistently higher accuracy than the ResNet baseline, with substantial gains exceeding 20% for receivers such as B200, B200-mini, and B210. These improvements confirm the advantage of adversarial feature learning when receiver diversity in the training data is limited. In contrast, Figure 4.4b shows the results for the heterogeneous scheme, where both methods perform more stably due to increased receiver diversity during training. Nevertheless, Triad-GAN maintains a consistent margin of improvement over ResNet, demonstrating its effectiveness in extracting transmitter-intrinsic features that generalize well to unseen receivers. Collectively, the results in Figure 4.4 highlight the robustness and scalability of Triad-GAN across different training strategies in multi-receiver RFFI.

To evaluate the robustness of the proposed method under receiver drift, we first examine its performance, as shown in Figure 4.5, and the test datasets comprise RTL-6 and N210-1 signals collected across four different days. We present the classification accuracy of the proposed Triad-GAN framework compared to the baseline



(A) Classification accuracy (homogeneous).

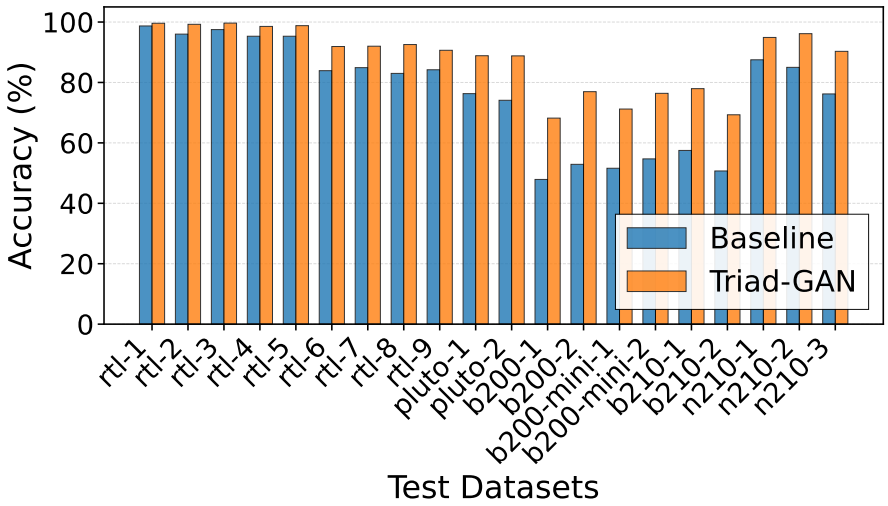


(B) Classification accuracy (heterogeneous).

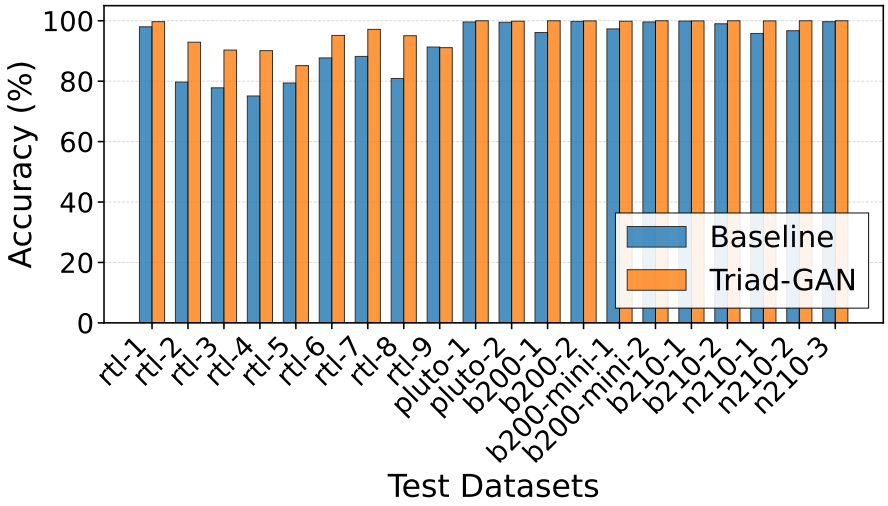
FIGURE 4.5: LoRa classification accuracy under receiver drift.

method under the same conditions, for both homogeneous and heterogeneous training schemes. In the homogeneous case (Figure 4.5a), Triad-GAN consistently outperforms the baseline across all test datasets, with particularly notable improvements of about 35% accuracy higher on devices RTL-6-day1. This demonstrates Triad-GAN's enhanced ability to extract transmitter-intrinsic features that are robust to temporal drifts in the receiver. In the heterogeneous case (Figure 4.5b), both methods achieve generally higher and more stable accuracies, but Triad-GAN maintains a slight advantage over the baseline across the entire set of test SDRs. The reduced performance gap is attributable to the fact that in highly diverse receiver environments, receiver-specific artifacts dominate, making the task inherently easier. Although this training strategy simplifies the task, it does so by amplifying receiver diversity and thus reducing the influence of subtle receiver-induced artifacts. Nonetheless, the consistent gains achieved by Triad-GAN highlight its effectiveness in mitigating receiver-induced domain shift and ensuring reliable authentication in practical multi-receiver deployments. While this training strategy simplifies the task by increasing receiver diversity and reducing the influence of subtle receiver-induced artifacts, the consistent gains of Triad-GAN nonetheless demonstrate its strong ability to mitigate receiver-induced domain shift and deliver reliable authentication in real-world multi-receiver scenarios.

To evaluate the robustness of Triad-GAN under receiver change, we evaluate its performance as shown in Figure 4.6, and the test datasets include data from all 20



(A) Classification accuracy (homogeneous).

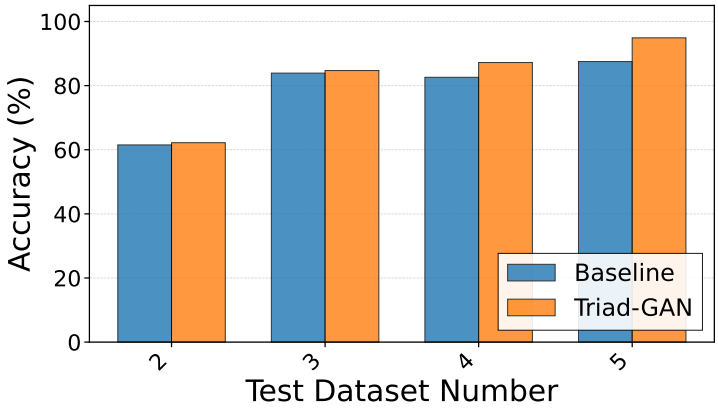


(B) Classification accuracy (heterogeneous).

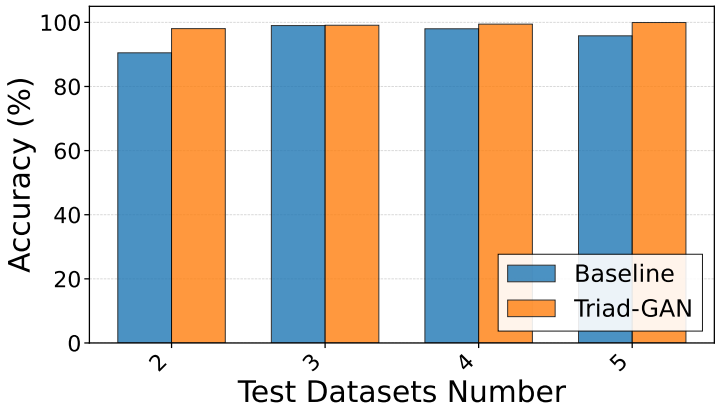
FIGURE 4.6: LoRa classification accuracy under receiver change.

receivers. The classification accuracy of the Triad-GAN is compared to the baseline method in these test datasets, including both homogeneous and heterogeneous training schemes. In the homogeneous case (Figure 4.6a), Triad-GAN consistently outperforms the baseline across all receiver datasets. The performance advantage is particularly pronounced for certain devices, where the baseline method suffers substantial drops in accuracy due to the limited hardware diversity. In contrast, Triad-GAN maintains higher and more stable results, achieving gains of over 20% in the B200, B200-mini, and B210 receiver datasets. In the heterogeneous scenario (Figure 4.6b), both methods achieve generally higher classification accuracies. However, Triad-GAN still exhibits clear superiority even in a heterogeneous scheme, specifically on the RTL-4 dataset, the Triad-GAN outperforms the baseline by more than 15% accuracy. This demonstrates the strong generalization capability of Triad-GAN in environments characterized by limited receiver diversity.

The classification results on the LoRa dataset under different receiver numbers are shown in 4.7. In the homogeneous scenario 4.7a, where training and testing are conducted on the same type of receivers, both the baseline and Triad-GAN improve



(A) Classification accuracy (homogeneous).

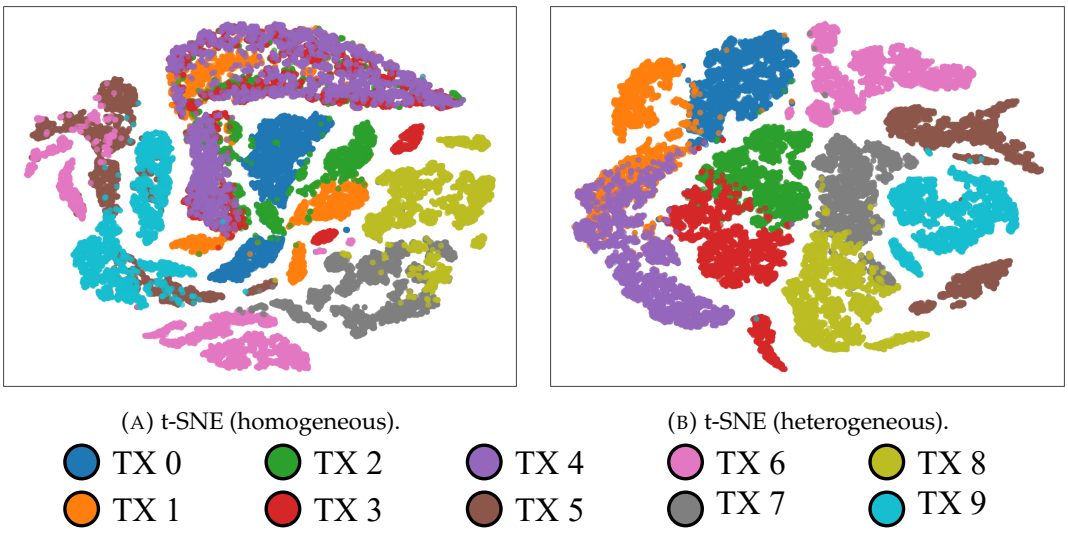


(B) Classification accuracy (heterogeneous).

FIGURE 4.7: LoRa classification accuracy under RX number.

in accuracy as the number of receivers increases. Nevertheless, Triad-GAN consistently surpasses the baseline, with the advantage becoming more evident in later test dates, indicating better robustness against temporal variations. In the heterogeneous scenario 4.7b, where training and testing receivers differ, overall accuracy is higher due to increased receiver diversity. Yet Triad-GAN maintains superior performance over the baseline across all test conditions. These results demonstrate that receiver diversity plays a critical role in transmitter identification and further highlight the strength of Triad-GAN in extracting receiver-invariant and discriminative features, thereby achieving robust generalization in multi-receiver environments.

To further investigate the discriminative power and transmitter-invariance of the learned features, we visualize the feature embeddings using t-distributed Stochastic Neighbor Embedding (t-SNE) under different conditions. We show the t-SNE visualizations of the extracted feature representations for LoRa signals under both heterogeneous and homogeneous schemes in Figure 4.8. The homogeneous scheme (Figure 4.8a) exhibits notable overlap and dispersion among certain transmitter classes. Although most clusters remain distinct, the reduced receiver diversity limits the effective disentanglement of transmitter identities, resulting in greater intra-class variability and a potential decline in classification accuracy. In comparison, the heterogeneous scheme (Figure 4.8b) yields feature embeddings for different transmitters (TX 0–TX 9) that form well-separated and compact clusters, and the presence of clear inter-class boundaries further underscores the robustness of Triad-GAN in mitigating receiver-induced variations. Overall, these t-SNE visualizations validate



(A) t-SNE (homogeneous). (B) t-SNE (heterogeneous).

FIGURE 4.8: t-SNE visualization of LoRa signal features.

the effectiveness of the proposed method in extracting transmitter-intrinsic features that are robust to receiver heterogeneity. The results also highlight the importance of receiver diversity for promoting improved generalization and class separability in practical RFFI systems.

4.5.3 Experimental Results and Analysis on WiSig

This section presents an evaluation of Triad-GAN on the WiSig dataset, focusing on receiver drift, receiver change, and the impact of receiver diversity on feature distribution.

FIGURE 4.9: WiSig classification accuracy comparison.

As shown in Figure 4.9, the classification accuracy of both methods decreases as the number of transmitter classes increases from 5 to 15, reflecting the increased difficulty of the task. Triad-GAN consistently outperforms the ResNet baseline across all class settings, with particularly notable gains at larger scales. For example, with 10 transmitter classes, Triad-GAN achieves an accuracy of approximately 83%, compared to 73% for ResNet, and with 15 classes, it improves performance by about 7% over the baseline. These results demonstrate that Triad-GAN is more effective at maintaining discriminative power as the classification task becomes more complex,

highlighting its robustness and scalability in large-class multi-receiver RFFI scenarios.

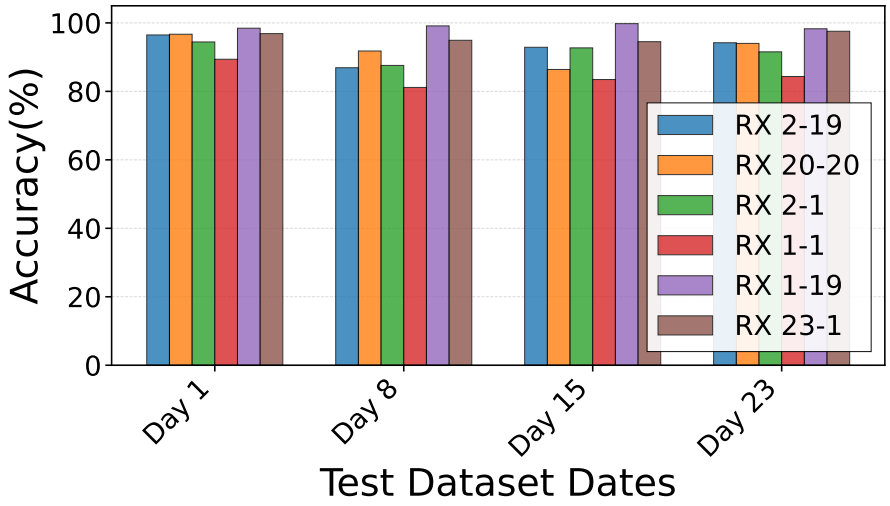


FIGURE 4.10: WiSig classification accuracy under receiver drift.

To evaluate the robustness of Triad-GAN under temporal variations, we first examine its performance in receiver drift. In this evaluation, Triad-GAN is trained on signals from 10 transmitters and 22 receivers, and tested on 6 unseen receivers. In Figure 4.10, we illustrate the classification accuracy of Triad-GAN on the WiSig dataset under receiver drift, where the model is evaluated on test datasets on different dates. Across all dates, Triad-GAN maintains high accuracy for different receiver devices. This robustness demonstrates the effectiveness of the learned feature representations in capturing transmitter-intrinsic signatures that remain stable despite temporal changes in the environment. Such temporal invariance is critical for real-world RFFI systems.

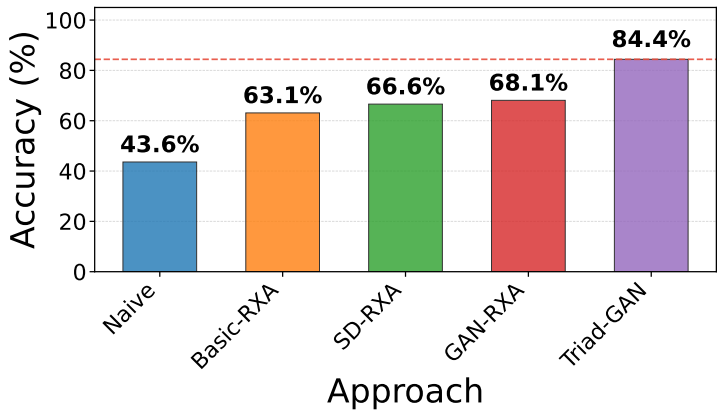


FIGURE 4.11: WiSig classification accuracy under receiver change.

Specifically, *Closed-Set Classification* means that the model is trained and evaluated on signals from the same set of transmitters, To evaluate the generalization capability of Triad-GAN for receiver-invariant transmitter fingerprinting under receiver change, we train Triad-GAN using 10 transmitters and 26 receivers, with testing conducted on 2 unseen receivers, and performance is measured by classification accuracy. Thus, aligning our tests with prior study [213], we perform this evaluation on the WiSig dataset and compare it with Naive, Basic-RXA, SD-RXA, and

GAN-RXA baselines from the study [213], which we selected as it achieves state-of-the-art closed-set classification performance in large-scale experiments on the WiSig dataset and implements multiple baselines used in our comparison. In Figure 4.11, we present an accuracy comparison among several baselines on the WiSig dataset. The results clearly indicate that Triad-GAN significantly outperforms all baseline methods, achieving an accuracy of 84.4%, compared to 68.1% for GAN-RXA and 43.6% for the Naive approach. This substantial performance improvement highlights the effectiveness of the Triad-GAN adversarial architecture in learning highly discriminative and receiver-invariant features. Together with the demonstrated stability under receiver drift, these results validate the practical applicability and reliability of Triad-GAN for large-scale, dynamic wireless authentication scenarios.

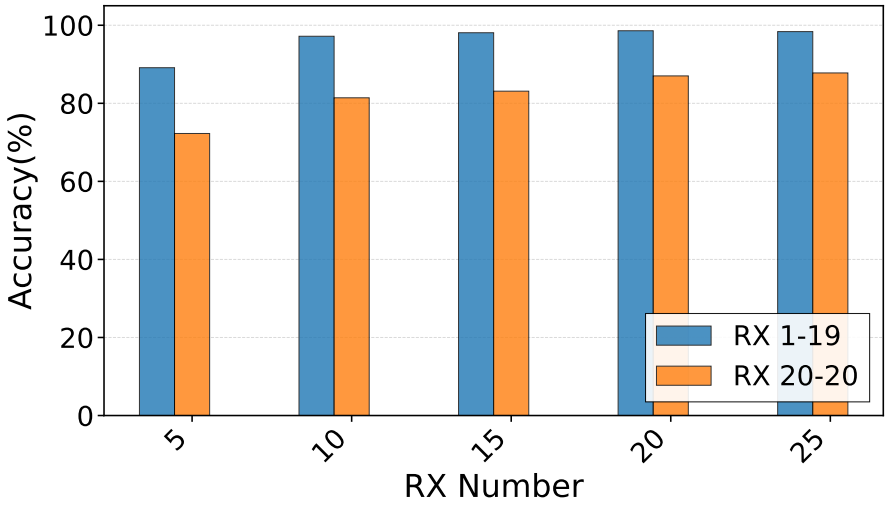
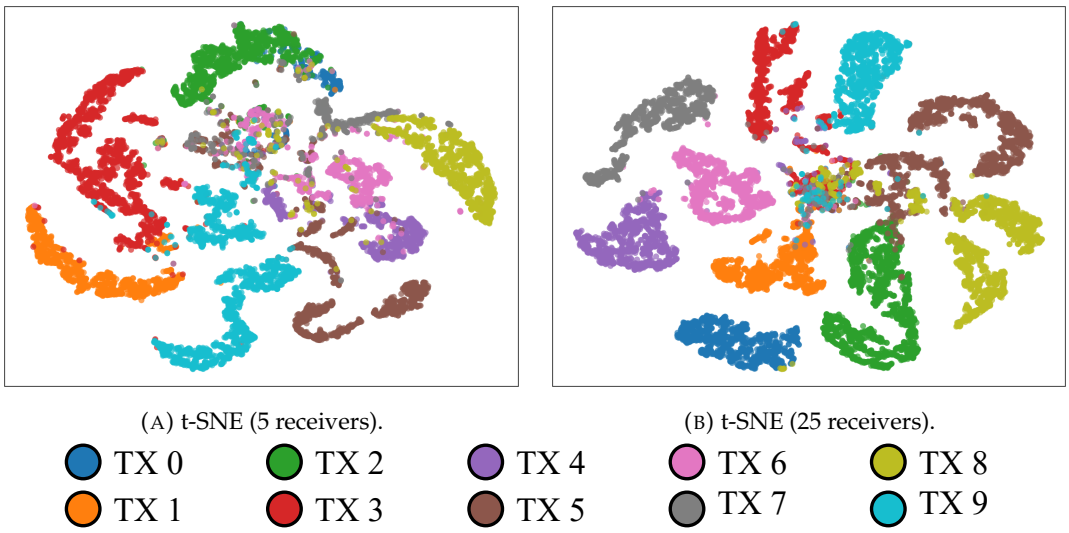


FIGURE 4.12: WiSig classification accuracy under RX number.

These results provide additional insights beyond those presented before. Supplementary experimental results on the WiSig dataset, as shown in Figure 4.12, illustrate how the number of receivers included in the training dataset influences transmitter classification accuracy. As the number of training receivers increases from 5 to 25, classification performance improves for both evaluation groups (RX 1-19 and RX 20-20). Furthermore, models trained with RX 1-19 consistently achieve higher accuracy than those trained with RX 20-20. These results confirm that both the quantity and selection of receivers in the training set play an important role in enhancing generalization. Overall, this experiment demonstrates the benefit of increasing receiver diversity in the training data for robust RFFI under different deployment conditions.

To illustrate the effect of receiver diversity on feature learning, we visualize the transmitter feature embeddings with t-SNE under different receiver settings. In Figure 4.13, we present the t-SNE visualizations of feature embeddings for the WiSig dataset under two receiver scenarios: (a) 5 receivers and (b) 25 receivers. In the 5-receiver scenario (Figure 4.13a), the features corresponding to different transmitters (TX 0–TX 9) exhibit reasonable separation, yet several clusters partially overlap or display elongated, dispersed structures. This suggests that while the feature extractor is capable of capturing some degree of transmitter-specific information, the limited receiver diversity may constrain the model's ability to fully disentangle device identities. In contrast, the 25-receiver scenario (Figure 4.13b) reveals much more compact and well-separated clusters for each transmitter. The significant reduction



(A) t-SNE (5 receivers). (B) t-SNE (25 receivers).

FIGURE 4.13: t-SNE visualization of WiSig signal features.

in intra-class variability and increased inter-class separation demonstrate the effectiveness of multi-receiver training in promoting robust, transmitter-intrinsic representations. The ability to maintain clear class boundaries in the presence of high receiver diversity further validates the generalization capability of the proposed feature learning approach. Overall, these t-SNE visualizations highlight the critical role of receiver diversity in enabling reliable and scalable RFFI in practical deployments.

4.6 Conclusion

In this paper, we introduced **Triad-GAN**, a novel feature-level GAN specifically designed to tackle the challenges posed by receiver variations in RFFI. By leveraging a triadic adversarial framework consisting of a feature extractor, generator, and discriminator, Triad-GAN effectively learns receiver-invariant and transmitter-discriminative features. Extensive experiments on datasets from LoRa and Wi-Fi protocols highlight the significant performance gains of our method, achieving classification accuracy improvements compared to state-of-the-art techniques under realistic LoRa and WiSig multi-receiver datasets and a more clear feature distribution visualization. Extensive experiments on LoRa and Wi-Fi datasets demonstrate that our method achieves classification accuracy improvements over state-of-the-art techniques in realistic multi-receiver scenarios. In addition, Triad-GAN provides clearer and compact feature distribution visualizations, further validating its effectiveness for receiver-invariant representation learning. These results underscore the robustness, scalability, and practicality of Triad-GAN for real-world IoT authentication systems. We hope that the advances presented in this work will inspire further exploration and innovation in this important yet underexplored domain.

Chapter 5

Open-Set Extension for RFFI (OS-TriadGAN)

Radio frequency fingerprint identification (RFFI) has emerged as a promising physical-layer technique for authenticating wireless devices by exploiting hardware-induced imperfections in their transmitted signals. While recent deep learning methods achieve near-perfect closed-set accuracy, they remain vulnerable under open-set conditions, where previously unseen or rogue devices may attempt unauthorized access. This work proposes **OS-TriadGAN**, an open-set RFFI framework that integrates adversarial feature-space alignment with calibration-aware training and validation-driven thresholding. The proposed model leverages a triadic adversarial structure, which combines a generator, a feature extractor, and a discriminator, together with label smoothing and temperature scaling, to produce transmitter-discriminative and receiver-invariant embeddings that are more separable from unknown devices. To ensure robust deployment, we adopt a principled F1-based calibration procedure that selects the rejection threshold on a validation split, balancing false acceptance and rejection errors. Extensive experiments on LoRa and WiSig benchmarks demonstrate that OS-TriadGAN achieves state-of-the-art closed-set accuracy (up to 99.7% and 99.5%) while consistently improving open-set detection, yielding 1 11% accuracy improvements on both datasets in accuracy over baselines. These results highlight the practicality and scalability of adversarially calibrated feature learning for secure wireless authentication in real-world IoT environments.

5.1 Introduction

In recent years, research on radio frequency fingerprint identification (RFFI) has provided promising new directions for securing next-generation wireless devices. A large number of studies have delivered solid results, offering a foundation for the development and potential large-scale deployment of RFFI. Considerable progress has been made across different problem settings and modeling strategies. However, among the many challenges in this field, open-set recognition remains one of the most practical yet unresolved problems. This challenge is crucial to determining whether an RFFI system can be trusted in real-world scenarios, and thus represents a fundamental research problem requiring high reliability and strong guarantees. It is evident that tackling this challenge will not only influence the applicability of current systems but will also shape the future prospects of RFFI, which has drawn increasing attention from researchers exploring novel, reliable, and efficient solutions.

With the rapid development of information technology, wireless communication has become indispensable due to its convenience and efficiency. However, the openness of the wireless medium inevitably exposes communication systems to security risks such as eavesdropping, spoofing, and unauthorized access [208]. Traditional encryption methods provide protection at the upper layers but face growing computational costs, particularly for resource-constrained IoT devices. As a complementary strategy, physical layer authentication (PLA) leverages the unique physical attributes of communication channels or hardware to enhance security with low overhead. Typical examples include channel state information (CSI) [186, 46], received signal strength indicator (RSSI) [52], and notably radio frequency fingerprinting (RFF) [144, 199]. RFFI exploits the intrinsic hardware imperfections introduced during the manufacturing process of RF components. These imperfections, although negligible for normal communication, are difficult to replicate and thus serve as unique device fingerprints [153]. The concept was first introduced by Hall et al. in 2003, who demonstrated transmitter classification based on fractal trajectories derived from phase variance analysis [64]. Since then, RFFI research has mainly followed two trajectories: (i) manual feature extraction combined with classical classifiers, and (ii) deep learning methods for automated representation learning. While early studies focused on expert-crafted features, more recent works have demonstrated the strong feature extraction capability of deep neural networks. For instance, [154] showed that CNNs applied to short-time fourier transform (STFT) spectrograms could achieve accuracy above 95%, and [115] introduced multidimensional fingerprint enhancement with attention fusion, pushing closed-set accuracy to 99.3%.

Despite such impressive performance, most RFFI studies are conducted under the closed-set assumption, where the devices in training and testing belong to the same set of identities. This assumption is unrealistic in practice, since rogue or previously unseen devices may attempt to access the network. Under such conditions, deep learning models often misclassify unknown devices as known ones with high confidence, leading to severe security risks. To address this issue, Gritsenko et al. [59] were the first to propose an unknown-class RFFI scheme, using slice-level confidence aggregation to detect new devices. Later, Hanna et al. [70] systematically introduced open-set recognition and anomaly detection into RFFI, evaluating the robustness of multiple models under open-set conditions. Other works have explored novel preprocessing strategies, such as constellation trace figures combined with ResNet backbones, and compared decision layers such as SoftMax and OpenMax for open-set classification [205]. Results consistently showed that while SoftMax excels in closed-set tasks, OpenMax or other threshold-based strategies outperform in open-set detection.

Building on these insights, we leverage the proposed Triad-GAN architecture, which is described in Chapter 4, to enhance open-set RFFI. While prior work has mainly focused on refining preprocessing pipelines or classifier heads, our approach strengthens the *feature extraction process itself*. By introducing adversarial feature-space alignment, Triad-GAN produces embeddings that are both transmitter-discriminative and more separable from unknown devices. This reduces the overlap of feature distributions, thereby mitigating the reliance on finely tuned thresholds. To enhance decision reliability in open-set recognition, we adopt a validation-based calibration strategy. Rather than fixing thresholds manually or relying on heuristic rules, the proposed system determines the decision boundary on a held-out validation set. This design allows consistent adaptation to new deployment environments without retraining and reduces the sensitivity of performance to

arbitrary threshold choices. Building on these improvements, we extend Triad-GAN into the *OS-TriadGAN* framework, specifically tailored to address the challenges of open-set RFFI. The key contributions of this work can be summarized as follows:

- We introduce **OS-TriadGAN**, an open-set RFFI framework that integrates adversarial feature-space alignment with label smoothing and temperature scaling, thereby enhancing the separability of known and unknown devices.
- We propose a **validation-driven calibration procedure** for threshold selection, which reduces sensitivity to manual tuning and ensures robustness across heterogeneous deployment scenarios.
- We conduct extensive evaluations on both the LoRa and WiSig RFFI datasets under diverse open-set conditions, demonstrating that OS-TriadGAN outperforms existing approaches by 1 11% in accuracy and achieves more stable precision, recall, and F1 performance.

The remainder of this chapter is organized as follows. Section 5.2 reviews prior work on closed-set RFFI, open-set recognition, and learning paradigms for RF device authentication. Section 5.3 formalizes the open-set RFFI system model and defines the problem. Section 5.4 details the proposed OS-TriadGAN framework, including adversarial feature learning, calibration strategies, and decision rules. Section 5.5 presents experimental results on LoRa and WiSig datasets, evaluating both closed-set accuracy and open-set detection performance. Finally, Section 5.6 concludes with key findings and outlines future research directions.

5.2 Related Work

5.2.1 From Closed-Set RFFI to Open-Set Recognition

Radio Frequency Fingerprinting Identification (RFFI) authenticates devices by exploiting hardware imperfections that are difficult to clone. The concept was introduced by Hall *et al.*, who demonstrated transmitter classification from phase–variance based fractal trajectories [64]. Subsequent work progressed along two main lines: hand-crafted features with classical classifiers, and deep learning based representation learning. In particular, convolutional networks on time–frequency views such as STFT spectrograms have achieved strong closed-set accuracy [154], with later designs enhancing separability through fingerprint enrichment and attention mechanisms [115]. However, early evaluations were almost exclusively conducted under a closed-set protocol, assuming that training and test identities coincide. This assumption is unrealistic in practice, where previously unseen or rogue devices may appear at test time. Gritsenko *et al.* first exposed this vulnerability through slice-level confidence aggregation [59], while Hanna *et al.* formally framed RFFI as an open-set problem and benchmarked multiple scoring rules, showing that standard Softmax classifiers often over-confidently mislabel unknowns as known [70]. Inspired by developments in computer vision, follow-up studies compared OpenMax with Softmax and demonstrated improved unknown detection while preserving closed-set performance [205, 17]. Modern open-set RFFI typically follows a common recipe: learning a discriminative embedding from raw I/Q samples, spectrograms, or constellation traces; modeling known classes in this space through logits, class centers, or prototypes; computing an unknownness score at inference time, such as one minus the maximum softmax probability (MSP), logit

energy, or distance to class representatives; and selecting a threshold on a validation split using criteria such as the F1-optimal point [147], equal error rate [121], or Youden's J statistic [49]. While effective, this approach remains sensitive to calibration and to partial overlap among legitimate devices, motivating the development of alternative learning paradigms.

5.2.2 Learning Paradigms for Open-Set RFFI

In response to these challenges, several major research directions have emerged. CNN backbones on time–frequency inputs remain a strong baseline [154], and refinements such as multi-branch processing, attention-based fusion, and fingerprint enhancement further improve separability under channel variability [115]. Beyond these baselines, open-set RFFI research has advanced along multiple complementary fronts. One line of work focuses on *confidence and activation design*, replacing the terminal Softmax with alternatives such as OpenMax, while calibration techniques including temperature scaling and label smoothing mitigate over-confidence on out-of-distribution inputs. Another direction emphasizes *metric and prototype learning*, introducing objectives such as center loss, Siamese networks, and triplet learning to compact intra-class representations and enlarge inter-class margins; verification is then cast as a distance test that naturally rejects samples outside an acceptance radius. A third area explores *adversarial and generative alignment*, addressing domain shift across channels and receivers by encouraging receiver-invariant embeddings and reshaping the manifold to discourage high confidence away from the data support. A further stream investigates *hybrid closed–open formulations*, which combine class prediction with an anomaly score, enabling inference to adaptively select the more reliable decision. Finally, the community has developed *evaluation protocols and metrics* that standardize open-set assessment by holding out transmitters as unknowns and reporting closed-set accuracy, AUROC, EER, precision–recall, and composite open-set accuracy, with thresholds selected via F1, EER, or Youden's J. These developments move RFFI research beyond closed-set performance on curated datasets toward integrated open-set pipelines that combine representation learning, calibrated scoring, and principled threshold selection. Early demonstrations of open-set risk [59, 70] catalyzed the adaptation of techniques such as OpenMax-style activations [17, 205], confidence calibration, compact prototype learning, and adversarial alignment, laying the foundation for current approaches that reshape embedding spaces and reduce threshold sensitivity while sustaining strong device identification accuracy.

5.3 Open-Set RFFI Scenario and Problem Formulation

5.3.1 Open-Set RFFI System Model

Radio Frequency Fingerprinting Identification seeks to authenticate wireless transmitters by exploiting unique hardware-induced artifacts embedded in their emitted signals. These imperfections, such as I/Q imbalance, phase noise, or nonlinear distortion, arise from manufacturing variability and are difficult to replicate even across devices of the same model. To formalize the setting, we describe the underlying signal model, the structure of the dataset, and the distinction between the *training* and *deployment* stages, as illustrated in Figure 5.1.

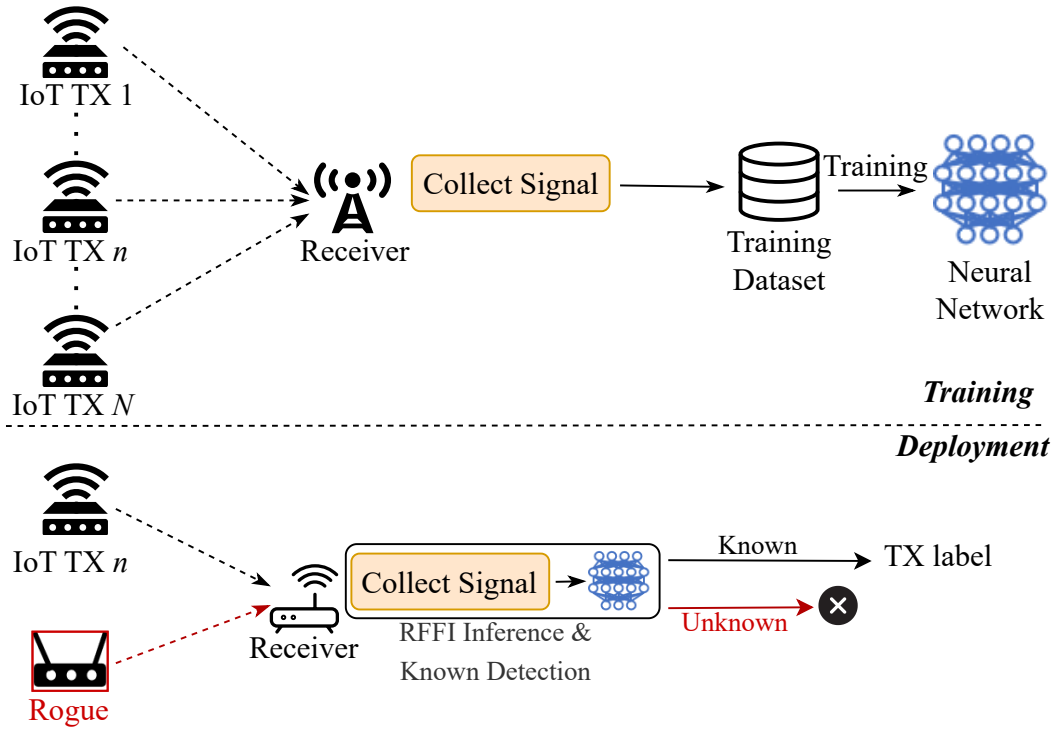


FIGURE 5.1: Open-Set RFFI system.

During system development, a labeled dataset is constructed from K known transmitters. The training set is defined as

$$\mathcal{D}_{\text{train}} = \{(y_i(t), \mathbf{tx}_i)\}_{i=1}^{N}, \qquad \mathbf{tx}_i \in \mathcal{Y} = \{0, 1, \ldots, K-1\}, \tag{5.1}$$

where each sample $y_i(t)$ represents a received RF signal and $\mathbf{tx}_i$ is its transmitter label. In practice, packets are converted into spectrograms or other structured representations before being batched for neural network optimization. This RF signal serves as the foundation for supervised learning, allowing the system to map raw signals to transmitter identities. Where the received signal $y_i(t)$ at time t can be expressed as

$$y(t) = f_{\text{rx}}\left(h * f_{\text{tx}}(x(t))\right) + n(t), \tag{5.2}$$

where $x(t)$ denotes the original transmitted baseband waveform, $f_{\text{tx}}(\cdot)$ and $f_{\text{rx}}(\cdot)$ capture hardware distortions from the transmitter and receiver respectively, h is the wireless channel impulse response, $n(t)$ is additive noise, and $*$ denotes convolution. The composite effect of these components produces subtle but stable radio frequency (RF) fingerprints that can be extracted for device identification.

During the *training* stage, receivers collect signals from all K known transmitters under varied channel and environmental conditions. These samples are preprocessed, labeled, and used to train a deep neural network classifier. The model learns to associate transmitter-specific distortions with identity labels by minimizing classification loss on the training dataset. At this stage, the problem is inherently *closed-set*, since all transmitters that will be encountered during training are fully represented in the dataset.

During the *deployment* stage, the trained model is applied to an operational wireless environment. Receivers continuously collect signals from devices attempting

to access the network. Each incoming signal is embedded by the feature extractor, scored by the calibration algorithm, and compared against known classes. Legitimate signals are accepted and associated with their transmitter identity, while illegitimate signals should ideally be rejected. Unlike the training stage, the deployment stage introduces the possibility of encountering *previously unseen* transmitters, including rogue devices attempting to impersonate legitimate nodes. These signals do not belong to the training label set $\mathcal{Y}$ and must therefore be treated as belonging to the *open set*.

Overall, the divergence between training and deployment settings gives rise to the open-set RFFI problem. While training operates under a closed-set assumption, real-world deployment requires the system to perform two simultaneous tasks: (i) Accurately classify packets originating from known transmitters in $\mathcal{Y}$; (ii) Reliably detect and reject packets from unknown transmitters that were absent during training. Failure to reject such unknown or rogue transmitters leads to severe security risks, as unauthorized devices may be misclassified as legitimate and gain access to the IoT network. Consequently, open-set RFFI demands both discriminative feature extraction and reliable anomaly detection mechanisms to ensure secure and trustworthy operation in practical deployments.

5.3.2 Problem Formulation

In practice, when deploying an RFFI in the scenario, the RFFI system can always receive the signals from the known and unknown devices. To introduce the open-set challenge more standardly, the received signals are usually divided into known (legitimate) and unknown (rogue) device sets. Let $\mathcal{Y} = \{0, 1, \ldots, K-1\}$ denote the set of K legitimate transmitters observed during training. During deployment, however, the system may encounter packets originating from transmitters not included in $\mathcal{Y}$. These previously unseen devices are denoted collectively as the *open set*. Therefore, the authentication task in open-set RFFI is two-fold: (ii) **Closed-set classification:** correctly identify signals from transmitters in $\mathcal{Y}$; (ii) **Open-set rejection:** detect signals from devices outside $\mathcal{Y}$ and prevent their unauthorized access.

A common strategy is to associate each input signal y with an *unknownness score*

$$s(y) = g(y), \tag{5.3}$$

where $g(\cdot)$ stands the converting processes and quantifies the likelihood that y belongs to the open set. Typical choices include the MSP, logit energy scores, or distances to class prototypes. A global threshold τ is then used to define the decision rule:

$$\text{Decision} = \begin{cases} \text{Classify as legitimate device,} & \text{if } s(y) \geq \tau, \\ \text{Reject as unknown device,} & \text{if } s(y) < \tau. \end{cases} \tag{5.4}$$

This formulation highlights the central challenge of open-set RFFI: the system must balance accurate closed-set classification with reliable rejection of an *unbounded* number of unknown devices. Addressing the open-set RFFI problem requires more than a simple score–threshold mechanism. Effective solutions must reshape the embedding space to enlarge margins between known and unknown distributions, provide calibrated confidence estimates, and remain robust under realistic distributional shifts.

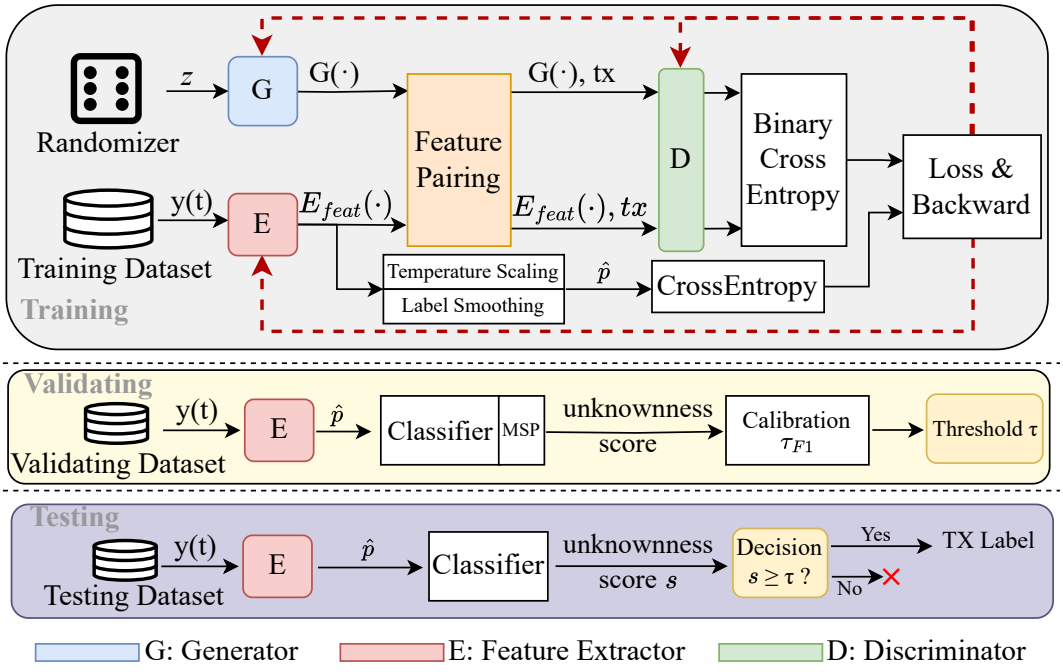


FIGURE 5.2: Overall workflow of the OS-extension Triad-GAN framework.

5.4 Proposed Method

5.4.1 Structure of OS-TriadGAN

The overall workflow of the proposed OS-TriadGAN framework is shown in Figure 5.2. It extends the conventional adversarial feature learning paradigm by coupling a generator (G), a feature extractor (E), and a discriminator (D) with a validation-driven calibration procedure. The framework operates in three stages: training, validating, and testing.

During the training stage, two sources of information are used jointly. First, a **randomizer** generates latent noise vectors z, which are concatenated with transmitter labels and input to the generator G. The generator synthesizes class-conditional pseudo-features $G(z, tx)$ that mimic real transmitter embeddings. Second, the training dataset provides observed signals $y(t)$, which are mapped by the feature extractor E into two outputs: a feature embedding $E_{\text{feat}}(y(t))$ and a predicted probability vector $\hat{p}$. The real features $E_{\text{feat}}(y(t))$ and the synthetic features $G(z, tx)$ are then paired with their corresponding transmitter labels and passed to the discriminator D. The discriminator learns to distinguish real from generated embeddings using a binary cross-entropy (BCE) loss, while the generator is trained adversarially to produce embeddings indistinguishable from real ones. In parallel, the predicted probabilities $\hat{p}$ are supervised by a cross-entropy loss with the true transmitter label, where **label smoothing** is applied to soften the target distribution and mitigate overconfidence. This triadic adversarial game jointly drives (i) the generator to enrich the feature distribution, (ii) the discriminator to enforce authenticity, and (iii) the extractor to produce transmitter-discriminative embeddings that are compact and robust.

After training, the framework introduces a validating phase to address the open-set challenge. Samples from a held-out validation set, which contains both known

devices (seen during training) and auxiliary unknown devices (never used in training or in testing), are processed by the extractor and classifier. The predicted probabilities $\hat{p}$ are first adjusted using **temperature scaling**, which calibrates the confidence of Softmax outputs. From these calibrated probabilities, **unknownness scores** are derived using MSP, defined as the complement of the largest Softmax output. A global rejection threshold τ is then calibrated on the validation split by maximizing the F1-score with the unknown class treated as positive. This procedure yields a principled, data-driven operating point that balances false acceptance and false rejection, while reducing the sensitivity of the system to arbitrary threshold choices.

During the testing stage, the trained feature extractor E and classifier are applied to incoming packets from both legitimate and potentially rogue transmitters. Each signal $y(t)$ is embedded, classified, and converted into a probability vector $\hat{p}$, which is again calibrated through **temperature scaling**. The corresponding MSP score is then compared against the fixed threshold τ: if the score $s < \tau$, the signal is rejected as *unknown* device; otherwise, it is classified into one of the known transmitter classes. This design ensures that only devices from the closed set are granted access to the IoT network, while previously unseen or rogue devices are reliably detected and blocked. Because the threshold τ is fixed once during validation and does not require retraining, the system can be easily adapted to new environments by recalibrating with a small validation slice.

In summary, the OS-extension Triad-GAN integrates adversarial feature learning with validation-based calibration. The adversarial interplay among G, E, and D encourages embeddings that maximize separability between transmitters while discouraging overconfidence on out-of-distribution signals. The combination of **label smoothing**, **temperature scaling**, and MSP+τ calibration then provides a principled open-set decision rule. Together, these components yield a framework that is both highly discriminative in closed-set classification and robust in open-set rejection, offering a practical solution for real-world RFFI deployment.

5.4.2 Adversarial Feature Learning with Supervised Classification

The objective of the proposed **OS-TriadGAN** is to enable reliable open-set RFFI by learning feature representations that are both transmitter-invariant and robust to distribution shifts. Unlike conventional models that overfit to known devices or specific receivers, OS-TriadGAN is explicitly designed to generalize to previously unseen devices while preserving strong discriminability among known classes. This is achieved by integrating supervised classification with adversarial training in the feature space, thereby encouraging the extractor to produce embeddings that are simultaneously receiver-invariant and transmitter-discriminative. In addition, the framework incorporates calibration-aware mechanisms — such as label smoothing and temperature scaling — to improve decision confidence and enhance the separability between known and unknown devices. The optimization of OS-TriadGAN is guided by three complementary loss components that jointly align feature distributions, enforce class separability, and promote robust open-set detection.

- **Transmitter Classification Loss with Label Smoothing and Temperature Scaling:** The feature extractor $E(\cdot)$ maps each received signal $y(t)$ directly into

a probability distribution $\hat{p} \in \mathbb{R}^N$ over the transmitter classes. To improve calibration for open-set detection, the distribution is obtained with temperature-scaled Softmax:

$$\hat{p}_{i,k} = \frac{\exp\left(E'_{i,k}/T\right)}{\sum_{j=1}^{N} \exp\left(E'_{i,j}/T\right)}, \tag{5.5}$$

where $E'_{i,k}$ denotes the k-th pre-activation output of E and $T > 0$ is the temperature parameter. Furthermore, we apply *label smoothing* to soften the one-hot ground-truth distribution $p_{i,k}$: for a true class tx,

$$p_{i,k} = \begin{cases} 1-\epsilon, & k = tx, \\ \epsilon/(N-1), & k \neq tx, \end{cases}$$

where $\epsilon \in [0, 1)$ is the smoothing factor. The classification loss is thus

$$\mathcal{L}_{\text{cls}}^{E} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{k=1}^{N} p_{i,k} \log\left(\hat{p}_{i,k}\right), \tag{5.6}$$

which encourages the model to avoid overconfident predictions and improves the reliability of threshold-based open-set rejection.

- **Generator Adversarial Loss:** The generator $G(\cdot)$ synthesizes pseudo-feature embeddings $G(z, tx)$ from a latent vector z and a transmitter label tx. The discriminator $D(\cdot, tx)$ outputs the probability that an input feature vector is real given the transmitter label tx. The generator adversarial loss is

$$\mathcal{L}_{\text{adv}}^{G} = -\frac{1}{B} \sum_{i=1}^{B} \log D(G(z_i, tx_i), tx_i), \tag{5.7}$$

which compels G to generate embeddings that align with the manifold of real transmitter features.

- **Discriminator Adversarial Loss:** The discriminator $D(\cdot, tx)$ is trained to distinguish between real features $E_{\text{feat}}(y_i(t))$ and generated features $G(z_i, tx_i)$, conditioned on the transmitter label tx_i. Its binary cross-entropy loss is

$$\mathcal{L}_{\text{adv}}^{D} = -\frac{1}{B} \sum_{i=1}^{B} \Big[\log D(E_{\text{feat}}(y_i(t)), tx_i) + \log\Big(1 - D(G(z_i, tx_i), tx_i)\Big) \Big]. \tag{5.8}$$

This ensures that D enforces the authenticity of E's embeddings while continuously pressuring G to improve feature realism.

- **Overall Objective:** The total loss integrates all terms:

$$\mathcal{L}_{\text{total}} = \lambda_E \mathcal{L}_{\text{cls}}^{E} + \lambda_G \mathcal{L}_{\text{adv}}^{G} + \lambda_D \mathcal{L}_{\text{adv}}^{D}, \tag{5.9}$$

where λ_E, λ_G, and λ_D balance the contributions of classification, generator adversarial, and discriminator adversarial objectives. By incorporating *label*

smoothing and *temperature scaling* into the classification loss, the open-set TriadGAN not only achieves strong closed-set accuracy but also produces well-calibrated confidence estimates, improving robustness under threshold-based rejection.

5.4.3 Threshold Calibration and Testing Application

In the OS-TriadGAN, the MSP is adopted as the *unknownness score*. Given the temperature-scaled Softmax distribution, the MSP score for a signal $y(t)$ is defined as

$$s_{\mathrm{MSP}}(y(t)) = 1 - \max_{k\in\{1,\ldots,N\}} \hat{p}_k^{(T)}, \tag{5.10}$$

where $\hat{p}_k^{(T)}$ is the k-th probability after temperature scaling. The decision rule is then defined as

$$\text{Decision} = \begin{cases} \arg\max_k \hat{p}_k^{(T)}, & \text{if } s_{\mathrm{MSP}}(y(t)) \geq \tau, \\ \text{Reject as unknown device}, & \text{if } s_{\mathrm{MSP}}(y(t)) < \tau, \end{cases} \tag{5.11}$$

with τ denoting a global rejection threshold calibrated on a validation set.

During validation, we construct a held-out split containing both known devices and devices from the unseen device set, which serve only as unknowns for calibration. The threshold τ is selected once by maximizing the F1-score, treating unknown devices as the positive class:

$$\tau_{F1} = \arg\max_t \mathrm{F1}(\mathbb{1}[s_{\mathrm{MSP}} \geq t], \text{unknown}) \,. \tag{5.12}$$

This criterion directly optimizes the precision–recall trade-off, which is more suitable than ROC-based rules under the inherent imbalance between the finite closed set and the unbounded open world. In practice, OS-TriadGAN adversarial feature shaping enlarges the separation between known and unknown distributions, producing a broad plateau of high F1 around τ_{F1} and thereby reducing sensitivity to small calibration errors.

For deployment, each received packet is processed by the feature extractor E to produce the probability vector $\hat{p}$, from which the MSP score is computed. The rejection condition is expressed as

$$s_{\mathrm{MSP}} \geq \tau_{F1}, \tag{5.13}$$

which is algebraically equivalent to

$$p_{\max} \leq 1 - \tau_{F1}, \tag{5.14}$$

where $p_{\max} = \max_k \hat{p}_k^{(T)}$ denotes the maximum Softmax probability with temperature scaling. Accordingly, the threshold τ_{F1} in MSP space directly maps to the conventional probability threshold

$$\tau = 1 - \tau_{F1}, \tag{5.15}$$

ensuring fair comparison with prior work that reports thresholds in the $p_{\max}$ domain. All thresholds τ are reported in the $p_{\max}$ domain, obtained from the MSP threshold τ_{F1} via the conversion $\tau = 1 - \tau_{F1}$.

Finally, temperature scaling mitigates overconfidence in the classifier outputs, while label smoothing prevents memorization of training samples. Together, these

techniques stabilize MSP scores, making threshold-based rejection more reliable across varying environments. This design supports efficient *plug-in calibration* using a small validation slice without retraining, while maintaining strict separation between calibration and test devices.

5.5 Experiments and Analysis

5.5.1 Dataset Description

To comprehensively evaluate the proposed OS-TriadGAN, we conduct experiments on two representative large-scale RF fingerprinting datasets: a LoRa-based COTS device dataset and the publicly available WiSig Wi-Fi dataset. These datasets differ in protocol, hardware diversity, and collection environment, thereby providing complementary validation of both closed-set classification accuracy and open-set device detection performance. In all experiments, the data is partitioned into training, validation, and testing sets using an 8:1:1 ratio unless otherwise specified. The LoRa dataset emphasizes cross-manufacturer generalization in a single-protocol, SDR-based setup, while the WiSig dataset stresses large-scale, multi-receiver robustness under Wi-Fi signals. Both datasets support closed-set classification experiments as well as open-set detection tasks, enabling a thorough validation of the proposed framework.

LoRa Dataset

The LoRa dataset contains RF signals collected from 45 commercial off-the-shelf (COTS) devices of the Pycom LoPy4 series, each equipped with the Semtech SX1276 chip. Signals were captured at 868.1 MHz using a USRP N210 with a sampling rate of 1 MHz in a fixed indoor line-of-sight (LOS) setting, where each device transmitted periodically every 0.3 s. The data is organized into multiple HDF5 subsets, with IQ samples stored as real and imaginary branches along with corresponding device labels. To enhance generalization, data augmentation (multipath fading and Doppler shift) is applied to the training data.

For closed-set evaluation, devices indexed 1–30 are treated as legitimate transmitters, providing up to 1000 packets per device. For open-set evaluation, devices 31–45 are designated as rogue transmitters and are entirely excluded from training. To preserve the integrity of the evaluation, the open-set group is further divided into two disjoint subsets: one subset is used exclusively during validation to calibrate the rejection threshold, while the remaining unseen devices are reserved for testing. In this way, the model is trained and validated only on legitimate devices together with auxiliary unknowns that never appear in the test set, ensuring a fair assessment of generalization. The final open-set test set, therefore, includes both legitimate transmitters and a separate group of rogue transmitters, simulating a realistic deployment scenario where the system must correctly identify known devices while reliably rejecting previously unseen ones.

WiSig Dataset

WiSig is the largest publicly available Wi-Fi fingerprinting dataset, containing over 10 million packets from 174 transmitters and 41 receivers across four capture days. In this work, we focus on the pre-packaged `SingleDay` subset provided by the authors, which is particularly suited for controlled open-set evaluation.

The `SingleDay` subset contains signals from 28 transmitters and 10 receivers captured within a single day, with each Tx–Rx pair contributing 800 IEEE 802.11a/g preamble signals at 2 Msps. Signals are standardized into 256-sample IQ sequences and processed with MMSE equalization to mitigate channel effects. For closed-set experiments, all 28 transmitters are considered legitimate, with disjoint splits used for training, validation, and testing (following the 8:1:1 ratio). For open-set evaluation, we partition the 28 transmitters into two groups: 26 devices serve as the legitimate set, while the remaining 2 devices are held out entirely as rogue devices. The model is trained and validated exclusively on the legitimate set, while the open-set test set includes test samples from both the legitimate devices and the rogue devices. This setup emulates a realistic deployment scenario where the system must authenticate known transmitters while rejecting previously unseen ones.

5.5.2 Closed-Set Classification Accuracy

Results: LoRa for Closed-Set

To establish a performance baseline, we first evaluate the closed-set classification accuracy of different approaches, where all transmitters appearing in the test phase are also present during training. Table 5.1 shows the confusion matrices for three representative systems: the scalable channel-robust RFFI baseline (SC-RFFI [153]), the joint RFF processing with spectrogram calibration (JRFFP-SC [28]), and our proposed OS-TriadGAN.

SC-RFFI achieves an overall accuracy of 95.33%, which confirms its scalability and robustness under controlled channel conditions (Table 5.1). JRFFP-SC further improves the performance to 98.47% (Table 5.1), benefiting from enhanced feature representation and calibration. In contrast, our GAN-based framework achieves the highest accuracy of 99.74% (Table 5.1 and Figure 5.3), with almost perfect diagonal dominance and negligible off-diagonal misclassifications.

The superior performance of our method can be attributed to two key design choices. First, the GAN-driven adversarial feature learning extracts highly discriminative latent representations that capture subtle hardware-specific artifacts. By shaping the feature distribution through generator–discriminator competition, the embeddings become more separable across devices, reducing confusion even among transmitters of the same chipset family. Second, the integration of validation-based threshold calibration, specifically using the F1-optimal threshold, ensures that the decision boundary is well aligned with the open-set extension of the problem. Although the results in this subsection focus on the closed-set case, the consistent application of the thresholding strategy provides a principled mechanism for extending to open-set rejection without sacrificing closed-set accuracy.

To prevent information leakage between training and testing, the test dataset was carefully partitioned such that no packet instance in the test set had been seen during either training or validation. This ensures a fair evaluation of generalization and demonstrates that the proposed model does not overfit to specific signal realizations but instead captures underlying device-level features. These results highlight that the GAN-augmented framework not only preserves high closed-set classification performance but also builds a strong foundation for reliable open-set recognition in dynamic wireless environments.

TABLE 5.1: Closed-set classification accuracy comparison among representative RFFI methods.

Method	Accuracy (%)
SC-RFFI	95.33
JRFFP-SC	98.47
OS-TriadGAN	**99.74**

Results: WiSig for Closed-Set

To further validate the generalization of the proposed framework, we evaluate closed-set classification performance on the `SingleDay` subset of the WiSig dataset. This benchmark is particularly challenging due to its multi-receiver setup, yet it also provides a consistent environment for fair comparison with prior work. Table 5.2 reports the classification accuracy of several representative baselines alongside our proposed method.

Early convolutional models such as CNN [48] achieved an accuracy of 85%, confirming the feasibility of deep learning for Wi-Fi fingerprinting but leaving substantial room for improvement. Later studies introduced optimization for embedded deployment, with CNN-TFLite [80] achieving 99% accuracy. Transformer-based designs also demonstrated strong performance, with Transformer-ResNet [162] reaching 93% and Transformer-TFLite [80] achieving 98%.

Our OS-TriadGAN framework achieves an accuracy of 99.54%, surpassing all baselines reported in Table 5.2. This performance demonstrates that OS-TriadGAN not only preserves nearly perfect closed-set accuracy but also offers a generalizable architecture capable of addressing open-set rejection. The results highlight that the proposed framework delivers state-of-the-art performance in closed-set scenarios while maintaining the added advantage of robustness to unseen devices, which is the central challenge in open-set RFFI.

TABLE 5.2: Closed-set classification accuracy on the WiSig `SingleDay` subset. Best results in **bold**.

Method	ACC (%)
CNN	85
CNN-TFLite	99
Transformer-ResNet	93
Transformer-TFLite	98
OS-TriadGAN	**99.54**

5.5.3 Open-Set and Unknown Device Detection

Results: LoRa for Open-Set

The open-set device detection results on the LoRa dataset are summarized in Table 5.3, where we compare four representative methods: the Siamese-based SIA-RFF [24], the scalable channel-robust SC-RFFI [153], the joint representation

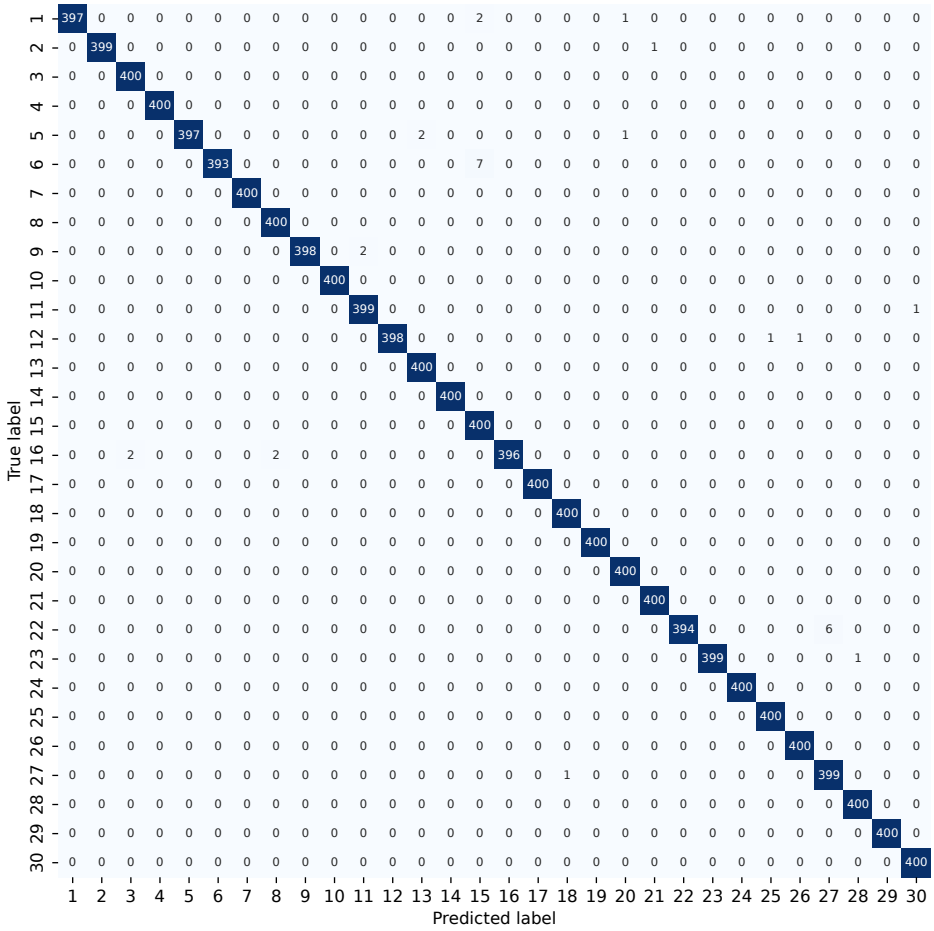


FIGURE 5.3: Closed-set classification result of OS-TriadGAN (accuracy = 99.74%).

and spectrogram calibration framework JRFFP-SC [28], and our proposed OS-TriadGAN. Evaluation follows the open-set protocol in which a subset of transmitters is held out during training and later introduced at test time as unknown devices. Metrics include accuracy, precision, recall, F1-score, and inference latency.

Among all methods, OS-TriadGAN achieves the best overall balance across metrics. In terms of accuracy, our framework reaches 96.9%, outperforming JRFFP-SC (93.9%), SC-RFFI (91.6%), and SIA-RFF (85.3%). While JRFFP-SC attains the highest recall (93.9%) and SC-RFFI achieves the highest F1-score (93.9%), OS-TriadGAN provides a more balanced trade-off, with precision of 90.3% and recall of 89.4%, resulting in a competitive F1-score of 92.8%. This balance is crucial in open-set RFFI, where both false acceptance of unknown devices and false rejection of legitimate ones must be minimized. By contrast, SC-RFFI exhibits strong precision (96.3%) but lower recall (91.6%), leading to less favorable performance under class imbalance.

The superior performance of OS-TriadGAN stems from its calibration strategy: MSP is used as the unknownness score, and a rejection threshold τ_{F1} is selected on a held-out validation set to maximize the F1 metric. This approach explicitly balances false positives and false negatives, ensuring stable and reproducible open-set detection.

Inference efficiency further highlights the practicality of the proposed method. OS-TriadGAN processes each packet in 0.1038 s, which is significantly faster than SC-RFFI (2.73 s) and JRFFP-SC (1.47 s), while remaining moderately slower than the

TABLE 5.3: Comparison of open-set detection performance on the LoRa dataset.

Method	Accuracy	Precision	Recall	F1-Score	Time (s)
SIA-RFF	0.853	0.708	0.853	0.774	**0.0043**
SC-RFFI	0.916	**0.963**	0.916	**0.939**	2.7305
JRFFP-SC	0.939	0.864	**0.939**	0.900	1.4749
OS-TriadGAN	**0.969**	0.903	0.894	0.928	0.1038

extremely lightweight SIA-RFF (0.0043 s). This latency remains acceptable for real-time wireless authentication and reflects a favorable balance between computational efficiency and robustness.

Overall, these results demonstrate that OS-TriadGAN offers a well-calibrated balance of accuracy, robustness, and efficiency. By uniting adversarial feature learning with validation-driven calibration, the framework provides reliable open-set detection, scales effectively to diverse deployment scenarios, and achieves inference efficiency suitable for practical wireless security systems.

Results: WiSig for Open-Set

TABLE 5.4: Comparison of Open-Set RFFI Results on the SingleDay Dataset. All thresholds τ are reported in the $p_{\max}$ domain, obtained from the MSP threshold τ_{F1} via the conversion $\tau = 1 - \tau_{F1}$.

Model	**Temperature T**	**Threshold τ**	**Accuracy (%)**
HyDRA (TDSE)	1.6	0.999	94.67
HyDRA (TDSE)	1.0	1.0	$\sim$45
OS-TriadGAN	0.95	0.9561	**96.11**

The open-set detection performance on the WiSig `SingleDay` subset is presented in Table 5.4, where we compare HyDRA (TDSE) [112] with our proposed OS-TriadGAN. HyDRA represents a state-of-the-art Wi-Fi fingerprinting model that introduces temporal–spectral enhancement to strengthen discriminability under channel variations. Although highly optimized for Wi-Fi scenarios, its performance is strongly affected by calibration choices. Specifically, when evaluated with the default temperature scaling $T = 1.0$, HyDRA exhibits severe overconfidence, reaching only about 45% accuracy. Increasing the temperature to $T = 1.6$ alleviates this limitation, improving the accuracy to 94.67%. These results underline both the strength of HyDRA's architecture and its sensitivity to calibration in multi-receiver environments with substantial distributional shifts.

Overall, these results demonstrate that OS-TriadGAN achieves state-of-the-art open-set detection on Wi-Fi signals by jointly leveraging adversarial feature learning and calibration-aware mechanisms. The framework not only surpasses specialized baselines such as HyDRA but also establishes a more generalizable and stable approach for separating known and unknown devices in large-scale, multi-receiver environments.

Summary

Across both the LoRa and WiSig benchmarks, the experiments provide complementary evidence of the effectiveness of the proposed framework. For LoRa, the main

challenge lies in distinguishing devices within the same chipset family, where OS-TriadGAN achieves the highest overall accuracy and delivers a well-balanced trade-off between precision, recall, and F1-score. For WiSig, the difficulty arises from large-scale, multi-receiver variability, yet our model surpasses specialized baselines such as HyDRA, demonstrating superior accuracy and stability even under significant distributional shifts. Taken together, these results confirm that OS-TriadGAN not only achieves state-of-the-art performance in closed-set evaluations but also extends reliably to open-set detection across diverse wireless environments, thereby offering a practical and scalable solution for real-world authentication tasks.

5.6 Conclusion

This work presents the *OS-TriadGAN*, a framework that advances open-set RFFI by combining adversarial feature shaping with principled calibration. The triadic adversarial structure (generator–extractor–discriminator) encourages the learning of transmitter-discriminative and receiver-robust embeddings, while the integration of temperature-scaled MSP and an F1-optimized global threshold provides a simple yet effective decision rule for distinguishing known from unknown devices. Extensive evaluations on the **LoRa** dataset, which emphasizes device-level generalization within the same chipset family, and the **WiSig** `SingleDay` dataset, which stresses multi-receiver variability, demonstrate that the proposed approach consistently achieves high closed-set accuracy while substantially improving open-set detection. The results confirm both the scalability of the framework and its suitability for real-time deployment, supported by its efficient calibration strategy that stabilizes decision boundaries across diverse environments. Moreover, the algebraic equivalence between s_{MSP} and $p_{\max}$ ensures direct comparability with prior work, further strengthening its practical relevance. Looking ahead, future research will investigate threshold-free uncertainty quantification, continual device enrollment, and lightweight domain adaptation techniques, with the aim of further enhancing robustness and ensuring long-term applicability of open-set RFFI in dynamic and heterogeneous wireless ecosystems.

Part II

Network Layer Security

Chapter 6

Graph Anomaly Detection (Catch-All)

Wi-Fi networks are ubiquitous in modern life, yet their continuous technological evolution exposes new vulnerabilities, making effective anomaly detection a critical task. Traditional rule-based or machine-learning approaches that treat traffic data in a tabular format as independent of each other fail to capture complex node-to-node relationships in dynamic wireless environments, thus leading to sub-optimal performance and missed detections. This may cause attacks to go undetected, thus exposing the network to security and privacy threats. In this paper, we propose a novel anomaly detection framework for Wi-Fi networks based on Graph Neural Networks (GNNs) to address the limitations of existing anomaly detection schemes. By modeling the network as a graph where nodes and edges carry features extracted from raw PCAP traces, our method learns representations that encapsulate both local behavior and global context, thus improving detection performance compared to traditional approaches. We implement our approach on three different classic GNNs (i.e., GCN, GAT, GraphSAGE), test them on NS3-simulated datasets, and compare them with non-graph-based machine learning approaches. Our results demonstrate that GNN-based detectors significantly outperform baseline methods in diverse conditions, showing improved recall, precision, and overall accuracy. In addition, we introduce feature- and topology-based adversarial attacks to evaluate the robustness of our framework. These findings underscore the promise of graph-centric approaches for real-world Wi-Fi anomaly detection.

6.1 Introduction

Wireless networks have rapidly become integral to modern communication infrastructures, from home environments to large enterprises and industrial systems. While wireless network protocols based on the IEEE 802.11 (Wi-Fi) technology have evolved significantly in speed and security, inherent vulnerabilities, such as unauthorized access [89], man-in-the-middle attacks [168], and denial-of-service incidents [45], pose persistent risks. These threats underscore the need for robust anomaly detection mechanisms that can pinpoint malicious activities in real-time.

Existing anomaly detection strategies for Wi-Fi networks mainly rely on rule-based, machine learning-based, and deep learning-based approaches. Rule-based methods use static thresholds [170], while machine learning algori2005newthms often operate on hand-crafted features extracted from packet headers or flow-level data. Deep learning-based methods [169] automatically learn feature representations, yet tend to treat traffic in a flat, tabular manner and typically require extensive labeled datasets. These methods disregard the *relational* or *graph-based* nature of

wireless networks, losing the context of *which* nodes are communicating, *how* they are interconnected, and *where* anomalies might propagate. Consequently, some research [106] introduced Graph Neural Network (GNN) in the wireless network scenario to utilize the relationships of each network device and the traffic logs. GNN provides a promising paradigm by directly incorporating the underlying topological structure of communication links. Unlike traditional methods, GNN operate on graph representations where nodes (wireless devices) and edges (direct links or flows) contain rich information. As demonstrated in [222, 201], relational embeddings learned by GNN can reveal anomalies that are otherwise missed when data is treated in isolation.

Our objective is to leverage Packet Capture (PCAP) files, that offer detailed traffic recordings for network operation, exported from Wi-Fi networks to design and validate a robust GNN-based anomaly detection framework. We aim to (i) leverage the topological information inherent in the data and (ii) assess the resilience of the proposed approach against adversarial attacks. In short, our contributions are:

- We design a GNN-based anomaly detection framework for Wi-Fi networks, integrating node and edge features for fine-grained analysis.
- We propose a PCAP traffic data conversion method for Wi-Fi networks, processing them as graph structure datasets.
- We evaluate the performance of GNN-based anomaly detection and demonstrate that our proposed approach remains robust even under adversarial attacks.

The remainder of the study is structured as follows: Section 6.2 summarizes recent related works; Section 6.3 presents our system and threat model; Section 6.4 gives the introduction of our proposed method; Section 6.5 presents experimental results and associated discussions; Section 6.6 provides the conclusion and recommendations for further work.

6.2 Background and Related Work

In this section, we delve into the fundamentals of wireless network protocols and graph neural networks, discussing their core concepts and inherent limitations. Additionally, we explore the various attack methodologies, specifically focusing on feature perturbation attacks and topology attacks.

6.2.1 Graph Neural Network in Wireless Networks

Graph Neural Networks (GNNs) have emerged as one of the most prominent research directions in graph machine learning, offering powerful frameworks for learning from non-Euclidean, relational data structures. Inspired by the success of traditional neural networks, Gori et al. [58] first generalized neural architectures to graph-structured inputs, laying the foundation for subsequent developments that have rapidly attracted attention in both academia and industry. Over the past decade, the GNN family has evolved into two primary categories: *spectral-based methods* and *spatial-based methods*, each characterized by distinct ways of defining graph convolution operations. Spectral GNNs operate in the Fourier domain of the graph, leveraging spectral representations to capture structural information. Bruna et al. [26] pioneered this line of research by introducing convolution operations from

Euclidean domains into graph domains via spectral construction. However, the reliance on eigendecomposition results in prohibitively high computational costs for large-scale graphs, limiting practical deployment. To address this, Defferrard et al. [41] proposed using Chebyshev polynomials to approximate spectral filters, significantly reducing parameterization overhead. Building on this idea, Kipf and Welling [94] introduced the Graph Convolutional Network (GCN), which further simplified the spectral filtering process and established a widely adopted baseline for semi-supervised learning on graphs. In contrast, Spatial GNNs define convolution operations directly in the graph domain by aggregating information from local neighborhoods. Kipf et al. [95] extended the autoencoder paradigm to graphs, leading to the Graph Auto-Encoder framework, which enabled unsupervised representation learning for high-dimensional graph data. Hamilton et al. [65] developed Graph Sample and Aggregate (GraphSAGE), a general inductive framework that learns node embeddings by sampling and aggregating features from local neighbors, making it well-suited for large and dynamic graphs. Veličković et al. [175] further advanced the field by introducing Graph Attention Networks (GAT), which incorporate attention mechanisms to adaptively learn the importance of neighboring nodes, thereby capturing heterogeneous relationships more effectively. More recently, Miao et al. [125] proposed GPINN, which interprets GNNs as representation learning models and enhances embedding utilization through principled graph-aware regularization. The versatility and expressive power of these architectures have made GNNs increasingly attractive for wireless communications, where graph-structured data naturally arise from the interactions among devices, sensors, and network nodes. In this domain, GNN-based methods have been explored extensively for *anomaly detection* and *security applications*. Wu et al. [183] investigated GNN-based approaches for anomaly detection in the Industrial Internet of Things (IIoT), demonstrating their effectiveness in capturing complex inter-device relationships. Blaž et al. [18] proposed transforming univariate wireless measurements such as RSS into graphs via the Visibility Graph (VG) method, enabling the detection of physical-layer anomalies in edge networks. Guo et al. [63] addressed the trade-off between accuracy and energy efficiency in IoT anomaly detection, focusing on graph-level classification of sensor data under resource constraints. Blaž et al. [19] further extended time-series to graph transformations for link-layer anomaly detection in large-scale IoT deployments, surpassing traditional shallow and deep learning baselines. Beyond IoT, Jiang et al. [84] and Tian et al. [171] applied GNNs to smart grid security, successfully identifying stealthy invisible attacks by leveraging the topological structure of the grid.

6.2.2 Adversarial Attacks to GNN

Adversarial attacks are methods designed to deceive machine learning and deep learning models by injecting meticulously designed perturbations into the input data. GNNs are vulnerable to adversarial attacks, with minor alterations in the graph structure [83] or node attributes [31] resulting in substantial performance degradation. Attacks refer to deliberate manipulations of input data aimed at misleading the predictions of a learned model [164]. While initially explored in the context of image processing—where imperceptible pixel perturbations can drastically alter classification outcomes—adversarial attack concepts have since expanded to other domains [203], including graph-structured data. In graph scenarios, attackers often aim to modify the graph topology [99, 216] (e.g., adding or removing edges) or

node features (e.g., tampering with attribute values) to induce erroneous predictions from an GNN.

Inspired by the existing works, the following are adversarial threats we performed for GNNs:

- **Feature Perturbation.** Modifying or disguising pivotal node features can distort the learned embeddings within the network. An attacker could manipulate traffic statistics or encryption-type attributes to make a malicious node appear more like a legitimate one, thus evading detection.
- **Topology Attacks.** Through the insertion of new edges or the deletion of critical links, adversaries disrupt the neighborhood aggregation process. For instance, by linking a malicious node to key highly influential hubs, an attacker might increase the node's significance and conceal any anomalous behavior.

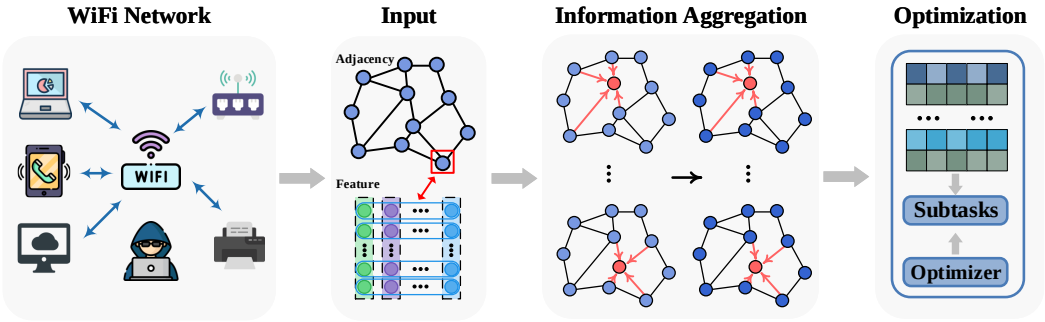


FIGURE 6.1: The framework of data collecting and model training.

6.3 System and Threat Model

6.3.1 Network Model

Wi-Fi is widely deployed in consumer and enterprise environments, driving numerous studies on more realistic simulation approaches. Nekovee et al. [130] used specialized data structures (adapted from molecular dynamics) to handle frequent topology changes in large-scale ad hoc networks. Michaloliakos et al. [126] focused on enterprise-scale setups with dense Access Point (AP), introducing analytical frameworks for channelization, power allocation, and scheduling. Fontes et al. [50] proposed Mininet-Wi-Fi, an SDN emulation tool that leverages lightweight virtualization for flexible prototyping.

Below is a brief overview of the standard node types simulated in our Wi-Fi networks:

- **AP**: Central hub in Infrastructure Mode, managing authentication, association, and channel utilization.
- **Routers**: Often integrated with AP functionality, provide routing and firewall/packet-filtering rules to connect local networks with external ones.
- Ad-Hoc Nodes: Operate without a centralized AP, facilitating direct peer-to-peer communication.

- **Stationary Devices**: Typically fixed in place (e.g., security cameras, Internet of Things sensors), yielding predictable traffic patterns but vulnerable if poorly secured.
- **Mobile Nodes**: Roaming devices (e.g., smartphones) that dynamically update parameters such as channels and encryption based on signal conditions.
- Wireless Bridges/Repeaters: Extend coverage by forwarding traffic between separated segments, introducing extra hops in the network topology.

6.3.2 Threat Model

To evaluate our GNN-based anomaly detection framework, we consider attackers delivering adversarial attacks to the dataset's core components. We consider two key adversarial attack strategies: feature perturbation, and topology attacks. Feature perturbations mimic real-world interference, where subtle modifications to node attributes can disguise malicious behavior, and topology attacks disrupt the inherent graph structure by altering edge connectivity. In particular: For **feature perturbation attacks**, in Wi-Fi networks, let $\mathbf{x}_i$ denote the original feature vector of node i. An attacker injects low-level interference by adding a perturbation δ_i to form a modified feature vector:

$$\mathbf{x}_i' = \mathbf{x}_i + \delta_i, \quad \text{with} \quad \|\delta_i\| \leq \epsilon, \tag{6.1}$$

where ϵ is the maximum allowable perturbation (attack budget). These subtle alterations in transmission metrics (e.g., signal strength, delay, or error rate) result in PCAP logs that record minor deviations, making a malicious node appear similar to a benign one [31, 164].

For **topology attacks**, let A denote the original adjacency matrix of the network graph. The attacker can modify the network connectivity by applying a perturbation matrix Δ to obtain a new adjacency matrix:

$$A' = A + \Delta, \tag{6.2}$$

where each entry in Δ indicates the insertion $(+1)$ or deletion (-1) of an edge between nodes. In practice, this can be achieved through MAC spoofing or exploiting protocol vulnerabilities to induce unauthorized connections or disconnections, thus disrupting the neighborhood aggregation process fundamental to GNN [83].

To quantify the robustness of our proposed approach, we systematically introduce these adversarial modifications into our network and evaluate the resulting performance degradation across various GNN architectures.

6.4 Our Proposed GNN Anomaly Detector

6.4.1 Network Graph Representation

In this paper, we apply the *graph-centric approach* for Wi-Fi anomaly detection to address the current limitation of existing methods that fail to harness the topology of the entire network. We define the wireless network as the attributed graph:

$$\mathcal{G} = (\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}}, \boldsymbol{X}), \tag{6.3}$$

where $\boldsymbol{\mathcal{V}} = \{v_1, v_2, \ldots, v_N\}$ is the set of nodes (e.g., smartphones, APs), $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ is the set of undirected edges that capture communication patterns or events, (v_i, v_j)

represents an edge between $node_i$ and $node_j$, $\boldsymbol{X} \in \mathbb{R}^{N \times d}$ is the feature matrix, in which each row $\boldsymbol{X}_i \in \mathbb{R}^d$ represents a d-dimensional feature vector for node v_i.

We focus on a semi-supervised node classification task on an unweighted, undirected graph. An unweighted graph has each edge indicating a connection without any associated strength or cost, thereby treating every link equally. Moreover, the graph is undirected, i.e., the relationships between nodes are reciprocal, with no inherent directionality. Notably, each node in our graph represents a Wi-Fi device (e.g., **APs, stationary node, jammer, and mobile node**). Its adjacency matrix is $\boldsymbol{A} \in \{0,1\}^{N \times N}$, where $\boldsymbol{A}_{ij} = 1$ if and only if $(v_i, v_j) \in \boldsymbol{\mathcal{E}}$ and $\boldsymbol{A}_{ij} = 0$ otherwise. This unified graph representation, including both topology (i.e., $\boldsymbol{A}$) and features (i.e., $\boldsymbol{X}$), This enables convenient modeling of the wireless network and supports effective anomaly detection.

6.4.2 Framework

In contrast to the techniques discussed in Section 6.2.1, our approach integrates a GNN-based anomaly detection mechanism at the node level with a traffic conversion scheme (e.g., from PCAP files). Instead of relying on sensor data or physical measurements, we focus on traffic logs and employ multivariate node features rather than univariate information.

An overview of the proposed GNN-based Wi-Fi anomaly detection framework is provided in Figure 6.1, which highlights the key components and their interactions within the system. The diagram depicts a variety of wireless-enabled devices, such as laptops, smartphones, and desktop computers, all connected to a central router or AP. Some nodes of these devices are highlighted as a potential security threat. The interactions between these devices, both legitimate and malicious, are represented as edges that together form the network's topology. The network's structure and the attributes of its nodes and edges, including traffic statistics, are encoded into two primary inputs: an *adjacency matrix* that contains node connections of the network and a *feature matrix* that contains various attributes such as packet counts and signal strength. The GNN processes these matrices, aggregating and learning from neighboring nodes' data and ultimately producing a *label* that classifies each node as either anomalous or benign. This comprehensive visualization shows how raw network data is transformed into a graph format suitable for GNN analysis and inference.

6.4.3 Graph-based Dataset Construction

We can gather raw network traffic data from Wi-Fi networks using monitoring utilities such as TCPDUMP or simulation toolkits. To process the collected data and use it in our GNN-based solution, we perform the following operations. However, the specific processes of the dataset construction will be described in Section 6.5.1.

- **Filtering:** Exclude non-essential network activities (e.g., broadcast, multicast) from the traffic.
- **Feature Extraction:** Compute features such as packet count, energy consumption, or time-based statistics, and record the node links via traffic logs.
- **Graph Construction:** Collect the final dataset in an adjacency matrix and a feature matrix, ensuring that node connection relation and node features are aligned.

The adjacency and feature matrices are then input into the GNNs for classification purposes. The adjacency matrix is a vital component of the dataset, as it delineates the connectivity between each node within the wireless network architecture.

6.4.4 The GNN-based Anomaly Detection

We adopt GCN, GAT, and GraphSAGE as GNNs to classify each node (or set of nodes) as normal or anomalous. GNNs generally follow the "propagation-transformation" paradigm to iteratively fuse neighbor node features as:

$$H_t^{(l)} \leftarrow \text{Transform}\left(\underset{\forall v_s \in \mathcal{N}_t}{\text{Propagate}}\left(H_s^{(l-1)}; H_t^{(l-1)}\right)\right), \tag{6.4}$$

where $H_t^{(l)}$ is the embedding of node v_t at layer l; $\mathcal{N}_t$ denotes the neighbors of node v_t; $H_s^{(l-1)}$ is the embedding of a neighbor v_s at layer $l-1$; and $H_t^{(l-1)}$ is the embedding of node v_t at the previous layer. Each node aggregates neighbor features and applies a transformation to update its embedding.

$$\textbf{Propagate}: M_t^{(l)} = \hat{A} H_t^{(l-1)}, \tag{6.5}$$

where $M_t^{(l)}$ represents the aggregated message for node v_t at layer l, $\hat{A}$ is the normalized adjacency matrix capturing the graph structure. Neighbor features are aggregated via multiplication with the normalized adjacency matrix $\hat{A}$.

$$\textbf{Transform}: H_t^{(l)} = \text{ReLU}\left(M_t^{(l)} W^{(l)}\right), \tag{6.6}$$

where $W^{(l)}$ is the learnable weight matrix at layer l, and $\text{ReLU}(\cdot)$ is the activation function that introduces non-linearity, mapping the transformed message $M_t^{(l)} W^{(l)}$ to the new embedding $H_t^{(l)}$. The aggregated message is linearly transformed and activated by ReLU for non-linear feature mapping.

$$\textbf{Loss}: \mathcal{L} = -\sum_{v \in V^L} y_v \log \hat{y}_v, \tag{6.7}$$

where $\mathcal{L}$ denotes the cross-entropy loss over the set V^L of labeled nodes; y_v is the ground truth label for node v; and $\hat{y}_v$ is the predicted probability for node v. The cross-entropy loss quantifies the prediction error over labeled nodes V^L.

The basic training scheme of the proposed GNN is illustrated in Figure 6.1. The framework follows a structured pipeline consisting of three main stages: input preparation, information aggregation, and optimization. First, an adjacency matrix and node feature matrix are extracted from the input graph, which encodes the graph topology, while the feature matrices capture attributes associated with each node. In the second stage—information aggregation—each node iteratively aggregates features from its neighbors. Integrating information along edges in multiple rounds makes these local embeddings more expressive representations of the global graph structure. Finally, the graph embeddings are fed into the optimization phase, where an objective function is minimized using gradient-based methods according to the specific downstream tasks. Specifically, this study leverages a GNN-based anomaly detection framework as a classifier to discriminate among various node types, including stationary devices, APs, mobile nodes, and malicious nodes. The

proposed approach systematically integrates both topological and feature information inherent in the network, thereby enabling a robust differentiation between benign and adversarial entities. This classification is critical for maintaining network integrity and ensuring the effective identification of anomalous behaviors within heterogeneous communication environments.

6.5 Numerical Results

This section describes the dataset used in the experiments and the relevant implement and the evaluation metrics. To confirm the abilities of the proposed GNN-based detectors, we tested three GNN-like models, e.g., GCN, GAT, GraphSAGE, and MLP in this paper. We introduced the GNNs to detect the anomaly nodes by using the traffic in a Wi-Fi network system and proposed a scheme to convert the PCAP files to graph-format datasets for training models. Other than in traditional approaches, if we want to apply it to Wi-Fi node anomaly detection, the mechanism is different from the GNNs scheme, so in order to confirm the abilities of the proposed GNN-based detectors, we trained and tested three GNN-like models, e.g., GCN, GAT, GraphSAGE, and MLP in this paper. We also showed the embedding distribution visualization illustrations of various experiment settings, alongside this section; in addition to the exhibition of the results, the embedding distributions are shown as well to an assist in understanding the models.

6.5.1 Dataset Construction

In this section, we describe the network simulation in Section 6.5.1, then we convert the PCAP traffic logs to graph-like data to train the GNN anomaly detection.

NS3 Simulation

NS3 [176] is a discrete-event network simulator for internet systems. We evaluate our proposed method on simulation datasets generated via NS3 simulation. Based on the description in 6.3.1, we simulate four node types—APs, stationary nodes, mobile nodes, and a jammer node (i.e., as an anomaly node)—in Wi-Fi networks, where the jammer interferes with normal communications.

The architecture of the simulator in the NS3 simulation environment is depicted in Figure 6.2. On the left-most part, the *Core* modules (e.g., ns3::Time, ns3::Scheduler, ns3::Simulator, ns3::Config) provide fundamental functions for discrete-event scheduling and configuration. These components collectively define how simulation events are initialized, orchestrated, and executed over simulated time. In the middle part, network functionalities such as *Wi-Fi*, *Energy*, and *Internet Applications* are assembled to construct a representative wireless scenario—e.g., setting up APs, defining energy consumption profiles, and generating network activities. On the right-most part, the *Trace* and *Monitor* modules enable detailed logging of packet-level data, link-layer events, and energy usage, ultimately producing traffic logs for post-processing or offline analysis. This architecture allows NS3 to simulate various wired and wireless networks while capturing essential performance metrics for thorough experimentation.

To facilitate controlled experimentation, we employ an *NS3*-based simulation environment that allows us to precisely configure the Wi-Fi network topology, traffic

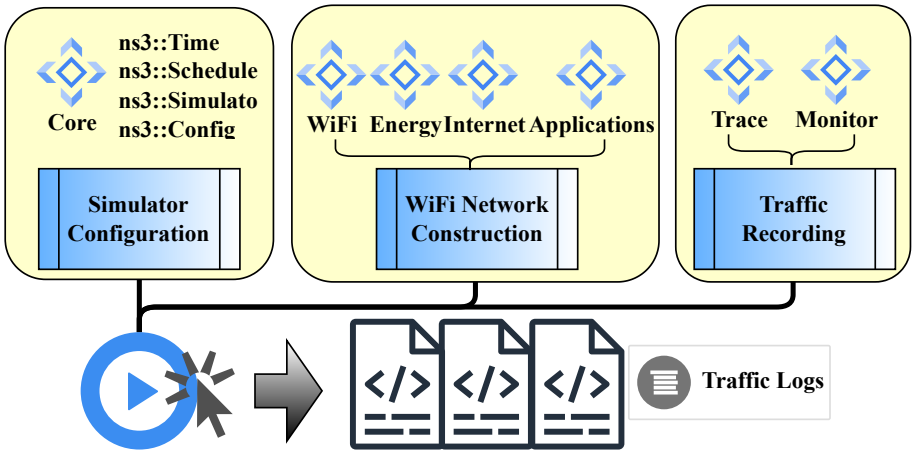


FIGURE 6.2: NS3-simulation introduction.

patterns, and node density. By leveraging NS3, we can systematically inject anomalies at predetermined intervals, yielding a labeled dataset suitable for both training and evaluation of our anomaly detection framework.

By modeling these diverse activities in a graph representation, each node's role and connection patterns become more transparent. This richer relational structure is highly beneficial for anomaly detection, as it captures both typical communication flows and potential deviations introduced by malicious actors or compromised devices.

Using this structured simulation approach, we can label each traffic flow or node activity based on whether it exhibits normal or anomalous behavior. These labels serve as ground truth for training and benchmarking our Graph Neural Network models, ensuring robust performance measurements in both benign and adversarial contexts.

Threat Module Implementation

To introduce anomalies into the generated dataset, we designed a malicious node type and deployed it on selected nodes within the Wi-Fi network. These nodes were configured to perform background Denial of Service (DoS) attacks under predefined parameters, thereby disrupting normal communications by inducing packet loss, delays, and channel congestion. The resulting interference forced legitimate nodes to retransmit packets, which increased traffic load and further reduced overall reliability. These adverse conditions were captured in the traffic logs, where anomalies manifested as irregular packet inter-arrival times, frequent retransmissions, elevated packet loss rates, and abnormal communication patterns. Such features serve both as indicators of potential security breaches and as valuable evidence for analyzing the impact of malicious activity on network performance.

At the implementation level, we employed several NS3 modules (e.g., *Core*, *Wi-Fi*, *Internet*, and *Applications*), as depicted in Figure 6.2. The *Core* module instantiated basic Wi-Fi devices, the *Wi-Fi* module configured network parameters for each node, and the *Internet* module established communication protocols and topology. Finally, the *Applications* module defined and executed the malicious behaviors of the attack nodes, enabling controlled injection of anomalies into the dataset.

PCAP File Properties

The datasets used in this project are derived from PCAP files, which provide a comprehensive record of network traffic at the packet level. Each packet in a PCAP file typically includes:

- Timestamps: Precise timing information indicating when each packet was captured, facilitating chronological analysis and event correlation.
- Link-Layer Headers: Low-level data such as MAC addresses (source/destination) and Wi-Fi/Ethernet fields, crucial for identifying device-level interactions.
- Network-Layer Headers: IP addresses, protocols (TCP, UDP, ICMP), and QoS markings, enabling higher-level analysis of routes, flows, and anomalies.
- Transport-Layer Headers: Port numbers, sequence numbers, and TCP/UDP flags (e.g., SYN, ACK), providing insight into connection states and handshakes.
- Payload Data (Optional): Packet contents may be recorded, aiding deep packet inspection but raising privacy concerns; partial payloads often suffice.
- Metadata for Wireless Traffic: Additional Wi-Fi attributes (channel frequency, RSSI, data rates) offering deeper context for diagnosing wireless-specific issues.

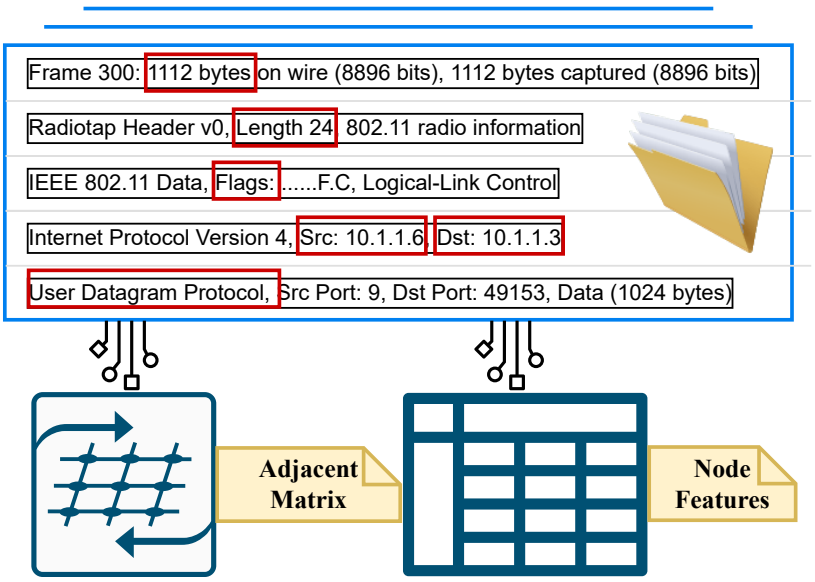


FIGURE 6.3: PCAP conversion.

As shown in Figure 6.3, it demonstrates how raw packet captures (PCAP files) can be transformed into graph-structured data. By inspecting source (e.g., Src:10.1.1.6) and destination (e.g., 10.1.1.3) fields, one can identify relationships (edges) between nodes in the network and construct an *adjacency matrix*, where each unique device/IP address becomes a node. Meanwhile, node features may include

aggregated statistics derived from the captured frames, such as total bytes, packet count, or protocol distribution (e.g., 1112 bytes, length 24, Flags, User Datagram Protocol, and so on); these could be the relevant features to improve the quality of graph-like datasets. This conversion process allows packet-level information to be encoded in a *graph form*, enabling graph-based models (e.g., GNNs) to analyze network traffic and detect anomalies or other complex patterns.

As introduced above, such granularity enables both *feature-level* and *structural* analyses of the network. When transformed into graph representations, the link-layer and IP-layer information become node identifiers and edge relationships, while header fields and derived statistics form node or edge features for the anomaly detection pipeline.

6.5.2 Experimental Results Analysis

TABLE 6.1: Precision of different models under Balanced and Imbalanced datasets.

Model	Balanced					Imbalanced				
	50	100	250	500	1000	50	100	250	500	1000
MLP	0.28	0.34	0.28	0.34	0.34	0.20	0.27	0.10	0.27	0.15
GCN	0.30	0.52	0.42	0.45	0.43	0.42	0.49	0.44	0.35	0.37
GAT	0.50	0.55	0.41	0.50	0.48	0.56	0.53	0.57	0.39	0.27
GraphSAGE	0.88	0.83	0.92	0.84	0.85	0.83	0.88	0.94	0.77	0.56

TABLE 6.2: Recall of different models under Balanced and Imbalanced datasets.

Model	Balanced					Imbalanced				
	50	100	250	500	1000	50	100	250	500	1000
MLP	0.35	0.45	0.33	0.44	0.32	0.33	0.40	0.13	0.42	0.25
GCN	0.44	0.44	0.41	0.50	0.34	0.29	0.48	0.38	0.31	0.27
GAT	0.40	0.48	0.34	0.37	0.44	0.50	0.48	0.46	0.27	0.27
GraphSAGE	0.83	0.75	0.91	0.71	0.86	0.89	0.92	0.93	0.78	0.60

TABLE 6.3: F1-score of different models under Balanced and Imbalanced datasets.

Model	Balanced					Imbalanced				
	50	100	250	500	1000	50	100	250	500	1000
MLP	0.20	0.36	0.21	0.36	0.25	0.25	0.30	0.12	0.29	0.19
GCN	0.35	0.42	0.39	0.46	0.30	0.33	0.45	0.39	0.33	0.23
GAT	0.37	0.45	0.34	0.32	0.42	0.49	0.46	0.47	0.22	0.23
GraphSAGE	0.79	0.73	0.91	0.74	0.84	0.82	0.87	0.93	0.77	0.58

TABLE 6.4: Accuracy of different models under Balanced and Imbalanced datasets.

Model	Balanced					Imbalanced				
	50	100	250	500	1000	50	100	250	500	1000
MLP	0.30	0.50	0.34	0.53	0.40	0.60	0.50	0.56	0.50	0.52
GCN	0.60	0.55	0.58	0.52	0.43	0.50	0.65	0.58	0.56	0.62
GAT	0.50	0.60	0.56	0.38	0.48	0.70	0.65	0.66	0.57	0.61
GraphSAGE	0.80	0.80	0.94	0.81	0.85	0.90	0.90	0.96	0.88	0.86

Xu et al. [192] present both theoretical and empirical evidence of the node classification power of GNN. In contrast to Multilayer Perceptron (MLP), which processes each node independently, and Convolutional Neural Network (CNN), which relies on local spatial patterns in grid-like data, GNNs recursively aggregate feature information from adjacent nodes. This adjacency-aware approach captures both node attributes and their relational dependencies, enabling better performance on tasks such as node classification, graph classification, and link prediction, where topology is crucial. Therefore, we evaluate three GNN-based architectures—GCN, GAT, and GraphSAGE—with a standard MLP serving as the baseline comparison.

We consider the following: True Positives (TP), positive cases correctly predicted as positive; False Positives (FP), negative cases incorrectly predicted as positive; False Negatives (FN), positive cases incorrectly predicted as negative; True Negatives (TN). negative cases are correctly predicted as negative. Based on these, we define the following metrics:

- **Precision (Prec.)**: $\text{Prec.} = \text{TP}/(\text{TP}+\text{FP})$;
- **Recall (Rec.)**: $\text{Rec.} = \text{TP}/(\text{TP}+\text{FN})$;
- **F1-Score (F1)**: $\text{F1} = 2 \times (\text{Prec.} \cdot \text{Rec.})/(\text{Prec.} + \text{Rec.})$;
- **Accuracy (Acc.)**: $\text{Acc.} = (\text{TP}+\text{TN})/(\text{TP}+\text{TN}+\text{FP}+\text{FN})$.

In Table 6.1, Table 6.2, Table 6.3, and Table 6.4, we summarize the performance of four different models (**MLP**, **GCN**, **GAT**, **GraphSAGE**) across datasets of various sizes and balance conditions. The datasets are labeled as Balanced, when classes have roughly equal representation; Imbalanced, when half of the nodes are stationary devices. Each column (e.g., *(Balanced, 50)*, *(Imbalanced, 1000)*) indicates both the class distribution in the datasets and the number of nodes.

Overall, incorporating graph structure via GCN, GAT, or GraphSAGE leads to more robust performance than MLP models, especially on large-scale or balanced data. However, imbalanced scenarios remain challenging, as reflected by dips in Precision and Recall.

GNN with different dataset scale

As the node count increases (from 50 to 1000), we observe that graph-based models (*GCN, GAT, GraphSAGE*) typically benefit from richer connectivity. In a larger graph, nodes have more neighbors or deeper neighborhoods, providing these models with additional context for their message-passing processes.

- **MLP's Lower Performance at Larger Scales:** Because MLP lacks explicit graph awareness, it must rely solely on the raw features of each instance, disregarding any relational structure. This often restricts its ability to discriminate complex patterns and causes lower precision, recall, and F1 when compared to graph-based methods.
- **GraphSAGE's Advantage in Large Graphs:** GraphSAGE samples local neighborhoods and aggregates information more flexibly than standard GCN, which can help avoid over-smoothing in deeper neighborhoods. As a result, GraphSAGE tends to maintain stronger performance as the graph grows in size.

In a wireless network comprising 50 nodes, GraphSAGE achieved superior performance compared to the other models, whereas the MLP, GCN, and GAT all exhibited accuracies below 60%, rendering them inadequate as reliable anomaly detectors. Notably, as the dataset's size increased, GCN and GraphSAGE models' performance remained stable performances, indicating that larger network topologies inherently provide richer feature representations that facilitate more accurate anomaly detection. In general, the larger the network, the easier it becomes to train a robust detector using graph-based methods.

GNN using different label distribution

In a balanced class distribution dataset, we want to discuss the effects of the imbalanced dataset (e.g., stationary devices, AP, mobile stations, and malicious nodes account for 60%, 20%, 10% and 10% respectively). The model should learn the anomaly nodes' behavior features from small part nodes. Due to this limitation, the model commonly got the worst results compared to the balanced label dataset with the same graph size. However, the GraphSAGE is better at these scenarios than others.

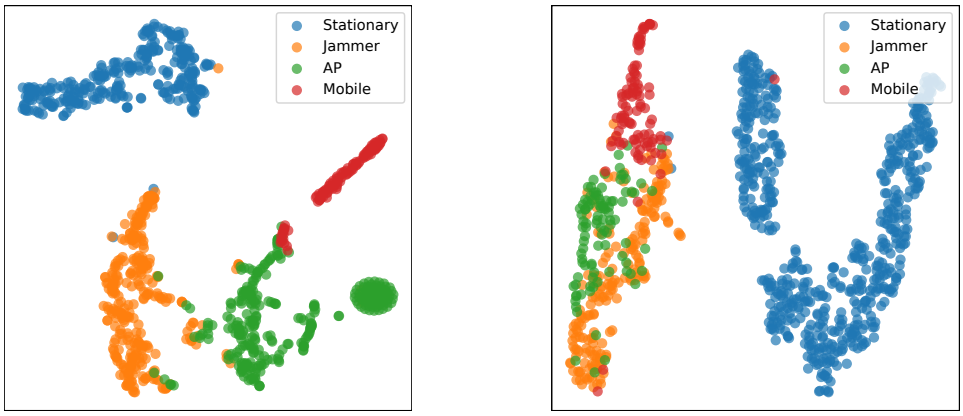
For the *balanced* datasets, each class is roughly equally represented, making it easier for models to learn discriminative features across all classes. This explains the higher F1 and accuracy values, particularly for GraphSAGE and GAT, which effectively propagate label information through the graph structure. By contrast, the *Imbalanced* datasets show reduced recall and precision in most models, indicating challenges in correctly identifying minority classes:

- **Minority Class Overlooked:** When one or two classes dominate, models can inflate their *accuracy* by predicting mostly the majority class. However, such behavior leads to lower *precision* and *recall* for the underrepresented classes, thus reducing the overall F1 score.
- **Accuracy vs. Precision/Recall/F1:** It is common to see higher accuracy (e.g., correctly classifying many majority-class samples) alongside lower precision/recall/F1 scores. Accuracy measures the overall rate of correct predictions, but precision, recall, and F1 highlight how well each class is handled. In an imbalanced setting, a model may achieve moderate or high accuracy by focusing on the dominant class, thereby neglecting the minority class and driving down the class-specific metrics.

As shown in Figure 6.4, the distribution of extracted node features for different label distribution datasets with 1000 nodes is depicted; the different color scatters stand for **APs, jammer, stationary node, and mobile node**, respectively. These visualizations further support the findings in this section, suggesting that an imbalanced dataset offers more majority-class feature information, thereby making the anomaly detectors recognize the class difficulty. As Figure 6.4a showed, the border of each class is more apparent than in Figure 6.4b.

6.5.3 Adversarial Attack Robustness Evaluation

In this section, we evaluate how different adversarial attacks affect our model's performance. We systematically modify node features or edge connections and compare classification metrics before and after the attack. To rigorously assess the robustness of our proposed GNN-based anomaly detection framework, we introduce controlled adversarial attacks in Section 6.2.2 that mirror real-world attack strategies.



(A) Balanced (B) Imbalanced

FIGURE 6.4: Embedding distribution.

TABLE 6.5: Precision under feature perturbation and topology attack.

Model	250			500			1000		
	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.
MLP	0.32	0.34	–	0.30	0.31	–	0.34	0.34	–
GCN	0.34	0.34	0.50	0.39	0.40	0.40	0.44	0.43	0.45
GAT	0.55	0.36	0.39	0.59	0.49	0.36	0.45	0.48	0.36
GraphSAGE	0.83	0.76	0.77	0.87	0.81	0.83	0.85	0.82	0.81

TABLE 6.6: Recall under feature perturbation and topology attack.

Model	250			500			1000		
	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.
MLP	0.39	0.32	–	0.41	0.44	–	0.38	0.35	–
GCN	0.36	0.37	0.42	0.40	0.41	0.45	0.39	0.35	0.38
GAT	0.54	0.37	0.36	0.46	0.48	0.37	0.41	0.38	0.40
GraphSAGE	0.83	0.81	0.74	0.86	0.81	0.78	0.81	0.83	0.77

TABLE 6.7: F1-score under feature perturbation and topology attack.

Model	250			500			1000		
	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.
MLP	0.29	0.24	–	0.32	0.32	–	0.30	0.28	–
GCN	0.33	0.32	0.42	0.38	0.37	0.42	0.36	0.33	0.35
GAT	0.52	0.34	0.74	0.46	0.45	0.35	0.37	0.37	0.38
GraphSAGE	0.82	0.77	0.74	0.86	0.81	0.80	0.82	0.82	0.78

TABLE 6.8: Accuracy under feature perturbation and topology attack.

Model	250			500			1000		
	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.	Origin	Perturb.	Topo.
MLP	0.46	0.40	–	0.44	0.35	–	0.46	0.43	–
GCN	0.50	0.42	0.44	0.51	0.47	0.48	0.49	0.46	0.45
GAT	0.54	0.50	0.46	0.51	0.48	0.47	0.50	0.49	0.48
GraphSAGE	0.88	0.82	0.80	0.84	0.79	0.80	0.83	0.81	0.79

By evaluating performance in these adversarially crafted conditions, we understand the model's tolerance to malicious interference. Furthermore, comparative

analyses under varying degrees of perturbation severity highlight the GNN's capacity to maintain accurate anomaly detection outcomes even under non-ideal experimental environments.

In Table 6.5, Table 6.6, Table 6.7, and Table 6.8, each row refers to a method used to experiment, while each column stands for a graph size (e.g., 250, 500, 1000 nodes) with different modifications, and the Tables report Precision, Recall, F1-score, and Accuracy, respectively. We compares four models (MLP, GCN, GAT, GraphSAGE) under three dataset variations:

- **Origin**: The unmodified baseline dataset.
- **Perturb**: A version subject to feature-based attacks (e.g., noise or adversarial perturbations on node attributes).
- **Topo**: A dataset compromised by graph-structure manipulation (e.g., edge insertions/removals).

A key finding from our experiments concerns the varying sensitivities of models to feature perturbations and topology attacks. When node features are altered (feature perturbation), an MLP, which relies exclusively on node features and does not incorporate graph structure, exhibits a more pronounced drop in F1 and accuracy than graph-based models. In particular, GraphSAGE, due to its neighborhood-sampling strategy, remains comparatively robust under moderate feature perturbations, preserving higher precision and recall in most of these settings.

In contrast, attacks involving the removal or addition of edges (topology attacks) tend to result in performance degradation for models such as GCN and GAT, both of which rely on propagating information through connected neighbors. By selectively disrupting edges, an adversary can undermine the neighborhood aggregation process, leading to notable declines in precision, recall, and thus F1. Interestingly, because an MLP is agnostic to graph connectivity, its performance generally deteriorates less under topology attacks relative to GCN or GAT.

Comparing model-specific trends, a MLP typically under-performs on the original dataset (i.e., in the absence of attacks) since it cannot leverage the graph structure. Under topology attacks, however, a MLP experiences a smaller performance drop than its graph-based counterparts, reflecting its independence from edge connectivity. GCN and GAT often yield higher performance in benign conditions but exhibit sharper declines when adversaries manipulate the topology. GraphSAGE, on the other hand, delivers relatively stable results across varying dataset sizes and under both feature and topology-based attacks, although it still suffers if the topology perturbations become severe.

Lastly, the size of the graph strongly influences these outcomes. In larger networks (e.g., 1000 nodes), GCN, GAT, and GraphSAGE have access to more abundant contextual information, alleviating some of the adverse effects of feature and topology manipulations. Conversely, in smaller graphs (e.g., 250 nodes), even minimal disruptions—such as the removal of key edges or corruption of a small set of crucial features—can considerably degrade overall performance.

Overall, these results illustrate that **feature perturbations** have a greater impact on MLP, while **topology attacks** can significantly degrade the performance of graph-based methods like GCN and GAT. GraphSAGE exhibits relatively balanced robustness in both scenarios, particularly in larger graphs.

6.6 Conclusion

In this study, we introduced a GNN-based anomaly detection pipeline tailored for Wi-Fi networks. By converting raw PCAP traffic logs into graph data, the GNNs leverage both node-level and edge-level attributes. Our experiments show that models like GCN, GAT, and GraphSAGE all benefit from larger, more balanced datasets, where more affluent node neighborhoods lead to more accurate embeddings. However, even in smaller or imbalanced settings, GNN-based methods still outperform a standard MLP baseline that ignores network topology. When subjected to adversarial attacks that modify node features or topologies, GNNs experience varying degrees of performance degradation; nonetheless,GraphSAGE consistently demonstrates stronger resilience.

Despite these promising results, several avenues remain for future work. Additional real-world datasets would further validate the robustness and scalability of the framework. Second, exploring advanced defense strategies against adversarial edge or feature modifications can strengthen the model's reliability. Finally, extending the scope to more complex scenarios by incorporating enriched edge-level features could further enhance the effectiveness of GNN-based detection methods.

Chapter 7

GAN-based Data Augmentation (GANSec)

Data augmentation techniques show potential in various domains, yet their application to enhance robustness in wireless anomaly detection remains underexplored. Wireless datasets often suffer from anomaly scarcity and class imbalance, hindering the training of reliable detection models. This work introduces GANSec, a novel conditional Generative Adversarial Networks (GAN) framework specifically designed to augment wireless time-series data. We investigate different neural network architectures (MLP, LSTM, CNN) and two conditional training objectives (Embedded Conditional, Classification Oriented) within GANSec, evaluating the framework using real-world 5G measurements for jamming anomaly detection. For evaluation, we train the downstream anomaly detector exclusively on GANSec-generated data and test its performance in a cross-scenario setting. Our evaluation demonstrates that models trained this way significantly outperform those trained on original or baseline augmentation data when tested under unseen network conditions. Specifically, our approach achieved up to 92.13% accuracy on the unseen dataset (i.e., data collected from a different distribution reflecting network conditions distinct from the training set), compared to 78% for models trained on raw data and 83.33% for the best performing baseline, exhibiting substantially enhanced robustness and generalization.

7.1 Introduction

Wireless networks form the backbone of modern communication, supporting applications from personal devices to critical infrastructure, making their security paramount [181, 91]. However, the open nature of wireless communication inherently leaves these networks vulnerable to a diverse array of security threats and anomalies [4], including sophisticated attacks targeting the physical layer, such as jamming [167]. Anomaly detection—the identification of significant deviations from normal behavior—is therefore pivotal for safeguarding wireless environments against security breaches, network malfunctions, and other detrimental activities [189, 150, 204]. Accurate and early identification of these irregularities is crucial for maintaining the security, performance, and reliability of wireless services. This need becomes even more critical as reliance on wireless connectivity escalates across sectors like the Internet of Things (IoT), which introduces numerous potential entry points for malicious actors [181, 91, 189]. Robust anomaly detection mechanisms, capable of adapting to the dynamic nature of wireless channels, represent a critical defense against this expanding threat landscape.

However, developing effective anomaly detection models for wireless security faces significant data-related hurdles, including anomaly scarcity, class imbalance, environmental non-stationarity, and the high cost of data acquisition. These challenges, which impede model training and generalization, are discussed in detail in Section 7.2.

Data augmentation emerges as a potential strategy, with Generative Adversarial Networks (GAN) offering a sophisticated approach compared to simpler techniques like SMOTE [34] and its variants [39]. GAN have been investigated for augmenting network and wireless signal data in security contexts [4, 16, 93, 93, 180, 181]. However, significant limitations persist in prior work, particularly concerning the lack of rigorous validation for robustness and generalization in diverse wireless conditions, and often limited control in generation. These challenges are further elaborated in Section 7.3.3.

In this work, we propose GANSec to address these challenges: a novel conditional GAN framework specifically designed and tailored for augmenting wireless time-series data. Within GANSec, we investigate different neural network backbones (MLP, LSTM, CNN) and conditional training objectives (Embedded Conditional, Classification-Oriented). To assess GANSec's effectiveness, we perform a challenging cross-scenario evaluation tackling jamming anomalies in 5G networks. In this evaluation (Section 7.5), we train the downstream anomaly detection model using GANSec-generated data and test its performance on unseen data from different network conditions. Our results demonstrate that models trained using GANSec significantly outperform baseline methods when tested on this unseen data, showcasing enhanced robustness and generalization capabilities.

This work makes the following contributions to robust wireless anomaly detection using generative augmentation:

- **Novel Conditional GAN Framework (GANSec):** Designed and implemented GANSec, tailored for augmenting labeled wireless time-series data to generate high-fidelity samples.
- **Architectural & Objective Impact Analysis:** Systematically analyzed the influence of backbones (MLP, LSTM, CNN) and conditional objectives (EC, CO) within GANSec on data quality and downstream performance.
- **Demonstrated Robustness & Generalization Enhancement:** Showcased via cross-scenario 5G evaluations that GANSec significantly improves detector robustness against distribution shifts, outperforming baselines.

The remainder of this chapter is organized as follows. Section 7.2 motivates the need for generative augmentation in wireless anomaly detection, focusing on data scarcity, quality limitations, and robustness to distribution shifts. Section 7.3 reviews prior work on anomaly detection and generative augmentation for wireless data, highlighting open gaps. Section 7.4 presents the proposed *GANSec* framework, detailing the architecture and training mechanisms (Embedded Conditional and Classification-Oriented), together with the wireless-specific design choices and the synthetic data generation strategy. Section 7.5 describes the experimental setup (datasets, baselines, and downstream detector) and reports results on synthetic data quality and cross-scenario detection performance. Section 7.6 discusses implications, limitations, and avenues for extension. Finally, Section 7.7 concludes the chapter.

7.2 Motivation

The pursuit of robust anomaly detection in wireless security faces significant hurdles related to data acquisition and quality, motivating the development of sophisticated data augmentation techniques. We argue that GANS offer a particularly compelling approach to overcome these challenges and enhance detection capabilities, driven by the following key factors:

7.2.1 The Prohibitive Cost of Wireless Data Acquisition

A significant practical barrier hinders the development of robust wireless anomaly detection models: acquiring large-scale, diverse, and representative real-world data. Collecting comprehensive datasets proves particularly challenging, especially for capturing sufficient instances of the rare anomalies crucial for security monitoring, often demanding prohibitive investments in time and resources [97, 138]. Standard measurement intervals dictate that accumulating datasets adequate for training complex machine learning models can require days, weeks, or even months of continuous monitoring per location [143]. This process frequently requires deploying specialized hardware and software, incurring substantial capital and operational costs [37, 97, 138]. Furthermore, operating this measurement equipment continuously over extended periods consumes considerable energy, adding to the overall expense and environmental footprint of large-scale data acquisition campaigns [8]. Capturing high-quality data within dynamic wireless environments, subject to fluctuating channel conditions like multipath fading and interference, adds considerable complexity. This inherent difficulty in obtaining sufficient real-world data, especially for infrequent anomaly events, directly impedes the development and reliable training of robust anomaly detection models. Consequently, cost-effective and scalable alternatives are needed. GAN-based data augmentation offers such an alternative, enabling the generation of large volumes of high-quality synthetic wireless data to supplement limited real-world samples without requiring prolonged and expensive physical data collection campaigns.

7.2.2 Data Quality Limitations and the Naivety of Traditional Augmentation

Beyond scarcity, real-world wireless data frequently suffers from quality limitations [91, 143]. It can be incomplete, noisy, challenging to model accurately, and may fail to represent the full spectrum of potential normal and anomalous behaviors. These quality issues compound the difficulties caused by data scarcity, further complicating the training of reliable detection models.

Data augmentation techniques, often applying in computer vision domain (e.g., random noise injection, rotation), are generally inadequate for the intricate nature of wireless time-series data [180]. Such naive transformations typically disregard crucial characteristics, such as temporal dependencies between measurements and the underlying statistical properties of wireless signals, potentially generating unrealistic or misleading samples. Similarly, techniques designed for tabular data, like SMOTE [34] or DeepSMOTE [39], primarily interpolate between existing samples. While helpful for addressing class imbalance, this interpolation approach often fails to capture the complex, non-linear features characteristic of anomalous wireless signals or network traffic, essentially creating averages rather than truly novel and representative examples [93].

In contrast, GAN [57] offer a more sophisticated approach. GAN can learn the complex, high-dimensional probability distributions underlying wireless data. This capability allows them to generate synthetic samples—including plausible representations of rare anomalies—that are statistically similar to real-world data yet potentially more diverse than simple interpolations. By learning the data distribution, GAN can overcome the limitations of simpler augmentation methods and provide more valuable and realistic training signals for anomaly detection models.

7.2.3 Enhancing Model Generalization and Robustness

A primary challenge for wireless anomaly detection systems is ensuring they generalize and remain robust across diverse operational conditions [189]. Supervised models, while potentially accurate under training conditions, often struggle when deployed in real-world wireless settings due to the environment's inherent dynamism [120]. Factors such as moving obstacles, electromagnetic interference, user mobility, and changing weather continuously perturb the physical communication channel [223].

Consequently, a model trained under one set of environmental conditions (e.g. outdoor conditions, obstacles, etc) frequently experiences significant performance degradation when encountering data reflecting different circumstances [137]. The model's learned decision boundary becomes suboptimal as the underlying data distribution shifts, leading to increased false positives (mistaking environmental variations for anomalies) and false negatives (missing true anomalies whose characteristics are altered or were unrepresented in training). This constant flux makes defining a stable baseline for "normal" behavior difficult. Therefore, developing anomaly detection methods that exhibit enhanced robustness and generalization is essential for reliable operation. This necessitates exploring techniques, such as advanced data augmentation strategies, capable of equipping models to handle the inevitable data shifts induced by environmental variability.

7.3 Related Work

The challenges outlined in Section 7.2—data scarcity, quality limitations, and the need for robust anomaly detection in dynamic wireless environments—motivate a review of existing approaches and their limitations. While various anomaly detection techniques exist, the unique characteristics of wireless data necessitate specialized solutions, increasingly pointing towards advanced data augmentation methods like GAN.

7.3.1 Anomaly Detection in Wireless Security

Anomaly detection in wireless security aims to identify deviations from normal operational behavior, which could signify faults or attacks [189]. However, accurately modeling "normal" behavior is difficult due to the dynamic nature of wireless systems, constantly shifting baselines, and unique physical layer vulnerabilities like jamming [150, 204, 137]. Classical methods, including statistical modeling, rule-based systems, and traditional machine learning [150], often struggle with the high dimensionality [165, 159], non-stationarity [178, 77, 132], severe class imbalance [34, 92, 93], and the need for continuous updates or extensive feature engineering common in wireless security datasets [181]. Even deep learning techniques like

autoencoders [39, 87], while powerful, traditionally require vast amounts of representative data and can be biased towards abundant normal samples, potentially missing subtle or rare anomalies – challenges directly related to the data scarcity and quality issues discussed previously (Section 7.2.1, 7.2.2).

7.3.2 Data Augmentation Techniques for Wireless Anomaly Detection

Data augmentation offers a compelling strategy to address the data limitations hindering effective wireless anomaly detection. Among various techniques, GAN [57] have garnered significant attention due to their ability to learn complex data distributions and generate high-fidelity synthetic samples [93, 180, 4]. While other generative models like diffusion models [87, 37] also show promise, we focus on GAN due to their established use in related tasks and architectural flexibility. The fundamental architecture of GAN is detailed in Section 7.4 where we describe our proposed framework.

Researchers have applied GAN to augment various types of data in wireless security.

Network Traffic Data:

GAN generate synthetic network flows or packet data, primarily to enrich datasets for Intrusion Detection Systems (IDS) [93]. This helps balance classes by creating more examples of underrepresented attacks and can potentially simulate novel attack variants based on learned patterns.

Wireless Signal Data:

GAN augment raw wireless signal measurements (e.g., IQ samples, channel state information) for tasks such as physical layer authentication [4] or detecting spectrum anomalies [181, 180], including jamming or unauthorized device transmissions.

Specific examples illustrate the active research in this area. TAnoGan [16] was proposed for unsupervised time-series anomaly detection using GAN to model normal behavior. Variants like Wasserstein GAN with Gradient Penalty (WGAN-GP) [61, 88] have been employed specifically to combat class imbalance in IDS datasets (e.g., NSL-KDD), improving classifier performance by generating high-quality minority class samples. Other studies [93, 180, 81] report improved accuracy in cyber-attack detection by training models on GAN-augmented data. These applications highlight GANs' versatility in addressing data scarcity and imbalance challenges (Section 7.2.1, 7.2.2).

7.3.3 Challenges and Limitations in GAN-based Augmentation

Despite this progress, applying GAN to enhance the robustness and generalization of wireless anomaly detection systems—critical requirements identified in Section 7.2.3—reveals significant challenges and limitations in existing practices.

Insufficient Robustness and Generalization Validation:

A primary concern is the frequent lack of rigorous evaluation regarding the robustness of models trained on GAN-generated data. Many studies do not adequately test performance against unseen data or network conditions significantly different from the GAN's training set [181, 81, 4, 180, 150]. Ensuring models generalize beyond the

learned distribution is crucial for deployment in dynamic real-world environments, yet this aspect often remains underexplored.

Limited Generation Control and Model Sophistication:

Many existing studies employ basic or non-conditional GAN architectures [4, 181, 180]. These often lack fine-grained control over the generation process (e.g., conditioning on specific labels, environmental factors, or attack parameters). This limits the ability to generate targeted, context-aware synthetic data, which is crucial for training models to handle complex, condition-dependent anomalies.

Oversimplified Evaluation Settings:

Some evaluations rely on datasets with limited features or utilize simplified simulated wireless environments [4, 93]. Such settings may not capture the full complexity of real-world deployments, potentially overstating performance and limiting the practical applicability of the findings.

These identified gaps—specifically concerning the validation of robustness and generalization in dynamic scenarios—underscore the necessity for more advanced, tailored generative frameworks. Existing approaches often fall short of producing the diverse, high-fidelity, and condition-specific synthetic data required to train wireless anomaly detectors that are truly robust to unseen conditions. This directly motivates our proposed framework, GANSec, detailed in Section 7.4. GANSec is specifically designed to address these limitations by incorporating conditional generation tailored for wireless time-series data, investigating architectures adept at capturing temporal dependencies, and rigorously evaluating performance with a focus on cross-scenario generalization.

7.4 The GANSec Framework

To address the challenges of data scarcity, quality, and the need for robustness in wireless anomaly detection highlighted previously, we propose GANsec:05: a tailored GAN framework designed specifically for augmenting labeled wireless data. GANSec aims to generate high-fidelity synthetic time-series data that captures the complex characteristics of wireless signals, including anomalous events, thereby improving the training and generalization capabilities of downstream anomaly detection models.

The overall workflow of the proposed approach is illustrated in Figure 7.1. It begins with real-world data acquisition, potentially sourced from distinct phases or environments (experiment and production). Raw signal measurements captured within these domains, along with relevant contextual information c, first undergo standard preprocessing steps such as data labeling, cleaning, and feature extraction to yield features x. The subsequent temporal aggregation step structures these extracted features over time into 2D time-series tensors.

In the baseline conventional approach, the processed 2D tensors from experiment domain would be used directly to train the anomaly detector. In contrast, we use these time series to train the conditional GAN for data augmentation. GANSec adapts the standard GAN paradigm [57], comprising a Generator ($\mathcal{G}$) and a Discriminator ($\mathcal{D}$), implemented as neural networks. $\mathcal{G}$ learns to transform random noise vectors z into synthetic data samples that resemble real wireless signals, while $\mathcal{D}$

learns to distinguish these synthetic samples from real-world data. Through adversarial training, both networks improve, enabling $\mathcal{G}$ to produce increasingly realistic synthetic wireless data.

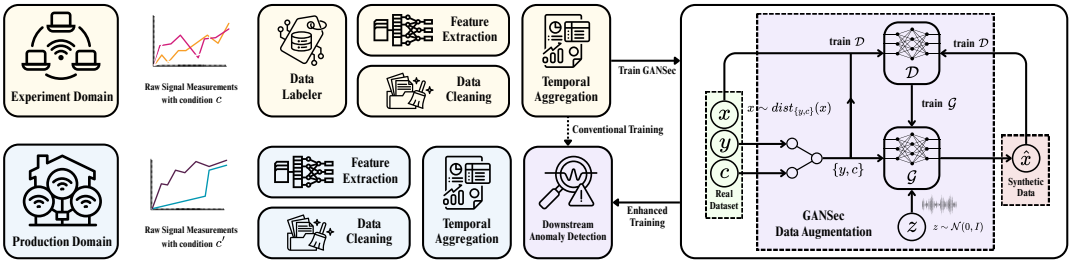


FIGURE 7.1: GAN-based Data Augmentation for Anomaly Detection in Wireless Security: Workflow Overview.

7.4.1 GANSec Architecture and Training Mechanisms

A detailed view of the GANSec architecture is provided in Figure 7.2. $\mathcal{G}$ takes a random noise vector z, sampled from a standard normal distribution ($\mathcal{N}(0, I)$) with dimensions defined by the desired latent space size, concatenated with conditioning information—specifically the target label y and potentially other contextual conditions c—as input. These inputs are processed through embedding layers and a neural network backbone (which we explore using MLP, LSTM, or CNN architectures) to produce synthetic features $\hat{x}$. $\mathcal{D}$ receives either a real data sample x or a synthetic sample $\hat{x}$, along with the corresponding conditions $\{y, c\}$, and outputs a likelihood on authenticity; this score challenges $\mathcal{G}$ to produce increasingly realistic data.

Within this conditional framework, we designed and investigated two primary $\mathcal{D}$ mechanisms for training $\mathcal{G}$, each utilizing different objective functions.

Embedded Conditional (EC) Discriminator

In this configuration, the primary objective is generating realistic data samples that accurately match the provided conditions $\{y, c\}$. $\mathcal{D}$ acts solely as a real/fake classifier for the given condition, outputting a probability score indicating authenticity. $\mathcal{G}$, in turn, aims to produce data $\hat{x}$ that the $\mathcal{D}$ classifies as real for the specified condition $\{y, c\}$. This approach emphasizes fidelity to the conditional data distribution. The training process follows Algorithm 2.

Classification Oriented (CO) Discriminator

This mechanism extends the $\mathcal{D}$'s role beyond simple real/fake classification, as depicted by the CO-Discriminator path in Figure 7.2. Here, $\mathcal{D}$ simultaneously performs two tasks: it predicts whether the input data (x or $\hat{x}$) is real or synthetic, and it predicts the data's class label y based on condition c. Correspondingly, $\mathcal{G}$'s loss function encourages the generation of synthetic data $\hat{x}$ that is not only classified as real by $\mathcal{D}$ but is also classified with the intended target label y. The intuition is that by optimizing for class discriminability alongside realism, $\mathcal{G}$ learns to generate samples particularly useful for the downstream classification task, potentially enhancing the anomaly detector's performance. The training follows Algorithm 3.

Comparing these two mechanisms allows us to evaluate distinct strategies for generating effective synthetic data: one focused primarily on conditional realism

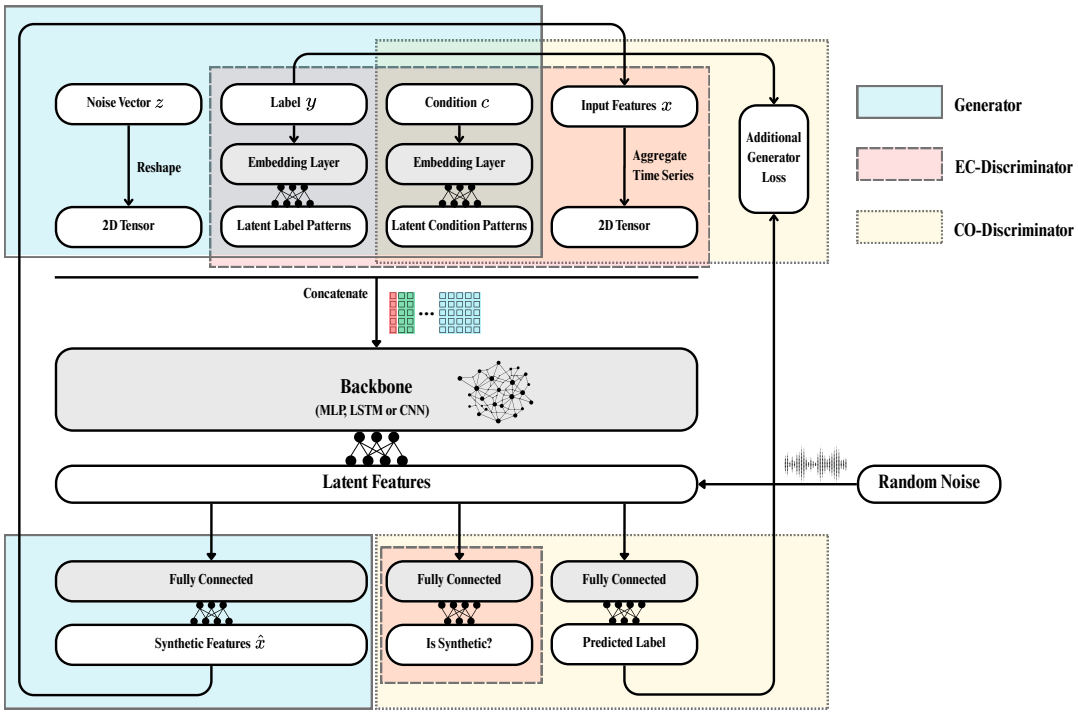


FIGURE 7.2: Architecture of the GANSec framework. The diagram shows distinct paths and objectives for the Embedded Conditional (EC-Discriminator) and Classification Oriented (CO-Discriminator) training mechanisms.

Algorithm 2 Embedded Conditional GAN Training: $\mathcal{D}$ focuses on distinguishing real samples from synthetic samples based on the provided context (label y and condition c), guiding $\mathcal{G}$ to produce realistic conditional samples

Require: Feature set $x \sim dist_{\{y,c\}}(x)$ with label y and condition c
Ensure: Discriminator $\mathcal{D}$, Generator $\mathcal{G}$

1: Initialize number of epochs ε, loss function $\mathcal{L}$, loss weights ω_i
2: Concatenate $\{y, c\}$
3: **for** current_epoch $\leftarrow 1, 2, ..\varepsilon$ **do**
4: Sample noise $z \sim \mathcal{N}(0, I)$
5: Generate synthetic data using $\mathcal{G}$: $\hat{x} \leftarrow \mathcal{G}(z, \{y, c\})$
6: Discriminate real or synthetic data using $\mathcal{D}$

$$output\_real \leftarrow \mathcal{D}(x, \{y, c\})$$

$$output\_synthetic \leftarrow \mathcal{D}(\hat{x}, \{y, c\})$$

7: Compute loss for $\mathcal{G}$ and $\mathcal{D}$

$$loss_{\mathcal{D}} \leftarrow \omega_0 \times \mathcal{L}(1, output\_real) + \omega_1 \times \mathcal{L}(0, output\_synthetic)$$

$$loss_{\mathcal{G}} \leftarrow \omega_2 \times \mathcal{L}(1, output\_synthetic)$$

8: **end for**

(EC-GAN) and the other directly incorporating the downstream task objective (CO-GAN). While the CO-GAN might offer advantages for improving classification accuracy by optimizing for class discriminability, it could also be more sensitive to class

Algorithm 3 Classification Oriented GAN Training: $\mathcal{D}$ performs a dual role: it assesses the realism of the input sample and predicts its class label. $\mathcal{G}$ is trained to produce realistic and discriminative samples on given label y

Require: Feature set $x \sim dist_{\{y,c\}}(x)$ with label y and condition c
Ensure: Discriminator $\mathcal{D}$, Generator $\mathcal{G}$
1: Initialize number of epochs ε, loss functions $\mathcal{L}_i$, loss weights ω_i
2: Concatenate $\{y, c\}$
3: **for** current_epoch $\leftarrow 1, 2, ..\varepsilon$ **do**
4: Sample noise $z \sim \mathcal{N}(0, I)$
5: Generate synthetic data using $\mathcal{G}$: $\hat{x} \leftarrow \mathcal{G}(z, \{y, c\})$
6: Discriminate real or synthetic data and predict label using $\mathcal{D}$

$$\{output\_real, y'\} \leftarrow \mathcal{D}(x, c)$$

$$\{output\_synthetic, \hat{y}\} \leftarrow \mathcal{D}(\hat{x}, c)$$

7: Compute loss for $\mathcal{G}$ and $\mathcal{D}$

$$loss_{\mathcal{D}} \leftarrow \omega_0 \times \mathcal{L}_0(1, output\_real) + \omega_1 \times \mathcal{L}_1(y, y') +$$

$$\omega_2 \times \mathcal{L}_0(0, output\_synthetic) + \omega_3 \times \mathcal{L}_1(y, \hat{y})$$

$$loss_{\mathcal{G}} \leftarrow \omega_4 \times \mathcal{L}_0(1, output\_synthetic) + \omega_5 \times \mathcal{L}_1(y, \hat{y})$$

8: **end for**

imbalance in the original training data.

7.4.2 Tailoring GANSec for Wireless Data

Generating realistic synthetic wireless data requires specific adaptations beyond standard GAN implementations, as naive approaches often fail to capture the unique characteristics of wireless signals (as discussed in Section 7.2.2). GANSec incorporates several design considerations to handle these intricacies effectively.

Handling Temporal Dependencies:

Wireless signal measurements are time-series data with strong temporal correlations crucial for context [178, 16]. To capture these, GANSec employs backbones suited for sequential data. LSTMs are used for their ability to model long-range dependencies, while 2D CNNs effectively learn local, time-invariant patterns within the signal sequences. An MLP baseline is included for comparison, requiring flattening of the time-series segments into 1D vectors, which potentially obscures some temporal relationships.

Preserving Statistical Properties:

Beyond temporal structure, wireless signals possess specific statistical properties vital for accurate representation. For instance, metrics like Reference Signal Received Power (RSRP) are often measured on a logarithmic scale (dBm). This logarithmic nature, while convenient for representing a wide dynamic range, introduces challenges for neural networks [35]. The non-linearity can interact suboptimally with standard activation functions if not handled carefully, and common loss functions like

Mean Squared Error behave differently in the logarithmic domain compared to the linear power domain [128, 190]. Furthermore, noise and interference cause disproportionately larger changes in dBm values for weaker signals compared to stronger ones [119]. Therefore, generating synthetic data that accurately reflects these statistical properties, especially for log-scale metrics, is non-trivial. While GANSec does not explicitly apply the specialized preprocessing techniques discussed for discriminative Received Signal Strength Indicator (RSSI) models, its generator must learn transformations that account for these statistical intricacies, and the discriminator's evaluation implicitly pushes the generated data towards realistic distributions, effectively addressing the challenges posed by logarithmic scales within the generative framework.

Incorporating Domain Knowledge:

The conditional nature of GANSec allows us to explicitly guide the generation process using domain-specific information. The condition vector c can include parameters representing known environmental factors, device types, protocol states, specific interference signatures, or indicators of known vulnerabilities, enabling the generation of contextually relevant data.

7.4.3 Synthetic Data Generation and Usage Strategy

Once GANSec is trained on the available real data using either the EC or CO mechanism, we utilize the trained $\mathcal{G}$ to produce synthetic data. To generate synthetic samples corresponding to a specific class label and condition $\{y, c\}$, we sample a random noise vector z and provide the tuple $\{z, y, c\}$ as input to the trained $\mathcal{G}$. This process, detailed in Algorithm 4, allows us to generate any desired number of synthetic samples for each class, potentially including optional post-processing or validation steps (e.g., using the $\mathcal{D}$ to filter generated samples based on a quality threshold t).

Algorithm 4 Synthetic Data Generation: $\mathcal{G}$ produces regulated synthetic feature sets given a target label $\hat{y}$ and condition $\hat{c}$, including an optional step for validating the quality of generated samples using the $\mathcal{D}$ with a threshold t

Require: Generator $\mathcal{G}$, Discriminator $\mathcal{D}$, label $\hat{y}$ and condition $\hat{c}$
Ensure: Synthetic feature set $\hat{x}$
1: Initialize regulations $regul(x)$, discriminator threshold t
2: Concatenate $\{y, c\}$
3: **while** $p < t$ **do**
4: Sample noise $z \sim \mathcal{N}(0, I)$
5: Generate synthetic data using $\mathcal{G}$: $\hat{x} \leftarrow \mathcal{G}(z, \{y, c\})$
6: Regulate synthetic data: $\hat{x}' \leftarrow regul(\hat{x})$
7: Validate using $\mathcal{D}$: $p \leftarrow \mathcal{D}(\hat{x}')$
8: **end while**

A key aspect of our proposed methodology involves training the downstream anomaly detection model exclusively on the synthetic data generated by GANSec. Employing a synthetic-only training strategy offers several significant advantages compared to using the original real-world data: (i) it allows for the generation of perfectly balanced datasets, mitigating issues caused by original class imbalance; (ii) if the GAN captures the underlying distribution well, the synthetic data can exhibit greater diversity than the initial samples, potentially enhancing detector robustness

and generalization; and (iii) it offers potential privacy preservation benefits, as training the anomaly detector solely on synthetic data avoids direct exposure of potentially sensitive real-world measurements during this critical phase, which is particularly relevant when dealing with user data or proprietary network information.

7.5 Evaluation

This section details the experimental evaluation conducted to validate the effectiveness of our proposed GANSec framework. Following the methodology outlined in Section 7.4, our evaluation rigorously assesses GANSec from two critical perspectives. First, we quantitatively measure the fidelity and statistical similarity of the synthetic wireless data generated by different GANSec configurations compared to real-world data. Second, and crucially for addressing the challenges motivated in Section 7.2.3, we evaluate the effectiveness of using GANSec-generated data for training robust anomaly detection models. Particular focus is placed on the model's ability to generalize to unseen data collected under different network conditions, assessed through a challenging cross-scenario evaluation.

7.5.1 Experimental Setup

Data Description:

The data used in our experiments consists of real-world 5G wireless measurements collected within the indoor environment illustrated in Figure 7.3a. The hardware setup, shown in Figure 7.3b, included a OnePlus Nord 2T 5G smartphone acting as the user equipment (UE), two Nuand bladeRF 2.0 micro SDRs configured as a 5G gNodeB and a jammer to introduce anomalies.

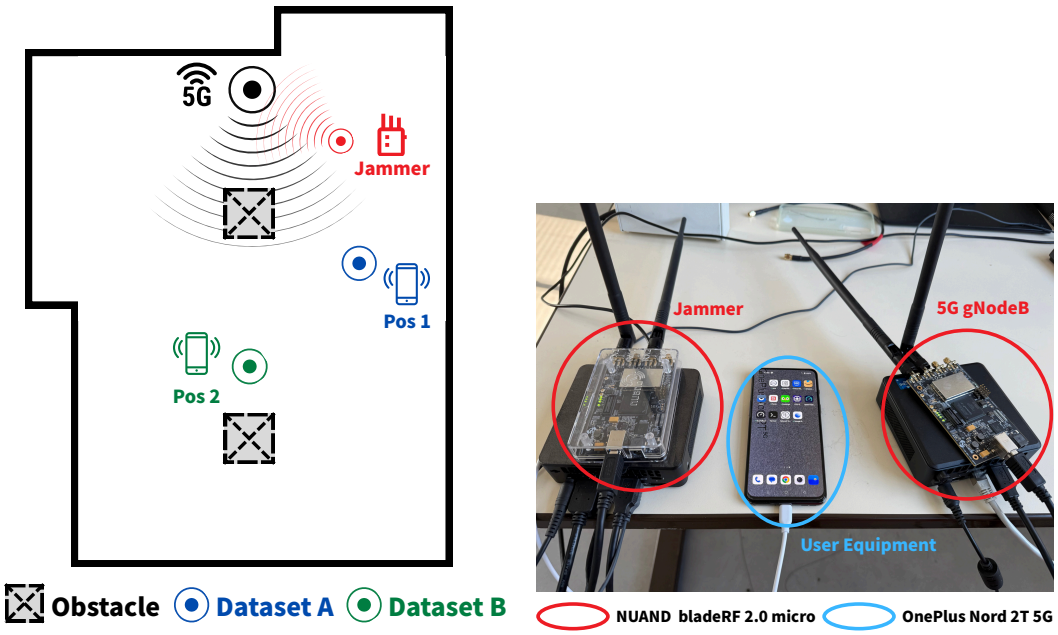
The jammer operated in the 5G n2 uplink frequency band (1850-1910MHz) and was controlled using a sinusoidal power modulation pattern to create disruptive signals. This type of power-modulated jamming, with its continuously time-varying pattern, can be particularly challenging to detect comparing to simplistic constant or reactive jammers [167]. Preliminary tests confirmed its effectiveness, as the jamming activity was observed to significantly impact the network latency of the UE during iperf3 testing.

We collected radio buffer messages from the Android smartphone via the Android Debug Bridge (ADB), capturing key signal quality indicators. The primary features extracted for analysis were SINR, RSRP, and RSRQ. To ensure data consistency and suitability for model training, we parsed the raw logs to extract these numerical features, handled missing values using interpolation, ensured uniform time steps, and normalized the feature values. We recorded the timestamps when the jammer was activated and deactivated to create the ground truth labels y. Each aggregated time-series segment was then labeled as 'anomalous' if the jammer was active during the corresponding time window, and 'normal' otherwise. We also constructed a condition vector c associated with each data sample. This vector included specifically: the UE's distance to the gNodeB, the device model (OnePlus Nord 2T 5G), and the smartphone's modem temperature readings obtained during data collection. Following this process, we established two distinct datasets reflecting different conditions or collection locations within the setup shown in Figure 7.3a:

- Scenario A: This dataset comprises 3000 samples collected over approximately 10 minutes. It includes instances of both normal operation and anomalous behavior caused by the jamming interference introduced in the setup. The class

distribution in Scenario A is relatively balanced, with jamming anomalies representing 46.37% of the samples. Data from Scenario A was used for training the GANSec models and the baseline anomaly detection models.

- Scenario B: This dataset contains 4559 samples collected under different UE position than Scenario A, having an additional obstacle to the gNodeB. It features the same set of measurements but has a different class distribution, with anomalies constituting 65.08% of the samples. Scenario B serves exclusively as the unseen test set to evaluate the generalization capabilities of the anomaly detection models trained using data derived from Scenario A.



(A) 5G Indoor Environment (B) Hardware Setup

FIGURE 7.3: Experimental Setup for Collecting Datasets A and B.

Baselines for Comparison

We compare the performance achieved using GANSec-augmented data against models trained using three baseline data strategies:

- Raw Data Training: The downstream anomaly detection model trained directly on the original Scenario A dataset.
- Classical Oversampling: The downstream model trained on the Scenario A dataset augmented using DeepSMOTE [39], an advanced implementation combining SMOTE [34] principles with deep learning representations.
- Unconditional GAN: The downstream model trained on data generated by TAnoGAN [16], representing another GAN-based approach for time-series anomaly detection, used here as a generative baseline.

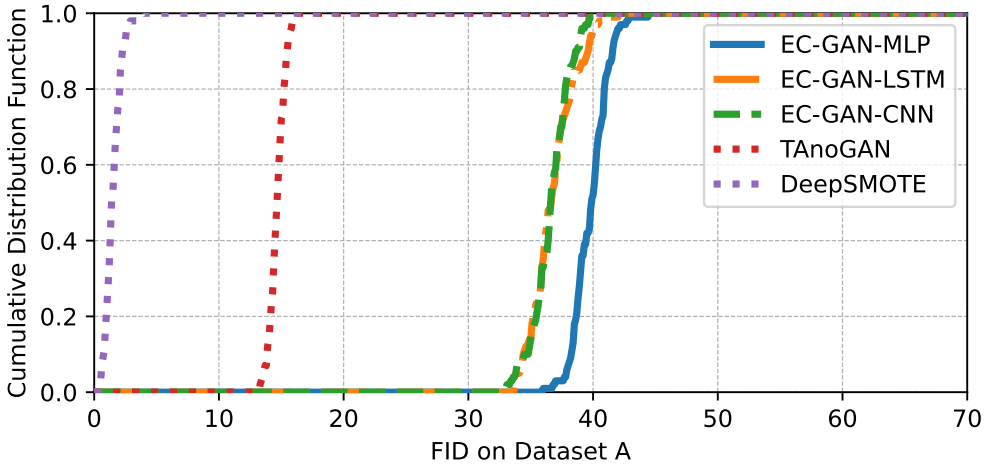
Downstream Anomaly Detection Model

To evaluate the utility of the generated data, we employed a downstream anomaly detection model tasked with classifying wireless measurements as 'normal' or

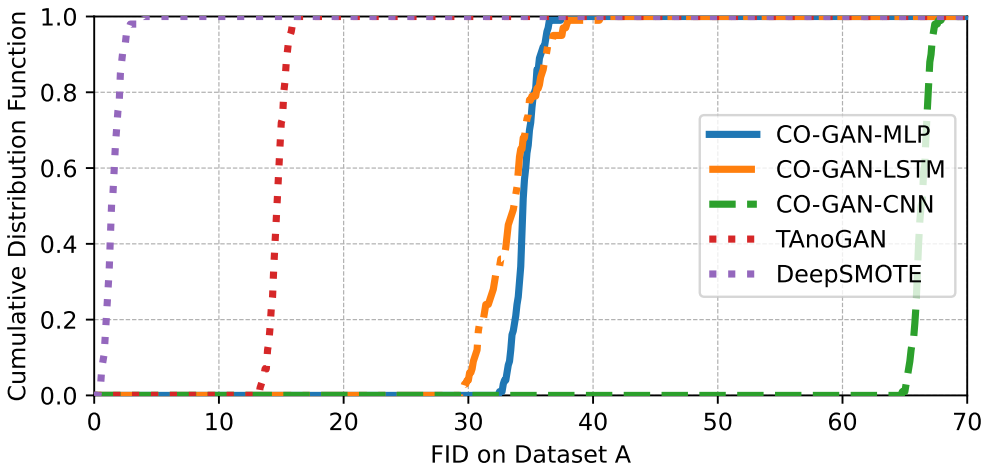
'anomalous' (jamming). We utilized a 3-layer Long Short-Term Memory (LSTM) network for this purpose, chosen for its ability to capture temporal dependencies in the signal data. Each LSTM layer contained 128 hidden units, followed by a dropout layer with a rate of 0.3 to prevent overfitting, and a final dense layer with a sigmoid activation function for classification. The model was trained using the Adam optimizer with a learning rate of 0.0001 for 128 epochs, utilizing binary cross-entropy as the loss function.

7.5.2 Synthetic Data Quality Analysis

We analyze the quality of synthetic wireless data generated by GANSec variants and baseline methods using Fréchet Inception Distance (FID) [76] and Maximum Mean Discrepancy (MMD) [23] with an RBF kernel. Lower scores indicate higher similarity to the real Scenario A data distribution. To assess the distribution and stability of these metrics, we performed 100 independent generation iterations for each model configuration, creating a synthetic dataset equivalent in size to Dataset A in each iteration. Figure 7.4 and Figure 7.5 presents the Cumulative Distribution Functions (CDFs) of these scores.

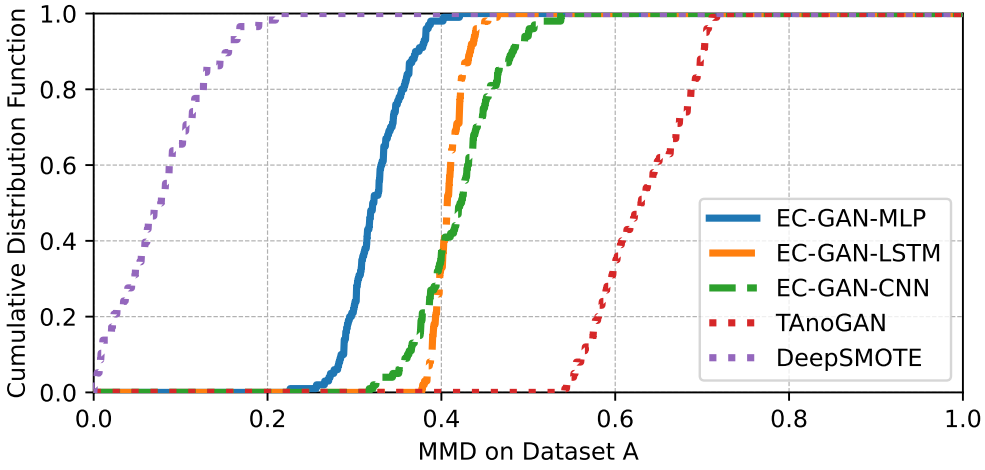


(A) FID Score (EC-GAN).

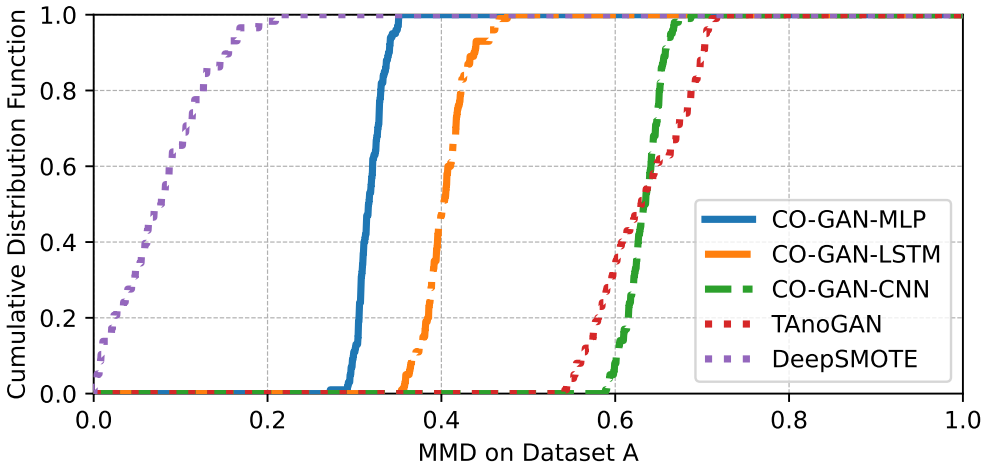


(B) FID Score (CO-GAN).

FIGURE 7.4: Distribution Analysis on Synthetic Dataset (FID).



(A) MMD Score (EC-GAN).



(B) MMD Score (CO-GAN).

FIGURE 7.5: Distribution Analysis on Synthetic Dataset (MMD).

The DeepSMOTE [39] oversampling method yields very low FID and MMD scores (e.g., average FID 1.603, MMD 0.164), indicating generated samples closely mirror the original data but offer limited diversity due to its interpolative nature (as argued in Section 7.2.2). In contrast, the TAnoGAN [16] baseline shows low FID (14.762) but high MMD (0.632), suggesting it captures global data structure but struggles with local sample quality or diversity.

Generally, our proposed GANSec variants exhibit higher FID and MMD scores than DeepSMOTE. We interpret these higher scores not merely as lower fidelity, but rather as reflecting GANSec's ability to generate a more diverse set of synthetic samples, while still capturing key signal patterns. This increased diversity is hypothesized to be crucial for improving model robustness and generalization, which we test in Section 7.5.3. In generative modeling, FID scores for GAN can vary widely depending on the complexity of the data and the model; values in the range observed (e.g., up to 60-70 for CNN backbones) can be considered reasonable [76].

Within GANSec, the choice of neural network architecture influences synthetic data characteristics. MLP-based GAN tend towards lower MMD scores, while LSTM-based GAN show stable behavior across metrics. CNN-based GAN appear more "aggressive," producing potentially more diverse outputs (reflected in higher

scores), which might explain the better cross-scenario performance observed later. This suggests a potential trade-off: architectures like CNN might sacrifice exact distributional match (higher FID/MMD) to achieve greater diversity beneficial for downstream task robustness.

7.5.3 Anomaly Detection Performance: Cross-Scenario Evaluation

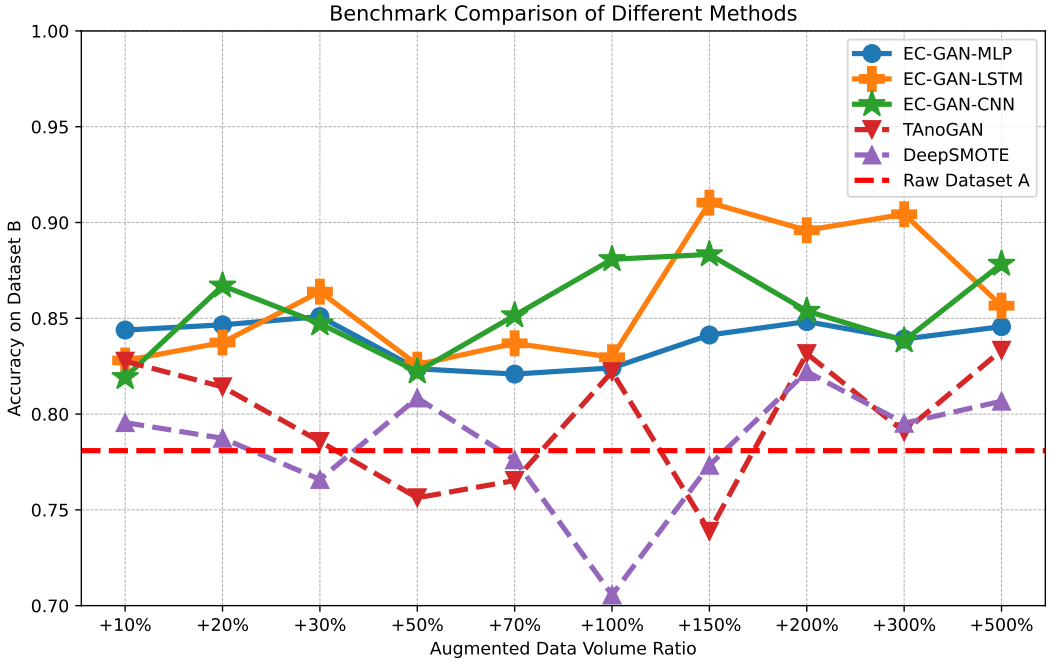
Having analyzed the synthetic data quality, we now evaluate our primary hypothesis regarding GANSec's ability to enhance anomaly detector robustness and generalization. We conducted a cross-scenario evaluation, training the LSTM anomaly detector (detailed in Section 7.5.1) on data derived from Scenario A and testing on unseen Scenario B data. Results are averaged over 10 independent iterations to mitigate randomness. Notably, all models performed well (>98% accuracy) on Scenario A data itself, suggesting the synthetic-only training does not cause catastrophic forgetting. Figure 7.6 illustrates key accuracy results.

The generalization challenge is highlighted by the baseline LSTM model trained on raw Scenario A data: its accuracy dropped from 92% when tested on Scenario A to 78% on Scenario B. This performance degradation underscores both the detection difficulty posed by the dynamic, power-modulated jammer implemented in our experiments and the substantial generalization gap faced by standard anomaly detection methods when encountering different network conditions.

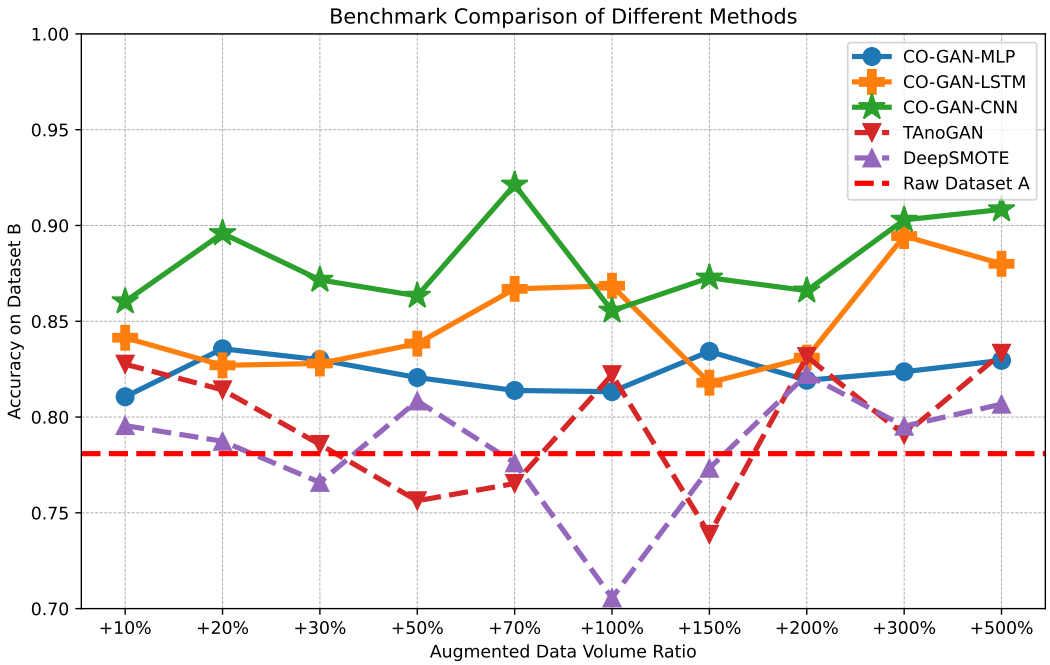
The results clearly demonstrate the advantage of the GANSec framework. Models trained exclusively on data generated by most GANSec variants significantly outperform those trained on the original Raw data or data augmented using DeepSMOTE or TAnoGAN when evaluated on the unseen Scenario B. This superior performance highlights the enhanced generalization achieved through GANSec. Models trained on GANSec data learned more fundamental and generalizable anomaly features, enabling better identification under different network conditions compared to models trained on the original distribution or simpler augmentation methods.

In contrast, the results clearly demonstrate the advantage of the GANSec framework. Models trained exclusively on GANSec-generated data significantly outperformed all baseline methods on the unseen Scenario B, showcasing enhanced generalization by learning more fundamental anomaly features. Consistent with data quality analysis, LSTM and CNN backbones generally outperformed MLPs, reinforcing the importance of architectures that effectively model temporal signal characteristics. Comparing training objectives, the Classification-Oriented (CO-GAN) objective showed a slight edge over the Embedded Conditional (EC-GAN) variant, suggesting that directly incorporating the classification objective during generation may produce more discriminative synthetic samples.

Crucially, the superior cross-scenario performance underscores the effectiveness of using tailored GAN with a synthetic-only training strategy to build more robust wireless anomaly detection systems capable of handling unseen conditions, directly addressing a key limitation of prior work.



(A) EC-GAN



(B) CO-GAN

FIGURE 7.6: Impact of the augmentation ratio (relative size of synthetic data used for training) on the accuracy achieved on the unseen Dataset B.

7.6 Discussion

This work demonstrates that GANSec, a tailored conditional GAN framework, significantly enhances the robustness and generalization of wireless anomaly detectors, particularly in challenging cross-scenario evaluations. Our results advocate for moving beyond generic augmentation towards methods designed for wireless data

intricacies, where sequence-aware architectures (LSTMs, CNNs) offer advantages. The viability of augmented training also presents practical benefits regarding data imbalance, collection costs, and potential privacy enhancement.

Despite these promising results, we acknowledge limitations. Our current evaluation focused on jamming anomalies within a specific 5G context. However, it is important to note that the framework demonstrated effectiveness against a sophisticated, power-modulated jammer exhibiting dynamic time-varying patterns (as described in Section 7.5.1), suggesting its potential applicability extends beyond this specific scenario to other challenging wireless anomalies where capturing complex temporal dynamics is crucial. Nevertheless, further investigation is required to assess GANSec's applicability to other anomaly types and wireless technologies (e.g., Wi-Fi, IoT protocols). Additionally, while GAN can generate diverse data, risks remain regarding the potential amplification of biases present in the initial real dataset or the failure to generate truly novel "black swan" anomalies. The computational expense of training advanced GAN models is also a practical consideration.

Future research should prioritize extending GANSec to a wider spectrum of anomalies and network contexts, potentially exploring hybrid approaches combining real and synthetic data. Developing techniques to explicitly encourage the generation of rare or edge-case anomalies and further refining generative architectures (e.g., using attention mechanisms or investigating alternatives like diffusion models, despite their current computational complexity concerns) are promising directions. While diffusion models show potential, they were not included as baselines in this study due to concerns regarding their computational complexity for large-scale deployment in this specific context. Crucially, validating enhanced approaches in real-time, operational settings is essential. Finally, while generative models can reduce the environmental impact associated with extensive physical data collection, the energy cost of training these complex models warrants consideration and comparison against traditional methods in future analyses.

7.7 Conclusion

Addressing the need for robust wireless anomaly detection despite data limitations, this work introduced GANSec, a tailored conditional GAN framework. We evaluated a novel strategy using exclusively GANSec-generated synthetic data for training the downstream detector. Cross-scenario evaluations confirmed this approach significantly enhances model generalization and robustness, effectively addressing the substantial performance degradation observed when anomaly classifiers, trained on limited data, encounter unseen network conditions. Our findings indicate that tailored generative models employing synthetic-only training offer a highly promising and effective path towards developing more reliable wireless security systems.

Conclusion and Future Directions

Chapter 8

Conclusion and Future Directions

8.1 Conclusion

This thesis advances wireless security through a coordinated, cross-layer program spanning the physical and network layers of the communication stack. We develop generative, adversarial, and calibration-aware methods tailored to the unique vulnerabilities of each layer. The main contributions are summarized as follows:

1. **Capodoglio (Physical Layer, Chapter 2).** We formalize adaptive jammers as online/offline MAB learners and analyze their interaction with frequency-hopping systems. Building on this model, we design three lightweight defenses: *Speed-Up* (temporal acceleration of hopping), *Helper-Node* (decoy transmissions to poison the attacker's reward model), and *Mirror-MAB* (a defender-side bandit policy). These mechanisms collectively stabilize packet-delivery ratios while clarifying the trade-offs between stability and overhead.

2. **DOA-CGAN (Physical Layer, Chapter 3).** We reformulate anti-jamming DOA estimation as generative interference removal in covariance space. A conditional GAN with an *unsupervised generator* and a *supervised discriminator* learns to "evaporate" structured interference, restoring lawful covariance structure for classical estimators such as MUSIC. This approach reduces estimation error relative to learning-based baselines while preserving interpretability.

3. **Triad-GAN (Physical Layer, Chapter 4).** We design a feature-level triadic adversarial framework composed of an extractor, generator, and discriminator. This interaction encourages receiver-invariant yet transmitter-discriminative embeddings, enabling more reliable authentication in multi-receiver RFFI scenarios. Experiments on LoRa and WiSig confirm robustness against receiver drift, deployment changes, and scaling.

4. **OS-TriadGAN (Physical Layer, Chapter 5).** Building on Triad-GAN, we propose a calibration-aware open-set extension. By integrating *label smoothing*, *temperature scaling*, and an *F1-optimal MSP threshold*, the framework enables reliable rejection of unknown devices while maintaining closed-set accuracy. An algebraic mapping further improves reproducibility and deployment readiness.

5. **Catch-All (Network Layer, Chapter 6).** We introduce a graph-structured abstraction of Wi-Fi traffic in which GNNs are trained with topological features and traffic statistics data. This framework improves precision–recall balance in imbalanced datasets and reduces false alarms under non-stationary

conditions, demonstrating the value of structural information in traffic-level anomaly detection.

6. **GANSec (Network Layer, Chapter 7).** We develop a classification-oriented conditional GAN that synthesizes realistic wireless time-series to mitigate domain shift and data scarcity. Sequence-aware backbones such as LSTMs and CNNs benefit most from this augmentation, achieving stronger cross-scenario generalization. Synthetic-only training further demonstrates feasibility in data-limited environments.

Overall, these contributions demonstrate how generative, adversarial, and calibration-aware strategies can be systematically applied across layers to build wireless systems that are resilient, adaptive, and deployment-ready.

8.2 Future Directions

While the methods presented advance the state of the art, several important avenues remain open for future research:

- **Testbed validation.** Extending Capodoglio and DOA-CGAN to hardware testbeds would allow evaluation of latency, energy consumption, and coexistence costs under real-world propagation and interference, revealing trade-offs not visible in simulations.
- **Adaptive and open-world security.** Future adversaries are likely to co-adapt across time and layers. Game-theoretic analysis and meta-reinforcement learning could enhance defenses against policy-tracking jammers, while open-world RFFI requires continual enrollment, drift adaptation, and graceful degradation mechanisms.
- **Federated and privacy-preserving learning.** Multi-receiver deployments motivate federated training with secure aggregation, differential privacy, and drift-aware model updates. Lightweight test-time adaptation and on-device calibration may help sustain performance under variability in hardware and environment.
- **Uncertainty and calibration.** Beyond MSP and temperature scaling, Bayesian methods, deep ensembles, and conformal prediction offer rigorous coverage guarantees and threshold-free operation, which are particularly important for high-stakes open-set rejection.
- **Synthetic data foundations.** Developing principled metrics and benchmarks for RF generative models, including fidelity, diversity, and downstream utility, will standardize evaluation. Generators that integrate protocol states, channel effects, and hardware impairments may further improve controllability.
- **Cross-layer co-design.** Joint optimization across layers could unlock new synergies. For instance, physical-layer interference fingerprints may regularize network-layer anomaly detectors, while anomaly detection outputs could dynamically steer channel-hopping defenses.

- **Efficiency and deployment readiness.** Future research should emphasize efficiency, including pruning, quantization, distillation with security constraints, and hardware-friendly architectures. Profiling floating-point operations (FLOPs) per joule and memory footprints alongside security metrics will facilitate real-world deployment.
- **Physical-layer location privacy (complementary work).** In parallel to the thesis contributions, we conducted a separate study, *Confundus*, which leverages fake multipath injection to mislead TDOA-based localization. Although not part of the thesis itself, this work highlights privacy as a distinct physical-layer objective and points to future research on tunable privacy guarantees that coexist with reliability and efficiency.

Collectively, these open directions underscore the need for continued research that not only advances theoretical understanding but also translates into practical, efficient, and privacy-preserving wireless security solutions ready for real-world deployment.

Bibliography

[1] AGRAWAL, S., AND GOYAL, N. Analysis of thompson sampling for the multi-armed bandit problem. In *Proceedings of the 25th Annual Conference on Learning Theory* (2012), vol. 23, pp. 39.1–39.26.

[2] AHMED, M., NASER MAHMOOD, A., AND HU, J. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications 60* (Jan. 2016), 19–31.

[3] AIMI, A., GUILLEMIN, F., ROVEDAKIS, S., AND SECCI, S. Packet delivery ratio guarantees for differentiated lorawan services. In *2022 IEEE Global Communications Conference (GLOBECOM)* (2022), pp. 2014–2019.

[4] ALHORAIBI, L., ALGHAZZAWI, D., AND ALHEBSHI, R. Generative adversarial network-based data augmentation for enhancing wireless physical layer authentication. *Sensors* (2024).

[5] ALOM, M. Z., TAHA, T. M., YAKOPCIC, C., WESTBERG, S., SIDIKE, P., NASRIN, M. S., VAN ESESN, B. C., AWWAL, A. A. S., AND ASARI, V. K. The history began from Alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164* (2018).

[6] ANDREWS, J. G., BUZZI, S., CHOI, W., HANLY, S. V., LOZANO, A., SOONG, A. C. K., AND ZHANG, J. C. What will 5g be? *IEEE Journal on Selected Areas in Communications 32*, 6 (2014), 1065–1082.

[7] ANDREWS, S., GERDES, R. M., AND LI, M. Crowdsourced measurements for device fingerprinting. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks* (2019), p. 72–82.

[8] ANG, K. L.-M., SENG, J. K. P., AND ZUNGERU, A. M. Optimizing energy consumption for big data collection in large-scale wireless sensor networks with mobile collectors. *IEEE Systems Journal 12*, 1 (2018), 616–626.

[9] ARJOUNE, Y., AND FARUQUE, S. Smart Jamming Attacks in 5G New Radio: A Review, Sept. 2020.

[10] ARJOVSKY, M., CHINTALA, S., AND BOTTOU, L. Wasserstein generative adversarial networks. In *Proceedings of the 34th International Conference on Machine Learning* (2017), vol. 70, pp. 214–223.

[11] BAHER SAFA HANBALI, S. Technique to counter improved active echo cancellation based on ISRJ with frequency shifting. *IEEE Sensors Journal 19*, 20 (2019), 9194–9199.

[12] BAHER SAFA HANBALI, S., AND KASTANTIN, R. Technique to counter active echo cancellation of self-protection ISRJ. *Electronics Letters 53*, 10 (2017), 680–681.

[13] BALLESTER, P., AND ARAUJO, R. M. On the performance of googlenet and alexnet applied to sketches. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2016), pp. 1124–1128.

[14] BAO, J., XIE, X., LU, Z., HONG, J., AND HUA, C. Receiver-agnostic radio frequency fingerprinting based on two-stage unsupervised domain adaptation and fine-tuning. In *GLOBECOM 2023 - 2023 IEEE Global Communications Conference* (2023), pp. 6085–6090.

[15] BARTHELME, A., AND UTSCHICK, W. DoA estimation using neural network-based covariance matrix reconstruction. *IEEE Signal Processing Letters 28* (2021), 783–787.

[16] BASHAR, M. A., AND NAYAK, R. Tanogan: Time series anomaly detection with generative adversarial networks. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)* (2020), pp. 1778–1785.

[17] BENDALE, A., AND BOULT, T. E. Towards open set deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 1563–1572.

[18] BERTALANIČ, B., HRIBAR, J., AND FORTUNA, C. Visibility graph based wireless anomaly detection for digital twin edge networks. *IEEE Open Journal of the Communications Society* (2024).

[19] BERTALANIČ, B., MOHORČIČ, M., AND FORTUNA, C. Gnn for wireless link anomaly detection. In *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops* (2023), pp. 1–2.

[20] BOODAI, J., ALQAHTANI, A., AND FRIKHA, M. Review of Physical Layer Security in 5G Wireless Networks. *Applied Sciences 13*, 12 (Jan. 2023), 7277.

[21] BOUT, E., BOUT, V., BRIGHENTE, A., CONTI, M., AND LOSCRI, V. Evaluation of channel hopping strategies against smart jamming attacks. In *ICC 2023-IEEE International Conference on Communications* (2023), IEEE, pp. 5210–5215.

[22] BOUT, E., BRIGHENTE, A., CONTI, M., AND LOSCRI, V. Folpetti: A novel multi-armed bandit smart attack for wireless networks. In *Proceedings of the 17th International Conference on Availability, Reliability and Security* (2022), pp. 1–10.

[23] BRIOL, F.-X., BARP, A., DUNCAN, A. B., AND GIROLAMI, M. Statistical inference for generative models with maximum mean discrepancy, 2019.

[24] BROMLEY, J., GUYON, I., LECUN, Y., SÄCKINGER, E., AND SHAH, R. Signature verification using a "siamese" time delay neural network. In *Proceedings of the 7th International Conference on Neural Information Processing Systems* (1993), p. 737–744.

[25] BROWN, C. N., MATTEI, E., AND DRAGANOV, A. Charrnets: Channel robust representation networks for rf fingerprinting, 2021.

[26] BRUNA, J., ZAREMBA, W., SZLAM, A., AND LECUN, Y. Spectral networks and deep locally connected networks on graphs. In *2nd International Conference on Learning Representations* (2014), pp. 1–14.

[27] CAI, C., ZHENG, R., AND LUO, J. Ubiquitous acoustic sensing on commodity iot devices: A survey. *IEEE Communications Surveys & Tutorials 24*, 1 (2022), 432–454.

[28] CAI, D., SHAN, J., GAO, N., HE, B., CHEN, Y., JIN, S., AND FAN, P. Open set rf fingerprinting identification: A joint prediction and siamese comparison framework. *arXiv preprint arXiv:2501.15391* (2025).

[29] CAO, X., TAN, W., GAO, Q., HU, Z., AND LI, C. Intml-knn: A few-shot radio frequency fingerprint identification scheme for lora devices. *IEEE Signal Processing Letters 32* (2025), 2259–2263.

[30] CAPON, J. High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE 57*, 8 (1969), 1408–1418.

[31] CECI, E., AND BARBAROSSA, S. Small perturbation analysis of network topologies. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing* (2018), pp. 4194–4198.

[32] CHALAPATHY, R., AND CHAWLA, S. Deep Learning for Anomaly Detection: A Survey, Jan. 2019.

[33] CHANDOLA, V., BANERJEE, A., AND KUMAR, V. Anomaly detection: A survey. *ACM Computing Surveys 41*, 3 (2009).

[34] CHAWLA, N. V., BOWYER, K. W., HALL, L. O., AND KEGELMEYER, W. P. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research 16* (June 2002), 321–357.

[35] CHEN, B., MA, J., ZHANG, L., ZHOU, J., FAN, J., AND LAN, H. Research Progress of Wireless Positioning Methods Based on RSSI. *Electronics 13*, 2 (Jan. 2024), 360. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[36] CHENG, S., LING, X., AND ZHU, L. Deep Reinforcement Learning-Based Anti-Jamming Approach for Fast Frequency Hopping Systems. *IEEE Open Journal of the Communications Society 6* (2025), 961–971.

[37] CHI, G., YANG, Z., WU, C., XU, J., GAO, Y., LIU, Y., AND HAN, T. X. Rf-diffusion: Radio signal generation via time-frequency diffusion. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking* (New York, NY, USA, 2024), Association for Computing Machinery, p. 77–92.

[38] CHOI, Y., CHOI, M., KIM, M., HA, J.-W., KIM, S., AND CHOO, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 8789–8797.

[39] DABLAIN, D., KRAWCZYK, B., AND CHAWLA, N. V. Deepsmote: Fusing deep learning and smote for imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems 34*, 9 (2023), 6390–6404.

[40] DANEV, B., ZANETTI, D., AND CAPKUN, S. On physical-layer identification of wireless devices. *ACM Computing Surveys 45*, 1 (Dec. 2012), 6:1–6:29.

[41] DEFFERRARD, M., BRESSON, X., AND VANDERGHEYNST, P. Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems 29* (2016), 1–9.

[42] DONG, X., ZHANG, X., ZHAO, J., AND SUN, M. DOA estimation for coprime array with mixed noise scenario via phased fractional low-order moment. *IEEE Wireless Communications Letters 10*, 11 (2021), 2567–2571.

[43] DU, Z.-Y., AND WANG, Q. Digital infrastructure and innovation: Digital divide or digital dividend? *Journal of Innovation & Knowledge 9*, 3 (2024), 100542.

[44] DUTTA, H., KUMAR BHUYAN, A., AND BISWAS, S. Using multi-armed bandit learning for thwarting mac layer attacks in wireless networks. *IEEE/ACM Transactions on Networking 33*, 1 (2025), 327–339.

[45] ELSADIG, M. A. Detection of denial-of-service attack in wireless sensor networks: A lightweight machine learning approach. *IEEE Access 11* (2023), 83537–83552.

[46] FANG, H., WANG, X., AND HANZO, L. Learning-aided physical layer authentication as an intelligent process. *IEEE Transactions on Communications 67*, 3 (2018), 2260–2273.

[47] FENG, J., FANG, S., AND FAN, Y. Cross-receiver radio frequency fingerprint identification based on domain adaptation with dynamic distribution alignment. *IEEE Internet of Things Journal* (2025).

[48] FENG, J., TANG, X., ZHANG, B., AND REN, Y. Lightweight cnn-based rf fingerprint recognition method. In *2023 8th International Conference on Computer and Communication Systems (ICCCS)* (2023), pp. 1031–1035.

[49] FLUSS, R., FARAGGI, D., AND REISER, B. Estimation of the youden index and its associated cutoff point. *Biometrical Journal: Journal of Mathematical Methods in Biosciences 47*, 4 (2005), 458–472.

[50] FONTES, R. R., AFZAL, S., BRITO, S. H. B., SANTOS, M. A. S., AND ROTHENBERG, C. E. Mininet-wifi: Emulating software-defined wireless networks. In *2015 11th International Conference on Network and Service Management* (2015), pp. 384–389.

[51] GANIN, Y., USTINOVA, E., AJAKAN, H., GERMAIN, P., LAROCHELLE, H., LAVIOLETTE, F., MARCH, M., AND LEMPITSKY, V. Domain-adversarial training of neural networks. *Journal of Machine Learning Research 17*, 59 (2016), 1–35.

[52] GAO, N., MENG, S., LI, C., MENG, S., TANG, W., JIN, S., AND MATTHAIOU, M. Ris-assisted wireless link signatures for specific emitter identification. *IEEE Transactions on Wireless Communications 23*, 12 (2024), 17872–17883.

[53] GE, S., LI, K., AND RUM, S. N. B. M. Deep Learning Approach in DOA Estimation: A Systematic Literature Review. *Mobile Information Systems 2021*, 1 (2021), 6392875.

[54] GIEST, S. The intersection of digital and social infrastructures in (a)spatial policymaking. *Policy Sciences 58*, 2 (June 2025), 369–384.

[55] GIL-MARTÍNEZ, A., POVEDA-GARCÍA, M., LÓPEZ-PASTOR, J. A., SÁNCHEZ-AARNOUTSE, J. C., AND GÓMEZ-TORNERO, J. L. Wi-fi direction finding with frequency-scanned antenna and channel-hopping scheme. *IEEE Sensors Journal 22*, 6 (2022), 5210–5222.

[56] GONG, J., XU, X., AND LEI, Y. Unsupervised specific emitter identification method using radio-frequency fingerprint embedded infogan. *IEEE Transactions on Information Forensics and Security 15* (2020), 2898–2913.

[57] GOODFELLOW, I. J., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A., AND BENGIO, Y. Generative adversarial nets. *Advances in Neural Information Processing Systems 27* (2014).

[58] GORI, M., MONFARDINI, G., AND SCARSELLI, F. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks* (2005), vol. 2, pp. 729–734.

[59] GRITSENKO, A., WANG, Z., JIAN, T., DY, J., CHOWDHURY, K., AND IOANNIDIS, S. Finding a 'new' needle in the haystack: Unseen radio detection in large populations using deep learning. In *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)* (2019), pp. 1–10.

[60] GROVER, K., LIM, A., AND YANG, Q. Jamming and anti-jamming techniques in wireless networks: a survey. *International Journal of Ad Hoc and Ubiquitous Computing 17*, 4 (2014), 197–215.

[61] GULRAJANI, I., AHMED, F., ARJOVSKY, M., DUMOULIN, V., AND COURVILLE, A. C. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems* (2017), I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30, Curran Associates, Inc.

[62] GUO, F., YU, F. R., ZHANG, H., LI, X., JI, H., AND LEUNG, V. C. M. Enabling massive iot toward 6g: A comprehensive survey. *IEEE Internet of Things Journal* (2021).

[63] GUO, H., ZHOU, Z., ZHAO, D., AND GAALOUL, W. Egnn: Energy-efficient anomaly detection for iot multivariate time series data using graph neural network. *Future Generation Computer Systems 151* (2024), 45–56.

[64] HALL, J., BARBEAU, M., KRANAKIS, E., ET AL. Detection of transient in radio frequency fingerprinting using signal phase. *Wireless and optical communications 9* (2003), 13.

[65] HAMILTON, W., YING, Z., AND LESKOVEC, J. Inductive Representation Learning on Large Graphs. *Advances in Neural Information Processing Systems 30* (2017), 1–11.

[66] HAN, F., YE, P., DUAN, S., AND WANG, L. Ada-iD: Active Domain Adaptation for Intrusion Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia* (New York, NY, USA, Oct. 2024), MM '24, Association for Computing Machinery, pp. 7404–7413.

[67] HAN, G., XIAO, L., AND POOR, H. V. Two-dimensional anti-jamming communication based on deep reinforcement learning. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (2017), IEEE, pp. 2087–2091.

[68] HAN, X., CHEN, X., AND LIU, L.-P. Gan ensemble for anomaly detection. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2021), pp. 4090–4097.

[69] HAN, Z., XIAO, J., ZHAO, Q., CUI, Z., WANG, Y., ZHANG, D., AND DING, W. Open-world radio frequency fingerprint identification via augmented semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2025), pp. 264–272.

[70] HANNA, S., KARUNARATNE, S., AND CABRIC, D. Open set wireless transmitter authorization: Deep learning approaches and dataset considerations. *IEEE Transactions on Cognitive Communications and Networking 7*, 1 (2021), 59–72.

[71] HANNA, S., KARUNARATNE, S., AND CABRIC, D. Wisig: A large-scale wifi signal dataset for receiver and channel agnostic rf fingerprinting. *IEEE Access 10* (2022), 22808–22818.

[72] HASSAN, W. H., ET AL. Current research on internet of things (iot) security: A survey. *Computer networks 148* (2019), 283–294.

[73] HASSIJA, V., CHAMOLA, V., SAXENA, V., JAIN, D., GOYAL, P., AND SIKDAR, B. A survey on iot security: Application areas, security threats, and solution architectures. *IEEE Access 7* (2019), 82721–82743.

[74] HAVIL, J. Gamma: exploring euler's constant. In *Gamma*. Princeton University Press, 2010.

[75] HE, K., ZHANG, X., REN, S., AND SUN, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016), pp. 770–778.

[76] HEUSEL, M., RAMSAUER, H., UNTERTHINER, T., NESSLER, B., AND HOCHREITER, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems* (2017), vol. 30, Curran Associates, Inc.

[77] HOELTGEBAUM, H., ADAMS, N., AND FERNANDES, C. Estimation, forecasting, and anomaly detection for nonstationary streams using adaptive estimation. *IEEE Transactions on Cybernetics 52*, 8 (2022), 7956–7967.

[78] HUANG, R., PENG, X., CHAI, Z., LI, M., REN, J., AND YANG, X. Radio frequency fingerprint extraction and authentication towards open set in noisy channels. *Digital Signal Processing 146* (Mar. 2024), 104363.

[79] HUANG, X., YANG, X., CAO, L., AND LU, W. Pseudo noise subspace based DOA estimation for unfolded coprime linear arrays. *IEEE Wireless Communications Letters 10*, 11 (2021), 2335–2339.

[80] HUSSAIN, A. M., ABUGHANAM, N., AND PAPADIMITRATOS, P. Edge ai-based radio frequency fingerprinting for iot networks, 2024.

[81] IFTIKHAR, U., AND ALI, S. A. Enhanced cyber threat detection system leveraging machine learning using data augmentation. *International Journal of Advanced Computer Science and Applications 16*, 2 (2025).

[82] JAGANNATH, A., JAGANNATH, J., AND KUMAR, P. S. P. V. A Comprehensive Survey on Radio Frequency (RF) Fingerprinting: Traditional Approaches, Deep Learning, and Open Challenges, Sept. 2022.

[83] JI, C., WANG, R., AND WU, H. Perturb more, trap more: Understanding behaviors of graph neural networks. *Neurocomputing 493*, C (2022), 59–75.

[84] JIANG, F., WU, H., TANG, Y., LIU, W., REN, W., AND YANG, Y. Graph attention networks for invisible attack identification in smart grids. In *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (2024), pp. 3924–3929.

[85] JIANG, Q., AND SHA, J. Radio frequency fingerprint identification based on variational autoencoder for gnss. *IEEE Geoscience and Remote Sensing Letters 21* (2024), 1–4.

[86] JIANG, W. Graph-based Deep Learning for Communication Networks: A Survey. *Computer Communications 185* (Mar. 2022), 40–54.

[87] JIANG, X., LIU, S., GEMBER-JACOBSON, A., BHAGOJI, A. N., SCHMITT, P., BRONZINO, F., AND FEAMSTER, N. Netdiffusion: Network data augmentation through protocol-constrained traffic generation. *Proceedings of the ACM on Measurement and Analysis of Computing Systems 8*, 1 (2024), 1–32.

[88] JIN, Q., LIN, R., AND YANG, F. E-wacgan: Enhanced generative model of signaling data based on wgan-gp and acgan. *IEEE Systems Journal 14*, 3 (2020), 3289–3300.

[89] KADHIM, A. N., AND SADKHAN, S. B. Security threats in wireless network communication-status, challenges, and future trends. In *2021 International Conference on Advanced Computer Applications* (2021), pp. 176–181.

[90] KARAMAN, B., BASTURK, I., TASKIN, S., ZEYDAN, E., KARA, F., BEYAZIT, E. A., CAMELO, M., BJÖRNSON, E., AND YANIKOMEROGLU, H. Solutions for Sustainable and Resilient Communication Infrastructure in Disaster Relief and Management Scenarios, Oct. 2024.

[91] KARKOUCH, A., MOUSANNIF, H., MOATASSIME, H. A., AND NOEL, T. Data quality in internet of things: A state-of-the-art survey. *Journal of Network and Computer Applications 73* (2016), 57–81.

[92] KIM, J., JEONG, K., CHOI, H., AND SEO, K. Gan-based anomaly detection in imbalance problems. In *Computer Vision – ECCV 2020 Workshops* (Berlin, Heidelberg, 2020), Springer-Verlag, p. 128–145.

[93] KIM, K. Gan based augmentation for improving anomaly detection accuracy in host-based intrusion detection systems. *International Journal of Engineering Research and Technology 13* (11 2020), 3987.

[94] KIPF, T. N., AND WELLING, M. Semi-Supervised Classification with Graph Convolutional Networks. In *International Conference on Learning Representations* (2016), pp. 1–14.

[95] KIPF, T. N., AND WELLING, M. Variational graph auto-encoders. *NIPS Workshop on Bayesian Deep Learning* (2016).

[96] KONG, R., AND CHEN, H. Deepcrf: Deep learning-enhanced csi-based rf fingerprinting for channel-resilient wifi device identification. *IEEE Transactions on Information Forensics and Security 20* (2025), 264–278.

[97] KOUSIAS, K., RAJIULLAH, M., CASO, G., ALI, U., ALAY, O., BRUNSTROM, A., DE NARDIS, L., NERI, M., AND DI BENEDETTO, M.-G. A large-scale dataset of 4g, nb-iot, and 5g non-standalone network measurements. *IEEE Communications Magazine 62*, 5 (2024), 44–49.

[98] KRIM, H., AND VIBERG, M. Two decades of array signal processing research: the parametric approach. *IEEE Signal Processing Magazine 13*, 4 (1996), 67–94.

[99] LAI, Y., LIN, X., YANG, R., AND WANG, H. Efficient topology-aware data augmentation for high-degree graph neural networks. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (2024), p. 1463–1473.

[100] LEE, H., KIM, Y., SEOL, S., AND CHUNG, J. Deep learning-based Direction-of-arrival estimation for far-field sources under correlated near-field interferences. *ICT Express 9*, 4 (Aug. 2023), 741–747.

[101] LI, C., XU, K., ZHU, J., LIU, J., AND ZHANG, B. Triple generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence 44*, 12 (2022), 9629–9640.

[102] LI, J., AND STOICA, P. *Robust adaptive beamforming*. John Wiley & Sons, 2005.

[103] LI, J., STOICA, P., AND WANG, Z. On robust capon beamforming and diagonal loading. *IEEE Transactions on Signal Processing 51*, 7 (2003), 1702–1715.

[104] LI, S., LIU, Y., YOU, L., WANG, W., DUAN, H., AND LI, X. Covariance matrix reconstruction for DOA estimation in hybrid massive MIMO systems. *IEEE Wireless Communications Letters 9*, 8 (2020), 1196–1200.

[105] LI, X., CHEN, J., LING, X., AND WU, T. Deep Reinforcement Learning-Based Anti-Jamming Algorithm Using Dual Action Network. *IEEE Transactions on Wireless Communications 22*, 7 (July 2023), 4625–4637.

[106] LI, Y., FANG, R., SONG, Q., AND YANG, X. Stgcn-based link flooding attack detection and mitigation in software-defined network. In *2024 IEEE 23rd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2024), pp. 346–353.

[107] LI, Y., PODILCHAK, S. K., ZHANG, J., COTTON, S. L., RATNARAJAH, T., AND DING, Y. Rffi protocols using antenna mutual coupling and power amplifier nonlinear memory effects. *IEEE Communications Letters* (2025).

[108] LI, Y., XU, Y., LI, G., GONG, Y., LIU, X., WANG, H., AND LI, W. Dynamic Spectrum Anti-Jamming Access With Fast Convergence: A Labeled Deep Reinforcement Learning Approach. *IEEE Transactions on Information Forensics and Security 18* (2023), 5447–5458.

[109] LICHTMAN, M., RAO, R. M., MAROJEVIC, V., REED, J. H., AND JOVER, R. P. 5G NR Jamming, Spoofing, and Sniffing: Threat Assessment and Mitigation, Apr. 2018.

[110] LIM, W., YONG, K. S. C., LAU, B. T., AND TAN, C. C. L. Future of generative adversarial networks (GAN) for anomaly detection in network security: A review. *Computers & Security 139* (Apr. 2024), 103733.

[111] LIU, F., AND SHROFF, N. Data poisoning attacks on stochastic bandits. In *International Conference on Machine Learning* (2019), PMLR, pp. 4042–4050.

[112] LIU, H., HUANG, Y., GONG, Y., ZHAI, Y., AND LU, J. Hydra: A hybrid dual-mode network for closed-and open-set rffi with optimized vmd. *arXiv preprint arXiv:2507.12133* (2025).

[113] LIU, X., XU, Y., JIA, L., WU, Q., AND ANPALAGAN, A. Anti-jamming communications using spectrum waterfall: A deep reinforcement learning approach. *IEEE Communications Letters 22*, 5 (2018), 998–1001.

[114] LOSCRI, V., AND BIAGI, M. Libero: Light bias as effective countermeasure against eavesdropper attacks. *IEEE Transactions on Communications 72*, 12 (2024), 7882–7893.

[115] LU, Q., YANG, Z., ZHANG, H., CHEN, F., AND XIAN, H. Mrfe: A deep-learning-based multidimensional radio frequency fingerprinting enhancement approach for iot device identification. *IEEE Internet of Things Journal 11*, 18 (2024), 30442–30454.

[116] LV, Q., QUAN, Y., FENG, W., SHA, M., DONG, S., AND XING, M. Radar deception jamming recognition based on weighted ensemble CNN with transfer learning. *IEEE Transactions on Geoscience and Remote Sensing 60* (2022), 1–11.

[117] MA, D., WANG, Y., AND WU, S. Against jamming attack in wireless communication networks: A reinforcement learning approach. *Electronics 13*, 7 (2024), 1209.

[118] MA, J., WANG, M., CHEN, Y., AND WANG, H. Deep convolutional network-assisted multiple direction-of-arrival estimation. *IEEE Signal Processing Letters 31* (2024), 576–580.

[119] MADURANGA, M., ORUTHOTA, U., LAKMAL, H., AND KULATUNGA, S. Rssi-based indoor localization using deep learning with a custom loss function. In *2024 8th SLAAI International Conference on Artificial Intelligence (SLAAI-ICAI)* (2024), pp. 1–5.

[120] MARFIEVICI, R., MURPHY, A. L., PICCO, G. P., OSSI, F., AND CAGNACCI, F. How environmental factors impact outdoor wireless sensor networks: A case study. In *2013 IEEE 10th International Conference on Mobile Ad-Hoc and Sensor Systems* (2013), pp. 565–573.

[121] MARTIN, A., DODDINGTON, G., KAMM, T., ORDOWSKI, M., AND PRZYBOCKI, M. The det curve in assessment of detection task performance. In *Proceedings of the Fifth European Conference on Speech Communication and Technology (Eurospeech 1997)* (Rhodes, Greece, September 1997), pp. 1895–1898.

[122] MERCHANT, K., AND NOUSAIN, B. Toward receiver-agnostic rf fingerprint verification. In *2019 IEEE Globecom Workshops (GC Wkshps)* (2019), pp. 1–6.

[123] MERCHANT, K., REVAY, S., STANTCHEV, G., AND NOUSAIN, B. Deep learning for rf device fingerprinting in cognitive communication networks. *IEEE Journal of Selected Topics in Signal Processing 12*, 1 (2018), 160–167.

[124] MESTRE, X., AND VALLET, P. On the resolution probability of conditional and unconditional maximum likelihood DoA estimation. *IEEE Transactions on Signal Processing 68* (2020), 4656–4671.

[125] MIAO, Y., LI, H., AND MANDIC, D. P. GPINN: physics-informed neural network with graph embedding. In *International Joint Conference on Neural Networks* (2024), pp. 1–8.

[126] MICHALOLIAKOS, A., ROGALIN, R., ZHANG, Y., PSOUNIS, K., AND CAIRE, G. Performance modeling of next-generation wifi networks. *Computer Networks 105* (2016), 150–165.

[127] MIRZA, M., AND OSINDERO, S. Conditional generative adversarial nets, 2014.

[128] NAGHDI, S., AND O'KEEFE, K. Combining Multichannel RSSI and Vision with Artificial Neural Networks to Improve BLE Trilateration. *Sensors 22*, 12 (Jan. 2022), 4320. Number: 12 Publisher: Multidisciplinary Digital Publishing Institute.

[129] NAMVAR, N., SAAD, W., BAHADORI, N., AND KELLEY, B. Jamming in the internet of things: A game-theoretic perspective. In *2016 IEEE Global Communications Conference (GLOBECOM)* (2016), pp. 1–6.

[130] NEKOVEE, M., AND SAKSENA, R. S. Simulations of large-scale wifi-based wireless networks: Interdisciplinary challenges and applications. *Future Generation Computer Systems 26*, 3 (2010), 514–520.

[131] NENG-JING, L., AND YI-TING, Z. A survey of radar ECM and ECCM. *IEEE Transactions on Aerospace and Electronic Systems 31*, 3 (1995), 1110–1120.

[132] O'REILLY, C., GLUHAK, A., IMRAN, M. A., AND RAJASEGARAR, S. Anomaly detection in wireless sensor networks in a non-stationary environment. *IEEE Communications Surveys & Tutorials 16*, 3 (2014), 1413–1432.

[133] PANG, G., SHEN, C., CAO, L., AND HENGEL, A. V. D. Deep Learning for Anomaly Detection: A Review. *ACM Computing Surveys 54*, 2 (Mar. 2021), 38:1–38:38.

[134] PAPAGEORGIOU, G. K., SELLATHURAI, M., AND ELDAR, Y. C. Deep networks for direction-of-arrival estimation in low SNR. *IEEE Transactions on Signal Processing 69* (2021), 3714–3729.

[135] PARK, D., YANG, E., AHN, S., AND CHUN, J. Adaptive beamforming for low-angle target tracking under multipath interference. *IEEE Transactions on Aerospace and Electronic Systems 50*, 4 (2014), 2564–2577.

[136] PENG, L., ZHANG, J., LIU, M., AND HU, A. Deep learning based rf fingerprint identification using differential constellation trace figure. *IEEE Transactions on Vehicular Technology 69*, 1 (2020), 1091–1095.

[137] PHAM, T. M. T., PREMKUMAR, K., NAILI, M., AND YANG, J. Time to retrain? detecting concept drifts in machine learning systems, 2024.

[138] PHUNG, C.-D., YELLAS, N.-E.-H., RUBA, S. B., AND SECCI, S. An open dataset for beyond-5g data-driven network automation experiments. In *2022 1st International Conference on 6G Networking (6GNet)* (2022), pp. 1–4.

[139] PIRAYESH, H., AND ZENG, H. Jamming Attacks and Anti-Jamming Strategies in Wireless Networks: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials 24*, 2 (2022), 767–809.

[140] POLAK, A. C., DOLATSHAHI, S., AND GOECKEL, D. L. Identifying wireless users via transmitter imperfections. *IEEE Journal on Selected Areas in Communications 29*, 7 (2011), 1469–1479.

[141] QU, Q., WEI, S., LIU, S., LIANG, J., AND SHI, J. JRNet: Jamming recognition networks for radar compound suppression jamming signals. *IEEE Transactions on Vehicular Technology 69*, 12 (2020), 15035–15045.

[142] QU, Y., MA, H., JIANG, Y., AND BU, Y. A Network Intrusion Detection Method Based on Domain Confusion. *Electronics 12*, 5 (Jan. 2023), 1255.

[143] RAZA, U., CAMERRA, A., MURPHY, A. L., PALPANAS, T., AND PICCO, G. P. Practical data prediction for real-world wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering 27*, 8 (2015), 2231–2244.

[144] ROY, D., MUKHERJEE, T., CHATTERJEE, M., BLASCH, E., AND PASILIAO, E. Rfal: Adversarial learning for rf transmitter identification and classification. *IEEE Transactions on Cognitive Communications and Networking 6*, 2 (2019), 783–801.

[145] RUSSO, D., VAN ROY, B., KAZEROUNI, A., OSBAND, I., AND WEN, Z. A tutorial on thompson sampling. *arXiv preprint arXiv:1707.02038* (2017).

[146] RYOO, D., YOO, Y., PAEK, J., AND BAHK, S. Spade: Secure periodic advertising using coded time-channel rendezvous for ble audio. In *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)* (2023), pp. 39–46.

[147] SAITO, T., AND REHMSMEIER, M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one 10*, 3 (2015), e0118432.

[148] SANKHE, K., BELGIOVINE, M., ZHOU, F., ANGIOLONI, L., RESTUCCIA, F., D'ORO, S., MELODIA, T., IOANNIDIS, S., AND CHOWDHURY, K. No radio left behind: Radio fingerprinting through deep learning of physical-layer hardware impairments. *IEEE Transactions on Cognitive Communications and Networking 6*, 1 (2020), 165–178.

[149] SCHMIDT, R. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation 34*, 3 (1986), 276–280.

[150] SCHUMMER, P., DEL RIO, A., SERRANO, J., JIMENEZ, D., SANCHEZ, G., AND LLORENTE, A. Machine learning-based network anomaly detection: Design, implementation, and evaluation. *AI* (2024).

[151] SHAO, G., CHEN, Y., AND WEI, Y. Deep fusion for radar jamming signal classification based on CNN. *IEEE Access 8* (2020), 117236–117244.

[152] SHEN, G., ZHANG, J., AND MARSHALL, A. Deep learning - powered radio frequency fingerprint identification: Methodology and case study. *IEEE Communications Magazine 61*, 9 (2023), 170–176.

[153] SHEN, G., ZHANG, J., MARSHALL, A., AND CAVALLARO, J. R. Towards scalable and channel-robust radio frequency fingerprint identification for lora. *IEEE Transactions on Information Forensics and Security 17* (2022), 774–787.

[154] SHEN, G., ZHANG, J., MARSHALL, A., PENG, L., AND WANG, X. Radio frequency fingerprint identification for lora using spectrogram and cnn. In *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications* (2021), pp. 1–10.

[155] SHEN, G., ZHANG, J., MARSHALL, A., WOODS, R., CAVALLARO, J., AND CHEN, L. Towards receiver-agnostic and collaborative radio frequency fingerprint identification. *IEEE Transactions on Mobile Computing 23*, 7 (2024), 7618–7634.

[156] SHEN, G., ZHANG, J., WANG, X., AND MAO, S. Federated Radio Frequency Fingerprint Identification Powered by Unsupervised Contrastive Learning. *IEEE Transactions on Information Forensics and Security 19* (2024), 9204–9215.

[157] SHEN, Y., LUO, P., YAN, J., WANG, X., AND TANG, X. Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 821–830.

[158] SHI, F., ZHOU, Z., GUO, J., LI, R., ZHANG, Z., LI, S., LIU, Q., AND BAO, X. Lightrl-ad: A lightweight online reinforcement learning approach for autonomous defense against network attacks. In *2024 IEEE 23rd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2024), pp. 1614–1621.

[159] SONG, H., JIANG, Z., MEN, A., AND YANG, B. A hybrid semi-supervised anomaly detection model for high-dimensional data. *Computational Intelligence and Neuroscience 2017*, 1 (2017), 8501683.

[160] STADDON, E. *Threat detection, identification and quarantine in wireless IoT based Critical Infrastructures*. PhD thesis, Université de Lille, 2022.

[161] STOICA, P., AND NEHORAI, A. Music, maximum likelihood, and cramer-rao bound. *IEEE Transactions on Acoustics, Speech, and Signal Processing 37*, 5 (1989), 720–741.

[162] SUN, C., CHEN, X., WANG, W., AND YIN, H. A transformer-based multi-feature extraction neural network for bluetooth devices identification. In *2023 9th International Conference on Computer and Communications (ICCC)* (2023), pp. 1469–1473.

[163] SUN, L., WANG, X., HUANG, Z., AND LI, B. Radio-frequency fingerprint extraction based on feature inhomogeneity. *IEEE Internet of Things Journal 9*, 18 (2022), 17292–17308.

[164] SZEGEDY, C., ZAREMBA, W., SUTSKEVER, I., BRUNA, J., ERHAN, D., GOODFELLOW, I. J., AND FERGUS, R. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014* (2014).

[165] TALAGALA, P. D., HYNDMAN, R. J., AND SMITH-MILES, K. Anomaly Detection in High-Dimensional Data. *Journal of Computational and Graphical Statistics* (June 2021).

[166] TANG, X., REN, P., AND HAN, Z. Jamming mitigation via hierarchical security game for iot communications. *IEEE Access 6* (2018), 5766–5779.

[167] TEDESCHI, P., OLIGERI, G., AND DI PIETRO, R. Localization of a Power-Modulated Jammer. *Sensors 22*, 2 (Jan. 2022), 646. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[168] THANKAPPAN, M., RIFÀ-POUS, H., AND GARRIGUES, C. A signature-based wireless intrusion detection system framework for multi-channel man-in-the-middle attacks against protected wi-fi networks. *IEEE Access 12* (2024), 23096–23121.

[169] THING, V. L. L. Ieee 802.11 network anomaly detection and attack classification: A deep learning approach. In *2017 IEEE Wireless Communications and Networking Conference* (2017), pp. 1–6.

[170] THOTTAN, M., AND JI, C. Anomaly detection in ip networks. *IEEE Transactions on Signal Processing 51*, 8 (2003), 2191–2204.

[171] TIAN, T., GONG, Y., JIANG, B., LIU, J., FENG, H., AND LU, Z. Insider threat detection based on heterogeneous graph neural network. In *2023 IEEE 22nd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2023), pp. 628–635.

[172] TOLDOV, V., CLAVIER, L., LOSCRÌ, V., AND MITTON, N. A thompson sampling approach to channel exploration-exploitation problem in multihop cognitive radio networks. In *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)* (2016), IEEE, pp. 1–6.

[173] TZENG, E., HOFFMAN, J., SAENKO, K., AND DARRELL, T. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 2962–2971.

[174] UPADHYAYA, B., SUN, S., AND SIKDAR, B. Machine learning-based jamming detection in wireless iot networks. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)* (2019), pp. 1–5.

[175] VELIČKOVIĆ, P., CUCURULL, G., CASANOVA, A., ROMERO, A., LIÒ, P., AND BENGIO, Y. Graph Attention Networks. In *International Conference on Learning Representations* (2018), pp. 1–12.

[176] WANG, A., AND JIANG, W. Research of teaching on network course based on ns-3. In *2009 First International Workshop on Education Technology and Computer Science* (2009), vol. 2, pp. 629–632.

[177] WANG, H., CHEN, B., YE, Q., AND HUANG, F. Mrednet: A Multi-Residual Encoder-Decoder Network for recovering LFM interfered by intra-pulse retransmission interferences. *Signal Processing 217* (2024), 109329.

[178] WANG, S., LI, C., AND LIM, A. A model for non-stationary time series and its applications in filtering and anomaly detection. *IEEE Transactions on Instrumentation and Measurement 70* (2021), 1–11.

[179] WANG, X., XU, Y., CHEN, J., LI, C., LIU, X., LIU, D., AND XU, Y. Mean field reinforcement learning based anti-jamming communications for ultra-dense internet of things in 6g. In *2020 International Conference on Wireless Communications and Signal Processing (WCSP)* (2020), IEEE, pp. 195–200.

[180] WEN, J., KANG, J., NIYATO, D., ZHANG, Y., WANG, J., SIKDAR, B., AND ZHANG, P. Generative ai for data augmentation in wireless networks: Analysis, applications, and case study, 2024.

[181] WONG, C., WANG, M., HOSSAIN, M. F., AND MUNASINGHE, K. Internet of things anomaly detection enhancement using gan-based data augmentation. In *2024 17th International Conference on Signal Processing and Communication System (ICSPCS)* (2024).

[182] WU, L., LIU, Z.-M., AND HUANG, Z.-T. Deep convolution network for direction of arrival estimation with sparse prior. *IEEE Signal Processing Letters 26*, 11 (2019), 1688–1692.

[183] WU, Y., DAI, H.-N., AND TANG, H. Graph neural networks for anomaly detection in industrial internet of things. *IEEE Internet of Things Journal 9*, 12 (2021), 9214–9231.

[184] XIANG, H. Steering vector embedding complex-valued generative adversarial nets for radar signal generation and large-scale phase enhancement learning for super-resolution DOA estimation. *Applied Intelligence* (2024), 1–12.

[185] XIANG, H., CHEN, B., YANG, M., AND XU, S. Angle separation learning for coherent DOA estimation with deep sparse prior. *IEEE Communications Letters 25*, 2 (2021), 465–469.

[186] XIAO, L., GREENSTEIN, L. J., MANDAYAM, N. B., AND TRAPPE, W. Using the physical layer for wireless authentication in time-variant channels. *IEEE Transactions on Wireless Communications 7*, 7 (2008), 2571–2579.

[187] XIAO, L., WAN, X., SU, W., TANG, Y., ET AL. Anti-jamming underwater transmission with mobility and learning. *IEEE Communications Letters 22*, 3 (2018), 542–545.

[188] XIE, L., PENG, L., ZHANG, J., AND HU, A. Radio frequency fingerprint identification for Internet of Things: A survey. *Security and Safety 3* (2024), 2023022.

[189] XIE, M., HAN, S., TIAN, B., AND PARVIN, S. Anomaly detection in wireless sensor networks: A survey. *Journal of Network and Computer Applications 34*, 4 (2011), 1302–1325. Advanced Topics in Cloud Computing.

[190] XU, J., DAI, H., AND YING, W.-H. Multi-layer neural network for received signal strength-based indoor localisation. *IET Communications 10* (03 2016).

[191] XU, J., LOU, H., ZHANG, W., AND SANG, G. An intelligent anti-jamming scheme for cognitive radio based on deep reinforcement learning. *IEEE Access 8* (2020), 202563–202572.

[192] XU, K., HU, W., LESKOVEC, J., AND JEGELKA, S. How powerful are graph neural networks? In *International Conference on Learning Representations* (2019).

[193] XU, Q., ZHENG, R., SAAD, W., AND HAN, Z. Device fingerprinting in wireless networks: Challenges and opportunities. *IEEE Communications Surveys & Tutorials 18*, 1 (2016), 94–104.

[194] XU, S., BRIGHENTE, A., CHEN, B., CONTI, M., CHENG, X., AND ZHU, D. Deep neural networks for direction of arrival estimation of multiple targets with sparse prior for line-of-sight scenarios. *IEEE Transactions on Vehicular Technology 72*, 4 (2023), 4683–4696.

[195] XU, W. Jamming attack defense. In *Encyclopedia of Cryptography, Security and Privacy*. Springer, 2025, pp. 1318–1325.

[196] XU, W., MA, K., TRAPPE, W., AND ZHANG, Y. Jamming sensor networks: attack and defense strategies. *IEEE Network 20*, 3 (2006), 41–47.

[197] XU, W., TRAPPE, W., ZHANG, Y., AND WOOD, T. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing* (2005), pp. 46–57.

[198] XU, Y., WANG, C., LIANG, J., YUE, K., LI, W., ZHENG, S., AND ZHAO, Z. Deep Reinforcement Learning Based Decision Making for Complex Jamming Waveforms. *Entropy 24*, 10 (Oct. 2022), 1441.

[199] XUE, C., LI, T., LI, Y., RUAN, Y., AND ZHANG, R. Radio frequency identification for drones using spectrogram and cnn. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference* (2022), pp. 4564–4569.

[200] YAMAGUCHI, S., KANAI, S., AND EDA, T. Effective data augmentation with multi-domain learning gans. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2020), pp. 6566–6574.

[201] YANG, J., MEDYA, S., AND YE, W. Incorporating heterophily into graph neural networks for graph classification. In *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (2024), pp. 1544–1551.

[202] YANG, L., LI, Q., REN, X., FANG, Y., AND WANG, S. Mitigating receiver impact on radio frequency fingerprint identification via domain adaptation. *IEEE Internet of Things Journal 11*, 13 (2024), 24024–24034.

[203] YANG, L., ZHOU, W., PENG, W., NIU, B., GU, J., WANG, C., CAO, X., AND HE, D. Graph neural networks beyond compromise between attribute and topology. In *Proceedings of the ACM Web Conference 2022* (2022), p. 1127–1135.

[204] YANG, M., AND ZHANG, J. Data anomaly detection in the internet of things: A review of current trends and research challenges. *International Journal of Advanced Computer Science and Applications 14* (01 2023).

[205] YANG, T., ZHAO, J., WANG, X., AND XU, F. Deep learning based rff recognition with differential constellation trace figure towards closed and open set. In *2022 IEEE/CIC International Conference on Communications in China (ICCC)* (2022), pp. 908–913.

[206] YANG, W., ZHAO, J., XUE, S., WANG, X., AND NI, X. A new radio frequency fingerprint identification method based on ceemdan. In *2024 International Wireless Communications and Mobile Computing (IWCMC)* (2024), pp. 109–113.

[207] YANG, Z., ZHENG, S., ZHANG, L., ZHAO, Z., AND YANG, X. Adversarial attacks on deep learning-based DOA estimation with covariance input. *IEEE Signal Processing Letters 30* (2023), 1377–1381.

[208] ZHANG, J., SHEN, G., SAAD, W., AND CHOWDHURY, K. Radio frequency fingerprint identification for device authentication in the internet of things. *IEEE Communications Magazine 61*, 10 (2023), 110–115.

[209] ZHANG, J., WOODS, R., SANDELL, M., VALKAMA, M., MARSHALL, A., AND CAVALLARO, J. Radio frequency fingerprint identification for narrowband systems, modelling and classification. *IEEE Transactions on Information Forensics and Security 16* (2021), 3974–3987.

[210] ZHANG, Q., CHEN, B., ZHANG, T., CAO, K., DING, Y., GAO, T., AND ZHAO, Z. Generative Adversarial Network-Based Anomaly Detection and Forecasting with Unlabeled Data for 5G Vertical Applications. *Applied Sciences 13*, 19 (Jan. 2023), 10745.

[211] ZHANG, Y., LI, Q., LIU, H., YANG, L., AND YANG, J. Domain generalization for cross-receiver radio frequency fingerprint identification, 2025.

[212] ZHANG, Z., AND KRUNZ, M. Sigtam: A tampering attack on wi-fi preamble signaling and countermeasures. In *2022 IEEE Conference on Communications and Network Security (CNS)* (2022), pp. 1–9.

[213] ZHAO, T., SARKAR, S., KRIJESTORAC, E., AND CABRIC, D. Gan-rxa: A practical scalable solution to receiver-agnostic transmitter fingerprinting. *IEEE Transactions on Cognitive Communications and Networking 10*, 2 (2024), 403–416.

[214] ZHAO, Z., SINGH, S., LEE, H., ZHANG, Z., ODENA, A., AND ZHANG, H. Improved consistency regularization for gans. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2021), pp. 11033–11041.

[215] ZHENG, R., SUN, S., LIU, H., CHEN, H., SOLTANALIAN, M., AND LI, J. Antenna failure resilience: Deep learning-enabled robust doa estimation with single snapshot sparse arrays. In *2024 58th Asilomar Conference on Signals, Systems, and Computers* (2024), pp. 84–89.

[216] ZHENG, X., WU, B., ZHANG, A. X., AND LI, W. Improving robustness of GNN-based anomaly detection by graph adversarial training. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (2024), ELRA and ICCL, pp. 8902–8912.

[217] ZHOU, H., WANG, L., AND GUO, Z. Recognition of radar compound jamming based on convolutional neural network. *IEEE Transactions on Aerospace and Electronic Systems 59*, 6 (2023), 7380–7394.

[218] ZHOU, Q., LI, Y., AND NIU, Y. Intelligent anti-jamming communication for wireless sensor networks: A multi-agent reinforcement learning approach. *IEEE Open Journal of the Communications Society 2* (2021), 775–784.

[219] ZHOU, Z., DONG, C., MO, D., AND ZHENG, P. Privacy-preserving decision making based on q-learning in cloud computing. In *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2022), pp. 727–732.

[220] ZHU, J.-Y., PARK, T., ISOLA, P., AND EFROS, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision* (2017), pp. 2242–2251.

[221] ZHU, W., LI, J., WANG, Z., LI, Q., LIN, J., AND XIAOXIAO, S. Radio frequency fingerprints identification based on gan networks. In *2023 IEEE 24th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)* (2023), pp. 51–55.

[222] ZHU, W., LONG, B., AND XIAO, J. Spatial-temporal retentive heterogeneous graph convolutional network for traffic flow prediction. In *International Joint Conference on Neural Networks* (2024), pp. 1–8.

[223] ŁUKASZ LEMIESZEWSKI, HANNEBAUER, D., REMISZEWSKI, G., ET AL. Vulnerability of wi-fi wireless network to signal interference. *Journal of Engineering 360 1*, 1/24 (2024), 58–65.