

Article

A Hybrid Cognitive Radio and Multi-Agent Reinforcement Learning Framework for Jamming Resilience in Integrated FANET–IoT–IoV Systems

Rizwan Raza ¹, Zahoor-ur-Rehman ^{1,2}, Muddasar Naeem ^{3,*}, Farhan Aadil ^{1,4}, Faheem Shehzad ⁵
and Antonio Coronato ³

¹ Computer Science Department, COMSATS University Islamabad, Attock Campus, Attock 43600, Pakistan; sp24-pcs-002@cuiatk.edu.pk (R.R.); zahoor@sakarya.edu.tr (Z.-u.-R.); farhan.aadil@sivas.edu.tr (F.A.)

² Department of Computer Engineering, Faculty of Computer and Information Science, Sakarya University, Serdivan 54187, Turkey

³ AIR Lab, Università Giustino Fortunato, 82100 Benevento, Italy; a.coronato@unifortunato.eu

⁴ Department of Computer Engineering, Sivas University of Science and Technology, Sivas 58000, Turkey

⁵ Department of Electrical Engineering and Information Technology (DIETI), University of Naples “Federico II”, 80138 Naples, Italy; faheem.shehzad@unina.it

* Correspondence: m.naeem@unifortunato.eu

Abstract

Flying Ad-Hoc Networks (FANETs), Internet of Things (IoT), and Internet of Vehicles (IoV) are critical enablers of intelligent transportation and smart city ecosystems. Their reliance on shared wireless channels, however, exposes them to diverse jamming attacks that threaten communication reliability, mission effectiveness, and safety. This paper presents a comprehensive study of jamming threats in integrated FANET–IoT–IoV environments and analyzes conventional and advanced anti-jamming techniques across physical, link/MAC, spectral, spatial, temporal, and hybrid domains. To address the challenges posed by heterogeneous and dynamic network conditions, we propose a cross-layer anti-jamming framework that integrates Cognitive Radio (CR) for dynamic spectrum access and Multi-Agent Reinforcement Learning (MARL) for cooperative, adaptive decision-making. The framework employs a Perception Engine for local anomaly detection, a Cognitive Engine for constructing a collaborative jamming map, and a Decision and Action Engine for multi-agent DRL-based mitigation. Simulation results demonstrate that the proposed CR–MARL framework significantly improves packet delivery ratio, reduces latency, and adapts efficiently to varying jamming strategies, while maintaining low energy and computational overhead, making it suitable for resource-constrained UAVs, vehicles, and IoT sensors.

Keywords: FANET; IoT; IoV; 6G; jamming resilience; cognitive radio; multi-agent reinforcement learning



Academic Editor: Tianwen Zhang

Received: 23 April 2026

Revised: 6 July 2026

Accepted: 8 July 2026

Published: 10 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The convergence of FANETs, IoT, and IoV forms a complex and highly dynamic wireless network. FANETs consist of drones or UAVs that communicate with each other and with ground systems. IoT includes sensors, smart infrastructure, and monitoring devices deployed across cities and remote areas. IoV connects vehicles, roadside units, and traffic management systems. Together, these technologies support smart cities, intelligent transportation, disaster response, surveillance, and logistics operations [1].

This integrated network operates in an open wireless radio environment. Wireless channels are shared and broadcast in nature. Any nearby device can transmit signals on the same frequency. This makes the system vulnerable to interference. The problem becomes more severe due to the high mobility of UAVs and vehicles. Network topology changes frequently. Links are created and broken very fast. Maintaining stable and reliable communication is already challenging even without attacks [2].

Another major issue is resource heterogeneity. UAVs have limited battery life and computing power. IoT sensors are often low-cost and energy-constrained. Vehicles have stronger hardware but must support real-time and safety-critical communication. Because of these differences, a single security or defense mechanism cannot work equally well for all nodes. These characteristics make the integrated FANET–IoT–IoV [3] system a prime target for jamming attacks. Jamming is an intentional interference where an attacker transmits signals to block or disturb legitimate communication. The attacker does not need to break encryption. The attacker only needs to disrupt the wireless channel [4].

Jamming can have serious consequences. In FANETs, jamming can break coordination among drone swarms. Drones may lose control signals or fail to share sensor data. This can lead to mission failure or even physical crashes [5,6]. In IoV systems, jamming can block safety messages between vehicles. This may cause delayed reactions, traffic congestion, or accidents. In IoT networks, jamming can disable large sensor grids. This can stop monitoring of traffic, environment, or critical infrastructure [7]. These risks show that jamming is not only a communication problem. It is also a safety and reliability threat. Therefore, effective and adaptive anti-jamming techniques are essential for secure and dependable FANET–IoT–IoV operations.

This paper addresses the jamming problem in integrated FANET, IoT, and IoV networks. The focus is on improving communication reliability under intentional interference. The objectives of this work are clearly defined and aligned with the challenges discussed earlier.

First, this paper systematically categorizes jamming attacks across different network layers. The aerial layer include drone or UAV swarms also known as FANETs. These nodes suffer from high mobility and have limited energy. The vehicular layer includes connected and autonomous vehicles. These nodes require low latency and high reliability. The terrestrial layer includes IoT sensors and infrastructure. These nodes are resource constrained and often static. Each layer experiences jamming in different ways. This paper identifies the attack types, attack targets, and attack impacts for each layer.

Second, this paper analyzes existing jamming detection and mitigation techniques. It reviews signal-based, protocol-based, and learning-based methods. Detection approaches based on Received Signal Strength Indicator (RSSI), packet loss, and Signal-to-Interference-plus-Noise Ratio (SINR) are examined. Mitigation techniques such as frequency hopping, power control, routing adaptation, and spread spectrum are compared. The strengths and weaknesses of each method are discussed. Special attention is given to their suitability for heterogeneous networks.

Third, this paper proposes a novel hybrid anti-jamming framework. The framework combines Cognitive Radio and Machine Learning. Cognitive Radio enables dynamic spectrum sensing and channel selection. It helps nodes avoid jammed frequencies in real time. Machine Learning is used to detect jamming patterns and classify attack behavior. Lightweight models are selected to support resource-limited nodes. The framework operates across aerial, vehicular, and terrestrial layers. This enables cross-layer coordination and improved resilience.

Finally, the proposed framework is evaluated using simulations. A realistic network scenario with UAVs, vehicles, and IoT sensors is considered. Different jamming strategies are applied during the experiments. Performance is measured using packet delivery ratio, end-to-end delay, and throughput. Simulation results show a clear improvement in packet delivery ratio under jamming conditions. Communication latency is also reduced. These results confirm the effectiveness of the proposed anti-jamming framework.

The remainder of this paper is organized as follows: Section 2 covers background and motivation for the current research, Section 3 presents the system model and jamming threat analysis. Section 4 comprehensively reviews jamming detection techniques. Section 5 discusses anti-jamming mitigation strategies. Section 6 details the proposed hybrid framework. Section 7 presents simulation results and performance evaluation. Section 8 outlines open challenges, and Section 9 concludes the paper.

Table 1 lists the main acronyms and abbreviations used in the paper.

Table 1. List of Acronyms and Abbreviations.

Acronym	Abbreviation
AI	Artificial Intelligence
CAM	Cooperative Awareness Message
CDF	Cumulative Distribution Function
DQN	Deep Q-Network
DSSS	Direct Sequence Spread Spectrum
CR	Cognitive Radio
CST	Carrier Sensing Time
CTDE	Centralized Training and Decentralized Execution
CTS	Clear to Send
DENM	Decentralized Environmental Notification Message
DRL	Deep Reinforcement Learning
DSA	Dynamic Spectrum Access
FANET	Flying Ad hoc Network
FHSS	Frequency Hopping Spread Spectrum
ISM	Industrial Scientific Medical
IoT	Internet of Things
IoV	Internet of Vehicles
ML	Machine Learning
JSS	Jamming Suspicion Score
KS	Kolmogorov–Smirnov
LSTM	Long Short-Term Memory
MAC	Media Access Control
MARL	Multi-agent Reinforcement Learning
PPO	Proximal Policy Optimization
QAM	Quadrature Amplitude Modulation
PBFT	Practical Byzantine Fault Tolerance
PDR	Packet Delivery Ratio
QPSK	Quadrature Phase-shift Keying
RTS	Request to Send
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
RSU	Roadside Unit
SINR	Signal-to-Interference-plus-Noise Ratio
UAV	Unmanned Aerial Vehicle

2. Background and Motivation

2.1. Background

The rapid advancement of Unmanned Aerial Vehicle (UAV) technologies has transformed modern wireless communication systems and reshaped operational paradigms across military, commercial, and civilian domains. In military applications, UAV swarms are increasingly deployed for intelligence, surveillance, reconnaissance, target tracking, electronic warfare, and coordinated mission execution, where reliable and low-latency communication is essential for mission success [8–11]. In the commercial sector, UAVs have become integral to emerging applications such as parcel delivery, infrastructure inspection, precision agriculture, traffic monitoring, and aerial imaging, all of which require continuous information exchange between aerial platforms and ground stations to maintain service quality [12]. Similarly, civilian applications, including disaster management, environmental monitoring, wildfire detection, and search-and-rescue operations, increasingly rely on UAV-enabled communication networks to provide timely situational awareness and effective emergency response [8].

The communication backbone supporting these applications is primarily provided by Flying Ad Hoc Networks (FANETs), which enable autonomous communication and cooperation among multiple UAVs without requiring fixed network infrastructure. Compared with conventional Mobile Ad Hoc Networks (MANETs) and Vehicular Ad Hoc Networks (VANETs), FANETs exhibit distinctive characteristics such as high node mobility, three-dimensional movement, rapidly changing network topology, intermittent connectivity, dynamic wireless channels, and stringent Quality-of-Service (QoS) requirements. These characteristics make FANETs highly flexible and scalable but simultaneously introduce significant challenges in maintaining reliable and secure wireless communication under highly dynamic operating environments [13].

The continuous evolution of smart cities and intelligent transportation systems has further expanded the operational scope of UAVs beyond standalone aerial networks. Modern UAVs increasingly cooperate with Internet of Things (IoT) devices for environmental sensing, data collection, and infrastructure monitoring while simultaneously interacting with Internet of Vehicles (IoV) platforms to support traffic management, autonomous driving, emergency communication, and intelligent transportation services. Consequently, future wireless communication systems are evolving toward heterogeneous FANET-IoT-IoV ecosystems in which aerial platforms, ground sensors, connected vehicles, and roadside infrastructure collaboratively exchange mission-critical information. Such heterogeneous integration significantly enhances sensing capability, communication coverage, operational efficiency, and service intelligence. However, it also introduces substantial challenges arising from heterogeneous communication protocols, diverse computational capabilities, varying mobility characteristics, resource constraints, and different Quality-of-Service requirements across interconnected network domains.

Despite these technological advances, the shared and broadcast nature of wireless communication makes UAV networks particularly vulnerable to cyber-physical attacks. Among the various security threats, jamming attacks remain one of the most severe because they directly target communication availability by intentionally transmitting high-power interference signals over legitimate communication channels. Such malicious transmissions degrade signal quality, reduce throughput, increase packet loss, interrupt channel access, and may ultimately disconnect communication links among cooperating UAVs. Since UAV communications rely on open wireless channels and rapidly changing network topologies, adversaries can exploit these characteristics to launch effective denial-of-service attacks with relatively limited resources [1,6,14].

The threat becomes even more critical in integrated FANET–IoT–IoV environments. UAVs continuously collect information from distributed IoT sensors while simultaneously exchanging data with connected vehicles and roadside infrastructure. Consequently, communication disruption within one network layer can propagate throughout the interconnected system, resulting in degraded situational awareness, increased transmission latency, reduced packet delivery ratio, interrupted mission execution, and compromised operational safety. Furthermore, the high mobility of UAVs continuously changes wireless channel characteristics, making communication links more susceptible to intentional interference and significantly increasing the complexity of spectrum management and communication resilience [15].

Traditional anti-jamming techniques, including Frequency-Hopping Spread Spectrum (FHSS) and Direct-Sequence Spread Spectrum (DSSS), have been widely employed to improve communication robustness in wireless networks. These techniques utilize predefined hopping sequences, spreading codes, or fixed threshold mechanisms to avoid interference and perform satisfactorily against simple and stationary jammers. However, they remain fundamentally reactive and cannot effectively adapt to intelligent adversaries capable of observing channel conditions, learning communication patterns, and dynamically modifying jamming strategies according to network behavior [16]. Recent studies have further demonstrated that modern jammers can selectively target critical communication links, exploit spectrum utilization patterns, and launch adaptive interference attacks that significantly degrade network performance, rendering conventional rule-based defense mechanisms increasingly ineffective [13,15].

These challenges have motivated a paradigm shift from conventional static defense mechanisms toward intelligent, adaptive, and learning-driven anti-jamming solutions. Recent advances in Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL), and Deep Reinforcement Learning (DRL) have demonstrated remarkable capabilities in extracting complex wireless signal characteristics, identifying hidden interference patterns, predicting malicious behaviors, and learning adaptive communication strategies directly from environmental observations [1,4,5,17]. Compared with conventional approaches, learning-based techniques enable proactive rather than reactive interference mitigation by continuously adapting to dynamic wireless environments and evolving attack strategies. Consequently, AI-driven communication intelligence has emerged as one of the most promising directions for improving communication resilience in future heterogeneous wireless systems.

Motivated by these developments, this work investigates intelligent anti-jamming mechanisms for integrated FANET–IoT–IoV environments. Rather than treating aerial, terrestrial, and vehicular networks independently, the proposed research aims to develop an adaptive and cooperative framework capable of jointly exploiting Cognitive Radio (CR) and Multi-Agent Reinforcement Learning (MARL) to provide distributed spectrum awareness, collaborative decision making, and cross-layer communication resilience under highly dynamic jamming environments. The following subsections critically review existing anti-jamming techniques, identify the remaining research challenges, and establish the motivation for the proposed Hybrid CR–MARL framework.

2.2. Limitations of Existing Anti-Jamming Techniques

The increasing sophistication of wireless attacks has motivated extensive research on anti-jamming techniques for UAV communications, IoT systems, and vehicular networks. Existing approaches can generally be categorized into conventional physical-layer techniques, Cognitive Radio (CR)-based spectrum management, machine learning and deep learning approaches, and reinforcement learning-based adaptive communication strategies.

Although these techniques have significantly improved communication robustness under specific operating conditions, several important challenges remain unresolved when they are applied to heterogeneous FANET–IoT–IoV environments.

Conventional anti-jamming techniques, including Frequency-Hopping Spread Spectrum (FHSS), Direct-Sequence Spread Spectrum (DSSS), adaptive modulation, channel coding, beamforming, and transmission power control, constitute the first line of defense against wireless interference. These approaches improve communication reliability by exploiting frequency diversity, signal spreading, or physical-layer redundancy. However, they generally rely on predefined communication patterns or fixed decision thresholds, making them effective primarily against static or predictable jamming attacks. As wireless adversaries become increasingly intelligent and adaptive, these static defense mechanisms are no longer sufficient to maintain reliable communication under rapidly changing spectrum conditions [15,16].

To overcome these limitations, Cognitive Radio (CR) has emerged as an effective paradigm for dynamic spectrum management. CR-enabled communication systems continuously monitor spectrum occupancy, identify available channels, and opportunistically utilize underutilized spectrum resources while avoiding congested or jammed frequency bands. Dynamic spectrum access significantly improves spectrum utilization and communication resilience compared with conventional static channel allocation techniques. Nevertheless, most existing CR-based anti-jamming solutions rely primarily on local spectrum observations and reactive channel switching strategies. Consequently, they often experience delayed adaptation under highly dynamic jamming environments and are unable to exploit cooperative intelligence across multiple heterogeneous network domains [6].

Recent advances in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) have introduced intelligent mechanisms capable of automatically learning complex wireless channel characteristics and identifying hidden interference patterns directly from communication data. Deep neural networks have demonstrated remarkable performance in wireless signal classification, interference detection, anomaly identification, and spectrum occupancy prediction. These learning-based approaches significantly outperform traditional rule-based methods in terms of detection accuracy and adaptability because they continuously improve their prediction capability using historical communication observations [1,5,17]. However, supervised learning methods generally require large labeled datasets and frequently require retraining whenever communication environments or attack strategies change significantly. Their generalization capability therefore remains limited in highly dynamic wireless environments.

Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) have further enhanced adaptive anti-jamming by enabling communication agents to learn optimal transmission policies through continuous interaction with the wireless environment. Instead of relying on predefined decision rules, RL agents dynamically optimize channel selection, transmission scheduling, routing decisions, and spectrum access according to experienced environmental rewards. Consequently, RL-based communication systems can progressively improve communication performance while adapting to changing network conditions and evolving jammer behaviors [4,16]. Nevertheless, many existing RL-based solutions are developed using single-agent learning architectures, where each communication node independently learns its optimal policy without explicitly considering cooperation among multiple distributed agents. Such independent learning often leads to locally optimal decisions, slower convergence, and limited scalability when applied to large-scale heterogeneous wireless networks.

The integration of FANETs, IoT, and IoV further amplifies these challenges. These networks exhibit substantially different communication protocols, node mobility characteristics, computational capabilities, energy constraints, latency requirements, and network topologies. Consequently, anti-jamming mechanisms designed for a single network domain cannot be directly deployed across heterogeneous communication systems without significant modifications. Furthermore, communication decisions made at one protocol layer frequently influence the performance of other layers. For example, spectrum sensing accuracy affects channel selection, channel allocation influences routing performance, while routing decisions directly impact communication latency and packet delivery ratio. Existing anti-jamming solutions generally optimize these communication functions independently, thereby overlooking the strong interdependence among physical, MAC, network, and application layers.

Another important limitation of current approaches is their limited capability for cooperative intelligence. Future integrated FANET–IoT–IoV systems consist of multiple autonomous entities that simultaneously observe different parts of the wireless environment. UAVs possess aerial sensing capability and wide communication coverage, IoT devices continuously monitor environmental conditions through dense sensing infrastructures, whereas connected vehicles provide mobility-aware communication information from dynamic road networks. Existing anti-jamming frameworks rarely exploit these complementary observations to construct a global interference perception model. Instead, most methods rely on isolated local observations, resulting in incomplete environmental awareness and suboptimal communication decisions under complex jamming scenarios.

Moreover, recent wireless adversaries have become increasingly intelligent. Modern jammers can continuously monitor spectrum utilization, identify critical communication links, predict channel switching behavior, and dynamically modify their interference strategies according to network responses. Such adaptive attacks significantly reduce the effectiveness of conventional anti-jamming mechanisms that depend on predefined communication policies or reactive channel switching. Therefore, future communication systems require proactive anti-jamming mechanisms capable of continuously learning evolving jammer behaviors, predicting future interference conditions, and coordinating distributed communication decisions among multiple autonomous agents.

These limitations demonstrate that, although significant progress has been achieved in Cognitive Radio, machine learning, deep learning, and reinforcement learning-based anti-jamming research, existing approaches still lack a unified framework capable of simultaneously supporting heterogeneous FANET–IoT–IoV environments, cooperative distributed intelligence, cross-layer optimization, dynamic spectrum awareness, and adaptive decision making. Consequently, there remains a strong need for an intelligent cross-layer framework that combines Cognitive Radio with Multi-Agent Reinforcement Learning to provide scalable, adaptive, and cooperative communication resilience against sophisticated jamming attacks in next-generation heterogeneous wireless networks.

2.3. Research Gap and Motivation

The preceding discussion demonstrates that considerable progress has been achieved in developing anti-jamming techniques for wireless communication systems. Conventional physical-layer approaches improve communication robustness through spectrum spreading and channel diversity, Cognitive Radio (CR) enables dynamic spectrum access, while Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL), and Reinforcement Learning (RL) have significantly enhanced intelligent spectrum management and adaptive communication. Despite these advancements, existing studies primarily focus on specific

communication scenarios and individual optimization objectives, leaving several important research challenges unresolved for next-generation heterogeneous wireless networks.

First, the majority of existing anti-jamming frameworks have been developed independently for FANETs, IoT systems, or vehicular networks. Since these network domains exhibit fundamentally different communication characteristics, mobility patterns, computational capabilities, and Quality-of-Service (QoS) requirements, solutions designed for one domain cannot be directly generalized to integrated FANET–IoT–IoV environments. Future intelligent communication systems require a unified framework capable of supporting heterogeneous aerial, terrestrial, and vehicular networks while preserving their individual operational characteristics.

Second, most existing studies perform optimization at a single protocol layer. Conventional approaches typically focus on physical-layer spectrum sensing, adaptive channel selection, routing optimization, interference detection, or transmission scheduling independently. However, communication performance in heterogeneous wireless systems depends on strong interactions among multiple protocol layers. For example, inaccurate spectrum sensing directly affects channel selection, channel allocation influences routing decisions, while routing performance ultimately determines packet delivery ratio, communication latency, and overall network reliability. Therefore, cross-layer optimization has become an essential requirement for resilient wireless communications.

Third, although recent reinforcement learning approaches have demonstrated promising results for adaptive anti-jamming, the majority of existing methods employ single-agent learning architectures. Independent learning agents often possess only partial environmental observations and cannot effectively coordinate distributed communication decisions among multiple autonomous nodes. Consequently, these methods frequently suffer from limited scalability, slower convergence, incomplete environmental awareness, and reduced adaptability under highly dynamic jamming environments. Multi-Agent Reinforcement Learning (MARL) provides a promising alternative by enabling cooperative policy learning through information sharing and distributed decision making; however, its integration with Cognitive Radio for heterogeneous FANET–IoT–IoV systems remains relatively unexplored.

Fourth, existing studies generally optimize communication performance against known or relatively simple jamming strategies. However, modern adversaries are becoming increasingly intelligent and adaptive. Contemporary jammers can continuously monitor spectrum utilization, identify communication patterns, predict channel switching behavior, and dynamically modify their attack strategies according to network responses. Such adaptive attacks require proactive learning mechanisms capable of continuously updating communication policies based on changing environmental conditions rather than relying on predefined decision rules or static optimization strategies.

Finally, relatively few studies simultaneously integrate dynamic spectrum awareness, cooperative Multi-Agent Reinforcement Learning, heterogeneous network abstraction, and cross-layer communication optimization within a unified anti-jamming framework. Consequently, there remains a significant research gap between existing intelligent anti-jamming techniques and the communication requirements of future integrated FANET–IoT–IoV systems.

Motivated by these limitations, this paper proposes a Hybrid Cognitive Radio and Multi-Agent Reinforcement Learning (CR–MARL) framework for adaptive jamming resilience in heterogeneous FANET–IoT–IoV environments. The proposed framework combines Cognitive Radio-based spectrum awareness with cooperative Multi-Agent Reinforcement Learning to enable distributed spectrum sensing, intelligent channel selection, cooperative decision making, and cross-layer communication optimization. Unlike exist-

ing approaches, the proposed framework introduces a unified cognitive abstraction layer that allows heterogeneous aerial, terrestrial, and vehicular nodes to exchange normalized environmental observations while preserving their domain-specific communication protocols. This enables scalable, adaptive, and cooperative anti-jamming capability for future intelligent wireless communication systems.

Table 2 summarizes representative anti-jamming studies discussed in this section and highlights their primary characteristics. Although existing approaches have substantially improved communication resilience within specific application domains, most of them remain limited to individual network environments or isolated communication functions. In contrast, the proposed Hybrid CR–MARL framework simultaneously supports heterogeneous FANET, IoT, and IoV environments while integrating Cognitive Radio, cooperative Multi-Agent Reinforcement Learning, and cross-layer optimization into a unified intelligent architecture. This capability distinguishes the proposed framework from existing state-of-the-art anti-jamming approaches and establishes the principal motivation and novelty of this research.

Table 2. Comparison of representative anti-jamming approaches and identified research gaps.

Reference	FANET	IoT	IoV	CR	AI/RL	Cross- Layer	Unified Framework
Mowla et al. [6]	✓	–	–	✓	–	–	–
Xu et al. [9]	✓	–	–	–	–	–	–
Khan et al. [12]	✓	–	–	–	–	–	–
Chang et al. [16]	✓	–	–	–	✓	–	–
Chhikara et al. [13]	✓	–	–	✓	✓	–	–
Chandran et al. [8]	✓	–	–	–	–	–	–
Meftah et al. [5]	✓	✓	–	–	✓	–	–
Irfan et al. [1]	–	✓	–	–	✓	–	–
Liu et al. [17]	✓	–	–	–	✓	–	–
Xia et al. [15]	✓	–	–	–	–	–	–
Jeon et al. [4]	–	–	✓	–	✓	✓	–
Ayegun et al. [14]	✓	–	–	–	–	–	–
Proposed Hybrid CR–MARL Framework	✓	✓	✓	✓	✓	✓	✓

3. System Model and Jamming Threat Analysis

3.1. Integrated FANET–IoT–IoV Architecture

The integrated FANET–IoT–IoV architecture is composed of three tightly connected tiers. Each tier has unique characteristics, constraints, and communication requirements. Together, they form a heterogeneous and multi-layer wireless system. Data flows vertically and horizontally across these tiers. This interaction enables complex applications but also increases vulnerability to jamming.

Figure 1 shows the integrated FANET–IoT–IoV architecture depicting aerial, vehicular and terrestrial tiers.

Integrated FANET-IoT-IoV Architecture

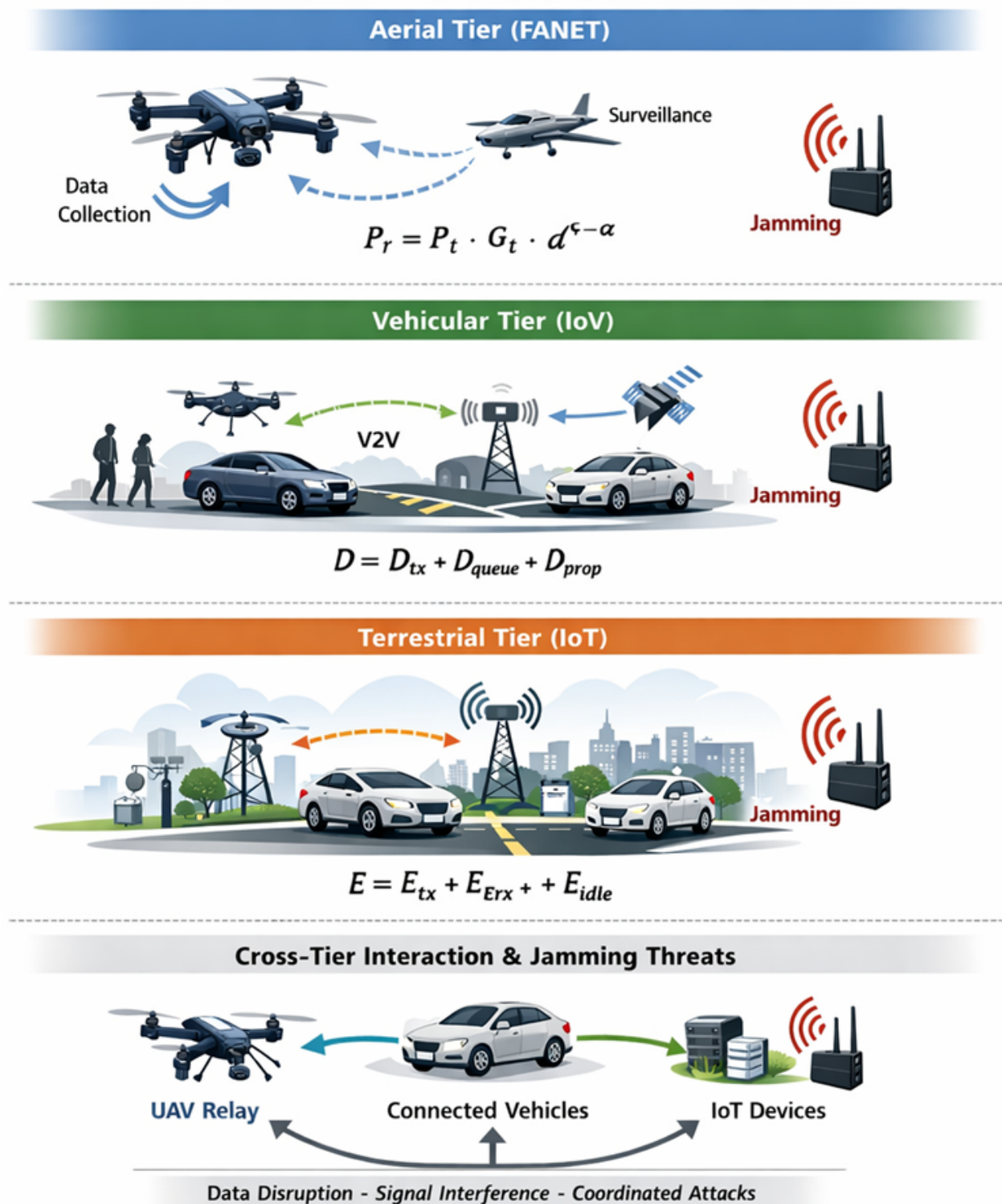


Figure 1. Integrated FANET-IoT-IoV Architecture.

3.1.1. Aerial Tier (FANET)

The aerial tier consists of UAVs forming a Flying Ad Hoc Network. These nodes operate in three-dimensional space. Their mobility is very high. Network topology changes frequently. Links are often short-lived. Most communications rely on line-of-sight propagation. This improves signal quality but also exposes links to interference [18,19]. UAVs use multiple communication links for different purposes. UAV-to-UAV links enable coordination and information sharing within a swarm. UAV-to-Ground links are used to upload sensor data or video streams to ground stations. UAV-to-Edge or UAV-to-Base Station links allow data aggregation, processing, and further dissemination to vehicles or the cloud. Each link type has distinct vulnerability to jamming, as critical data or control messages may be disrupted if specific links are attacked.

Communication in UAVs can be classified into control links and data links. Control links carry flight commands and coordination signals. Data links transmit sensor readings, video streams, and mission-related information. Control links are more critical because their disruption can result in mission failure or physical UAV crashes. Jamming attacks targeting control links are therefore particularly dangerous compared to attacks on data links. UAV mobility patterns affect network stability and vulnerability. UAVs may operate in swarm formations, leader-follower structures, or follow random waypoint mobility patterns. These formations influence link stability and the effectiveness of jamming attacks. Highly dynamic mobility can make the network more resilient to static jammers but may also complicate coordinated defense strategies.

UAV altitude affects both communication performance and vulnerability to jamming. Flying at higher altitudes increases coverage area and line-of-sight range, enabling broader surveillance and data collection. However, higher altitude also makes UAVs more visible and susceptible to interference. Adaptive altitude control can balance coverage needs and minimize exposure to jamming, enhancing overall network resilience. UAVs have limited battery capacity. Energy efficiency is a major concern. Communication, sensing, and computation all compete for power. UAVs often act as relays between ground nodes. They may also serve as data mules or mobile base stations. In surveillance missions, UAVs collect video and sensor data. In disaster scenarios, they provide temporary connectivity. The received signal power at a UAV can be modeled as

$$P_r = P_t \cdot G_t \cdot G_r \cdot d^{-\alpha}$$

where P_t is the transmit power, G_t and G_r are antenna gains, d is the distance, and α is the path loss exponent. Jamming increases interference power and reduces the SINR. This directly affects packet delivery. UAVs in the aerial tier support critical applications. Examples include disaster area connectivity restoration, where UAVs provide temporary communication when ground infrastructure is damaged. Border surveillance missions use UAVs to monitor wide areas for security purposes. Battlefield reconnaissance involves UAVs gathering real-time intelligence for operational decision-making. These scenarios highlight the importance of reliable and secure communication links and motivate the need for advanced anti-jamming techniques.

Figure 2 shows the aerial tier consisting of FANETs that provide high-mobility, and on-demand wireless connectivity. This tier supports low-latency links with terrestrial and vehicular tiers while being highly adaptive to dynamic topology changes and mission requirements.

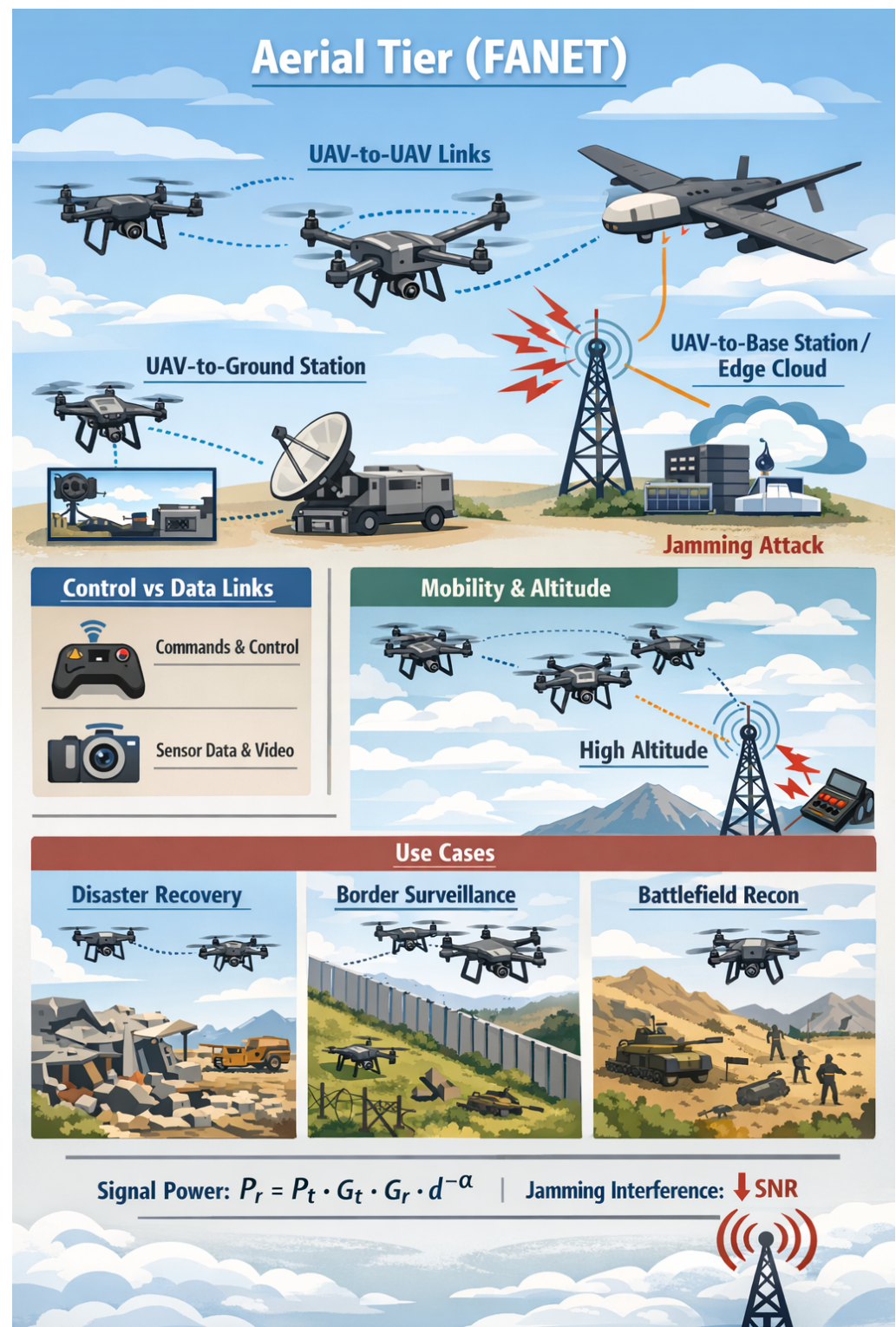


Figure 2. Formation of Aerial Tier (FANET).

3.1.2. Vehicular Tier (IoV)

The vehicular tier consists of connected and autonomous vehicles. It also includes road-side units and pedestrian devices. Vehicles move at high speed but follow predictable paths. Movement is constrained to road networks. Network density can be very high in urban areas. Traffic conditions cause frequent topology changes. Vehicles communicate through multiple modes. Vehicle-to-Vehicle (V2V) links allow direct exchange of information such as speed, position, and intentions between nearby vehicles. Vehicle-to-Infrastructure (V2I)

communication connects vehicles to RSUs or traffic lights for coordinated traffic management. Vehicle-to-Pedestrian (V2P) links alert pedestrians to nearby vehicles for safety. Vehicle-to-Network (V2N) communication connects vehicles to cloud or edge servers for navigation, traffic updates, and infotainment. Each communication mode has different latency and reliability requirements, and jamming can affect them differently.

Some messages are extremely time-sensitive and crucial for safety. Emergency brake warnings inform nearby vehicles of sudden deceleration. Collision avoidance messages help vehicles make split-second maneuvers to prevent accidents. Lane change alerts communicate intentions to surrounding vehicles. Jamming attacks targeting these messages can cause delays or loss of critical information, increasing the risk of accidents [20–22]. Network conditions and jamming impact are affected by traffic density. In sparse highway scenarios, fewer vehicles result in fewer available relays, making communication more vulnerable to interference. In dense urban intersections, high vehicle density may create channel congestion, increasing the effect of jamming. Traffic jams during peak hours can amplify interference due to multiple vehicles transmitting simultaneously [23].

Roadside units (RSUs) and edge servers play a key role in processing vehicular data locally. RSUs reduce latency by handling safety messages and local traffic coordination. Edge computing allows vehicles to receive processed data quickly without relying on distant cloud servers. Jamming attacks can isolate vehicles from RSUs or edge nodes, disrupting communication and decision-making. UAVs in the aerial tier may assist by relaying messages between vehicles and edge servers, making the IoV layer dependent on cross-tier communication [24]. The vehicular tier supports several real-world applications. Smart traffic light coordination uses V2I communication to optimize traffic flow. Autonomous driving assistance relies on timely exchange of safety and navigation messages. Emergency vehicle routing uses connected vehicle data to clear paths and reduce response time. These use cases highlight the critical need for reliable and secure communication in the IoV tier, motivating the development of adaptive anti-jamming strategies.

IoV communication supports safety messages, traffic updates, and infotainment services [25]. Many messages are delay-sensitive. Even small delays can lead to unsafe decisions. Vehicle-to-vehicle and vehicle-to-infrastructure links operate over shared spectrum. The communication delay in the vehicular tier can be expressed as

$$D = D_{tx} + D_{queue} + D_{prop}$$

where D_{tx} is the transmission delay required to place all packet bits onto the wireless channel, D_{queue} is the queuing delay caused by packet buffering and retransmissions at intermediate nodes, and D_{prop} is the propagation delay determined by the physical distance between the transmitter and receiver. Jamming increases packet collisions and retransmissions, which significantly increases D_{queue} and overall delay. Jamming increases queuing and retransmission delays. This degrades real-time performance.

In a jamming scenario, safety messages may not reach nearby vehicles. This can cause chain collisions or traffic congestion. When FANETs support IoV, UAVs may relay traffic information. Jamming at this tier can propagate to aerial and terrestrial layers. Figure 3 shows the vehicular tier which comprises of connected and autonomous vehicles, roadside units, and pedestrian devices. These systems communicate through V2V, V2I, and V2P links. Vehicular Tier enables low-latency, high-reliability data exchange for traffic safety and mobility services. It ensures all these while operating under dynamically changing network density and channel condition.

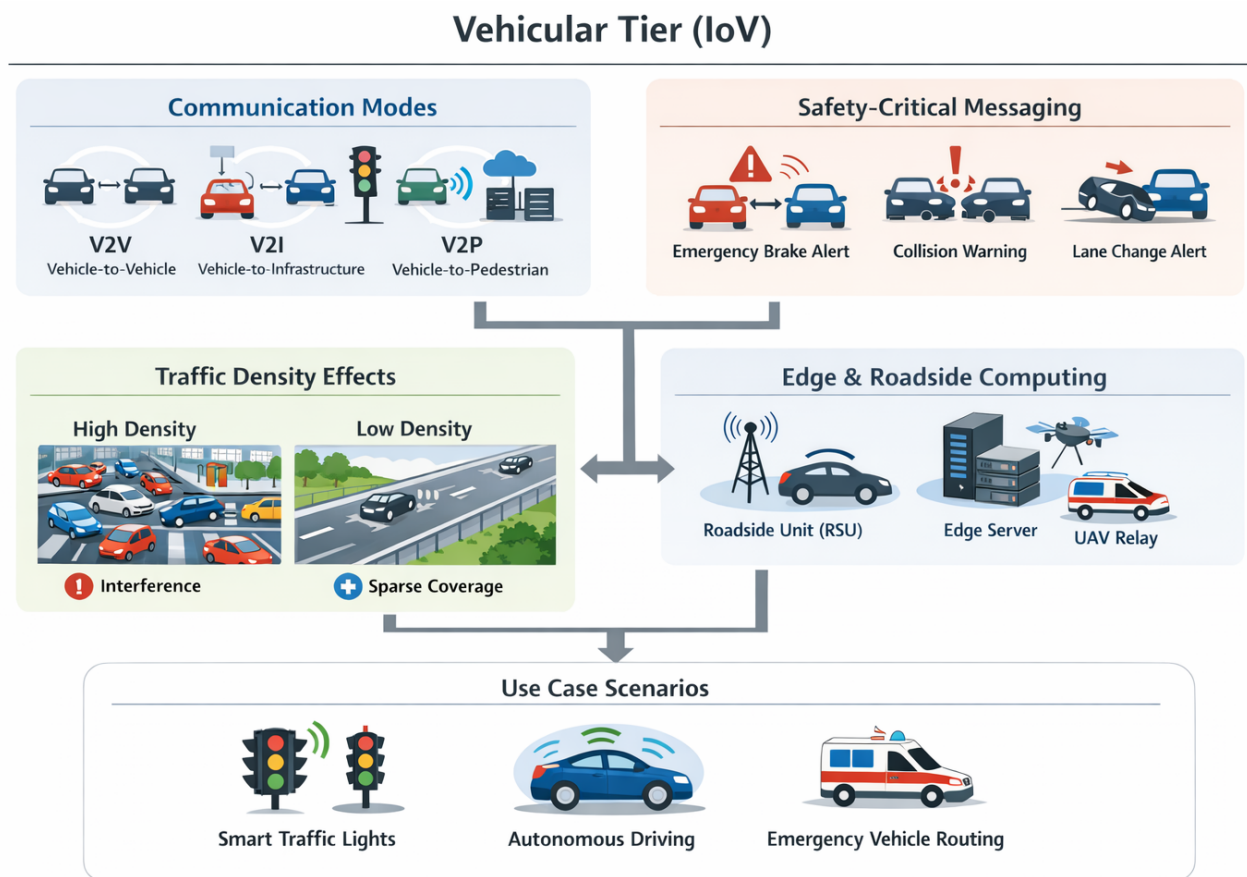


Figure 3. Formation of Vehicular Tier (IoV).

3.1.3. Terrestrial Tier (IoT)

The terrestrial tier consists of IoT sensors and smart infrastructure. Nodes are usually static or low mobility. They are densely deployed. Devices are low-cost and energy-constrained. Many nodes operate on batteries for long periods. This tier performs sensing and data collection. Sensors monitor traffic, weather, pollution, and infrastructure health. Data is forwarded to edge servers or UAVs. Communication protocols are lightweight. Computational capabilities are limited. IoT nodes can include environmental sensors for temperature, humidity, and pollution, traffic and motion sensors for vehicle detection, structural health sensors for bridges and buildings, and surveillance cameras for security and monitoring. Different sensor types generate different traffic patterns and require varied communication reliability and latency [24,26,27].

Terrestrial IoT nodes typically use lightweight communication protocols such as IEEE 802.15.4, ZigBee, LoRa, LoRaWAN, and NB-IoT. These protocols are designed for low power and low data rate applications but are inherently more vulnerable to jamming due to narrowband operation and limited interference handling capabilities. To reduce energy consumption and optimize communication, nodes often form clusters with designated cluster heads. Cluster heads aggregate data from member nodes and forward it to edge servers or UAVs. Jamming attacks targeting cluster heads or aggregation points can disrupt a large portion of the network and significantly degrade data collection efficiency. IoT nodes often operate with duty cycles, alternating between active and sleep states to conserve energy. Jamming attacks force repeated transmissions and prolonged channel sensing, increasing energy consumption and reducing network lifetime. Careful management of sleep schedules and adaptive wake-up strategies are essential for maintaining resilience under interference.

The terrestrial tier supports a variety of real-world applications. Smart city monitoring relies on distributed sensors for traffic flow, environmental conditions, and public safety. Flood and weather monitoring requires reliable sensor data collection. Industrial IoT systems monitor machinery and infrastructure for faults or failures. Jamming attacks on this tier can create blind spots in monitoring, delay critical decisions, and impact overall network reliability [5,28]. The energy consumption of an IoT node can be modeled as

$$E = E_{tx} + E_{rx} + E_{idle}$$

where E_{tx} is the energy consumed during data transmission, E_{rx} is the energy used for receiving packets, and E_{idle} is the energy spent while the node remains active but not transmitting or receiving. Under jamming attacks, repeated packet transmissions and prolonged channel sensing increase E_{tx} and E_{idle} , leading to faster battery depletion. Jamming forces repeated transmissions. This increases energy usage. Nodes may exhaust batteries early. A jammer can block sensor reports over a large area. This creates blind spots in monitoring systems. In smart cities, this affects decision making and automation. Figure 4 shows the Terrestrial Tier that visually represents IoT sensor nodes, smart infrastructure, edge devices, clustering, communication protocols, sleep/duty-cycle behavior, energy consumption, and smart city applications.

Terrestrial Tier (IoT)



Figure 4. Terrestrial Tier.

3.1.4. Cross-Tier Interdependence and Motivation for Intelligent Anti-Jamming

The three tiers of the integrated FANET-IoT-IoV architecture are highly interdependent. UAVs collect sensory data from terrestrial IoT nodes and forward it to higher layers. Vehicles receive timely updates through direct UAV relays or via roadside infrastructure. Edge computing nodes assist in coordinating data processing and decision making across tiers [29]. As a result, the performance of each tier directly affects the overall system reliability [30]. Jamming attacks on any single tier can propagate across the architecture.

Disruption in the IoT tier can degrade situational awareness at the aerial and vehicular layers. Jamming of UAV communication links can isolate vehicles from real-time updates. Vehicular jamming can increase latency and reduce safety-critical message delivery. These cascading effects significantly amplify the impact of adversarial interference. The heterogeneous nature of the architecture makes anti-jamming particularly challenging [18,31]. UAVs operate under high mobility and require rapid adaptation to dynamic interference. Vehicular networks demand ultra-low latency and high reliability for safety applications. IoT nodes are energy constrained and cannot support complex or centralized defense mechanisms. Consequently, a single static or tier-specific anti-jamming solution is insufficient.

This architecture necessitates cross-layer and cooperative anti-jamming strategies [32]. Defense mechanisms must jointly consider mobility, traffic density, and resource constraints. They must also adapt to intelligent and time-varying jamming behaviors. Cognitive and learning-based approaches are well suited for this purpose. They enable autonomous sensing, decision making, and continuous adaptation [10]. In this context, a multi-agent reinforcement learning (MARL) based framework is highly promising. Each tier can be modeled as an intelligent agent with local observations and objectives. UAVs, vehicles, and edge nodes can learn coordinated anti-jamming policies through interaction. Agents can jointly optimize channel selection, power control, routing, and relay decisions. Decentralized learning improves scalability and robustness against partial network failures [16]. Such a MARL-based framework enables distributed intelligence across the three tiers. It supports fast adaptation to jamming dynamics while respecting tier-specific constraints. It also facilitates cooperative defense without excessive signaling overhead. These characteristics directly address the limitations of conventional anti-jamming techniques. Therefore, this work is motivated by the need for intelligent, cross-tier, and learning-driven anti-jamming solutions in integrated FANET-IoT-IoV systems.

Table 3 summarizes the key characteristics, communication requirements, and jamming vulnerabilities of the three network tiers discussed above. This summary highlights their complementary capabilities and motivates the need for a unified cross-layer anti-jamming framework.

Table 3. Summary of characteristics of the integrated FANET-IoT-IoV architecture.

Network Tier	Mobility	Communication Characteristics	Major Constraints	Typical Jamming Impact	Role in Proposed Framework
FANET	Very High	UAV-UAV, UAV-Ground, UAV-Edge	Energy, dynamic topology, intermittent links	Control-link disruption, mission failure, relay interruption	Adaptive spectrum sensing, aerial relaying, cooperative learning
IoV	High	V2V, V2I, V2P, V2N	Ultra-low latency, high reliability	Delayed safety messages, collision risk	Mobility-aware communication and collaborative decision making
IoT	Low/Static	Sensor-to-edge, clustered communication	Limited energy, computation, bandwidth	Data loss, blind monitoring regions, battery depletion	Environmental sensing and distributed interference monitoring

3.2. Unified Cross-Domain Abstraction for Heterogeneous Networks

The integration of Flying Ad Hoc Networks (FANETs), the Internet of Things (IoT), and the Internet of Vehicles (IoV) presents significant challenges because these network domains

differ substantially in terms of mobility, communication protocols, computational resources, topology dynamics, latency requirements, and energy constraints. Rather than assuming homogeneous network characteristics, the proposed framework adopts a cross-domain abstraction strategy that enables cooperative anti-jamming intelligence while preserving the operational characteristics of each individual network.

The proposed architecture introduces a unified cognitive abstraction layer positioned between the heterogeneous communication infrastructure and the intelligent decision-making engine. Each network domain performs its native sensing and communication functions independently while exporting standardized contextual information to the abstraction layer. Consequently, the framework does not require identical communication protocols or hardware platforms across FANET, IoT, and IoV systems.

Specifically, each network node periodically reports a common set of observable features, including received signal strength (RSS), signal-to-interference-plus-noise ratio (SINR), channel occupancy, packet delivery ratio (PDR), residual energy, node mobility information, neighboring node density, and local interference measurements. Although these parameters are obtained using different sensing mechanisms depending on the underlying network technology, they are transformed into a normalized feature representation before being processed by the Cognitive Radio (CR) engine.

The CR engine performs spectrum sensing, interference identification, and dynamic channel availability estimation. The processed environmental information is subsequently forwarded to the Multi-Agent Reinforcement Learning (MARL) layer, where each intelligent agent constructs a local state representation using the normalized observations. Since all agents operate on a common feature space, cooperative policy learning becomes independent of the underlying communication technology while still preserving network-specific characteristics through feature encoding.

This abstraction strategy enables the proposed framework to generalize across heterogeneous environments without modifying the protocol stack of individual networks. UAVs operating within FANETs contribute highly dynamic aerial observations, IoT devices provide dense environmental sensing information, whereas connected vehicles continuously contribute mobility-aware spectrum observations. The cooperative exchange of these complementary observations significantly improves the global perception of the wireless environment, allowing the MARL agents to make coordinated anti-jamming decisions under highly dynamic interference conditions.

Consequently, the proposed framework should be viewed as a unified cognitive intelligence architecture rather than a unified communication protocol. The communication mechanisms remain domain-specific, whereas cognitive spectrum management, interference awareness, and cooperative decision-making are jointly optimized through the proposed CR-MARL framework. This hierarchical abstraction enables the framework to maintain scalability, adaptability, and interoperability across heterogeneous FANET–IoT–IoV deployments while reducing the need for network-specific anti-jamming algorithms.

3.3. Jamming Attack Models and Taxonomy

Jamming attacks are classified based on their operational strategy and capability. Figure 5 depicts Jamming attack models and taxonomy visually.

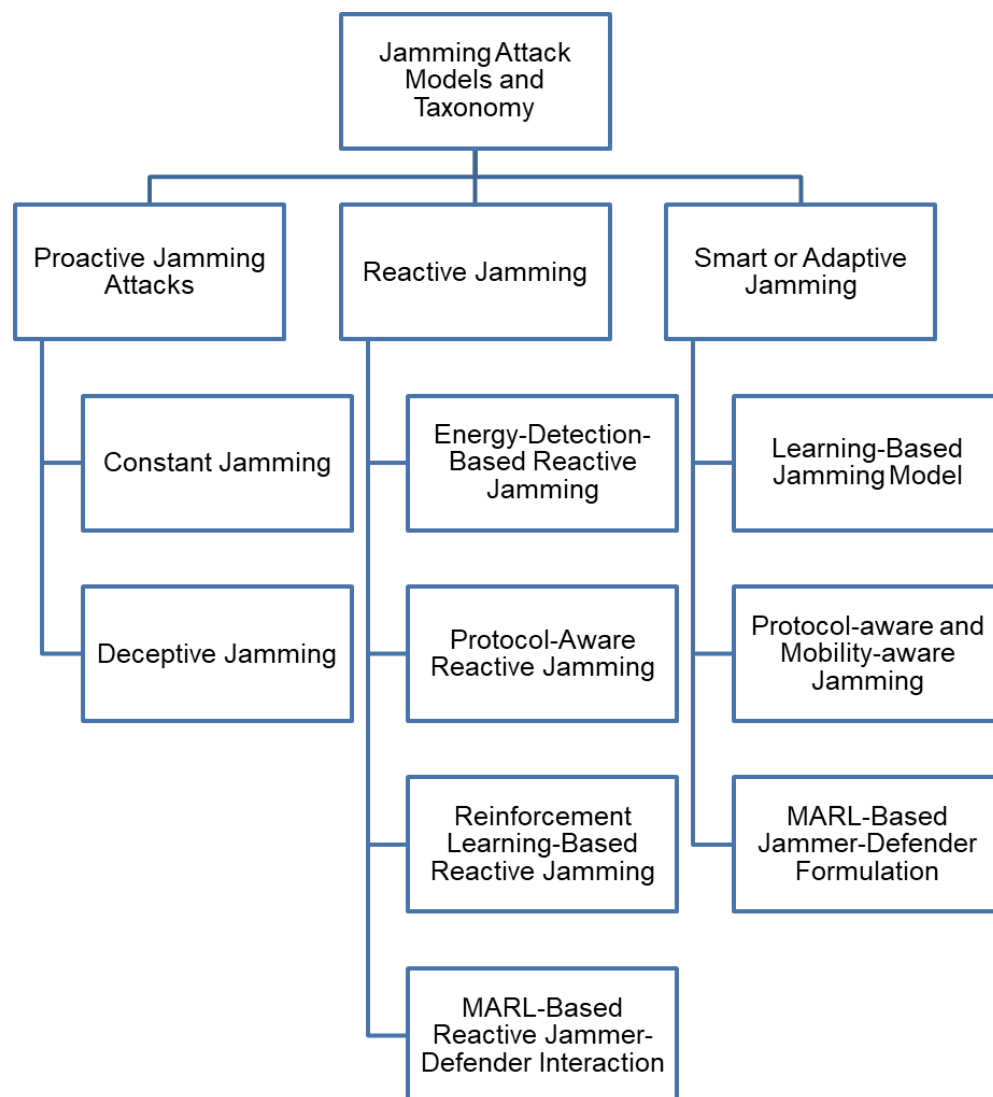


Figure 5. Overview of Jamming Attack Models and Taxonomy.

3.3.1. Proactive Jamming Attacks

Jamming attacks can be classified according to the jammer's operational strategy and capability. One important category is proactive jamming. In proactive jamming, the attacker continuously interferes with the wireless medium. The jammer does not wait for legitimate transmissions. Instead, it transmits regardless of channel activity. The goal is to degrade communication availability at all times [33,34]. Proactive jammers are particularly harmful in FANET–IoT–IoV systems. These systems rely on continuous data exchange. Examples include UAV command and control links, vehicular safety messages, and IoT sensing data. Persistent interference can severely reduce system reliability.

Constant Jamming

A constant jammer continuously emits radio frequency noise or interference signals. The jammer transmits with fixed or slowly varying power. It occupies a wide or narrow frequency band. No sensing or adaptation is performed. This makes the jammer simple to design and deploy [35]. At the physical layer, the received signal at a legitimate receiver can be modeled as

$$y(t) = s(t) + j(t) + n(t), \quad (1)$$

where $s(t)$ is the legitimate transmitted signal, $j(t)$ is the jamming signal, and $n(t)$ is additive noise. For constant jamming, $j(t)$ is present at all times. Its power P_j is typically constant. The resulting signal-to-interference-plus-noise ratio (SINR) is given by

$$\text{SINR} = \frac{P_s}{P_j + P_n}, \quad (2)$$

where P_s is the signal power and P_n is the noise power. As P_j increases, the SINR drops significantly. This leads to packet loss and link failure. A typical application of constant jamming is disabling a UAV command and control channel. The jammer targets the uplink or downlink frequency used by the ground control station. The UAV may lose navigation commands. In severe cases, the UAV may enter fail-safe mode or crash. From the attacker's perspective, constant jamming has several advantages. It is easy to implement. It does not require protocol knowledge. It is effective against poorly protected links. However, it also has limitations. Energy consumption is high. The jammer emits continuously. This makes it easy to detect using spectrum monitoring. Once detected, localization and mitigation become possible.

Deceptive Jamming

A deceptive jammer transmits signals that resemble legitimate protocol data. Instead of random noise, it injects structured packets. These packets conform partially or fully to protocol specifications. The objective is to confuse receivers rather than block the channel completely [36]. Deceptive jamming operates mainly at the physical and MAC layers. It may also exploit higher-layer semantics. The jammer may spoof headers, identifiers, or control fields. This makes detection more difficult. Mathematically, the received signal can be expressed as

$$y(t) = s(t) + j_d(t) + n(t), \quad (3)$$

where $j_d(t)$ is a deceptive jamming signal. Unlike constant noise, $j_d(t)$ is structured. It may have similar modulation and timing as $s(t)$. At the packet level, the jammer aims to maximize the probability of false packet acceptance. Let P_f denote the probability that a receiver accepts a fake packet. The attacker seeks to maximize

$$\max_{j_d} P_f, \quad (4)$$

subject to power and timing constraints. A common example occurs in vehicular networks. In Internet of Vehicles systems, Cooperative Awareness Messages and Decentralized Environmental Notification Messages are used for safety. A deceptive jammer can inject fake emergency brake messages. Vehicles may interpret these as genuine alerts. This can trigger unnecessary braking. Such behavior increases the risk of traffic congestion or collisions.

In FANET scenarios, deceptive jamming can spoof UAV coordination messages. Fake position or status updates can disrupt swarm formation. This degrades mission performance. Compared to constant jamming, deceptive jamming is energy efficient. It does not require continuous transmission. However, it requires protocol knowledge. It may also require precise synchronization. This increases attacker complexity. Detection is harder because signals appear legitimate.

3.3.2. Reactive Jamming

Reactive jamming is characterized by conditional interference. The jammer first observes the channel. It transmits only when legitimate activity is detected. This makes reactive jamming energy efficient and stealthy. It is widely studied in wireless and cognitive radio literature. Reactive jammers exploit predictable transmission behaviors. They do not

require full protocol decoding. They rely on fast sensing and response mechanisms. This makes them effective against time-sensitive links.

Energy-Detection-Based Reactive Jamming

The simplest form of reactive jamming uses energy detection. The jammer monitors the channel power level. It compares the measured energy with a threshold. Let E_t denote the measured energy in a sensing window. Channel activity is detected when

$$E_t > \gamma, \quad (5)$$

where γ is a detection threshold. Once activity is detected, the jammer transmits interference. The jamming duration may cover the entire packet. It may also target only the packet header. It depends on jammer capability. This technique is commonly used against UAV telemetry links. Short control packets are highly vulnerable. Partial jamming is often sufficient to corrupt decoding. Energy-detection-based jamming is simple. It requires minimal computation. However, it is sensitive to noise uncertainty. False alarms may trigger unnecessary transmissions.

Protocol-Aware Reactive Jamming

Protocol-aware reactive jamming exploits known protocol structures. The jammer decodes partial headers or preambles. It identifies packet type and timing. It reacts only to selected frames. Let T_p denote the packet duration. Let T_h denote the header duration. The jammer targets the interval $[0, T_h]$. This maximizes disruption with minimal energy. Such jammers are effective against MAC-layer control packets. Examples include RTS/CTS frames and synchronization beacons. In FANETs, coordination packets are primary targets. In IoV systems, safety beacons are vulnerable. Protocol-aware reactive jamming requires protocol knowledge. It also requires accurate synchronization. This increases complexity but improves efficiency.

Markov-Based Reactive Jamming

Reactive jamming behavior can be modeled using Markov processes. The jammer state depends on observed channel activity. State transitions capture sensing and reaction dynamics. Let the jammer state space be

$$\mathcal{S}_J = \{\text{Idle}, \text{Sensing}, \text{Jamming}\}. \quad (6)$$

State transitions follow a Markov chain:

$$\Pr(s_{t+1} | s_t). \quad (7)$$

For example, upon detecting activity, the jammer transitions from Sensing to Jamming. After transmission ends, it returns to Idle. Markov modeling allows analytical evaluation. Metrics include jamming probability and energy efficiency. Such models are widely used in cognitive radio studies.

Cognitive Radio-Based Reactive Jamming

Cognitive radio principles enhance reactive jamming. The jammer performs spectrum sensing. It learns channel occupancy patterns. It selects channels with high activity probability. Let π_f denote the probability that channel f is active. The jammer selects

$$f^* = \arg \max_f \pi_f. \quad (8)$$

Interference is applied only when activity is detected. This improves stealth and efficiency. In multi-channel FANET systems, this is highly effective. UAVs often use adaptive frequency selection. Cognitive reactive jammers can track these changes.

Reinforcement Learning-Based Reactive Jamming

Reinforcement learning improves reactive jamming decisions. The jammer learns when and how long to jam. It adapts sensing thresholds and reaction timing. The jammer is modeled as an agent. The environment is partially observable. The observation includes channel energy and timing. The action space includes

$$a_t^J \in \{\text{jam, do not jam, adjust threshold}\}. \quad (9)$$

The reward balances disruption and energy cost. Over time, the jammer learns optimal reaction policies. Such methods outperform static reactive jammers. They adapt to traffic load and mobility.

MARL-Based Reactive Jammer–Defender Interaction

Reactive jamming naturally extends to a multi-agent setting. Multiple jammers or defenders may coexist. Each agent has local observations. The interaction is modeled as a stochastic game. Defenders adapt transmission strategies. Jammers adapt reaction policies. Defender agents may randomize timing and frequency. Jammer agents learn reaction strategies. This creates a dynamic adversarial environment. MARL enables coordinated defense. UAVs, vehicles, and IoT nodes act as agents. They learn to avoid predictable patterns. This reduces reactive jamming effectiveness.

3.3.3. Smart or Adaptive Jamming

Smart or adaptive jamming represents the most advanced class of jamming attacks [13]. The jammer employs learning and optimization algorithms. It observes network behavior over time. It exploits protocol-level patterns. Its objective is to maximize disruption while minimizing exposure and energy use. Unlike reactive jammers, smart jammers do not rely on simple threshold sensing. They analyze traffic statistics, timing structures, and control signaling. They infer Medium Access Control schedules, and track channel hopping or frequency selection policies. They adapt their strategy dynamically as the network behavior changes.

Table 4 summarizes the operational characteristics and system-level impacts of different jamming strategies in integrated FANET–IoT–IoV architectures. Command and Control (C2) links mentioned in the table refer to the covert communication channels. They establish a connection between an attacker’s infrastructure (the C2 server) and a compromised device within a network. These covert communication channels allow attackers to issue instructions, download malicious payloads, and exfiltrate sensitive data without detection.

Smart jammers are well suited for heterogeneous FANET–IoT–IoV environments due to the reason that these systems use diverse protocols and dynamic configurations. In these environments, static jamming strategies are completely ineffective or less effective. Learning-based adaptation for jamming detection and applying anti-jamming measures becomes essential.

Table 4. Comparative Impact of Jamming Attacks in Integrated FANET–IoT–IoV Systems.

Jamming Type	Operational Behavior	Primary Impact	Most Affected Components
Proactive	Continuous or deceptive interference without sensing	Persistent availability loss; integrity degradation	UAV C2 links, IoT sensing channels
Reactive	Jams only upon detecting active transmissions	Selective disruption of control and safety packets	UAV-to-UAV links, vehicular safety messages
Smart/Adaptive	Learns protocol timing, hopping, and mobility patterns	Targeted mission degradation with low detectability	UAV video streams, cross-tier coordination links
Overall Impact	Exploitation of broadcast and interdependent architecture	Cascading reliability and latency degradation	Integrated FANET–IoT–IoV system

Learning-Based Jamming Model

The smart jammer can be modeled as a learning agent. The wireless environment is partially observable. The jammer does not have full knowledge of network states. Instead, it relies on observations. Let the jammer observation at time step t be

$$o_t^J = \{f_t, p_t, \Delta t, \eta_t\}, \quad (10)$$

where f_t is the observed frequency, p_t is the received signal power, Δt is inter-packet timing, and η_t represents protocol-specific features. These observations are extracted through passive monitoring. Based on observations, the jammer selects an action. The action space may include

$$a_t^J \in \{\text{jam, stay silent, change frequency, follow target}\}. \quad (11)$$

The jammer aims to maximize a cumulative reward. A typical reward function is

$$r_t^J = \alpha \cdot D_t - \beta \cdot P_t - \delta \cdot L_t, \quad (12)$$

where D_t measures communication disruption, P_t is energy consumption, and L_t is the probability of detection. The coefficients α , β , and δ balance these objectives. Reinforcement learning techniques are used to optimize this policy. Q-learning or deep reinforcement learning can be applied. The jammer updates its policy based on feedback from the environment.

Protocol-Aware and Mobility-Aware Jamming

Smart jammers exploit MAC-layer behavior. They learn time-slotted access patterns. They infer contention windows. They identify control packet periodicity. They focus jamming efforts on critical frames. Let T_s denote a MAC scheduling period. The jammer estimates T_s by observing packet arrivals. The estimation error decreases over time. Targeted interference is then applied at high-impact time slots. Channel hopping schemes are also vulnerable. Let $h(t)$ denote the hopping sequence used by the defender. The jammer attempts to estimate

$$\hat{h}(t+1) = \arg \max_f \Pr(h(t+1) = f \mid o_{1:t}^J). \quad (13)$$

Mobility further enhances smart jamming. A jammer mounted on a vehicle can track a target UAV. It adapts transmission power and position. It maintains proximity to maximize interference. A representative example is a mobile jammer mounted on a vehicle. The jammer targets a specific surveillance UAV. The UAV streams real-time video to a ground station. The jammer observes uplink and downlink traffic patterns. It identifies the video stream characteristics. It selectively jams only video frames. Control channels may be left

untouched. This avoids triggering fail-safe mechanisms. Such targeted attacks are highly effective. They degrade mission performance without total communication loss. They are difficult to detect using conventional methods.

MARL-Based Jammer–Defender Formulation

Smart jamming naturally leads to a multi-agent learning framework. Both jammer and defenders act as intelligent agents. Each tier contains multiple agents. UAVs, vehicles, and edge nodes cooperate as defenders. The jammer operates as an adversarial agent. The interaction can be modeled as a stochastic game. Let $\mathcal{N} = \mathcal{N}_D \cup \mathcal{N}_J$ denote the set of defender and jammer agents. Each agent i has a local observation o_t^i and action a_t^i . The global system state evolves as:

$$s_{t+1} \sim \mathcal{P}(s_{t+1} \mid s_t, a_t), \quad (14)$$

where a_t is the joint action of all agents. Defender agents aim to maximize communication reliability. Their reward can be expressed as:

$$r_t^D = \lambda \cdot \text{SINR}_t - \mu \cdot E_t - \nu \cdot \text{Delay}_t, \quad (15)$$

where E_t is energy consumption and Delay_t captures latency. The jammer aims to minimize defender rewards while optimizing its own objectives. This creates an adversarial learning environment. Multi-agent reinforcement learning is well suited for such interactions. Centralized training with decentralized execution can be employed. Each agent learns local policies. Global coordination emerges implicitly.

3.4. Impact and Implications of Jamming on Integrated FANET–IoT–IoV Systems

Proactive, reactive, and smart jamming attacks have distinct but complementary impacts on integrated FANET–IoT–IoV architectures. All three exploit the broadcast and shared nature of wireless channels. Their effects are amplified by the tight coupling among aerial, terrestrial, and vehicular tiers. Disruption in one tier can propagate to others. This leads to cascading performance degradation. Proactive jamming primarily targets availability. Constant jamming causes persistent denial of service. Deceptive jamming additionally compromises data integrity. Fake or spoofed packets can mislead decision-making processes. Such attacks are particularly harmful in heterogeneous systems. Different tiers use different protocols, frequencies, and timing constraints. This diversity complicates unified defense mechanisms.

Reactive jamming exploits transmission predictability. It selectively targets active communication periods. This makes it energy efficient and stealthy. Control packets and safety-critical messages are especially vulnerable. In FANETs, UAV command and coordination links are affected. In IoV systems, safety beacons and emergency messages are disrupted. In IoT networks, sporadic sensor transmissions suffer repeated failures. Detection becomes difficult due to the jammer's intermittent behavior.

Smart and adaptive jamming represent the most severe threat. These jammers learn protocol structures and traffic patterns. They exploit MAC schedules, channel hopping sequences, and mobility behavior. They apply interference only at high-impact moments. This maximizes damage while minimizing exposure. Mobile smart jammers can track specific targets. Examples include vehicle-mounted jammers following surveillance UAVs. Such attacks degrade mission objectives without causing total communication collapse.

Each tier exhibits unique vulnerabilities. IoT nodes are constrained in energy and computation. They cannot support complex defense algorithms. Vehicular networks demand ultra-low latency and high reliability. Even short disruptions can compromise

safety. UAVs rely on continuous command, control, and data links. Loss of connectivity can result in mission failure or unsafe operation. These challenges highlight the limitations of static or tier-specific defenses. Rule-based and centralized approaches do not scale. They fail to adapt to dynamic and intelligent adversaries. Effective protection requires cross-layer awareness. It also requires cooperation across tiers.

Learning-based and intelligent anti-jamming strategies are therefore essential. Cognitive methods enable sensing and adaptation. Reinforcement learning enables experience-driven optimization. Multi-agent reinforcement learning is particularly suitable. It models UAVs, vehicles, IoT nodes, and edge devices as cooperative agents. It also accommodates adversarial jammer behavior. Policies can be trained centrally and executed in a distributed manner. Such frameworks support scalability and robustness. They adapt to mobility, density variation, and protocol diversity. They enable coordinated defense without excessive signaling overhead. These properties directly address the combined threats posed by proactive, reactive, and smart jammers. This motivates the focus of this work on intelligent, cross-tier, and MARL-based anti-jamming solutions for integrated FANET-IoT-IoV systems.

Tables 5 describes corresponding learning-based defense mechanisms. The mapping highlights that static countermeasures are insufficient against adaptive adversaries. Multi-agent reinforcement learning enables coordinated, cross-tier defense by allowing nodes to jointly adapt transmission, routing, and spectrum access policies in response to dynamic jamming behavior.

Table 5. Mapping of Jamming Attacks to MARL-Based Defense Strategies.

Jamming Type	MARL-Based Defense Mechanism	Targeted Tier(s)
Proactive	Multi-agent channel selection and power adaptation	IoT, UAV, Vehicular
Reactive	Transmission timing randomization and cooperative sensing	UAV, Vehicular
Smart/Adaptive	Joint policy learning for frequency hopping, routing, and relay selection	Cross-tier (FANET-IoT-IoV)

3.5. Recent Advances in Cross-Layer and Cross-Domain Collaborative Anti-Jamming

Recent research has increasingly shifted from conventional single-layer anti-jamming techniques toward intelligent cross-layer and cross-domain collaborative defense mechanisms. Emerging studies published during 2024–2025 have investigated AI-enabled interference detection, Cognitive Radio-based dynamic spectrum management, integrated sensing and communication, blockchain-assisted spectrum security, and distributed reinforcement learning for next-generation heterogeneous wireless networks. These approaches recognize that future communication systems will simultaneously incorporate aerial platforms, vehicular networks, IoT devices, edge computing, and 6G communication technologies, making isolated anti-jamming solutions insufficient for highly dynamic operating environments.

Cross-layer approaches jointly optimize the physical, MAC, network, and application layers to improve spectrum utilization, communication reliability, latency, and energy efficiency. Similarly, cross-domain collaborative frameworks exploit cooperation among UAVs, connected vehicles, IoT devices, roadside units, and edge servers to improve global awareness of interference conditions and enable coordinated anti-jamming decisions. Recent surveys have also highlighted the growing importance of AI-driven cognitive networking, integrated sensing and communication (ISAC), and cooperative spectrum intelligence for resilient wireless communication in heterogeneous environments.

Despite these significant advances, existing studies primarily focus on either FANETs, IoT, IoV, or 6G-enabled networks independently. Very limited research has investigated a unified cognitive architecture that simultaneously integrates Cognitive Radio, cooperative

Multi-Agent Reinforcement Learning, and cross-layer optimization for coordinated anti-jamming across heterogeneous FANET–IoT–IoV systems. This limitation provides the primary motivation for the proposed Hybrid CR–MARL framework, which combines distributed spectrum awareness, cooperative learning, and adaptive decision making within a unified cross-domain architecture.

To position the proposed work with respect to the latest research, Table 6 summarizes representative studies published during 2024–2025 on AI-enabled anti-jamming, cooperative communication, and heterogeneous wireless networking. Although recent studies have made significant progress in applying machine learning and deep reinforcement learning to specific communication domains, most remain limited to single-network environments or address only individual protocol layers. In contrast, the proposed Hybrid CR–MARL framework integrates Cognitive Radio with cooperative Multi-Agent Reinforcement Learning within a unified cross-layer architecture for heterogeneous FANET–IoT–IoV systems.

Table 6. Comparison of recent (2024–2025) cross-layer and cross-domain anti-jamming approaches with the proposed framework.

Reference	FANET	IoT	IoV	Cross-Layer	Cross-Domain	AI/DRL	CR	MARL
[37]	✓	×	×	×	×	✓	×	×
[5]	×	✓	×	✓	×	✓	×	×
[2]	✓	×	×	×	×	✓	×	×
[38]	×	×	✓	✓	×	✓	×	×
[23]	×	×	✓	✓	×	×	×	×
[39]	×	✓	✓	✓	✓	✓	×	×
[40]	✓	✓	×	✓	✓	✓	×	×
[4]	✓	×	×	✓	✓	✓	×	✓
[7]	×	×	✓	✓	×	✓	×	×
[1]	×	✓	×	✓	×	✓	×	×
[17]	✓	×	×	×	×	✓	×	×
[14]	✓	×	×	✓	×	✓	×	×
[41]	✓	✓	✓	✓	✓	✓	×	✓
[22]	×	×	✓	✓	×	✓	×	×
[42]	✓	✓	✓	✓	✓	✓	×	×
Proposed Hybrid CR–MARL Framework	✓	✓	✓	✓	✓	✓	✓	✓

3.6. Contribution of the Proposed Framework over Existing Technologies

Although the enabling technologies discussed in this section, including Cognitive Radio, Multi-Agent Reinforcement Learning, cross-layer optimization, and intelligent spectrum management, have been extensively investigated individually, existing studies typically address these techniques in isolation or within a single communication domain. Consequently, they are insufficient for supporting future heterogeneous FANET–IoT–IoV environments characterized by dynamic topology, distributed intelligence, heterogeneous communication protocols, and adaptive jamming attacks.

The proposed Hybrid CR–MARL framework builds upon these established technologies and integrates them into a unified cognitive architecture. Rather than introducing a new Cognitive Radio algorithm or a new reinforcement learning algorithm, this work focuses on their coordinated integration through a common cross-layer framework. Cognitive Radio provides adaptive spectrum awareness and dynamic channel access, while Multi-Agent Reinforcement Learning enables distributed cooperative decision making among heterogeneous communication nodes. The proposed architecture further introduces a unified cognitive abstraction layer that normalizes heterogeneous observations from UAVs, IoT devices, and connected vehicles, allowing cooperative learning while preserving domain-specific communication protocols.

Therefore, the primary contribution of this work lies in the system-level integration of existing technologies into a scalable, cooperative, and cross-layer anti-jamming framework suitable for heterogeneous next-generation wireless communication systems. This

integrated perspective distinguishes the proposed framework from existing state-of-the-art approaches that generally optimize only individual communication components.

For clarity, Table 7 summarizes the principal analytical models introduced in this section together with their physical interpretation and their relevance to the proposed Hybrid CR–MARL framework.

Table 7. Summary of analytical models used in the system model.

Equation	Network Tier	Physical Meaning	Role in the Proposed Framework
Received Signal Power $P_r = P_t G_t G_r d^{-\alpha}$	FANET	Models wireless propagation and received signal strength.	Provides signal quality observations for Cognitive Radio spectrum sensing and jamming detection.
Communication Delay $D = D_{tx} + D_{queue} + D_{prop}$	IoV	Models end-to-end packet delay caused by transmission, queuing, and propagation.	Represents latency-sensitive communication performance used by MARL agents during policy optimization.
Energy Consumption $E = E_{tx} + E_{rx} + E_{idle}$	IoT	Models battery utilization of resource-constrained sensor nodes.	Allows MARL agents to balance anti-jamming performance and energy efficiency during decision making.
Normalized Observation Vector	Cross-tier	Combines RSS, SINR, PDR, residual energy, mobility, neighbor density, and channel occupancy.	Provides a common state representation enabling cooperative learning across heterogeneous FANET, IoT, and IoV environments.

The system model presented in this section establishes the communication characteristics, operational constraints, and jamming vulnerabilities of the integrated FANET–IoT–IoV architecture. Unlike existing studies that investigate these domains independently, this work models them within a unified heterogeneous communication environment. The summarized analytical models provide the physical foundation for the Cognitive Radio spectrum sensing process and the state representation employed by the proposed Multi-Agent Reinforcement Learning framework described in the following sections.

4. Jamming Detection Techniques: Algorithms and Applicability

4.1. Physical Layer Detection Techniques

Physical layer detection techniques operate directly on received signal properties. They do not require packet decoding. They are suitable for resource-constrained nodes. They are often the first line of defense against jamming attacks. However, their effectiveness depends on channel dynamics and mobility.

4.1.1. Energy Detection (Threshold-Based)

Energy detection is the simplest jamming detection method. It monitors the received signal strength or energy level. The detector compares the observed value with a predefined threshold. If the threshold is exceeded, jamming is declared. Let $r(t)$ denote the received baseband signal. The instantaneous received signal strength (RSS) over a sensing window T is given by

$$\text{RSS} = \frac{1}{T} \int_0^T |r(t)|^2 dt. \quad (16)$$

The decision rule is defined as

$$\text{If } \text{RSS} > \gamma, \text{ declare jamming,} \quad (17)$$

where γ is the detection threshold.

The threshold may be fixed or adaptive. A fixed threshold is determined from noise statistics. An adaptive threshold adjusts based on historical observations. The detection threshold is defined as

$$\gamma = \mu_{\text{RSS}} + k\sigma_{\text{RSS}} \quad (18)$$

where μ_{RSS} and σ_{RSS} denote the mean and standard deviation of the received signal strength, respectively, and k is a tunable scaling factor controlling the sensitivity of detection. Energy detection is highly effective against constant jamming. Constant jammers increase the average received energy. The method is widely used in static IoT sensor fields. Such environments exhibit limited mobility. RSS variations are relatively stable.

The computational complexity is very low. Only simple averaging and comparison are required. This makes the technique suitable for low-power IoT nodes. However, energy detection has significant limitations. In FANET and IoV environments, nodes are mobile. RSS fluctuates due to fading, shadowing, and Doppler effects. These fluctuations can resemble jamming behavior. As a result, the false positive rate increases. Reactive and smart jammers may also evade detection. They transmit intermittently. This keeps the average RSS close to normal levels.

Although received signal strength (RSS) provides a useful indication of wireless channel quality, it is insufficient by itself for reliable jamming detection in heterogeneous FANET-IoT-IoV environments. RSS measurements are influenced by mobility, path loss, multipath fading, shadowing, antenna orientation, and environmental obstacles, which may produce signal fluctuations unrelated to malicious interference. Therefore, the proposed Hybrid CR-MARL framework does not rely exclusively on RSS. Instead, RSS is combined with complementary observations, including the Signal-to-Interference-plus-Noise Ratio (SINR), Packet Delivery Ratio (PDR), channel occupancy, packet error rate (PER), residual energy, neighboring node density, node mobility, and historical interference statistics. These measurements are normalized within the Cognitive Radio engine to construct a robust multi-dimensional state representation that enables more accurate spectrum awareness and cooperative anti-jamming decision making.

4.1.2. Signal Feature Analysis

Signal feature analysis examines higher-order characteristics of the received signal. It goes beyond average energy. It analyzes statistical properties of signal samples. These properties change under jamming conditions. Let $x[n]$ denote discrete-time received signal samples. Under normal operation, $x[n]$ follows a known distribution. Jamming alters this distribution. Feature-based detection exploits this deviation. One commonly used feature is kurtosis. Kurtosis measures the peakedness of a distribution. It is defined as

$$\kappa = \frac{\mathbb{E}[(x - \mu)^4]}{\sigma^4}, \quad (19)$$

where μ and σ are the mean and standard deviation of $x[n]$. Gaussian noise has a kurtosis of three. Jamming signals often produce non-Gaussian behavior. This changes the kurtosis value. A significant deviation indicates possible jamming.

Another widely used method is the Kolmogorov–Smirnov (KS) test. The KS test compares the empirical distribution function of observed samples with a reference distribution. Let $F_n(x)$ be the empirical cumulative distribution function (CDF), and $F_0(x)$ be the reference distribution under normal conditions. The KS statistic is defined as

$$D = \sup_x |F_n(x) - F_0(x)|. \quad (20)$$

If D exceeds a predefined threshold, jamming is detected.

The Kolmogorov–Smirnov (K–S) statistic defined in Equation (20) measures the maximum deviation between the empirical cumulative distribution function (CDF) of the currently observed wireless measurements, denoted by $F_n(x)$, and the reference CDF $F_0(x)$ obtained under normal communication conditions. From a physical perspective, this statistic quantifies how much the current channel behavior differs from its expected interference-free operation. Under normal network conditions, observations such as RSS, SINR, packet delivery ratio, or channel occupancy exhibit statistical characteristics similar to the reference distribution, resulting in a small value of D . However, a jamming attack introduces abnormal interference that significantly alters these statistical distributions, causing the maximum deviation to increase. Therefore, when the K–S statistic exceeds a predefined threshold, the observed channel behavior is considered statistically abnormal, indicating the possible presence of malicious jamming.

Signal feature analysis is particularly effective for detecting reactive jamming. Reactive jammers introduce bursts of interference. These bursts alter packet inter-arrival times. They also distort signal amplitude distributions. Such changes may not significantly affect average RSS. However, they are clearly reflected in higher-order statistics.

In FANET systems, feature analysis can detect selective interference on UAV coordination packets. In IoV networks, it can identify abnormal variations in safety message timing. In IoT networks, it helps distinguish jamming from normal noise. The main advantage of signal feature analysis is improved detection accuracy. It reduces false positives compared to pure energy detection. However, computational complexity is higher. Statistical estimation requires buffering and processing samples. This may not be suitable for extremely constrained nodes.

Physical layer detection techniques are attractive due to simplicity. Energy detection is effective for strong and persistent jammers. Signal feature analysis improves robustness against stealthy attacks. However, no single method is sufficient across all tiers. Mobility and heterogeneity complicate detection. These limitations motivate adaptive and learning-based detection frameworks. Such approaches can combine multiple features and improve detection reliability.

4.2. Link and MAC Layer Detection Techniques

Link and MAC layer detection techniques rely on packet-level behavior. They operate above the physical layer. They observe communication performance rather than raw signal properties. These techniques are lightweight and widely applicable. They are especially useful when physical-layer access is limited. However, MAC-layer metrics are influenced by network dynamics. Congestion, mobility, and interference all affect observations. Careful interpretation is therefore required.

4.2.1. Packet Delivery Ratio (PDR)-Based Detection

Packet Delivery Ratio (PDR) is a commonly used metric. It measures communication reliability. It is defined as the ratio of successfully received packets to transmitted packets. Let N_{sent} denote the number of packets transmitted during an observation window. Let N_{recv} denote the number of packets successfully received. The PDR is computed as

$$\text{PDR} = \frac{N_{\text{recv}}}{N_{\text{sent}}}. \quad (21)$$

Under normal operating conditions, the PDR remains above a minimum acceptable level. Jamming attacks introduce packet corruption and collisions. This leads to a sharp reduction in PDR. A threshold-based detection rule is commonly applied:

$$\text{If } \text{PDR} < \delta, \text{ declare jamming,} \quad (22)$$

where δ is a predefined threshold. The threshold can be fixed or adaptive. A fixed threshold is chosen empirically. An adaptive threshold can be computed as

$$\delta = \mu_{\text{PDR}} - k\sigma_{\text{PDR}}, \quad (23)$$

where μ_{PDR} and σ_{PDR} represent the mean and standard deviation of PDR under normal conditions. PDR-based detection is universally applicable. It can be used in IoT, IoV, and FANET tiers. It is independent of modulation and coding schemes. It does not require access to physical-layer measurements. In IoT networks, a sudden PDR drop may indicate constant jamming. In vehicular networks, it may signal reactive jamming of safety beacons. In FANETs, it can reveal disruption of UAV coordination links.

The main advantage of PDR-based detection is simplicity. It has low computational overhead. It can be implemented on resource-constrained nodes. However, PDR degradation is not unique to jamming. Congestion can also reduce PDR. Mobility-induced path loss can cause similar effects. As a result, PDR-based detection alone cannot distinguish jamming from benign network conditions. This leads to detection ambiguity. False alarms may occur in dense or highly mobile environments.

4.2.2. Carrier Sensing Time (CST) Analysis

Carrier sensing time (CST) is a MAC-layer metric in CSMA/CA-based networks. It measures the duration for which the channel is sensed as busy. This metric is particularly relevant for IoT and IoV systems. These systems commonly use contention-based access. In the absence of jamming, CST follows a predictable pattern. Channel occupancy reflects legitimate transmissions and backoff behavior. Jamming introduces continuous or frequent interference. This prolongs the perceived busy state of the channel. Let T_{busy} denote the total busy time during an observation window T . The average carrier sensing time is given by

$$\text{CST}_{\text{avg}} = \frac{T_{\text{busy}}}{T}. \quad (24)$$

A jammer increases T_{busy} . This prevents legitimate nodes from accessing the channel. Nodes remain in backoff mode and Packet delays increase. Detection can be based on mean or variance analysis. A simple detection rule is

$$\text{If } \text{CST}_{\text{avg}} > \theta, \text{ declare jamming,} \quad (25)$$

where θ is a threshold derived from normal channel behavior. Variance-based detection improves robustness. Let σ_{CST}^2 denote the variance of CST over multiple windows. Jamming often causes abnormal variance patterns. A sudden increase indicates persistent interference. CST analysis is effective against proactive and reactive jamming. Proactive jammers keep the channel busy continuously. Reactive jammers repeatedly trigger busy states during transmissions.

In IoV systems, prolonged CST delays safety messages. This compromises real-time response. In IoT networks, it increases energy consumption. Nodes remain active while waiting for channel access. In FANETs, CST anomalies indicate disruption of shared coordination channels. The main advantage of CST-based detection is protocol awareness.

It directly reflects MAC-layer contention behavior. It does not require packet decoding. However, CST is also influenced by network load. High traffic density increases channel occupancy. Distinguishing congestion from jamming is challenging. This limitation motivates combining CST with other metrics.

Link and MAC layer detection techniques provide valuable indicators of jamming. PDR captures reliability degradation. CST reflects abnormal channel occupancy. Both are lightweight and widely applicable. However, both suffer from ambiguity. They cannot reliably distinguish jamming from congestion or mobility effects. These limitations motivate cross-layer and learning-based detection approaches. Combining MAC-layer metrics with physical-layer features improves detection accuracy.

4.3. Machine Learning-Based Detection Techniques

Machine learning-based jamming detection techniques learn patterns from data. They exploit correlations across multiple features. They are well suited for complex and dynamic environments. These techniques can distinguish jamming from benign network effects. They are increasingly used in FANET-IoT-IoV systems. Unlike rule-based methods, machine learning adapts to non-linear behavior. It can integrate physical, MAC, and network layer features. However, model design and training strategy are critical.

4.3.1. Supervised Learning-Based Detection

Supervised learning relies on labeled datasets. Each sample is associated with a class label. Typical labels include normal operation and jammed condition. The detector learns a decision boundary from examples. Let $\mathbf{x}_i \in \mathbb{R}^d$ denote a feature vector. Features may include RSS, PDR, CST, packet delay, and spectral statistics. Let $y_i \in \{0, 1\}$ denote the class label. The training dataset is

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N. \quad (26)$$

The objective is to learn a function

$$f: \mathbf{x} \rightarrow y, \quad (27)$$

that minimizes classification error.

Support Vector Machines

Support Vector Machines (SVMs) construct a separating hyperplane. The hyperplane maximizes the margin between classes. For a linear SVM, the decision function is

$$f(x) = \mathbf{w}^T x + b. \quad (28)$$

The optimization problem is

$$\min_{w,b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i, \quad (29)$$

subject to classification constraints. In this soft-margin SVM optimization problem, w is the weight vector and b is the bias scalar that define the separating hyperplane. The term $\frac{1}{2} \|\mathbf{w}\|^2$ is minimized to maximize the geometric margin between classes. The hyperparameter C controls the penalty trade-off between maximizing this margin and minimizing classification errors. The slack variable ξ_i quantifies the margin or misclassification violation for the i -th data point, summed over all N training samples. Kernel functions extend SVMs to non-linear detection. SVMs perform well in dense IoV environments. They

can separate jamming-induced anomalies from congestion effects. However, they require careful feature engineering.

Random Forests

Random Forests are ensemble classifiers. They consist of multiple decision trees. Each tree is trained on a random subset of data and features. The final decision is obtained by majority voting. Let $T_k(\mathbf{x})$ denote the output of the k -th tree. The final prediction is

$$\hat{y} = \text{mode}\{T_1(\mathbf{x}), \dots, T_K(\mathbf{x})\}. \quad (30)$$

Random Forests handle noisy features well. They are robust to overfitting. They are effective for heterogeneous FANET–IoT–IoV features. The main advantage of supervised learning is high detection accuracy. It can model complex relationships. The main limitation is data dependency. Labeled datasets are expensive to obtain. Models may degrade if network dynamics change.

4.3.2. Unsupervised Learning-Based Detection

Unsupervised learning does not require labeled data. The model learns normal behavior patterns. Deviations from these patterns are treated as anomalies. This approach is suitable for dynamic and unpredictable environments.

K-Means Clustering

K-means clustering partitions data into clusters. Each cluster represents a typical behavior pattern. The objective function is:

$$\min_{\{\mathcal{C}_k\}} \sum_{k=1}^K \sum_{x \in \mathcal{C}_k} \|x - \mu_k\|^2, \quad (31)$$

where μ_k is the centroid of cluster $\mathcal{C}_k$. Samples far from all centroids are flagged as anomalies. This may indicate jamming activity.

Autoencoder-Based Detection

Autoencoders are neural networks trained to reconstruct inputs. They learn a compressed representation of normal behavior. Let $f_\theta(\cdot)$ denote the encoder. Let $g_\phi(\cdot)$ denote the decoder. The reconstruction is

$$\hat{x} = g_\phi(f_\theta(x)). \quad (32)$$

The reconstruction error is

$$e = \|x - \hat{x}\|^2. \quad (33)$$

Under normal conditions, the error is small. Jamming causes abnormal feature patterns. This increases reconstruction error. If $e > \epsilon$, jamming is detected. Unsupervised learning is ideal for FANETs. Normal behavior is difficult to define explicitly. However, legitimate maneuvers may be misclassified. This leads to false alarms.

4.3.3. Deep Learning-Based Detection

Deep learning models capture temporal and spatial dependencies. They are suitable for large-scale and high-dimensional data.

Convolutional Neural Networks

CNNs extract local feature patterns. They are useful for spectrogram-based jamming detection. The input may be a time-frequency representation. CNNs learn spatial correlations caused by jamming signals.

Recurrent Neural Networks

RNNs and LSTMs capture temporal behavior. They model packet timing, CST evolution, and PDR trends. The hidden state update :

$$h_t = \Phi(x_t, h_{t-1}), \quad (34)$$

where $\Phi(\cdot)$ denotes a recurrent function. Deep learning offers high accuracy. However, training requires large datasets, and computation and energy costs are significant.

4.3.4. Federated Learning-Based Detection

Federated learning enables distributed model training. Raw data remains local. Only model updates are shared. Each node i trains a local model with parameters w_i . The global model is updated as:

$$w = \sum_{i=1}^M \frac{n_i}{N} w_i, \quad (35)$$

where n_i is the local dataset size. This equation defines the global weight vector w as a weighted average of local model weights w_i from M different clients in a federated or distributed learning setting. Each local client's weight contribution is scaled proportionally by its dataset fraction $\frac{n_i}{N}$, where n_i represents the local sample size and N is the total number of training samples across all clients. Federated learning preserves privacy. It reduces communication overhead. It is suitable for IoT and vehicular networks. However, it is vulnerable to Non-Independent and Identically Distributed (non-IID) data and stragglers. describes data points that are either correlated with one another or drawn from different probability distributions, rather than being uniform and random.

4.3.5. MARL-Based Cooperative Detection

Multi-agent reinforcement learning (MARL) enables distributed and cooperative jamming detection in heterogeneous FANET-IoT-IoV systems. Within this framework, each network entity acts as an autonomous learning agent that observes local radio and network features to learn effective detection policies through direct interaction with the environment. This cooperation allows agents to collectively detect stealthy and adaptive jammers that cannot be reliably identified using isolated, single-agent observations.

To formalize this setup, the cooperative detection problem is modeled as a stochastic Markov game. Let $\mathcal{N} = \{1, 2, \dots, N\}$ denote the set of autonomous agents, which may encompass heterogeneous nodes such as UAVs, vehicles, RSUs, and IoT gateways. The underlying Markov game is formally defined by the multi-component tuple:

$$\mathcal{G} = \langle \mathcal{N}, \mathcal{S}, \{\mathcal{A}^i\}, \mathcal{P}, \{r^i\}, \gamma \rangle, \quad (36)$$

where $\mathcal{S}$ represents the global environment state, $\mathcal{A}^i$ is the action space of agent i , $\mathcal{P}(s_{t+1}|s_t, \mathbf{a}_t)$ denotes the state transition probability to the next state given the current joint action, r^i is the reward function of agent i , and $\gamma \in (0, 1)$ serves as the temporal discount factor. Because the true global state is partially observable due to physical topology constraints, each agent receives a localized observation defined by:

$$o_t^i = \Omega^i(s_t), \quad (37)$$

where $\Omega^i(\cdot)$ represents the observation function mapping the true state to the agent's view. These local observations typically include metrics like RSS and SINR fluctuations, packet delivery ratios (PDR), carrier sensing time (CST), channel occupancy statistics, and neighbor alert messages.

At each time step t , agent i maps its current observation to a discrete selection by picking a localized action:

$$a_t^i \in \mathcal{A}^i. \quad (38)$$

Typical detection actions available to the nodes include NoAttack for normal operations, JamDetected to raise a local jamming alarm, ShareAlert to broadcast a suspicion to immediate neighbors, and Escalate to request cross-tier cooperative confirmation. Combining these individual choices yields the joint action vector:

$$\mathbf{a}_t = (a_t^1, a_t^2, \dots, a_t^N), \quad (39)$$

which represents the aggregate response of the network and ultimately drives the system transition. To guide this collective decision process, a carefully structured reward function is needed to encourage accurate classification while penalizing mistakes. A typical reward formulation for a given agent is given by:

$$r_t^i = \begin{cases} +R_d, & \text{correct jamming detection} \\ -R_f, & \text{false alarm} \\ -R_m, & \text{missed detection} \\ -R_c, & \text{excessive signaling} \end{cases} \quad (40)$$

where R_d , R_f , R_m , and R_c are positive constants assigned to balanced detection parameters. To foster true cross-tier collaboration, global or shared rewards are often utilized via the summation:

$$r_t = \sum_{i=1}^N r_t^i, \quad (41)$$

which effectively encourages cooperative alignment across different network tiers by pooling the individual outcomes. During the optimization process, each individual agent learns a stochastic policy:

$$\pi^i(a^i|o^i; \theta^i), \quad (42)$$

which is parameterized by the internal neural network weights θ^i . The objective of this training process is to maximize the expected discounted return over time:

$$J^i = \mathbb{E} \left[\sum_{t=0}^{\infty} \gamma^t r_t^i \right]. \quad (43)$$

Value-based methods optimize this objective by approximating the expected return using an action-value function:

$$Q^i(o^i, a^i) = \mathbb{E}[J^i|o^i, a^i], \quad (44)$$

whereas policy-gradient methods directly optimize the policy parameters π^i , often utilizing actor–critic architectures to stabilize learning. Due to partial observability, MARL implementations typically adopt the centralized training and decentralized execution (CTDE) paradigm. During the centralized training phase, a centralized critic observes the joint

observations and actions to learn critical agent correlations and approximate the centralized action-value function:

$$Q(\mathbf{o}, \mathbf{a}), \quad (45)$$

which maps global information into an evaluation score. Conversely, during decentralized execution, the critic is discarded, and each individual agent acts independently by drawing its actions directly from its local policy:

$$a^i \sim \pi^i(o^i). \quad (46)$$

This asymmetric structure is uniquely suited for resource-constrained FANET-IoT-IoV systems because it completely avoids excessive communication overhead during runtime.

To visualize this in an integrated cross-tier cooperative detection example, consider a scenario where IoT nodes detect abnormal RSS elevation, vehicles simultaneously observe increased CST and packet delays, and UAVs detect macro-scale spatial interference patterns. Individually, these localized signals may be highly ambiguous or dismissed as normal network congestion. However, MARL enables these multi-tier agents to fuse evidence implicitly through their learned policies, leading to coordinated alerts that boost total detection confidence. This joint inference is exceptionally effective against reactive and smart jammers that continuously change their behavior by dropping power or jamming selectively to evade static detectors. Through continuous online policy updates, MARL agents track these jammer reaction patterns and adapt their detection thresholds dynamically.

While MARL scales naturally with expanding network sizes due to localized execution and lightweight message exchange, practical deployment requires addressing core challenges such as environment non-stationarity, multi-agent credit assignment, and training instability. These limitations are mitigated through the use of CTDE structures, shared rewards, and periodic policy synchronization. Ultimately, MARL-based cooperative detection provides a principled framework for intelligent jamming detection that successfully captures complex spatial and temporal dependencies across heterogeneous FANET-IoT-IoV tiers, forming a foundational component of next-generation learning-enabled anti-jamming systems.

5. Anti-Jamming Mitigation Techniques: Strategies and Evaluation

The objective of this section is not to propose new anti-jamming techniques but to systematically summarize the principal mitigation mechanisms that are applicable to heterogeneous FANET-IoT-IoV environments. These techniques constitute the action space available to the proposed Hybrid CR-MARL framework. Depending on the observed network conditions, jamming characteristics, communication requirements, and resource constraints, the MARL agents learn adaptive policies that select the most appropriate mitigation strategy rather than applying all techniques simultaneously. Consequently, this section establishes the design space from which the proposed framework performs intelligent and context-aware anti-jamming decision making.

5.1. Spatial Domain Techniques

Spatial domain anti-jamming techniques exploit spatial selectivity and node mobility. They mitigate jamming by controlling the direction of signal transmission or by physically relocating nodes. These techniques are particularly effective in FANET and IoV systems, where nodes are equipped with multiple antennas or exhibit high mobility.

5.1.1. Beamforming and Directional Antennas

Beamforming is a spatial signal processing technique that employs multiple antenna elements to concentrate the transmitted energy toward an intended receiver while suppressing interference from undesired directions. Unlike omnidirectional transmission, beamforming exploits the spatial characteristics of the wireless channel to improve communication reliability, increase received signal strength, and mitigate intentional jamming without increasing the transmission power. By appropriately adjusting the amplitude and phase of each antenna element, the transmitter can form highly directional beams toward legitimate users while placing spatial nulls in the direction of jammers. Consider a transmitter equipped with an M -element antenna array. The transmitted signal can be represented as

$$x(t) = ws(t), \quad (47)$$

where $x(t) \in \mathbb{C}^{M \times 1}$ denotes the transmitted signal vector, $s(t)$ represents the complex baseband information signal, and $w = [w_1, w_2, \dots, w_M]^T$ is the beamforming weight vector. Each complex weight w_m controls the amplitude and phase of the corresponding antenna element to steer the transmitted beam toward the desired direction. The received signal arriving from an angle θ is expressed as:

$$y(t, \theta) = a^H(\theta)ws(t) + n(t), \quad (48)$$

where $y(t, \theta)$ is the received signal, $a(\theta)$ is the array steering vector corresponding to the direction θ , $(\cdot)^H$ denotes the Hermitian transpose, and $n(t)$ represents additive white Gaussian noise (AWGN). The steering vector characterizes the phase shifts experienced by signals received across different antenna elements. The communication performance is commonly evaluated using the signal-to-interference-plus-noise ratio (SINR), which is given by

$$\text{SINR} = \frac{|a^H(\theta_s)w|^2 P_s}{|a^H(\theta_j)w|^2 P_j + \sigma^2}, \quad (49)$$

where θ_s denotes the direction of the desired receiver, θ_j represents the direction of the jammer, P_s is the desired signal power, P_j is the jammer power, and σ^2 denotes the noise variance. A higher SINR indicates stronger resistance against interference and improved communication reliability. To suppress interference from the jammer, beamforming employs a null-steering constraint given by:

$$a^H(\theta_j)w = 0, \quad (50)$$

which forces the antenna array to create a spatial null in the direction of the jammer, thereby minimizing the received interference while maintaining the desired communication link. In practical wireless systems, the beamforming weights are continuously adapted according to the radio environment. One of the most widely adopted techniques is the Minimum Variance Distortionless Response (MVDR) beamformer, which determines the optimal beamforming weights by solving

$$\begin{aligned} \min_w \quad & w^H R w \\ \text{s.t.} \quad & a^H(\theta_s)w = 1, \end{aligned} \quad (51)$$

where R denotes the interference-plus-noise covariance matrix. The optimization minimizes the interference and noise power while preserving unit gain toward the desired receiver, thereby maximizing communication quality under jamming conditions. As an illustrative example, consider a UAV equipped with a phased antenna array serving as an

aerial base station for a group of ground vehicles. A stationary jammer attempts to disrupt the downlink transmission by transmitting high-power interference. Using adaptive beamforming, the UAV dynamically steers its main beam toward the legitimate vehicles while simultaneously placing a spatial null toward the jammer. Consequently, the desired communication link is maintained with minimal interference, resulting in improved transmission reliability and enhanced anti-jamming capability.

Beamforming is particularly effective against directional and stationary jammers because it exploits the spatial separation between legitimate users and interference sources. It provides significant interference suppression, extends communication range, and improves spectral efficiency. However, its practical deployment requires multiple antenna elements, accurate channel state information, and real-time beamforming optimization. These requirements increase hardware complexity, computational overhead, and energy consumption, making beamforming less suitable for highly resource-constrained IoT devices.

5.1.2. Mobility Exploitation

Mobility exploitation mitigates jamming by physical relocation. Nodes move away from regions with strong interference. This technique is unique to mobile FANET and IoV systems. It requires no additional spectrum or antenna resources. Let the spatial jamming intensity be modeled as a field

$$J(\mathbf{p}) : \mathbb{R}^2 \rightarrow \mathbb{R}, \quad (52)$$

where $\mathbf{p} = (x, y)$ denotes the node position. Higher values of $J(\mathbf{p})$ indicate stronger jamming.

Path planning incorporates jamming awareness using artificial potential fields. The total potential is defined as

$$U(\mathbf{p}) = U_{\text{goal}}(\mathbf{p}) + U_{\text{jam}}(\mathbf{p}), \quad (53)$$

where U_{goal} attracts the node to its destination and U_{jam} repels it from jamming regions. A repulsive jamming potential is defined as

$$U_{\text{jam}}(\mathbf{p}) = \begin{cases} \eta \left(\frac{1}{d(\mathbf{p})} - \frac{1}{d_0} \right)^2, & d(\mathbf{p}) \leq d_0 \\ 0, & d(\mathbf{p}) > d_0 \end{cases} \quad (54)$$

where $d(\mathbf{p})$ is the distance to the jammer, d_0 is the influence radius, and η controls repulsion strength. The movement direction is computed as

$$\mathbf{v} = -\nabla U(\mathbf{p}). \quad (55)$$

Alternatively, the environment is discretized into a graph. Each edge cost includes jamming awareness:

$$C(e) = \alpha L(e) + \beta J(e), \quad (56)$$

where $L(e)$ is the physical distance and $J(e)$ is the average jamming intensity. Algorithms such as A* compute the minimum-cost path [43]. A UAV swarm encounters localized reactive jamming. Each UAV updates its local jamming heat map. The swarm reroutes to maintain connectivity. An emergency vehicle detects persistent jamming near an intersection. It changes lanes or reroutes to preserve safety message delivery.

Mobility-based mitigation is energy-efficient for UAVs and vehicles. It is ineffective for static IoT nodes. It may increase travel distance or mission time. Coordination is required to avoid network fragmentation. Spatial domain techniques provide powerful anti-jamming capabilities. Beamforming suppresses jammers spatially. Mobility avoids

jamming physically. These techniques are complementary. They are most effective when combined with learning-based detection and cooperative control mechanisms in integrated FANET-IoT-IoV systems.

5.2. Spectral Domain Techniques

Spectral domain anti-jamming techniques mitigate interference by exploiting frequency diversity and adaptive spectrum access. Instead of remaining on a fixed carrier, nodes dynamically change operating frequencies. This reduces the effectiveness of jammers that target specific bands. These techniques are fundamental in wireless security and are widely used in FANET, IoT, and IoV systems.

5.2.1. Frequency Hopping Spread Spectrum (FHSS)

Frequency hopping spread spectrum (FHSS) mitigates jamming by rapidly switching the carrier frequency according to a pseudo-random hopping sequence. The sequence is shared only between legitimate transmitter and receiver. A jammer that does not know the sequence cannot reliably jam the signal.

Let $\mathcal{F} = \{f_1, f_2, \dots, f_K\}$ denote the set of available channels. The hopping sequence is defined as

$$f(t) = \pi(t), \quad \pi(t) \in \mathcal{F}, \quad (57)$$

where $\pi(t)$ is a pseudo-random mapping generated from a shared seed. The transmitted signal is

$$x(t) = s(t) \cos(2\pi f(t)t), \quad (58)$$

where $s(t)$ is the baseband signal. A jammer transmitting on frequency f_j successfully interferes only when

$$f_j = f(t). \quad (59)$$

The probability of a successful jamming event is therefore

$$P_{\text{jam}} = \frac{1}{K}, \quad (60)$$

assuming uniform hopping and no sequence knowledge.

Hop Rate and Processing Gain

The hop duration is denoted by T_h . Fast hopping occurs when multiple hops happen within one symbol period. Slow hopping occurs when multiple symbols share one frequency. The processing gain is approximated as

$$G_p = K, \quad (61)$$

which improves resilience against narrowband jamming.

Synchronization Requirement

Receiver synchronization is required. The receiver must align its hopping pattern with the transmitter. Let Δt be synchronization error. Successful reception requires

$$\Delta t < T_h. \quad (62)$$

Synchronization overhead increases with hop rate. This is challenging in highly mobile FANET environments.

Vulnerability to Follow-On Jamming

Advanced jammers observe frequency usage over time. They estimate the hopping sequence. A follow-on jammer predicts future hops using

$$\hat{f}(t+1) = \arg \max_{f \in \mathcal{F}} P(f|f(t), f(t-1)). \quad (63)$$

Once prediction accuracy improves, jamming success increases. This limits the effectiveness of static FHSS schemes. Bluetooth-based IoT sensors use FHSS in the 2.4 GHz ISM band. A narrowband jammer affects only a subset of hops. Most packets remain unaffected. This improves robustness in dense environments. FHSS is simple and effective against fixed-frequency jammers. It provides frequency diversity without complex hardware. However, it requires tight synchronization. Sequence management introduces overhead. It is vulnerable to intelligent and learning-based jammers.

5.2.2. Cognitive Radio and Dynamic Spectrum Access (DSA)

Cognitive radio enables dynamic spectrum access by sensing and adapting to spectrum availability. Nodes identify underutilized frequency bands. They opportunistically transmit without interfering with licensed users. Let B denote the total spectrum. It is partitioned into

$$B = B_p \cup B_s, \quad (64)$$

where B_p is occupied by primary users and B_s consists of spectrum holes.

Spectrum Sensing Model

Each channel is modeled as a binary hypothesis test:

$$\begin{cases} \mathcal{H}_0 : y(t) = n(t), & \text{channel idle} \\ \mathcal{H}_1 : y(t) = s_p(t) + n(t), & \text{primary user active} \end{cases} \quad (65)$$

where $s_p(t)$ is the primary signal. Energy detection is commonly used:

$$E = \frac{1}{T} \int_0^T |y(t)|^2 dt. \quad (66)$$

The sensing decision is

$$E \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\geq}} \lambda, \quad (67)$$

where λ is a threshold.

Dynamic Spectrum Selection

To achieve spectral agility, available channels are dynamically ranked based on a multi-objective quality metric that evaluates the suitability of a given frequency band:

$$Q(f) = w_1 \text{SINR}(f) + w_2 \text{Availability}(f) - w_3 J(f), \quad (68)$$

where $\text{SINR}(f)$ is the Signal-to-Interference-plus-Noise Ratio measuring link quality, $\text{Availability}(f)$ represents the temporal vacancy of the band to minimize collisions, and $J(f)$ is the estimated jamming activity targeting that frequency. The non-negative weighting coefficients w_1 , w_2 , and w_3 prioritize these metrics depending on operational requirements, while the negative sign before w_3 ensures that highly jammed channels explicitly penalize and reduce the overall quality score $Q(f)$. By selecting the channel that maximizes this

aggregate quality score, cognitive radios can proactively migrate to cleaner bands, thereby mitigating the impact of jammers that typically target congested unlicensed spectrum.

For instance, public safety UAVs can exploit this agility by avoiding highly contested Industrial, Scientific, and Medical (ISM) bands and operating in TV white space bands instead. This intelligent migration strategy renders targeted ISM jamming attempts ineffective and significantly improves mission reliability. However, this flexibility requires secondary users to strictly protect licensed primary users from harmful cross-tier interference. To mathematically guarantee this operational safety boundary, the probability of missed detection during spectrum sensing must satisfy the strict regulatory threshold:

$$P_{\text{md}} \leq \epsilon, \quad (69)$$

where P_{md} represents the statistical probability that an active primary user goes undetected by the secondary network, and $\epsilon \in (0, 1)$ is the maximum allowable violation constraint mandated by regulatory bodies. This upper bound limits sensing aggressiveness, ensuring that the secondary network remains silent or vacates the channel whenever a primary transmission is likely present. Consequently, satisfying this constraint introduces a critical trade-off between aggressive spectrum access and hardware or regulatory complexity, a challenge that is further exacerbated by high node mobility which complicates the accurate prediction of temporal spectrum availability.

Ultimately, these spectral-domain techniques provide essential frequency agility to counter malicious interference. While Frequency Hopping Spread Spectrum (FHSS) offers architectural simplicity against stationary, narrowband interference, cognitive radio enables intelligent, wideband spectrum avoidance based on real-time channel conditions. Nevertheless, both standalone approaches remain vulnerable to sophisticated, adaptive jammers capable of tracking frequency changes. Their overall defensive effectiveness increases significantly when combined with learning-based detection mechanisms and Multi-Agent Reinforcement Learning (MARL)-enabled spectrum coordination across integrated FANET-IoT-IoV systems, creating a robust, multi-tiered framework for secure wireless networking.

5.3. Temporal Domain Techniques

Temporal domain techniques mitigate jamming by adapting transmission behavior over time. Instead of avoiding frequencies or spatial directions, the transmitter modifies symbol rate, coding redundancy, or transmission timing. These methods are lightweight and compatible with existing protocols. They are especially useful for maintaining minimum connectivity under moderate interference.

5.3.1. Adaptive Rate Control and Channel Coding

Adaptive rate control is a physical-layer countermeasure that dynamically adjusts the transmission rate according to the current wireless channel conditions. During jamming attacks, interference reduces the quality of the received signal, making high-order modulation and high-rate channel coding unreliable. To maintain communication reliability, the transmitter continuously adapts its modulation order and coding rate based on the measured channel quality. Although this adaptation reduces the achievable throughput, it significantly improves communication robustness under adverse conditions. The quality of the communication channel is commonly represented by the instantaneous signal-to-

interference-plus-noise ratio (SINR), which quantifies the strength of the desired signal relative to the combined interference and background noise. The SINR is expressed as

$$\gamma = \frac{P_s}{P_j + \sigma^2}, \quad (70)$$

where γ denotes the instantaneous SINR, P_s is the received signal power, P_j represents the received jammer interference power, and σ^2 denotes the additive white Gaussian noise (AWGN) variance. As the jammer power increases, the SINR decreases, resulting in a less reliable communication channel. The maximum achievable transmission rate over a noisy wireless channel is bounded by the Shannon capacity theorem,

$$R \leq B \log_2(1 + \gamma), \quad (71)$$

where R is the achievable data rate (bits/s), B denotes the available channel bandwidth (Hz), and γ is the instantaneous SINR. This equation shows that the achievable data rate decreases as the SINR deteriorates due to jamming. Consequently, adaptive rate control lowers the transmission rate to maintain an acceptable packet error performance. Based on the measured SINR, the transmitter dynamically selects an appropriate modulation scheme. Higher-order modulation provides greater spectral efficiency but requires higher SINR, whereas lower-order modulation offers improved robustness against interference. The modulation adaptation policy can be represented as

$$M(\gamma) = \begin{cases} 64\text{-QAM}, & \gamma \geq \gamma_3, \\ 16\text{-QAM}, & \gamma_2 \leq \gamma < \gamma_3, \\ \text{QPSK}, & \gamma_1 \leq \gamma < \gamma_2, \\ \text{BPSK}, & \gamma < \gamma_1, \end{cases} \quad (72)$$

where $M(\gamma)$ denotes the selected modulation format, while γ_1 , γ_2 , and γ_3 are predefined SINR thresholds used to switch between different modulation schemes. As the SINR decreases, the transmitter progressively shifts from higher-order modulation (64-QAM) to more robust schemes such as QPSK and BPSK. In addition to adaptive modulation, Forward Error Correction (FEC) improves transmission reliability by introducing controlled redundancy into the transmitted data. The coding rate is defined as

$$r_c = \frac{k}{n}, \quad (73)$$

where k represents the number of information bits, n is the total number of transmitted bits after encoding, and r_c is the coding rate. A lower coding rate corresponds to greater redundancy, allowing the receiver to correct a larger number of transmission errors. The additional redundancy introduced by FEC reduces the effective user throughput, which can be expressed as

$$R_{\text{eff}} = R \times r_c, \quad (74)$$

where R_{eff} is the effective information throughput, R denotes the physical-layer transmission rate, and r_c is the coding rate. During severe jamming conditions, the system intentionally lowers the coding rate to improve error correction capability, accepting a reduction in throughput in exchange for enhanced communication reliability. The packet error probability can be approximated as

$$P_e \approx Q\left(\sqrt{2\gamma r_c}\right), \quad (75)$$

where P_e denotes the packet error probability, γ is the instantaneous SINR, r_c is the coding rate, and $Q(\cdot)$ represents the Gaussian Q-function, which gives the tail probability of the standard normal distribution. This relationship illustrates that stronger channel coding and higher SINR jointly reduce the probability of packet errors. In Internet of Vehicles (IoV) networks, reliable delivery of Basic Safety Messages (BSMs) is essential for collision avoidance and cooperative driving applications. When a jamming attack is detected, vehicles can automatically switch to robust modulation schemes such as BPSK together with lower-rate FEC coding. Although this adaptation increases transmission latency and decreases throughput, it preserves packet delivery reliability and ensures that safety-critical information continues to be exchanged. Therefore, adaptive rate control and channel coding provide a practical and low-complexity anti-jamming solution that does not require additional spectrum, antennas, or specialized hardware. Nevertheless, their effectiveness is limited under extremely powerful or wideband jamming attacks that completely saturate the communication channel.

5.3.2. Time Hopping and Unpredictable Transmission

Time hopping is a temporal-domain anti-jamming technique that improves communication resilience by randomizing the transmission instants of data packets. Instead of transmitting continuously or at fixed intervals, packets are sent as short bursts at pseudo-random time slots known only to legitimate communicating nodes. This unpredictability prevents reactive jammers from accurately detecting and disrupting ongoing transmissions, thereby exploiting the inherent sensing and reaction delay of the jammer. Assume that time is divided into equal-duration slots of length T_s . The transmitter selects the transmission instant for the k -th packet according to

$$t_k = \tau_k T_s, \quad (76)$$

where t_k denotes the transmission time of the k -th packet, T_s is the duration of each time slot, and τ_k is an index generated from a synchronized pseudo-random sequence shared between the legitimate transmitter and receiver. Because the transmission schedule is unpredictable, the jammer cannot anticipate the exact transmission instant. A reactive jammer first senses the wireless channel before emitting interference. Let T_j denote the jammer's sensing and reaction delay. Successful interference is only possible when the jammer reacts before the packet transmission has completed, which can be expressed as

$$T_j < T_s, \quad (77)$$

where T_j is the jammer reaction time and T_s is the transmission slot duration. If the jammer requires more time to react than the packet duration, the transmission is completed before jamming becomes effective. Accordingly, the probability that a packet is successfully jammed is

$$P_{\text{jam}} = \Pr(T_j < T_s), \quad (78)$$

where P_{jam} denotes the probability of successful jamming. A smaller value of P_{jam} indicates greater resistance against reactive jamming attacks. To further reduce the likelihood of successful jamming, the transmission slot duration should satisfy

$$T_s < \mathbb{E}[T_j], \quad (79)$$

where $\mathbb{E}[T_j]$ represents the expected jammer reaction delay. Under this condition, most packet transmissions finish before the jammer can generate effective interference, thereby decreasing the probability of successful attacks. In addition to reducing jamming probab-

ity, short transmission bursts also limit the energy expended by the jammer. The jammer energy consumed during an attack is approximately proportional to

$$E_j \propto P_j T_s, \quad (80)$$

where E_j denotes the jammer energy consumed during a transmission attempt, P_j is the jammer transmission power, and T_s is the burst duration. Consequently, reducing the transmission duration decreases the jammer's energy expenditure while simultaneously reducing the communication exposure time. As an illustrative example, two UAVs exchanging reconnaissance information in a contested environment can transmit encrypted packets as short pseudo-random bursts. Since the transmission timing is unpredictable, a reactive jammer is unable to synchronize its interference with the transmitted packets. Consequently, reliable and covert communication can be maintained even in the presence of intelligent reactive jammers. Overall, time hopping provides an effective and lightweight defense against reactive jamming by exploiting the jammer's response delay rather than relying on additional transmission power or spectrum resources. The technique requires accurate time synchronization and coordination between legitimate nodes and may introduce additional latency due to idle waiting intervals. Furthermore, its effectiveness is reduced against proactive or broadband jammers that continuously transmit interference across the entire communication period. Therefore, time hopping is most effective when integrated with complementary spectral, spatial, and protocol-layer anti-jamming techniques in future FANET-IoT-IoV communication architectures.

5.4. Hybrid and Proactive Techniques

Hybrid and proactive anti-jamming techniques explicitly anticipate adversarial behavior. They combine communication-theoretic defenses with decision-theoretic and system-level coordination mechanisms. These techniques are designed for environments where attackers adapt and respond strategically. They are particularly relevant for integrated FANET-IoT-IoV systems, where decentralized nodes must cooperate under uncertainty.

5.4.1. Game-Theoretic Approaches

Game-theoretic approaches model the interaction between the network and the jammer as a strategic game. The defender selects communication strategies. The jammer selects interference strategies. Each player aims to optimize its own objective. Let the defender and jammer be denoted by D and J . The strategy spaces are $\mathcal{S}_D$ and $\mathcal{S}_J$. The utility function of the defender is

$$U_D(s_D, s_J) = R(s_D, s_J) - C_D(s_D), \quad (81)$$

where $R(\cdot)$ is achievable throughput or reliability and $C_D(\cdot)$ is the cost of defense. The jammer utility is

$$U_J(s_D, s_J) = -R(s_D, s_J) - C_J(s_J), \quad (82)$$

where $C_J(\cdot)$ captures power or detection cost. In a zero-sum formulation,

$$U_D = -U_J. \quad (83)$$

A Nash equilibrium (s_D^*, s_J^*) satisfies

$$U_D(s_D^*, s_J^*) \geq U_D(s_D, s_J^*), \quad \forall s_D \in \mathcal{S}_D, \quad (84)$$

$$U_J(s_D^*, s_J^*) \geq U_J(s_D^*, s_J), \quad \forall s_J \in \mathcal{S}_J. \quad (85)$$

In many FANET scenarios, the interaction is hierarchical. The defender acts first. The jammer responds. This is modeled as a Stackelberg game. The defender solves

$$s_D^* = \arg \max_{s_D \in \mathcal{S}_D} U_D(s_D, s_J^*(s_D)), \quad (86)$$

where

$$s_J^*(s_D) = \arg \max_{s_J \in \mathcal{S}_J} U_J(s_D, s_J). \quad (87)$$

Typical defender strategies include channel selection, power control, coding rate, and beam selection. Jammer strategies include jamming power, timing, and target selection. In a distributed FANET, centralized decision-making is often impractical due to highly dynamic network topologies, limited communication range, and incomplete knowledge of the jammer's behavior. Consequently, each UAV independently solves a local game using only partial observations collected from its neighboring nodes. Since both the defender and the jammer operate under uncertainty, deterministic strategies can become predictable and vulnerable to exploitation. Therefore, mixed strategies are commonly employed, allowing each player to randomize its actions according to predefined probability distributions.

Let p_i denote the probability that the defender selects communication strategy (e.g., channel) i , and let q_j denote the probability that the jammer selects attack strategy j . The defender's expected utility is then given by

$$\mathbb{E}[U_D] = \sum_i \sum_j p_i q_j U_D(i, j), \quad (88)$$

where $\mathbb{E}[U_D]$ denotes the expected utility of the defender, p_i is the probability of selecting the i -th communication strategy, q_j is the probability that the jammer employs the j -th attack strategy, and $U_D(i, j)$ represents the utility obtained when the defender adopts strategy i while the jammer chooses strategy j . Physically, this equation computes the average communication performance that the defender can expect when both the defender and the jammer randomly select their strategies. Each possible defender–jammer strategy pair contributes to the overall utility according to the joint probability $p_i q_j$. By optimizing the probability distribution $\{p_i\}$ rather than committing to a single deterministic strategy, legitimate UAVs make their communication behavior less predictable, thereby reducing the ability of intelligent or adaptive jammers to anticipate future transmissions. As a result, mixed-strategy game formulations improve the robustness and resilience of distributed FANET communications against strategic jamming attacks. Game-theoretic approaches provide rigorous guarantees. However, equilibrium computation is complex. The jammer may not behave rationally. Imperfect information reduces solution optimality.

5.4.2. Blockchain for Trusted Coordination

Blockchain-based approaches address trust and coordination in distributed anti-jamming systems. Nodes collaboratively sense jamming and share decisions. Participants may not fully trust each other. A distributed ledger ensures integrity and accountability. Each node i reports an observation

$$o_i = \{f_i, \text{RSS}_i, \text{PDR}_i, t_i\}, \quad (89)$$

which is packaged into a transaction. Transactions are grouped into blocks. Each block contains a cryptographic hash of the previous block. Let B_k denote block k . Immutability is enforced by

$$\text{Hash}(B_k) = H(B_k \parallel \text{Hash}(B_{k-1})), \quad (90)$$

where $H(\cdot)$ is a cryptographic hash function. Consensus ensures agreement on the ledger state. To ensure that all participating nodes maintain a consistent view of the distributed ledger, a consensus mechanism is employed before appending a new block. Owing to the stringent latency requirements of FANET-IoT-IoV networks, lightweight consensus protocols such as Practical Byzantine Fault Tolerance (PBFT) are generally preferred over computationally intensive mechanisms. Under PBFT, a proposed block is accepted only when

$$N_{\text{agree}} \geq \frac{2}{3}N, \quad (91)$$

where N_{agree} denotes the number of validator nodes that approve the proposed block, and N represents the total number of participating validator nodes. This condition ensures that at least two-thirds of the validators reach agreement before the block is permanently added to the blockchain, thereby enabling the network to tolerate Byzantine (malicious or faulty) nodes while preserving the integrity and consistency of the distributed ledger. The blockchain ledger maintains a permanent and tamper-resistant record of verified jamming alerts, authenticated spectrum access decisions, and node reputation scores. These records provide a trustworthy history of network events, enabling legitimate nodes to validate collaborative sensing information, support secure spectrum coordination, and evaluate the reliability of participating entities during future anti-jamming decision-making processes. Reputation evolves over time:

$$R_i(t+1) = \alpha R_i(t) + (1-\alpha)\mathbb{I}_{\text{truth}}(i), \quad (92)$$

where $\mathbb{I}_{\text{truth}}(i)$ indicates correctness of reports. Blockchain enables tamper-resistant collaboration. It mitigates false alert injection and spoofing. It removes single points of failure. However, communication overhead grows with network size. Latency introduced by consensus may violate real-time constraints. Energy consumption limits applicability for IoT nodes. Hybrid and proactive techniques address strategic and trust-related challenges. Game-theoretic models provide formal adversarial reasoning. Blockchain enables secure and decentralized coordination. Both approaches complement learning-based detection and adaptive defenses. They are most effective when selectively applied at capable nodes such as UAVs and RSUs, while lightweight mechanisms protect constrained IoT devices.

Table 8 compares different learning paradigms used for jamming detection and mitigation in FANET-IoT-IoV systems. It highlights the limitations of conventional ML and deep learning approaches in handling dynamic, cross-tier interactions, while demonstrating the superior adaptability and coordination capabilities of MARL and federated learning.

Table 8. Comparison of Learning Paradigms for Anti-Jamming in FANET-IoT-IoV.

Method	ML	DL	MARL	FL
Learning Type	Centralized	Centralized	Distributed	Distributed
Model Complexity	Low	High	High	Medium–High
Handles Dynamics	Limited	Good	Excellent	Good
Cross-Tier Coordination	No	Limited	Yes	Yes
Data Requirement	Labeled	Large-scale	Online interaction	Local data only
Adaptation to Smart Jammers	Poor	Moderate	Strong	Moderate
Communication Overhead	Low	Medium	Medium–High	Low
Scalability	Moderate	Limited	High	Very High
Suitability for FANET	Low	Moderate	High	High
Suitability for IoT	High	Low	Moderate	High
Suitability for IoV	Moderate	High	High	High

The detection and mitigation capabilities of learning-based approaches are summarized in Table 9. The table shows that while supervised and unsupervised learning methods are effective for jamming detection, only MARL and CR-enabled learning frameworks can jointly support detection, decision-making, and cooperative mitigation in real time.

Table 9. Detection and Decision Capabilities of Learning-Based Techniques.

Technique	Detection	Mitigation	Online Learning	Cooperation	Privacy
Traditional ML	Yes	No	No	No	Low
Deep Learning	Yes	Limited	Limited	No	Low
MARL	Yes	Yes	Yes	Yes	Medium
Federated Learning	Yes	Indirect	Yes	Yes	High
CR-MARL (Hybrid)	Yes	Yes	Yes	Yes	Medium

Table 10 maps different jamming strategies to suitable learning-based defense mechanisms. The results indicate that proactive and reactive jammers can be partially handled by conventional learning methods, whereas smart and adaptive jammers require multi-agent and federated learning approaches for robust defense.

Table 10. Mapping of Jamming Types to Learning-Based Defense Paradigms.

Jamming Type	ML	DL	MARL	FL
Constant Jamming	Yes	Yes	Yes	Yes
Reactive Jamming	Limited	Moderate	Strong	Moderate
Deceptive Jamming	Poor	Good	Strong	Moderate
Smart/Adaptive Jamming	No	Limited	Excellent	Good
Mobile Jamming	Poor	Moderate	Excellent	Good
Cross-Tier Jamming	No	Limited	Excellent	Excellent

These comparative analyses collectively justify the need for an intelligent, cooperative, and spectrum-aware anti-jamming framework, motivating the CR–MARL-based architecture proposed in this work.

5.5. Role of Anti-Jamming Techniques in the Proposed CR–MARL Framework

Although the anti-jamming techniques discussed in this section are well established individually, they exhibit different strengths, limitations, computational requirements, and applicability across heterogeneous wireless environments. Therefore, the proposed Hybrid CR–MARL framework should not be interpreted as simultaneously employing every mitigation technique. Instead, these methods collectively define the candidate mitigation actions available to intelligent agents during policy learning.

The Cognitive Radio engine continuously monitors the wireless environment by performing spectrum sensing and interference characterization. Based on the observed network state, the Multi-Agent Reinforcement Learning agents cooperatively determine the most suitable mitigation strategy. For example, spectrum-aware channel switching may be selected under narrowband interference, adaptive beamforming may be preferred when directional jammers are detected, mobility-based relocation may be employed by UAVs operating in highly dynamic environments, while lightweight adaptive transmission techniques are more suitable for resource-constrained IoT devices. Consequently, the proposed framework performs context-aware anti-jamming optimization by intelligently selecting mitigation actions according to current environmental conditions instead of statically applying predefined defense mechanisms.

To improve readability, Table 11 summarizes the representative anti-jamming techniques discussed in this section, together with their applicability to different network tiers and their role within the proposed Hybrid CR–MARL framework. Rather than applying all mitigation mechanisms simultaneously, the proposed framework treats these techniques as candidate actions from which intelligent agents learn to select the most appropriate strategy according to the observed network state and jamming characteristics.

Table 11. Representative anti-jamming techniques and their role in the proposed Hybrid CR–MARL framework.

Category	Representative Techniques	Applicable Tier(s)	Primary Objective	Role in the Proposed Hybrid CR–MARL Framework
Spatial Techniques	Beamforming, UAV relocation, directional communication	FANET, IoV	Reduce jammer influence through spatial diversity	Selected dynamically by MARL agents when interference originates from specific spatial regions or highly directional jammers are detected.
Spectral Techniques	Frequency hopping, Dynamic spectrum access, Cognitive Radio	FANET, IoV, IoT	Avoid occupied or jammed channels	The CR engine continuously performs spectrum sensing, while MARL learns the optimal channel selection policy under dynamic interference conditions.
Temporal Techniques	Time hopping, adaptive retransmission, adaptive transmission scheduling	IoT, IoV	Reduce collision probability and improve communication reliability	Activated under intermittent or burst jamming conditions to minimize packet loss while satisfying latency constraints.
Network and Routing Techniques	Multi-hop routing, relay selection, topology adaptation	FANET, IoV	Maintain end-to-end connectivity	MARL agents cooperatively determine relay nodes and routing paths to bypass jammed communication regions.
Power Control Techniques	Adaptive transmission power, interference-aware power allocation	FANET, IoT	Improve communication reliability while conserving energy	Transmission power is optimized jointly with channel selection to balance anti-jamming performance and energy consumption.
Hybrid Intelligent Techniques	Game theory, Machine Learning, Deep Reinforcement Learning, Blockchain	Cross-tier	Adaptive and cooperative anti-jamming decision making	The proposed framework integrates Cognitive Radio with cooperative MARL to intelligently select the most appropriate mitigation strategy according to real-time network observations.

6. Proposed Hybrid Framework: CR-MARL for Integrated Resilience

6.1. Framework Architecture Overview

The proposed framework is a cross-layer, distributed intelligence architecture designed for the integrated FANET–IoT–IoV environment. It enables proactive and adaptive anti-jamming through continuous sensing, collaborative cognition, and intelligent decision-making. The framework consists of three tightly coupled engines operating in a closed feedback loop: a perception engine, a cognitive engine, and a decision and action engine. The architecture supports heterogeneous node capabilities while maintaining global coordination.

The motivation for integrating cognitive radio with multi-agent reinforcement learning is summarized in Table 12. This table illustrates how the proposed CR–MARL framework addresses key challenges of heterogeneity, mobility, scalability, and adversarial adaptation in integrated FANET–IoT–IoV environments.

Table 12. Motivation for CR–MARL Integration in FANET–IoT–IoV.

Challenge	CR–MARL Advantage
Heterogeneous tiers	Learns tier-specific and global policies
Dynamic topology	Online adaptation via reinforcement learning
Energy constraints	Reward-aware energy optimization
Smart jammers	Adversarial learning and policy evolution
Scalability	Distributed agents with local observations
Privacy concerns	Compatible with federated updates

6.1.1. Perception Engine (ML-Based)

The Perception Engine operates locally at each IoT device, vehicle, and UAV to provide the first line of defense against wireless jamming attacks. Its primary objective is to continuously monitor the communication environment and identify abnormal channel behavior at an early stage before network performance is significantly degraded. Since obtaining labeled datasets for all possible jamming scenarios is challenging in highly dynamic FANET–IoT–IoV environments, the proposed framework employs a lightweight unsupervised learning model based on an autoencoder. By learning only the characteristics of normal communication, the autoencoder can identify previously unseen jamming attacks through deviations from the learned behavior. Each node periodically collects a set of physical- and link-layer measurements over a sliding observation window to construct the local feature vector

$$\mathbf{x}_t = [\text{RSS}_t, \text{PDR}_t, \text{CST}_t, \text{SINR}_t], \quad (93)$$

where $\mathbf{x}_t$ denotes the feature vector at time instant t , RSS_t is the received signal strength, PDR_t represents the packet delivery ratio, CST_t denotes the channel sensing time (or channel busy duration), and SINR_t is the measured signal-to-interference-plus-noise ratio. These measurements collectively capture the current wireless channel conditions and are highly sensitive to the presence of jamming interference. The collected feature vector is processed by the encoder of the autoencoder network, which compresses the high-dimensional input into a compact latent representation according to

$$\mathbf{z}_t = f_\theta(\mathbf{x}_t), \quad (94)$$

where $\mathbf{z}_t$ denotes the latent feature vector, $f_\theta(\cdot)$ represents the encoder function, and θ is the set of trainable parameters (weights and biases) of the encoder. The latent representation preserves the most informative characteristics of normal network behavior while removing redundant information.

The decoder subsequently reconstructs the original feature vector from the latent representation as

$$\hat{\mathbf{x}}_t = g_\phi(\mathbf{z}_t), \quad (95)$$

where $\hat{\mathbf{x}}_t$ denotes the reconstructed feature vector, $g_\phi(\cdot)$ represents the decoder function, and ϕ is the set of trainable decoder parameters. During normal network operation, the reconstructed feature vector closely matches the original input because the autoencoder has been trained using only legitimate communication data.

The quality of reconstruction is quantified using the squared Euclidean reconstruction error

$$e_t = \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|_2^2, \quad (96)$$

where e_t denotes the reconstruction error at time instant t , $\mathbf{x}_t$ is the observed feature vector, $\hat{\mathbf{x}}_t$ is the reconstructed feature vector, and $\|\cdot\|_2$ denotes the Euclidean norm. Physically, the reconstruction error measures the difference between the observed channel conditions and the normal communication patterns learned by the autoencoder. Under normal operating conditions, this error remains small. However, during a jamming attack, significant changes in RSS, PDR, CST, and SINR produce feature vectors that differ substantially from the learned normal behavior, resulting in a larger reconstruction error.

To provide a normalized measure of abnormality that is less sensitive to natural channel variations, the reconstruction error is converted into a Jamming Suspicion Score (JSS), defined as

$$\text{JSS}_t = \frac{e_t - \mu_e}{\sigma_e}, \quad (97)$$

where JSS_t denotes the Jamming Suspicion Score at time t , e_t is the current reconstruction error, μ_e is the mean reconstruction error obtained from normal operating data during the training phase, and σ_e is the corresponding standard deviation. This normalization expresses the reconstruction error relative to the expected distribution of normal behavior, thereby reducing the influence of channel variations caused by mobility, fading, or environmental changes.

A local jamming alert is generated whenever

$$\text{JSS}_t > \tau, \quad (98)$$

where τ denotes the anomaly detection threshold. The threshold is determined during the training or validation stage using reconstruction errors obtained under normal operating conditions. A commonly adopted statistical criterion is to select $\tau = \mu_e + k\sigma_e$, where k is a design parameter that controls the trade-off between false alarms and missed detections. Consequently, only reconstruction errors that significantly deviate from normal communication behavior are classified as potential jamming events.

The proposed perception engine is computationally lightweight and requires only locally collected channel measurements, making it well suited for resource-constrained IoT devices while remaining equally applicable to vehicles and UAVs. Furthermore, because the autoencoder learns normal communication behavior rather than specific attack signatures, it can detect previously unseen jamming attacks without requiring explicitly labeled training data, enabling robust operation in dynamic FANET-IoT-IoV environments.

Local Perception Algorithm

Algorithm 1 summarizes the operation of the proposed local perception engine executed at each IoT device, vehicle, or UAV. First, the node continuously collects local channel measurements to construct the feature vector $\mathbf{x}_t$. The feature vector is then encoded into a latent representation and reconstructed by the autoencoder to estimate the corresponding

reconstruction error e_t . This error is normalized to compute the Jamming Suspicion Score (JSS), which quantifies the degree of deviation from normal communication behavior. If the computed JSS exceeds the predefined threshold τ , the node identifies the presence of a potential jamming attack and generates a local jamming alert for subsequent collaborative analysis and mitigation.

Algorithm 1 Local Perception and Jamming Suspicion Score Computation

- 1: Collect local channel measurements and construct feature vector $\mathbf{x}_t$
 - 2: Encode $\mathbf{x}_t$ into latent representation $\mathbf{z}_t$
 - 3: Reconstruct feature vector $\hat{\mathbf{x}}_t$
 - 4: Compute reconstruction error e_t
 - 5: Compute normalized Jamming Suspicion Score (JSS) JSS_t
 - 6: **if** $\text{JSS}_t > \tau$ **then**
 - 7: Generate local jamming alert
 - 8: **end if**
-

6.1.2. Cognitive Engine (CR-Based)

The cognitive engine aggregates local perceptions into a global view. It maintains a collaborative spatial–temporal jamming map. This engine may operate at a UAV, RSU, or distributed cluster head. Each node i reports

$$\mathcal{R}_i(t) = \{\mathbf{p}_i(t), \text{JSS}_i(t), t\}, \quad (99)$$

where $\mathbf{p}_i(t)$ is location. Reports are encrypted or anonymized. The jamming map is modeled as a continuous field:

$$J(\mathbf{p}, t) = \sum_{i=1}^N w_i K(\|\mathbf{p} - \mathbf{p}_i\|) \text{JSS}_i(t), \quad (100)$$

where $K(\cdot)$ is a spatial kernel and w_i reflects node reliability. Consensus-based fusion is used:

$$J^{(k+1)} = \alpha J^{(k)} + (1 - \alpha) \bar{J}, \quad (101)$$

where $\bar{J}$ is the mean estimate across participants. In high-security scenarios, a lightweight blockchain is used. Each report is stored immutably. Only validated alerts influence the map. This prevents false alert injection. The cognitive engine enables spectrum awareness, jammer localization, and prediction of attack evolution.

Collaborative Jamming Map Update Algorithm

Algorithm 2 summarizes the collaborative construction and update of the regional jamming map within the proposed framework. The process begins by collecting local jamming reports from neighboring IoT devices, vehicles, and UAVs. Each report is first validated through the distributed consensus mechanism to eliminate false or malicious observations before being incorporated into the global situational awareness process. The validated reports are then assigned reliability weights and fused using a spatial kernel function to estimate the regional jamming intensity. To suppress transient fluctuations caused by channel fading, mobility, or temporary interference, temporal smoothing is subsequently applied to the constructed map. Finally, the updated collaborative jamming map is disseminated to neighboring network entities, enabling all participating nodes to maintain a consistent view of the current jamming environment and support coordinated anti-jamming decision-making.

Algorithm 2 Collaborative Jamming Map Construction and Update**Require:** Local jamming reports $\mathcal{R}_i(t)$ from neighboring IoT nodes, vehicles, and UAVs**Ensure:** Updated collaborative jamming map $J_{\text{smooth}}(\mathbf{p}, t)$

- 1: Receive local jamming reports $\mathcal{R}_i(t)$
- 2: Validate reports using the distributed consensus mechanism
- 3: Assign reliability weights w_i to validated reports
- 4: Construct the spatial jamming map $J(\mathbf{p}, t)$ using kernel-based fusion
- 5: Apply temporal smoothing to obtain $J_{\text{smooth}}(\mathbf{p}, t)$
- 6: Broadcast the updated collaborative jamming map to neighboring nodes

6.1.3. Decision and Action Engine

The decision and action engine is the core intelligence unit. It selects optimal counter-measures. It is typically implemented at a cluster head. Execution is distributed.

The problem is modeled as a multi-agent Markov decision process. The global state is

$$s_t = \{J(\mathbf{p}, t), \mathbf{E}_t, \mathbf{M}_t, \mathbf{Q}_t\}, \quad (102)$$

where $\mathbf{E}_t$ is node energy, $\mathbf{M}_t$ is mobility state, and $\mathbf{Q}_t$ is traffic priority.

The action space $\mathcal{A}$ contains `switch_channel`, `adjust_power`, `initiate_beamforming`, `trigger_UAV_reroute`, and `change_modulation`

The reward function is

$$r_t = \lambda_1 \text{PDR}_t - \lambda_2 \text{Latency}_t - \lambda_3 \text{Energy}_t - \lambda_4 \text{ControlCost}_t. \quad (103)$$

A deep reinforcement learning agent learns the optimal policy

$$\pi^*(a|s) = \arg \max_{\pi} \mathbb{E} \left[\sum_t \gamma^t r_t \right]. \quad (104)$$

DQN approximates the action-value function:

$$Q(s, a; \theta) \approx \mathbb{E}[r_t + \gamma \max_{a'} Q(s_{t+1}, a')], \quad (105)$$

while PPO optimizes a clipped policy objective for stability.

Decision Engine Algorithm

Algorithm 3 summarizes the operation of the proposed DRL-based Decision and Action Engine. The algorithm begins by observing the current global network state, which includes the collaborative jamming map, node energy levels, mobility information, and traffic priorities collected from the FANET-IoT-IoV network. Based on this state, the trained policy network selects the optimal anti-jamming action, such as adaptive channel selection, transmission power adjustment, beamforming, modulation switching, or UAV trajectory adaptation. The selected action is then executed across the corresponding network nodes, after which the resulting network performance is observed to compute a reward reflecting communication reliability, latency, and energy efficiency. Finally, the DRL agent updates its policy using the observed reward and the next network state, enabling continuous learning and adaptation to dynamic jamming environments.

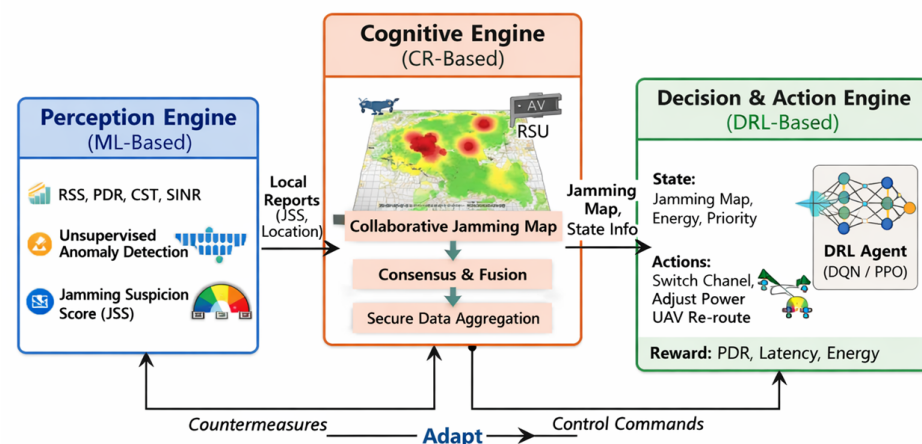
Algorithm 3 DRL-Based Decision and Action Selection**Require:** Current network state s_t **Ensure:** Optimal anti-jamming action a_t

- 1: Observe the global network state s_t
- 2: Select the optimal action $a_t \sim \pi_\theta(s_t)$
- 3: Execute the selected anti-jamming action across the network
- 4: Observe the immediate reward r_t and the next state s_{t+1}
- 5: Update the policy parameters θ using the observed transition (s_t, a_t, r_t, s_{t+1})

6.2. Framework Characteristics

The three engines form a closed control loop. Perception provides early warning. Cognition builds global awareness. Decision-making enables coordinated response. The framework adapts to mobility, heterogeneity, and intelligent jammers. It scales naturally across FANET-IoT-IoV tiers.

Figure 6 illustrates the proposed cross-layer anti-jamming framework for the integrated FANET-IoT-IoV environment. The Perception Engine operates locally at each node, collecting metrics such as RSS, PDR, and CST to compute a Jamming Suspicion Score (JSS). These local reports feed into the Cognitive Engine, which maintains a Collaborative Jamming Map by fusing spatial-temporal data from multiple nodes using consensus or lightweight blockchain mechanisms. The Decision and Action Engine then takes the global jamming map, node states, and traffic priorities as input to select optimal countermeasures, including channel switching, power adjustment, beamforming, UAV rerouting, or modulation changes, using a multi-agent deep reinforcement learning policy. Feedback loops ensure continuous adaptation and cross-tier coordination.

**Figure 6.** CR-MARL Framework for Jamming Resilience.**6.3. Operational Workflow**

The proposed framework operates as a closed-loop perception–cognition–decision cycle that continuously detects jamming attacks, evaluates network conditions, and adapts communication strategies across the integrated FANET-IoT-IoV architecture. As illustrated in Figure 6, intelligence is distributed across the sensing, edge, and aerial layers, enabling collaborative anti-jamming decisions in real time. The complete operational workflow consists of six sequential stages.

6.3.1. Stage 1: Local Sensing and Jamming Detection

The workflow begins at the IoT sensing layer, where each node periodically monitors wireless channel conditions using a lightweight, low-duty-cycle sensing mechanism. Local channel metrics, including Received Signal Strength (RSS), Packet Delivery Ratio (PDR),

Carrier Sense Time (CST), and other physical-layer indicators, are collected over predefined observation windows rather than through uninterrupted sensing. Furthermore, the sensing interval can be adaptively adjusted according to the node's residual energy, traffic priority, and perceived jamming risk, thereby reducing energy consumption while maintaining reliable jamming detection. These measurements are subsequently processed by the local Perception Engine to estimate the likelihood of abnormal channel behavior.

The degree of abnormality is quantified through the Jamming Suspicion Score (JSS), defined as:

$$\text{JSS}_t = \frac{\|\mathbf{x}_t - \hat{\mathbf{x}}_t\|_2^2 - \mu_e}{\sigma_e}, \quad (106)$$

where $\mathbf{x}_t$ denotes the observed feature vector at time t , $\hat{\mathbf{x}}_t$ is the corresponding reconstruction generated by the unsupervised anomaly detection model, while μ_e and σ_e represent the mean and standard deviation of the reconstruction error obtained during normal network operation. When the computed JSS exceeds a predefined threshold τ , the node identifies a potential jamming event and generates a local alarm.

6.3.2. Stage 2: Collaborative Information Reporting

After detecting a suspicious event, the affected node reports the anomaly to nearby vehicles, Road Side Units (RSUs), or UAVs depending on the current network topology. Each report contains the node location, detection confidence, and observation time, which together form the local sensing report

$$\mathcal{R}_i(t) = \{\mathbf{p}_i, \text{JSS}_i(t), t\}, \quad (107)$$

where $\mathbf{p}_i$ denotes the geographical position of node i , $\text{JSS}_i(t)$ represents its local jamming suspicion score, and t is the reporting time. Reports may be transmitted directly or aggregated through multi-hop communication to reduce communication overhead and energy consumption.

6.3.3. Stage 3: Regional Situation Awareness

The Cognitive Engine residing on UAVs or RSU cluster heads fuses reports received from multiple nodes to construct a regional collaborative jamming map. The spatial threat level at location $\mathbf{p}$ is estimated as

$$J(\mathbf{p}, t) = \sum_{i=1}^N w_i K(\|\mathbf{p} - \mathbf{p}_i\|) \text{JSS}_i(t), \quad (108)$$

where N is the number of reporting nodes, w_i denotes the reliability weight assigned to node i , and $K(\cdot)$ is a spatial kernel that models the influence of neighboring observations.

To suppress short-term fluctuations and emphasize persistent attacks, temporal smoothing is performed according to

$$J_{\text{smooth}}(\mathbf{p}, t) = \alpha J(\mathbf{p}, t) + (1 - \alpha) J(\mathbf{p}, t - 1), \quad (109)$$

where α is the temporal smoothing factor. The resulting map provides a comprehensive view of the spatial distribution and temporal evolution of jamming activities throughout the heterogeneous network.

6.3.4. Stage 4: Intelligent Decision Making

Based on the regional jamming map, the Decision and Action Engine constructs the current network state

$$s_t = \{J(\mathbf{p}, t), \mathbf{E}_t, \mathbf{M}_t, \mathbf{Q}_t\}, \quad (110)$$

where E_t denotes the residual energy of network nodes, M_t represents node mobility information, and Q_t corresponds to traffic priority levels.

The Multi-Agent Deep Reinforcement Learning (MADRL) policy π_θ selects the optimal anti-jamming action by maximizing the expected cumulative discounted reward,

$$a_t = \pi_\theta(s_t) = \arg \max_a \mathbb{E} \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \right], \quad (111)$$

where γ is the reward discount factor and r_t is the immediate reward. Depending on the network conditions, the selected action may involve adaptive channel allocation, transmission power control, UAV trajectory optimization, beamforming, modulation adaptation, stronger forward error correction, or traffic rerouting to bypass jammed regions.

6.3.5. Stage 5: Performance Evaluation and Policy Learning

After executing the selected action, the network continuously monitors its operational performance using communication metrics such as packet delivery ratio, latency, and energy consumption. These observations are combined into the reinforcement learning reward

$$r_t = \lambda_1 \text{PDR}_t - \lambda_2 \text{Latency}_t - \lambda_3 \text{Energy}_t, \quad (112)$$

where λ_1 , λ_2 , and λ_3 are weighting coefficients that balance communication reliability, delay, and energy efficiency. This reward enables the DRL agent to continuously refine its policy and improve future anti-jamming decisions under dynamic operating conditions.

6.3.6. Stage 6: Continuous Closed-Loop Adaptation

The above stages execute continuously throughout network operation, forming an adaptive perception–cognition–action loop. Information collected by distributed IoT sensors is fused at edge and aerial nodes, while intelligent control decisions are disseminated back to the network to mitigate newly emerging threats. The framework naturally accommodates the heterogeneous capabilities of different network tiers by exploiting the dense sensing capability of IoT devices, the computational resources of vehicles and RSUs, and the mobility and wide-area coverage provided by UAVs. Consequently, the proposed architecture delivers a scalable, resilient, and intelligent anti-jamming solution capable of protecting future integrated FANET–IoT–IoV communication systems against dynamic and coordinated jamming attacks.

6.4. Cooperative Learning Mechanism Among MARL Agents

The proposed framework adopts a Centralized Training and Decentralized Execution (CTDE) paradigm to enable cooperation among distributed agents while maintaining scalability in heterogeneous FANET–IoT–IoV environments. Each intelligent node, including UAVs, roadside units, IoT gateways, and edge servers, is modeled as an autonomous MARL agent. Every agent continuously observes its local wireless environment and constructs a state vector consisting of normalized network measurements such as RSS, SINR, channel occupancy, packet delivery ratio (PDR), residual energy, node mobility, neighboring node density, queue utilization, and local interference statistics.

During centralized training, agents periodically upload compact state representations, selected actions, and received rewards to a centralized training server located at the edge or cloud. The centralized critic aggregates information from all participating agents to estimate the global network state and optimize a shared cooperative policy. Consequently, the learned policy captures cross-tier dependencies that cannot be inferred from local observations alone. The cooperation among agents is achieved through periodic exchange

of lightweight contextual information rather than raw sensor data. Examples include interference alerts, local spectrum occupancy maps, neighboring channel availability, relay availability, and mobility updates. This information enables each agent to improve its local perception while avoiding excessive communication overhead.

Once training is completed, the optimized policy is distributed to all agents. During online deployment, each agent executes the learned policy independently using only locally available observations. Therefore, the framework satisfies the decentralized execution requirement while preserving the cooperative behavior learned during centralized training. The cooperative objective of the MARL framework is to maximize the long-term communication performance of the integrated network rather than optimizing individual node performance. Agents jointly learn spectrum selection, relay selection, transmission power adaptation, and routing decisions that collectively maximize packet delivery ratio, spectrum utilization, energy efficiency, and communication reliability while minimizing interference and jamming impact.

Algorithm 4 summarizes the cooperative learning procedure adopted by the proposed Hybrid CR-MARL framework. During centralized training, distributed agents periodically exchange lightweight contextual information while a centralized critic utilizes the aggregated global state to optimize cooperative policies. After convergence, only the trained actor networks are deployed to individual UAVs, IoT gateways, roadside units, and edge servers. During online operation, each agent independently selects anti-jamming actions using locally observed information, thereby preserving decentralized execution while benefiting from the cooperative knowledge acquired during centralized training.

Algorithm 4 Cooperative MARL Training Procedure Using Centralized Training and Decentralized Execution (CTDE)

Require: Set of agents $\mathcal{A} = \{a_1, a_2, \dots, a_N\}$, learning rate α , discount factor γ , replay buffer $\mathcal{D}$

```

1: Initialize actor network  $\pi_{\theta_i}$  for each agent  $a_i$ 
2: Initialize centralized critic network  $Q_\phi$ 
3: Initialize replay buffer  $\mathcal{D}$ 
4: for each training episode do
5:   Reset wireless environment
6:   for each decision interval  $t$  do
7:     for all agents  $a_i$  in parallel do
8:       Observe local state  $s_i^t$ 
9:       Measure RSS, SINR, PDR, residual energy, mobility, and channel occupancy
10:      Exchange lightweight observations with neighboring agents
11:      Execute action  $a_i^t = \pi_{\theta_i}(s_i^t)$ 
12:      Receive reward  $r_i^t$ 
13:      Observe next state  $s_i^{t+1}$ 
14:     end for
15:     Aggregate local observations to construct global state  $S^t = \{s_1^t, s_2^t, \dots, s_N^t\}$ 
16:     Store transition  $(S^t, A^t, R^t, S^{t+1})$  in replay buffer  $\mathcal{D}$ 
17:     Sample mini-batch from  $\mathcal{D}$ 
18:     Update centralized critic using global observations
19:     for all agents do
20:       Update actor policy using critic gradients
21:     end for
22:   end for
23: end for
24: Broadcast optimized actor parameters to all agents
25: for all agents do
26:   Execute learned policy independently using only local observations
27: end for

```

6.5. Information Exchange and Operational Workflow

The proposed Hybrid CR–MARL framework operates in discrete decision intervals denoted by $t_k = k\Delta t$, where Δt represents the sensing and decision update period. During each interval, the Perception Engine continuously monitors the wireless environment and collects local observations from UAVs, IoT nodes, roadside units (RSUs), and edge servers. Each sensing node generates a standardized observation record:

$$\mathbf{o}_i = [RSS, SINR, PDR, PER, E_r, M, N_d, C_o, T],$$

where RSS denotes the received signal strength, $SINR$ is the signal-to-interference-plus-noise ratio, PDR represents the packet delivery ratio, PER is the packet error rate, E_r denotes the residual energy, M represents the node mobility state, N_d is the neighboring node density, C_o denotes channel occupancy, and T is the timestamp.

Whenever the observed measurements violate predefined statistical thresholds or the anomaly detector identifies abnormal channel behavior, the Perception Engine immediately generates an anomaly report. Rather than transmitting raw sensing data, each report contains a compact feature vector together with the node identifier, geographical position, confidence score, and timestamp. These reports are periodically transmitted to the Cognition Engine through lightweight control messages, thereby minimizing communication overhead.

The Cognition Engine aggregates anomaly reports received from multiple agents and fuses them with historical observations stored in the global knowledge base. A dynamic interference map is then updated by estimating the spatial and temporal distribution of jammer activity. The interference map stores information regarding jammer location, interference intensity, affected channels, confidence level, and the corresponding validity period. As new observations become available, obsolete entries are discarded while recent measurements are incorporated using weighted temporal averaging to maintain an up-to-date representation of the wireless environment.

The updated interference map together with the current network observations forms the global state representation supplied to the Multi-Agent Reinforcement Learning (MARL) module. Based on this information, the Decision-Making Engine determines the optimal anti-jamming strategy, including channel switching, relay selection, transmission power adaptation, routing modification, or UAV repositioning. Finally, the selected mitigation actions are transmitted back to the corresponding network nodes as compact control commands. Each node executes the received action locally and reports the resulting communication performance during the next sensing interval. This closed-loop interaction enables continuous adaptation to dynamic jamming conditions while maintaining low communication overhead and scalable decentralized execution.

Table 13 summarizes the information exchange among the three functional engines of the proposed Hybrid CR–MARL framework. The Perception Engine continuously monitors the wireless environment and forwards normalized observations together with anomaly alerts to the Cognition Engine. Based on the aggregated observations received from multiple agents, the Cognition Engine maintains an updated interference map and constructs the global network state. The Decision-Making Engine subsequently utilizes this information to determine the optimal anti-jamming actions, including channel selection, relay selection, routing adaptation, transmission power control, and UAV repositioning. This sequential information flow enables timely detection of jamming attacks while supporting cooperative and adaptive decision making across heterogeneous FANET–IoT–IoV environments.

Table 13. Information exchange among the three functional engines.

Engine	Output Information	Destination
Perception Engine	RSS, SINR, PDR, PER, residual energy, mobility, anomaly alerts	Cognition Engine
Cognition Engine	Updated interference map, jammer confidence, channel occupancy, global state vector	Decision-Making Engine
Decision-Making Engine	Channel selection, routing updates, relay selection, power control, UAV repositioning	UAVs, IoT nodes, RSUs, vehicles

6.6. MARL Formulation and CTDE Implementation

The proposed anti-jamming framework models each UAV, roadside unit, edge server, or IoT gateway as an intelligent reinforcement learning agent. Each agent independently interacts with the wireless environment while cooperatively learning through the Centralized Training and Decentralized Execution (CTDE) paradigm.

State Space:

Each agent observes a local state vector that represents both communication quality and network conditions,

$$s_i = [RSS, SINR, PDR, PER, E_r, C_o, N_d, M, L], \quad (113)$$

where RSS denotes received signal strength, $SINR$ represents the signal-to-interference-plus-noise ratio, PDR is the packet delivery ratio, PER denotes the packet error rate, E_r represents residual energy, C_o is channel occupancy, N_d denotes neighboring node density, M represents node mobility, and L denotes the estimated jammer location or interference intensity.

Action Space:

Each agent selects one action from

$$A = \{a_1, a_2, \dots, a_n\}, \quad (114)$$

where the available actions include

- Channel switching,
- Transmission power adaptation,
- Relay selection,
- Routing update,
- UAV repositioning,
- Spectrum sensing interval adjustment.

The selected action depends on the current local state and the learned policy.

Reward Function:

The immediate reward jointly considers communication reliability, latency, energy consumption, and successful anti-jamming performance,

$$R = w_1 PDR + w_2 SINR - w_3 D - w_4 E, \quad (115)$$

where D denotes communication delay, E represents energy consumption, and $w_1, \dots, w_4$ are weighting coefficients satisfying

$$\sum_i w_i = 1.$$

Higher rewards correspond to reliable communication with low delay and reduced energy consumption under jamming conditions.

6.6.1. Communication Overhead During Centralized Training

During centralized training, individual agents periodically exchange only compact observation vectors instead of raw sensing data. Each observation contains normalized communication statistics including RSS, SINR, channel occupancy, residual energy, packet delivery ratio, and mobility information. Consequently, the communication overhead grows approximately linearly with the number of participating agents,

$$C_{train} = N_a \times S_o, \quad (116)$$

where N_a denotes the number of agents and S_o represents the size of the exchanged observation vector. Since only lightweight feature vectors are transmitted, the communication overhead remains significantly lower than exchanging raw spectrum measurements or complete environmental maps.

6.6.2. Synchronization During Decentralized Execution

After centralized training converges, the optimized actor parameters are broadcast to all participating agents. During online operation, each agent performs inference independently using only locally observed information without requiring continuous communication with the centralized critic. Parameter synchronization is performed periodically or whenever significant environmental changes are detected. This asynchronous synchronization mechanism substantially reduces communication overhead while maintaining policy consistency among distributed agents operating across the heterogeneous FANET-IoT-IoV environment.

Table 14 presents the key components of the proposed MARL formulation, including the learning agent, state representation, action space, reward function, training strategy, execution mechanism, and synchronization process. The formulation demonstrates how heterogeneous network nodes cooperate under the CTDE paradigm to learn coordinated anti-jamming policies while relying only on local observations during online deployment. This enables scalable, low-overhead, and adaptive decision making in integrated FANET-IoT-IoV networks.

Table 14. MARL formulation adopted by the proposed Hybrid CR-MARL framework.

Component	Description
Agent	UAV, RSU, IoT Gateway, Edge Node
State	RSS, SINR, PDR, PER, Energy, Mobility
Action	Channel, Relay, Power, Routing, UAV Position
Reward	Reliability + SINR – Delay – Energy
Training	Centralized Critic
Execution	Decentralized Actors
Synchronization	Periodic Parameter Broadcast

6.7. Computational Complexity and Deployment Feasibility

The proposed Hybrid CR-MARL framework follows the Centralized Training and Decentralized Execution (CTDE) paradigm, in which computationally intensive training

is performed offline at centralized edge or cloud servers, while only lightweight policy inference is executed by distributed network nodes. This separation significantly reduces the computational burden on resource-constrained UAVs and IoT devices.

During centralized training, the complexity depends on the number of agents, neural network parameters, and training episodes. Assuming N_a agents, network parameter size P , and E training episodes, the overall training complexity can be approximated as

$$\mathcal{O}_{train} = \mathcal{O}(E N_a P). \quad (117)$$

Since training is performed offline using dedicated computing resources, this complexity does not affect the real-time operation of the communication network.

During online deployment, each agent performs only a forward pass through its pretrained actor network to determine the optimal anti-jamming action. Consequently, the inference complexity is approximately

$$\mathcal{O}_{infer} = \mathcal{O}(P), \quad (118)$$

which is significantly lower than the corresponding training complexity because no gradient computation or parameter optimization is required.

Furthermore, only compact normalized observation vectors are exchanged during centralized training, while decentralized execution relies exclusively on locally available observations. This substantially reduces communication overhead and enables scalable deployment across heterogeneous FANET-IoT-IoV environments.

Table 15 summarizes the computational complexity of the proposed Hybrid CR-MARL framework. The analysis shows that the computationally intensive optimization process is confined to the centralized training phase, which is executed on edge or cloud computing infrastructure. During online deployment, each distributed agent performs only lightweight policy inference using pretrained neural network parameters, thereby significantly reducing computational and energy requirements. Consequently, the proposed framework is suitable for practical deployment on resource-constrained UAVs, roadside units, and IoT gateways while maintaining adaptive and cooperative anti-jamming capabilities.

Table 15. Computational complexity of the proposed Hybrid CR-MARL framework.

Component	Complexity
Centralized training	$\mathcal{O}(E N_a P)$
Online inference	$\mathcal{O}(P)$
Parameter synchronization	$\mathcal{O}(N_a)$
Observation exchange	$\mathcal{O}(N_a S_o)$
Deployment platform	Edge/Cloud
Inference platform	UAVs, RSUs, IoT Gateways

Secure Cooperative Learning and Trust Management

In the Hybrid CR-MARL framework, participating UAVs, roadside units, IoT gateways, and edge nodes are assumed to cooperate honestly while exchanging spectrum observations and interference information. However, in practical deployments, compromised or malicious nodes may intentionally transmit false interference alerts, manipulated spectrum measurements, or fabricated environmental observations in an attempt to mislead the learning agents and degrade network performance. To improve resilience against such attacks, the proposed framework can be extended with a lightweight trust management mechanism. Each participating agent maintains a dynamic trust score that reflects the historical reliability and consistency of its sensing reports. Trust values are continuously

updated by comparing locally received observations with neighboring reports and global interference estimates generated by the Cognitive Radio engine. The trust score of agent i is defined as

$$T_i(t+1) = \alpha T_i(t) + (1 - \alpha) C_i(t), \quad (119)$$

where $T_i(t)$ denotes the current trust value, $C_i(t)$ represents the consistency between the reported observation and the consensus of neighboring agents, and α is the trust update factor satisfying $0 < \alpha < 1$. During cooperative spectrum sensing, reports received from agents with lower trust values are assigned smaller weights during information fusion,

$$S = \sum_{i=1}^N T_i O_i, \quad (120)$$

where O_i denotes the observation reported by agent i , and S represents the aggregated interference estimate used by the Cognitive Radio engine. This weighted information fusion mechanism limits the influence of malicious or compromised nodes on interference detection and subsequent MARL decision making. Consequently, false interference alerts have significantly reduced impact on channel selection, routing decisions, and cooperative anti-jamming policies.

Although the present work focuses primarily on intelligent anti-jamming through Cognitive Radio and Multi-Agent Reinforcement Learning, integrating Byzantine-resilient federated learning, blockchain-assisted trust management, and secure distributed consensus represents an important direction for future research to further strengthen the security and robustness of heterogeneous FANET-IoT-IoV systems.

Table 16 summarizes the proposed trust-aware extension for secure cooperative learning. By continuously evaluating the reliability of participating agents and assigning trust-dependent weights to their sensing reports, the framework mitigates the impact of false interference alerts generated by compromised nodes. This lightweight mechanism enhances the robustness of cooperative spectrum sensing and MARL-based decision making without introducing significant computational or communication overhead, making it suitable for heterogeneous FANET-IoT-IoV deployments.

Table 16. Security mechanisms for mitigating malicious sensing reports.

Security Component	Function
Trust Score	Evaluate reporting reliability
Consistency Check	Compare neighboring observations
Weighted Fusion	Reduce influence of suspicious nodes
MARL State Update	Ignore low-trust observations
Future Extension	Byzantine FL and Blockchain

7. Simulation and Performance Evaluation

7.1. Simulation Setup and Parameters

To evaluate the effectiveness of the proposed CR-MARL framework, extensive simulations were conducted using a realistic heterogeneous networking environment that integrates FANET, IoT, and IoV communication systems. The simulation environment was developed using OMNeT++ with the INET and VEINS frameworks to model vehicular communications, while the Artery framework was employed to support V2X communication. A custom UAV mobility model was integrated to emulate the FANET layer and provide aerial communication, sensing, and relay capabilities. The simulation scenario consists of a 2 km urban highway segment populated with connected vehicles, roadside IoT sensors, and UAVs operating collaboratively to detect and mitigate jamming attacks.

Both stationary and mobile jamming scenarios were considered to evaluate the robustness of the proposed framework under different attack conditions. Network performance was assessed using standard communication and machine learning metrics, including reliability, latency, throughput, energy consumption, and jamming detection performance.

The key simulation parameters are summarized as follows:

- **Simulation Platform:** OMNeT++ integrated with the INET and VEINS frameworks for vehicular networking, Artery for V2X communication, and a custom UAV mobility model for the FANET layer.
- **Network Topology:** A 2 km urban highway consisting of 50 connected vehicles, 200 roadside IoT sensors, and 5 UAVs providing aerial communication coverage, sensing, and relay services.
- **Jamming Scenarios:** Two representative jamming models were implemented: (i) a stationary reactive jammer deployed near a busy road intersection, and (ii) a mobile smart jammer mounted on a moving vehicle capable of dynamically adapting its attack strategy.
- **Performance Metrics:** The proposed framework was evaluated using Packet Delivery Ratio (PDR), End-to-End Delay, Network Throughput, Energy Consumption per node, and jamming detection metrics including Precision, Recall, F1-Score, and Detection Accuracy.

7.2. Compared Techniques

The performance of the proposed CR-MARL framework was compared against representative baseline approaches commonly employed for jamming detection and mitigation in heterogeneous wireless networks. These baselines were selected to demonstrate the advantages of combining cognitive radio with multi-agent reinforcement learning over conventional machine learning and rule-based spectrum management techniques. All competing methods were evaluated under identical network configurations, mobility patterns, traffic loads, and jamming scenarios to ensure a fair and consistent comparison.

The compared techniques are summarized below:

- **Baseline 1 (Deep Learning—DL):** A centralized deep learning-based jamming detector that performs attack classification using neural network models without cognitive radio or reinforcement learning-based mitigation.
- **Baseline 2 (Cognitive Radio with Machine Learning—CR-ML):** A hybrid Cognitive Radio (CR) framework integrated with a conventional Machine Learning (ML) jamming detector. The ML model identifies potential jamming attacks based on observed channel features, while the cognitive radio performs rule-based spectrum management by selecting alternative communication channels according to predefined channel quality metrics (e.g., RSS or SINR), without reinforcement learning or cooperative multi-agent decision-making.
- **Baseline 3 (Machine Learning—ML):** A standalone Random Forest-based jamming detector combined with a simple rule-based mitigation strategy that switches communication channels whenever a jamming attack is detected.
- **Proposed Method (CR-MARL):** The proposed hybrid Cognitive Radio and Multi-Agent Reinforcement Learning framework, which jointly performs collaborative jamming detection, intelligent spectrum selection, adaptive mitigation, and coordinated cross-tier decision-making using a DRL-based Decision Engine.

7.3. Results and Discussion

To evaluate the effectiveness of the proposed CR-MARL framework, simulation-based results are generated and compared against baseline learning-based approaches,

namely ML, DL, and CR-ML, under dynamic jamming conditions. The performance is assessed using key metrics relevant to integrated FANET-IoT-IoV systems, including Packet Delivery Ratio (PDR), end-to-end latency, and energy consumption.

In Figure 7, the PDR comparison under jamming conditions demonstrates a clear and consistent advantage of the proposed CR-MARL framework. Conventional ML-based approaches exhibit the lowest PDR due to their limited adaptability and lack of coordinated spectrum awareness. DL-based methods show moderate improvement; however, they remain reactive and are unable to fully anticipate jammer behavior. The CR-ML approach benefits from cognitive radio sensing and dynamic spectrum access, achieving higher PDR by avoiding persistently jammed channels. Nevertheless, its decision-making remains locally optimized. In contrast, the CR-MARL framework achieves the highest and most stable PDR across time. This improvement is attributed to cooperative multi-agent learning, where UAVs, vehicles, and infrastructure nodes jointly adapt channel selection, power control, and routing strategies based on the shared jamming map. The results indicate that CR-MARL significantly enhances communication reliability in contested environments.

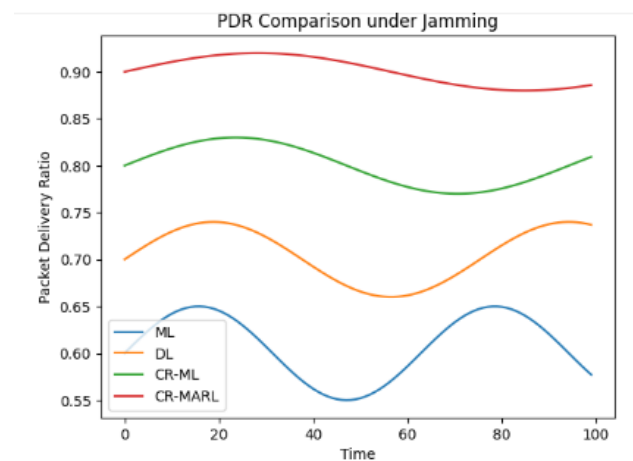


Figure 7. PDR Comparison under Jamming.

In Figure 8, Latency performance further highlights the benefits of coordinated intelligence. ML and DL methods experience higher and more fluctuating delays due to frequent retransmissions and suboptimal mitigation actions under jamming. The CR-ML approach reduces latency by opportunistically switching channels; however, the absence of global coordination can still lead to congestion and inefficient rerouting. The CR-MARL framework consistently achieves the lowest latency, as the decision engine proactively orchestrates cross-layer actions such as modulation adaptation, UAV repositioning, and prioritized scheduling of safety-critical traffic. The reduced latency is particularly important for delay-sensitive applications in IoV and safety-critical FANET missions.

Energy efficiency is a critical constraint in UAVs and IoT nodes. The results in Figure 9 show that ML and DL approaches consume the most energy due to repeated failed transmissions and reactive adjustments. CR-ML improves energy efficiency by avoiding heavily jammed channels, but still incurs overhead due to frequent spectrum sensing and isolated decision-making. The CR-MARL framework demonstrates the lowest energy consumption among all approaches. This gain is achieved through coordinated action selection, minimizing unnecessary retransmissions, and balancing power control decisions across agents. By jointly optimizing communication and mobility strategies, CR-MARL extends network lifetime while maintaining robust performance.

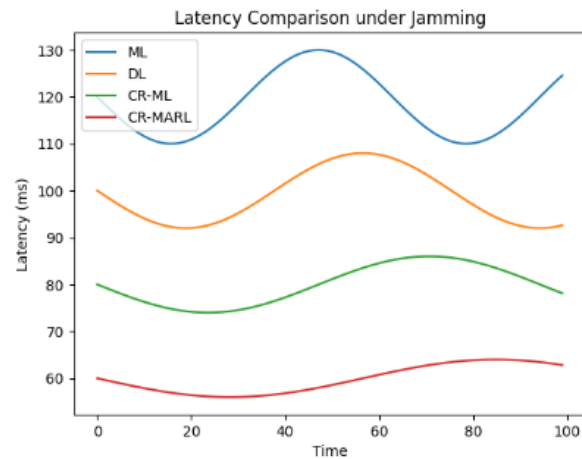


Figure 8. Latency Comparison under Jamming.

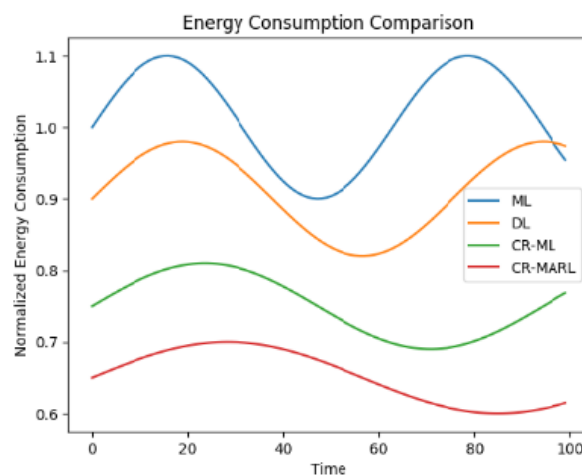


Figure 9. Energy Consumption Comparison under Jamming.

Overall, the simulation results confirm that integrating Cognitive Radio with Multi-Agent Reinforcement Learning provides substantial gains over standalone ML, DL, and CR-ML approaches. The proposed framework effectively transforms jamming mitigation from a reactive, node-centric problem into a proactive, system-level optimization process, making it well suited for large-scale, heterogeneous FANET-IoT-IoV deployments.

Figure 10 shows bar chart for PDR Comparison under Jamming. This figure compares the packet delivery ratio achieved by conventional ML, DL, CR-ML, and the proposed CR-MARL framework under jamming conditions. Traditional ML-based detection suffers from limited adaptability, resulting in lower PDR due to delayed or suboptimal mitigation decisions. DL improves resilience by modeling non-linear interference patterns; however, it remains largely reactive and node-centric. The CR-ML approach benefits from dynamic spectrum access, enabling nodes to avoid heavily jammed channels, which leads to a noticeable PDR improvement. The proposed CR-MARL framework achieves the highest PDR by jointly optimizing spectrum selection, power control, and cooperative mitigation actions across FANET, IoT, and IoV tiers. The result demonstrates that cross-layer intelligence combined with multi-agent coordination significantly enhances communication reliability under adversarial interference.

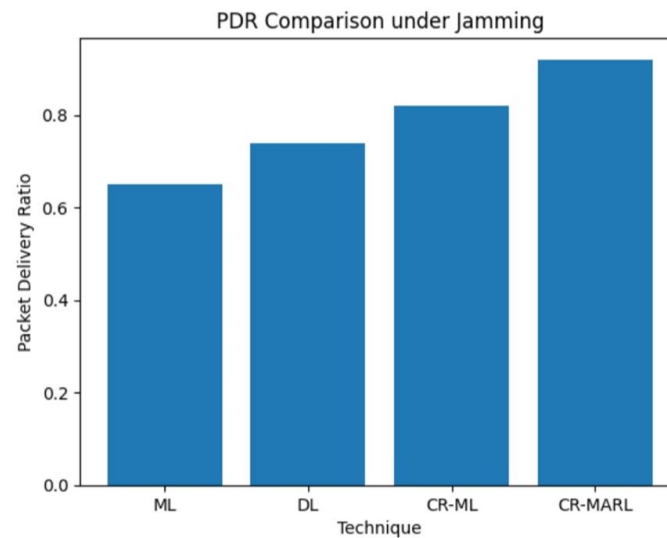


Figure 10. Bar Chart PDR Comparison under Jamming.

Figure 11 shows bar chart for Latency Comparison under Jamming. This figure illustrates the average end-to-end latency experienced by different anti-jamming strategies. ML-based approaches exhibit high latency due to frequent retransmissions and lack of predictive adaptation. DL reduces latency by improving detection accuracy but still relies on local decisions. CR-ML further lowers latency by proactively switching to less congested spectrum bands. The proposed CR-MARL framework achieves the lowest latency by leveraging global situational awareness from the jamming map and coordinated actions across agents, allowing time-critical IoV safety messages and FANET control links to be preserved even under sustained jamming.

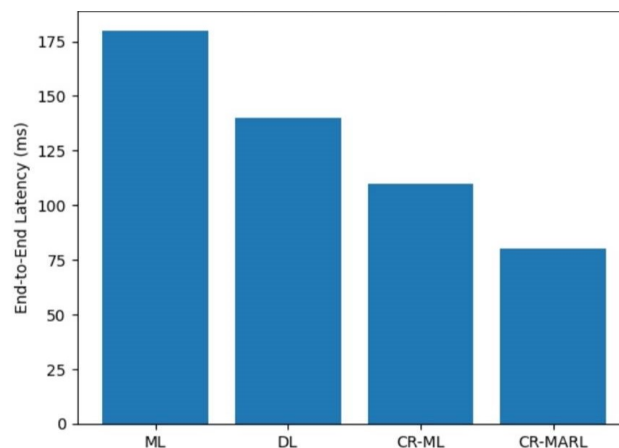


Figure 11. Bar Chart Latency Comparison under Jamming Techniques.

Figure 12 shows Energy Consumption Comparison. Energy efficiency is critical for battery-constrained IoT sensors and UAVs. This figure shows normalized energy consumption for each technique. ML and DL approaches consume higher energy due to repeated sensing, retransmissions, and inefficient mitigation. CR-ML reduces energy usage by avoiding persistently jammed bands. The CR-MARL framework further minimizes energy consumption by learning long-term optimal policies that balance transmission power, mobility, and spectrum usage, demonstrating its suitability for resource-constrained multi-tier networks.

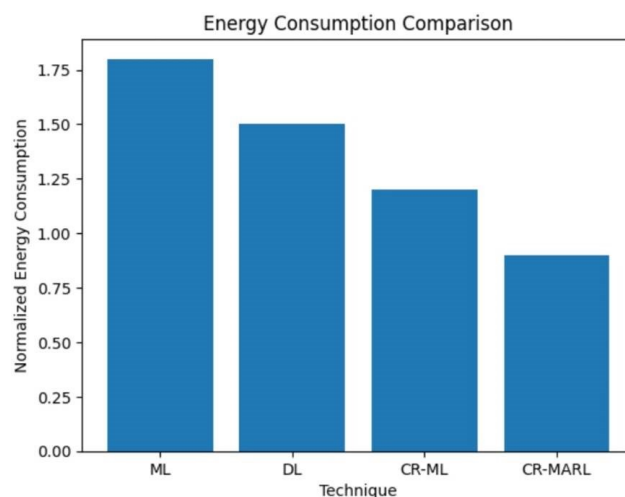


Figure 12. Bar Chart Energy Consumption Comparison under Jamming.

Figure 13 depicts Detection Performance Comparison focusing on Precision, Recall, and F1-Score.

This figure compares detection accuracy metrics across different learning paradigms. ML-based detection shows limited recall, failing to detect adaptive and reactive jammers consistently. DL improves both precision and recall by capturing complex signal patterns. CR-ML benefits from spectrum-aware features, resulting in further gains. The CR-MARL framework achieves the highest precision, recall, and F1-score due to cooperative detection, where agents share observations and jointly learn jammer behavior. This highlights the effectiveness of MARL in handling intelligent and adaptive adversaries.

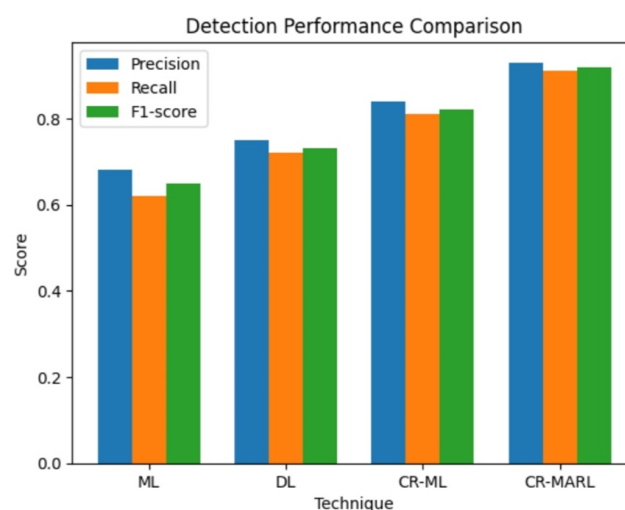


Figure 13. Detection Performance Comparison under Jamming.

Figure 14 visualizes Confusion Matrix analysis for CR-MARL Detection. The confusion matrix provides deeper insight into the detection capability of the proposed framework. A high number of true positives and true negatives indicates reliable identification of both jammed and non-jammed states. The low false alarm and missed detection rates confirm that CR-MARL maintains robustness against environmental dynamics while avoiding unnecessary mitigation actions. This balance is essential in FANET-IoT-IoV systems, where false alarms can trigger costly mobility or spectrum reconfiguration.

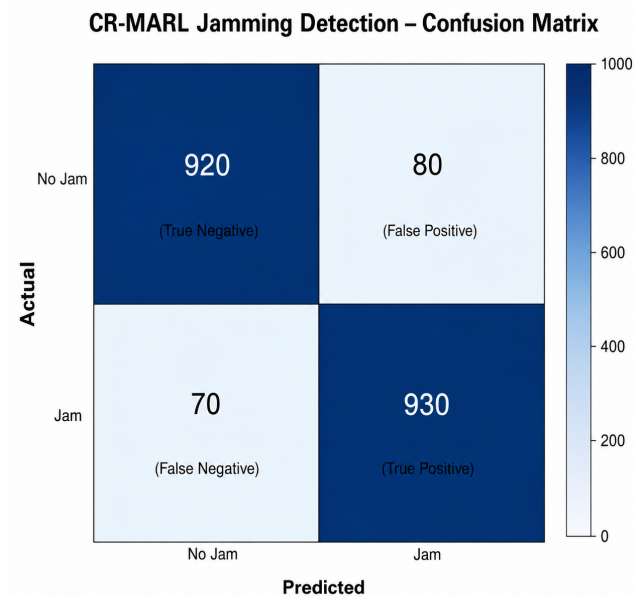


Figure 14. Confusion Matrix for CR-MARL Technique.

8. Open Challenges and Future Directions

Although significant progress has been made in developing intelligent anti-jamming techniques for integrated FANET-IoT-IoV networks, several technical challenges remain before such frameworks can be deployed in real-world large-scale systems. Future research should focus on improving the intelligence, scalability, security, explainability, and practical deployment of Cognitive Radio (CR) and Multi-Agent Reinforcement Learning (MARL)-based anti-jamming frameworks. The most promising research directions are discussed below.

8.1. AI-Powered Adversarial Jamming

Future jammers are expected to employ advanced Artificial Intelligence (AI) techniques, including deep reinforcement learning, generative adversarial networks (GANs), and large language model (LLM)-assisted decision-making, to continuously learn optimal attack strategies. Unlike conventional reactive or random jammers, AI-powered adversaries can dynamically predict spectrum occupancy, communication patterns, UAV trajectories, and routing decisions to launch highly targeted, low-probability-of-detection attacks. Developing MARL frameworks capable of anticipating and countering intelligent adversaries through adversarial reinforcement learning remains an important research challenge.

8.2. Robust MARL Under Partial Observability

In practical FANET-IoT-IoV deployments, individual agents rarely possess complete knowledge of the global network state. Communication delays, intermittent connectivity, and rapidly changing topologies introduce significant uncertainty into the learning process. Future research should investigate Partially Observable MARL (POMARL), belief-state learning, graph neural networks, and transformer-based cooperative learning to improve decision-making under incomplete and noisy observations.

8.3. Energy-Efficient Learning for Resource-Constrained IoT

Although MARL provides excellent adaptability, its computational and communication requirements remain challenging for battery-powered IoT devices. Lightweight neural architectures, model compression, knowledge distillation, edge intelligence, and adaptive inference mechanisms should be investigated to reduce energy consumption

while preserving anti-jamming performance. Energy-aware reward functions that jointly optimize communication reliability and battery lifetime also represent an important research direction.

8.4. Federated and Distributed Learning

Centralized model training may become infeasible due to privacy concerns, communication overhead, and limited bandwidth in heterogeneous networks. Federated Learning (FL) offers a promising solution by enabling distributed model training without sharing raw sensing data. Future research should investigate hybrid CR–MARL–FL architectures that combine collaborative reinforcement learning with privacy-preserving distributed intelligence while minimizing communication overhead.

8.5. Cross-Layer Semantic Jamming

Emerging jamming attacks increasingly target protocol semantics rather than only physical-layer transmissions. Examples include selectively jamming TCP acknowledgments, routing control packets, synchronization beacons, or TDMA scheduling messages to disrupt network operation while transmitting minimal interference energy. Future anti-jamming frameworks should jointly exploit information from the physical, MAC, network, and application layers to detect these sophisticated cross-layer attacks.

8.6. Explainable and Trustworthy AI

Current deep reinforcement learning models often operate as black boxes, making it difficult to understand why a particular anti-jamming action was selected. Explainable Artificial Intelligence (XAI) techniques can improve transparency by identifying the network observations and environmental factors influencing each decision. Developing interpretable MARL policies will be particularly important for mission-critical applications such as military UAV operations, intelligent transportation systems, and emergency response networks.

8.7. Blockchain-Enabled Trust Management

Collaborative anti-jamming decisions rely on trustworthy sensing information exchanged among heterogeneous network nodes. Future research should investigate lightweight blockchain architectures, reputation-aware consensus mechanisms, and smart-contract-based spectrum management to ensure secure and tamper-resistant coordination while minimizing latency and computational overhead.

8.8. Quantum-Secure Coordination

Future military and critical infrastructure applications may require communication systems that remain secure even against quantum-enabled adversaries. Quantum Key Distribution (QKD) and post-quantum cryptographic algorithms offer promising approaches for establishing secure control channels between UAVs, ground stations, and command centers. Integrating quantum-secure communication with intelligent anti-jamming frameworks remains an open research problem.

8.9. Digital Twins and Real-Time Network Simulation

Digital Twin technology can provide virtual replicas of integrated FANET–IoT–IoV networks for continuously evaluating jamming threats, testing anti-jamming strategies, and optimizing network configurations before deployment. Combining Digital Twins with MARL can significantly accelerate policy training while reducing the risks associated with real-world experimentation.

8.10. Standardization and Real-World Deployment

Despite considerable advances in intelligent anti-jamming techniques, the absence of standardized cross-tier communication protocols remains a significant obstacle to practical deployment. Future work should focus on developing interoperable standards for collaborative jamming detection, spectrum sharing, cognitive radio coordination, and multi-agent information exchange across UAVs, vehicles, roadside infrastructure, and IoT devices. Furthermore, extensive field experiments and large-scale testbeds are needed to validate CR-MARL frameworks under realistic mobility, interference, and environmental conditions before adoption in next-generation 6G intelligent transportation and aerial communication systems.

9. Conclusions

This paper has presented a comprehensive analysis of the jamming threat landscape in integrated FANET-IoT-IoV systems, which are foundational to future smart cities and intelligent transportation. We systematically reviewed and compared a wide array of jamming detection and mitigation techniques, highlighting their respective strengths, weaknesses, and applicability across the aerial, vehicular, and terrestrial tiers.

Recognizing that no single technique is sufficient for this complex, heterogeneous environment, we proposed a novel hybrid anti-jamming framework that synergistically combines the environmental awareness of Cognitive Radio with the adaptive intelligence of Machine Learning, specifically Deep Reinforcement Learning. The framework employs collaborative sensing to build a jamming map and uses a DRL agent to make optimal, context-aware mitigation decisions.

Simulation results demonstrated that the proposed framework significantly outperforms conventional and standalone intelligent methods in maintaining high Packet Delivery Ratio and low latency under various jamming attacks, while judiciously managing system overhead. While challenges such as AI-powered adversarial attacks and standardization remain, this work provides a concrete architectural blueprint and a step towards building robust, secure, and resilient integrated cyber-physical systems for the future.

Author Contributions: Conceptualization, R.R. and Z.-u.-R.; methodology, R.R., M.N. and F.S.; software, Z.-u.-R., F.A. and F.S.; validation, M.N., F.A. and A.C.; formal analysis, R.R. and F.A.; investigation, M.N. and A.C.; resources, M.N., F.S. and A.C.; data curation, R.R., Z.-u.-R. and F.A.; writing original draft preparation, R.R., Z.-u.-R. and M.N.; writing—review and editing, F.A., F.S. and A.C.; visualization, M.N. and A.C.; supervision, A.C.; project administration, R.R., Z.-u.-R. and F.S.; funding acquisition, M.N. and A.C. All authors have read and agreed to the published version of the manuscript.

Funding: This project has been partially funded by the “Programma Nazionale Ricerca, Innovazione e Competitività per la transizione verde e digitale 2021/2027 destinate all’intervento del FCS “Scoperta imprenditoriale”-Azione 1.1.4 “Ricerca collaborativa”-with the project SIAMO (Servizi Innovativi per l’Assistenza Medica a bOrdo) project number F/360124/01-02/X75.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Irfan, M.; Omri, A.; Hernandez Fernandez, J.; Sciancalepore, S.; Oligeri, G. Detecting Jamming in Smart Grid Communications via Deep Learning. *J. Cybersecur. Priv.* **2025**, *5*, 46. [\[CrossRef\]](#)
2. Li, Z.; Wang, J.; Chen, J.; Fang, Z.; Ren, Y. Flying Ad-Hoc Network Covert Communications with Deep Reinforcement Learning. *IEEE Wirel. Commun.* **2024**, *31*, 117–125. [\[CrossRef\]](#)

3. Raza, R.; Aadil, F. A comprehensive survey on FANET-IoT-IoV interactions: Challenges, opportunities, and future directions in 6G-enabled smart cities. *Computing* **2026**, *108*, 10.
4. Jeon, K.; Lee, Y.S.; Jung, B.C.; Lee, H. Cooperative Jamming for Secure Air-Ground Integrated Networks: A Hierarchical Distributed Deep Reinforcement Learning Approach. *IEEE Internet Things J.* **2025**, *12*, 52607–52620. [[CrossRef](#)]
5. Meftah, A.; Do, T.N.; Kaddoum, G.; Talhi, C. Federated Learning-Enabled Jamming Detection for Stochastic Terrestrial and Non-Terrestrial Networks. *IEEE Trans. Green Commun. Netw.* **2024**, *9*, 271–290.
6. Mowla, N.I.; Tran, N.H.; Doh, I.; Chae, K. Federated Learning-Based Cognitive Detection of Jamming Attack in Flying Ad-Hoc Network. *IEEE Access* **2019**, *8*, 4338–4350.
7. El-Shafai, W.; Azar, A.T.; Ahmed, S. AI-Driven Ensemble Classifier for Jamming Attack Detection in VANETs to Enhance Security in Smart Cities. *IEEE Access* **2025**, *13*, 50687–50713. [[CrossRef](#)]
8. Chandran, I.; Vipin, K. Multi-UAV networks for disaster monitoring: Challenges and opportunities from a network perspective. *Drone Syst. Appl.* **2024**, *12*, 1–28. [[CrossRef](#)]
9. Xu, C.; Zhang, K.; Jiang, Y.; Niu, S.; Yang, T.; Song, H. Communication aware UAV swarm surveillance based on hierarchical architecture. *Drones* **2021**, *5*, 33. [[CrossRef](#)]
10. Zeeshan, M.; Farooq, M.U.; Shahzad, K.; Akhunzada, A. 5G/SDR-Assisted Cognitive Communication in UAV Swarms: Architecture and Applications. *IT Prof.* **2022**, *24*, 28–34. [[CrossRef](#)]
11. Javed, S.; Hassan, A.; Ahmad, R.; Ahmed, W.; Ahmed, R.; Saadat, A.; Guizani, M. State-of-the-art and future research challenges in UAV swarms. *IEEE Internet Things J.* **2024**, *11*, 19023–19045. [[CrossRef](#)]
12. Khan, A.; Gupta, S.; Gupta, S.K. Emerging UAV technology for disaster detection, mitigation, response, and preparedness. *J. Field Robot.* **2022**, *39*, 905–955. [[CrossRef](#)]
13. Chhikara, P.; Tekchandani, R.; Kumar, N. Adaptive federated learning scheme for recognition of malicious attacks in an IoT network. *Computing* **2024**, *106*, 1177–1192.
14. Ayegun, O.M.; Akingbade, K.F.; Popoola, J.J.; Ubochi, B.C. Leveraging on Artificial Intelligence for Universal Jamming Attack Detection in Unmanned Aerial Vehicle Communication. *Comput. Electr. Eng.* **2025**, *127*, 110602. [[CrossRef](#)]
15. Xia, Z.; Lin, Y. Signal Bandwidth Estimation for Jamming Detection: Skip Connection and LARS-Augmented Mask R-CNN. *IEEE Access* **2025**, *13*, 167726–167743. [[CrossRef](#)]
16. Chang, X.; Li, Y.; Zhao, Y.; Du, Y.; Liu, D. An Improved Anti-Jamming Method Based on Deep Reinforcement Learning and Feature Engineering. *IEEE Access* **2022**, *10*, 69992–70000. [[CrossRef](#)]
17. Liu, C.; Ren, B.; Xie, Y.; Chen, F. Deep Learning-Based GNSS Composite Jamming Detection and Recognition Technology. *Front. Signal Process.* **2025**, *5*, 1567926. [[CrossRef](#)]
18. Sultan, L.; Anjum, M.; Rehman, M.; Murawwat, S.; Kosar, H. Communication among heterogeneous unmanned aerial vehicles (UAVs): Classification, trends, and analysis. *IEEE Access* **2021**, *9*, 118815–118836. [[CrossRef](#)]
19. Hemmati, A.; Zarei, M.; Souri, A. UAV-based internet of vehicles: A systematic literature review. *Intell. Syst. Appl.* **2023**, *18*, 200226. [[CrossRef](#)]
20. Zheng, T.; Wu, J.; Li, G. IoV data sharing scheme based on the hybrid architecture of blockchain and cloud-edge computing. *J. Cloud Comput.* **2023**, *12*, 99. [[CrossRef](#)]
21. Husnain, G.; Anwar, S.; Shahzad, F. An enhanced AI-enabled routing optimization algorithm for internet of vehicles (IoV). *Wirel. Pers. Commun.* **2023**, *130*, 2623–2643. [[CrossRef](#)]
22. Al-Quayed, F.; Tariq, N.; Humayun, M.; Aslam Khan, F.; Attique Khan, M.; Alnusairi, T.S. Securing the Road Ahead: A Survey on Internet of Vehicles Security Powered by a Conceptual Blockchain-Based Intrusion Detection System for Smart Cities. *Trans. Emerg. Telecommun. Technol.* **2025**, *36*, e70133. [[CrossRef](#)]
23. Rajashekara, K.; Koppera, S. Data and energy impacts of intelligent transportation—A review. *World Electr. Veh. J.* **2024**, *15*, 262. [[CrossRef](#)]
24. Saini, K.; Sharma, S. Smart Road Traffic Monitoring: Unveiling the Synergy of IoT and AI for Enhanced Urban Mobility. *ACM Comput. Surv.* **2025**, *57*, 1–45. [[CrossRef](#)]
25. Mishra, P.; Singh, G. Internet of Vehicles for Sustainable Smart Cities: Opportunities, Issues, and Challenges. *Smart Cities* **2025**, *8*, 93. [[CrossRef](#)]
26. Zaman, M.; Puryear, N.; Abdelwahed, S.; Zohrabi, N. A review of IoT-based smart city development and management. *Smart Cities* **2024**, *7*, 1462–1501. [[CrossRef](#)]
27. Cheng, N.; Wu, S.; Wang, X.; Yin, Z.; Li, C.; Chen, W.; Chen, F. AI for UAV-assisted IoT applications: A comprehensive review. *IEEE Internet Things J.* **2023**, *10*, 14438–14461. [[CrossRef](#)]
28. Marchese, M.; Moheddine, A.; Patrone, F. IoT and UAV integration in 5G hybrid terrestrial-satellite networks. *Sensors* **2019**, *19*, 3704. [[CrossRef](#)] [[PubMed](#)]
29. Larian, H.; Safi-Esfahani, F. InTec: Integrated things-edge computing: A framework for distributing machine learning pipelines in edge AI systems. *Computing* **2025**, *107*, 41.

30. Yazid, Y.; Ez-Zazi, I.; Guerrero-Gonzalez, A.; El Oualkadi, A.; Arioua, M. UAV-enabled mobile edge-computing for IoT based on AI: A comprehensive review. *Drones* **2021**, *5*, 148. [\[CrossRef\]](#)
31. Li, X.; Chen, J.; Ling, X.; Wu, T. Deep Reinforcement Learning-Based Anti-Jamming Algorithm Using Dual Action Network. *IEEE Trans. Wirel. Commun.* **2022**, *22*, 4625–4637. [\[CrossRef\]](#)
32. Garg, S.; Ihler, A.; Bentley, E.S.; Kumar, S. A Cross-Layer, Mobility, and Congestion-Aware Routing Protocol for UAV Networks. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *59*, 3778–3796.
33. Xu, D.; Zhu, H. Proactive eavesdropping via jamming over short packet suspicious communications with finite blocklength. *IEEE Trans. Commun.* **2022**, *70*, 7505–7519. [\[CrossRef\]](#)
34. Abdollahi, M.; Malekinasab, K.; Tu, W.; Bag-Mohammadi, M. Physical-Layer Jammer Detection in Multihop IoT Networks. *IEEE Internet Things J.* **2023**, *10*, 20574–20585. [\[CrossRef\]](#)
35. Shi, Y.; Lu, X.; An, K.; Li, Y.; Zheng, G. Efficient index-modulation-based FHSS: A unified anti-jamming perspective. *IEEE Internet Things J.* **2023**, *11*, 3458–3472.
36. Bouzabia, H.; Do, T.N.; Kaddoum, G. Deep learning-enabled deceptive jammer detection for low probability of intercept communications. *IEEE Syst. J.* **2022**, *17*, 2166–2177. [\[CrossRef\]](#)
37. AL-Syouf, R.A.; Bani-Hani, R.M.; AL-Jarrah, O.Y. Machine Learning Approaches to Intrusion Detection in Unmanned Aerial Vehicles (UAVs). *Neural Comput. Appl.* **2024**, *36*, 18009–18041. [\[CrossRef\]](#)
38. Zheng, R.; Sumper, A.; Aragüés-Peñalba, M.; Galceran-Arellano, S. Advancing power system services with privacy-preserving federated learning techniques: A review. *IEEE Access* **2024**, *12*, 76753–76780. [\[CrossRef\]](#)
39. Jiang, Y.; Ma, B.; Wang, X.; Yu, G.; Yu, P.; Wang, Z.; Ni, W.; Liu, R.P. Blockchain federated learning for internet of things: A comprehensive survey. *ACM Comput. Surv.* **2024**, *56*, 1–37. [\[CrossRef\]](#)
40. Zhou, L.; Leng, S.; Wang, Q.; Quek, T.Q.; Guizani, M. Cooperative digital twins for uav-based scenarios. *IEEE Commun. Mag.* **2024**, *63*, 40–46. [\[CrossRef\]](#)
41. Gao, Y.; Li, S.; Xu, T.; Lakshminarayana, S.; Bu, S.; Gu, C.; Ai, Q. Distributed Model Predictive Control Strategy for Multi-energy Virtual Power Plant Based on Digital Twin. *IEEE Trans. Smart Grid* **2025**, *17*, 1971–1984. [\[CrossRef\]](#)
42. Razafimanjato, M.; Saad, M.M.; Kim, D. Blockchain-based trust management systems in the Internet of Vehicles: A comprehensive survey. *ICT Express* **2025**, *11*, 1265–1285. [\[CrossRef\]](#)
43. Yodianto, W.; Warnars, H.L.H.S.; Warnars, L.L.H.S.; Ramadhan, A.; Siswanto, T. Searching routing using a-star (a*) search algorithm. In *Proceedings of the 2024 3rd International Conference on Creative Communication and Innovative Technology (ICCIT)*, Tangerang, Indonesia, 7–8 August 2024; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.