END OF STUDIES INTERNSHIP REPORT

Electrical, Electronics and Telecommunications Engineering

Secrecy Rate Maximization in RIS-Aided UAV ISAC Networks

AI-Driven Defense Against Jamming and Eavesdropping

Author:
Mahamadou DANSOKO
Academic Tutor:
Pr. Jamal EL AOUFI

Supervisors:
Pr. El Mehdi AMHOUD
Pr. Zouheir REZKI

Academic Year 2024/2025

August 27, 2025

Dedication

I dedicate this work, first, to my moms. Your belief in me, your endless encouragement, and your profound love have been my constant anchors. I wouldn't be here without you.

To my dear father, who is no longer with us: you sacrificed so much and gave us your absolute all. There isn't a day I don't wish you were here to share this moment, so I could tell you face-to-face how incredibly proud I am to carry your name and be your son.

To my elder brothers, my lifelong allies, your unwavering support and willingness to always lend a hand have been invaluable. I want you to know that my love and gratitude for you and for everything you do, are immense.

To all my beloved family, near and far, I hold an infinite love for each of you. You are my roots and my strength.

And to my friends, the family I have chosen, your daily inspiration, your belief in me, and your companionship help me grow and become a better person. Thank you for being in my life.

Acknowledgments

I want to start by thanking my school, the Mohammed VI Academy of Civil Aviation, for everything it has given me over the years. The knowledge, the discipline, and the mindset I carry today are the result of the education I received there. I'm especially thankful to Madame Khadija SOLDI, our Department Head, whose leadership has always been both strong and kind. I also want to thank my academic tutor, Mr. Jamal EL AOUFI, whose support and guidance during this internship made a real difference. And to all the teachers who've been part of my journey from the beginning, thank you. Each of you helped shape the person and student I am today.

I'm also incredibly grateful to the University Mohammed VI Polytechnic (UM6P), my host institution, for the warm welcome and the stimulating research environment it provided. Working there allowed me to grow in ways I didn't expect, both academically and personally

My sincere thanks go to Professor El Mehdi AMHOUD from UM6P and Professor Zouheir REZKI from Santa Cruz University, California, my supervisors for their time, insightful feedback, and continuous support throughout this project.

A heartfelt thank you to Abdoul Karim Saliah, PhD student at the College of Computing (UM6P), whose advice, availability, and encouragement have been of great help to me throughout my research.

To each of you, thank you for contributing to this experience and helping me grow as a student and as a person.

List of Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
B5G	Beyond 5G
CNN	Convolutional Neural Network
CSI	Channel State Information
CU	Communication User
DDPG	Deep Deterministic Policy Gradient
DFRC	Dual-Functional Radar-Communication
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
EKF	Extended Kalman Filter
Eve	Eavesdropper
eMBB	Enhanced Mobile Broadband
FL	Federated Learning
HAPS	High-Altitude Platform Station
IRIS	Intelligent Reflecting Surface
ISAC	Integrated Sensing and Communication
ISAC-UAV	ISAC-Enabled UAV
ISAGC	Integrated Space-Air-Ground Communication
J-UAV	Jamming UAV
LoS	Line-of-Sight
MEC	Multi-access Edge Computing
MIMO	Multiple-Input Multiple-Output
ML	Machine Learning
mMTC	Massive Machine-Type Communications

List of Abbreviations

Abbreviation	Meaning
NLoS	Non-Line-of-Sight
NR	New Radio (5G standard)
NSA	Non-Standalone (5G architecture)
QoE	Quality of Experience
QoS	Quality of Service
RIS	Reconfigurable Intelligent Surface
RL	Reinforcement Learning
SA	Standalone (5G architecture)
SCA	Successive Convex Approximation
SDR	Semidefinite Relaxation
SINR	Signal-to-Interference-plus-Noise Ratio
SNR	Signal-to-Noise Ratio
THz	Terahertz
UAV	Unmanned Aerial Vehicle
URLLC	Ultra-Reliable Low Latency Communication

Nomenclature

Symbol	Definition
$\mathbf{q}_C[n]$	3D position of the ISAC-UAV at time slot n
$\mathbf{q}_J[n]$	3D position of the Jamming UAV at time slot n
$\mathbf{q}_E[n]$	3D position of the Eavesdropper UAV at time slot n
$\mathbf{w}_k$	Position of the k-th communication user (CU)
$\mathbf{w}_S$	Position of the sensing target
$\mathbf{w}_R$	Position of the RIS
$\mathbf{h}_{Ck}[n]$	Direct channel from ISAC-UAV to CU k
$\mathbf{h}_{CR}[n]$	Channel from ISAC-UAV to RIS
$\mathbf{g}_{Rk}[n]$	Channel from RIS to CU k
$\mathbf{\Phi}[n]$	RIS phase shift matrix at time slot n
$\gamma_k[n]$	SINR at CU k
$\gamma_E[n]$	SINR at the eavesdropper
$R_{\text{sec},k}[n]$	Secrecy rate of CU k at time slot n
$SNR_S[n]$	Sensing Signal-to-Noise Ratio at time slot n
$P_C[n]$	Transmit power of ISAC-UAV at time slot n
$P_J[n]$	Transmit power of Jamming UAV at time slot n
M	Number of reflecting elements in the RIS
N	Total number of time slots
K	Number of legitimate users
α	Path loss exponent
κ	Rician factor (LoS to NLoS ratio)
τ	Duration of a time slot
$\mathbf{x}[n]$	Sensing signal transmitted by the ISAC-UAV at time slot n
$\mathbf{y}_{\text{CU},k}[n]$	Received signal at the k-th communication user at time n
$\mathbf{y}_{\text{E}}[n]$	Received signal at the eavesdropper at time n
$\mathbf{y}_{\text{S}}[n]$	Received sensing signal reflected from the sensing target
σ^2	Noise power
$\mathbb{E}[\cdot]$	Expectation operator
$\mathcal{L}$	Lagrangian function used in optimization
λ	Lagrange multiplier
$\mathcal{A}, \mathcal{S}, \mathcal{R}$	Action space, state space, and reward in MDP/DRL
μ	Policy function in reinforcement learning
$Q(s, a)$	Action-value function
$\mathcal{N}(0, \sigma^2)$	Gaussian distribution with zero mean and variance σ^2
η	Learning rate in the DRL algorithm
θ	Parameter of the neural network
τ'	Target update factor in soft updates

Abstract

The growing demand for secure and resilient **Integrated Sensing and Communication (ISAC) networks** has attracted significant attention, particularly in adverserial environments affected by **jamming and eavesdropping threats**. In this paper, we introduce a novel **Reconfigurable Intelligent Surface(RIS)-aided Unmanned Aerial Vehicle(UAV)-enabled ISAC framework**, designed to ensure the confidentiality and reliability or simultaneous **Communication and Sensing (C&S)** operations, under strict **secrecy constraints**.

Our primary objective is to **maximize the secrecy rate** by jointly optimizing critical parameters, including the **UAV's dynamic trajectory**, **RIS phase shift configuration**, **transmit power allocation**, and **user scheduling**. To address the resulting high-dimensional and complex optimization problem, we propose an **Arificial Intelligent(AI)-driven** solution based on the **Deep Deterministic Policy Gradeint (DDPG)** algorithm, a reinforcement learning technique adapted for continous control in dynamic environments.

The proposed model is capable of learning optimal policies that balance the trade-offs between **communication quality**, **secrecy enhancement**, and **robustness against jamming** in real-time. Furthermore, its ability to operate in both **online and offline** modes enhances its applicability on practical scenarios. The **simuation results demostrate the superiority of our framework** over benchmark scheles in termes of secrecy rate and overall systems resilience.

***Index Terms*—**ISAC, UAV, RIS, secrecy rate maximization, DDPG, trajectory optimization, anti-jamming.

Résumé

L'intérêt croissant pour les **réseaux de détection et de communication (ISAC) sécurisés** suscite une attention particulière, notamment face aux menaces de brouillage et d'écoute illicite. Dans ce travail, nous proposons un cadre innovant basé sur une **surface réfléchissante intelligente (RIS)** associée à un **véhicule aérien sans pilote (UAV)** doté de capacités ISAC, le tout sous de **strictes contraintes de sécurité**. L'objectif principal est de garantir des **fonctionnalités fiables de communication et de détection (C&S)**, tout en **maximisant le taux de secret**.

Pour ce faire, nous concevons une solution fondée sur un **algorithme d'apprentissage par renforcement profond (DRL)**, en l'occurrence le **Deep Deterministic Policy Gradient (DDPG)**, permettant d'optimiser simultanément plusieurs variables clés : **la trajectoire du drone, le déphasage des RIS, la répartition de puissance et l'ordonnancement des utilisateurs** dans un environnement dynamique.

Notre modèle apprend à s'adapter aux compromis entre **qualité de communication, confidentialité et résilience face au brouillage**, le tout en temps réel. De plus, il peut également être utilisé en mode **hors ligne** afin d'assurer une plus grande flexibilité. Les résultats de **simulation ont démontré une performance supérieure** en matière de **confidentialité**, ainsi qu'une forte **robustesse face aux attaques**.

Mots clés—ISAC, UAV, RIS, maximisation du taux de secret, DDPG, optimisation de trajectoire, anti-brouillage.

Table of Contents

List of Figures

List of Tables

Chapter 1

GENERAL INTRODUCTION

Despite the promise of Integrated Sensing and Communication (ISAC) in revolutionizing wireless networks, the capabilities of ***terrestrial networks remain inherently constrained*** by their static deployment. Ground-based ISAC infrastructure cannot adequately adapt to rapidly changing or high-mobility scenarios, such as those encountered in ***disaster relief operations***, ***military missions***, or ***urban traffic monitoring*** in next-generation smart cities. In these environments, stationary base stations fail to offer the flexibility, line-of-sight advantage, and dynamic coverage needed for effective communication and sensing.

This gap has catalyzed growing interest in aerial network platforms, such as Unmanned Aerial Vehicles (UAVs), which bring mobility, altitude, and adaptability to wireless communication. These platforms can be rapidly deployed to areas where terrestrial infrastructure is damaged, absent, or insufficient. Beyond emergency response and urban mobility, UAV-based ISAC systems are increasingly seen as critical enablers for agriculture monitoring, border surveillance, remote sensing, and environmental tracking, all scenarios where ground infrastructure either underperforms or is completely unfeasible.
This fundamental shift from static to mobile ISAC deployments introduces new challenges and opportunities in secure communication, resource allocation, and joint sensing-communication optimization, laying the groundwork for the innovations explored in this work.

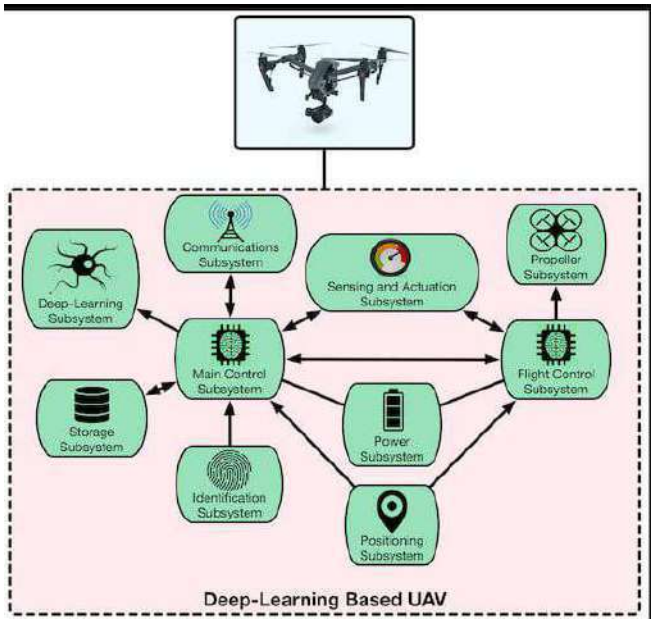

Figure 1.1: Deep Learning Based UAV Networks

1 Background on ISAC

Imagine a single piece of wireless tech that can both ”see” the world around it and ”talk” to other devices, and that, all at the same time. That’s the core idea behind Integrated Sensing and Communication (ISAC). It’s a real game-changer. Initially, engineers came up with ISAC to use our crowded airwaves more efficiently. But it turns out, it’s also great for things like better central control, super-quick data sharing, and saving energy.

Essentially, ISAC bundles sensing, communication, and even some computing power into one neat package. This means different devices in a network can work together much more smoothly. Take drone networks, for example. With ISAC, drones can spot and keep tabs on friendly drones nearby, while also sharing crucial info for flying safely, avoiding crashes, and managing all the network chatter. This ability to act like a mini radar and swap data at high speeds makes these ISAC-equipped drones incredibly useful for all sorts of jobs, like keeping an eye on the environment, conducting surveillance, helping out in disasters, or even making our transportation systems smarter.

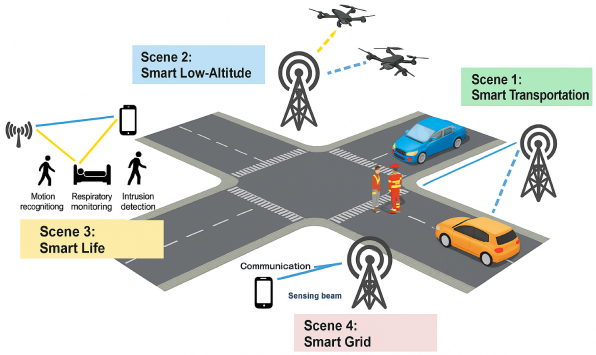

Figure 1.2: Scenarios of ISAC Systems.

However, there is a catch. Because wireless signals travel openly through the air, these smart ISAC-equipped UAV systems can be prime targets for troublemakers. We are talking about things like jamming, where someone deliberately blasts noise to disrupt signals, or eavesdropping, where they try to listen in on private communications. Since ISAC uses the same airwaves for both seeing and talking, attackers can exploit this, which means that we really need strong ways to fight back. Thankfully, researchers are already working on designs that can resist jamming and are smart about potential eavesdroppers.

To make these UAV-based ISAC systems even more robust, a cool technology called Reconfigurable Intelligent Surfaces (RIS) has entered the picture. Think of RIS as smart, programmable mirrors for radio waves, allowing us to precisely aim signals where we want them.

Additionally, to manage all the complex juggling, like balancing how well the system ”sees” versus how well it ”talks,” all while keeping things secure, Artificial Intelligence (AI) is stepping in. Clever AI methods like Reinforcement Learning (where systems learn from trial and error), Federated Learning (where learning happens across many devices without sharing raw data), and even Game Theory (which studies strategic decision-making) are being used more and more.

So, when you bring together ISAC’s dual power, the flexibility of drones (UAVs), the signal-shaping magic of RIS, and the intelligence of AI, you have got a really exciting foundation for the next wave of secure and intelligent wireless systems.

UAVs are already shaking up how we think about wireless networks. The are mobile, adaptable, and do not cost a fortune, making them perfect for everything from checking on our environment and conducting surveillance to restoring communication lines after an emergency or helping build intelligent transport systems.

To handle both the need for efficient operations and knowing what is happening around them, ISAC is perfect for UAVs, letting them sense like radar and communicate quickly. However, as we have said, the open nature of wireless signals and how drones are often out in the open makes these ISAC systems vulnerable to security threats like jamming and snooping.

This is where Reconfigurable Intelligent Surfaces (RIS) really shine as a way to create safer and smarter wireless zones. RIS can cleverly manipulate radio waves to boost the signals we want and squash the ones we do not (like interference from an attacker). If you mount an RIS on a UAV, you get fantastic, on-the-fly control over how signals travel, which is ideal for tricky or even hostile situations.

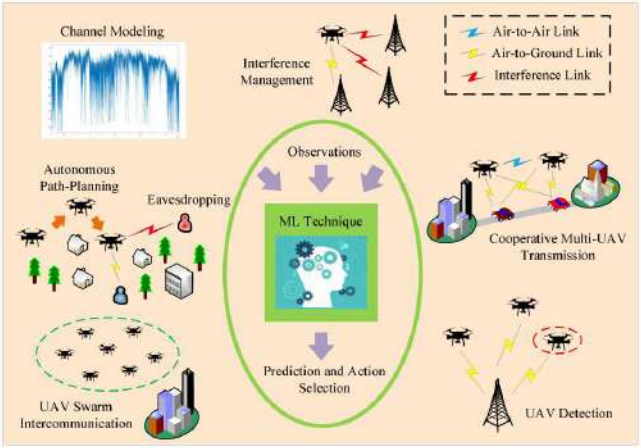

Figure 1.3: Machine Learning Techniques for UAV-based Communicaations

2 Security Challenges in UAV Networks

The open and broadcast nature of wireless communication in UAV networks makes them highly vulnerable to a variety of physical-layer attacks. Unlike terrestrial systems, UAVs often operate with dominant line-of-sight (LoS) [1] channels, which increases the risk of interception and manipulation. Common threats include *eavesdropping*, where adversaries intercept sensitive data; *jamming*, which disrupts signal transmission by injecting noise; and *spoofing*, where attackers generate fake signals to mislead or hijack UAV systems [2]. These attacks have been observed in real-world scenarios, such as GPS spoofing in military operations or drone signal jamming at airports, underlining the urgency for robust UAV security mechanisms.

These vulnerabilities are further exacerbated in dynamic and contested environments, where the UAV's high mobility and changing topology complicate secure communication and reliable resource allocation (e.g., bandwidth and power). In such scenarios, maintaining seamless and confidential data exchange is a significant challenge. The problem becomes even more critical in *multi-UAV* networks, where some UAVs may act maliciously or be compromised [3], operating as stealthy eavesdroppers without easy detection.

Moreover, integrating ISAC or ISACC into UAV networks, while efficient in terms of hardware reuse and spectrum optimization, introduces additional attack surfaces. Since the same waveform may be reused for both sensing and communication, adversaries can potentially exploit sensing signals for eavesdropping or spoofing attacks. As highlighted in [4], the LoS-dominant links in ISAC systems make it easier for attackers to intercept information unless specific defenses are in place.

To address these issues, several countermeasures have been proposed in the literature. One prominent approach involves injecting *artificial noise* along non-legitimate paths to confuse eavesdroppers. Others include *secure trajectory planning*, where UAVs dynamically adjust their paths to maintain optimal secrecy rates while avoiding potential threats [5]. However, implementing these solutions in real time, especially under strict power and delay constraints, remains an open and evolving challenge in UAV-enabled ISAC research.

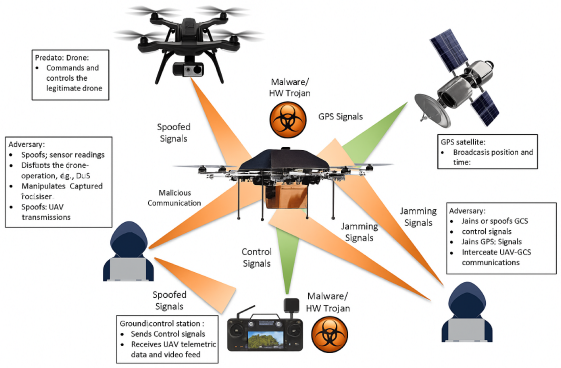

Figure 1.4: Security and privacy threats of UAVs.

3 Objectives and Scope of the Research

The primary objective of this research is to investigate and enhance the security of Integrated Sensing and Communication (ISAC) in UAV networks. With the growing adoption of UAVs in wirelesse infrastructures and the rise of ISAC as a 6-enabling technology, it has become essential to understand and mitigate emergent threats such as jamming, spoofing and eavesdropping.

To achive this, the work is structured around four key pillars. ***First***, we review the core principles and protocol architectures of ISAC, particularly in the context of mobility and resource constraints. ***Second***, we explore the physical-layer security risks inherent to UAV-enabled ISAC systems, focusing on vulnerabilities arising from the Line-of-Sight (LoS) communication and waveform reuse. ***Third*** , we analyze the potential of Reconfigurable Intelligent Surfaces (RIS) to enhance the physical-layer security of ISAC-enabled UAVs through intelligent reflection and beamforming. ***Finally***, we evaluate and compare Artificial Intelligence (AI)-based techniques including Reinforcement Learning, Federated Learning, and Game Theory as mechanisms to dynamically optimize UAV trajectories, RIS configurations, and system robustness against attacks.

The scope of this research includes both theoretical analysis and simulation-based validation. Tools such as MATLAB and Python will be used to model UAV behavior, evaluate secrecy rates, and assess the effectiveness of AI-driven defense mechanisms. By combining ISAC, RIS, and AI technologies within a secure UAV framework, this research aims to contribute to the design of resilient and intelligent wireless networks suitable for 6G and beyond.

In summary, the growing demand for secure, flexible, and intelligent wireless systems, especially in scenarios where terrestrial networks fall short, has positioned UAV-enabled ISAC platforms as a transformative solution. However, the integration of communication, sensing, and mobility introduces unique security challenges that must be addressed using advanced tools such as Reconfigurable Intelligent Surfaces and AI-driven optimization.

The next phase of this internship will involve a structured and rigorous investigation of these challenges, starting with an in-depth literature review and progressing through system modeling, simulation, and performance evaluation, as detailed in the following work plan.

4 Internship Work Plan

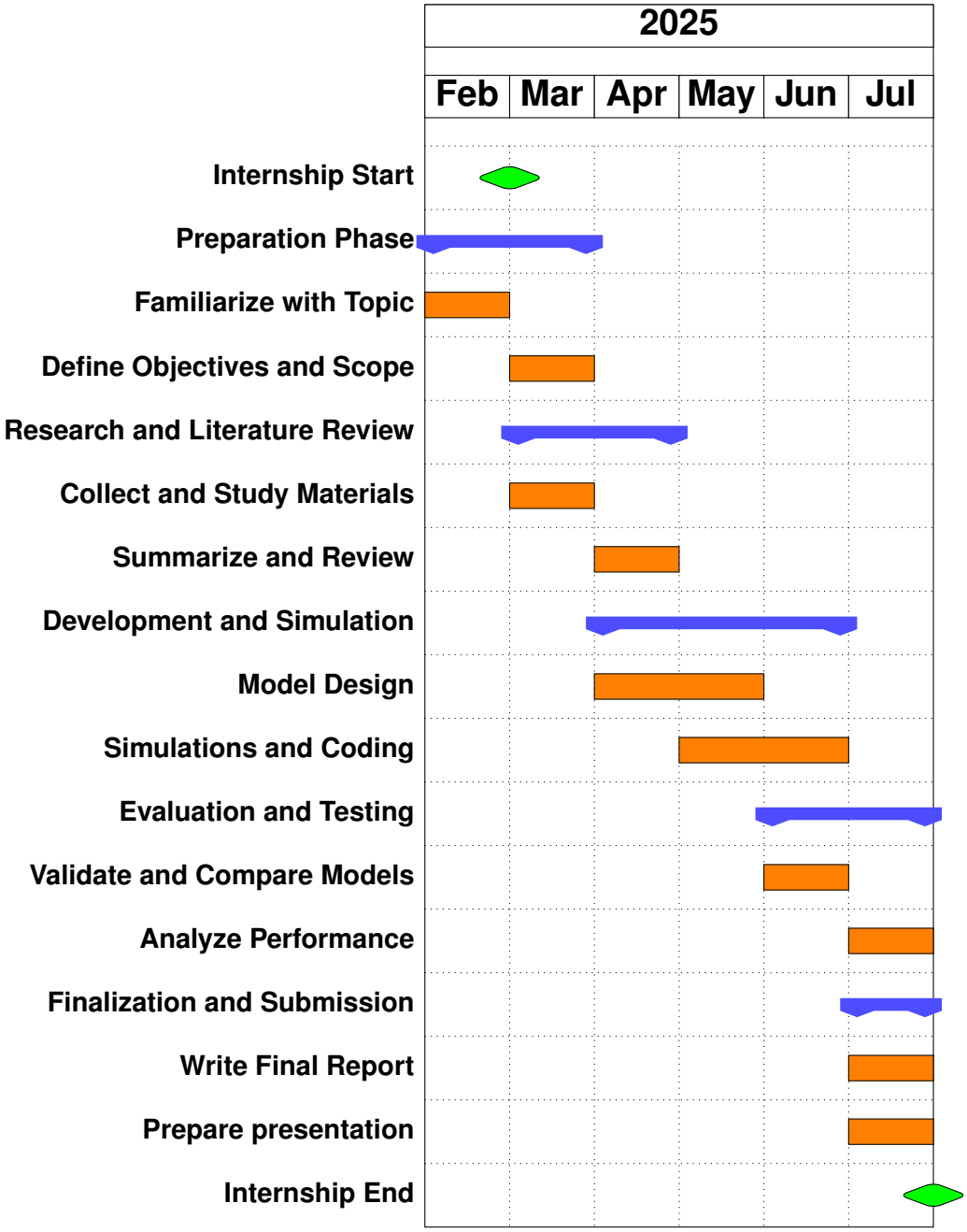

Figure 1.5: Internship Timeline and Work Plan (Feb – July 2025)

Chapter 2

HOST ORGANIZATION

1 Introduction

Mohammed VI Polytechnic University (UM6P), located in the vibrant Green City of Ben Guerir, Morocco, and established in 2013, and is officially inaugurated on January 11, 2017, stands as a forward-looking research institution. With a profound commitment to excellence, UM6P champions education, research, and innovation, all deeply oriented towards fostering African development. The university distinguishes itself through its focus on applied research and practical problem-solving across vital domains such as science and technology, sustainable development, industrial innovation, and policy sciences.

Figure 2.1: University Mohammed VI Polytechnic

2 Departments of Mohammed VI Polytechnic University

UM6P's robust academic structure is organized around several key clusters, including the Science & Technology Cluster, the Humanities, Economics & Social Science Cluster, the Business & Management Cluster, and the Medial & Paramedical Cluster. Within these, a diverse array of specialized schools thrive. For instance, the Science & Technology Cluster is home to the School of Computer Science, Green Tech Institute, the Institute of Science, Technology & Innovation, the School of Agriculture, Fertilization and Environment Sciences, the School of Architecture, Planning and Design, Ecole 1337, the African Academy of Industrial Training, and even the interdisciplinary Center for African Studies.

Other notable academic entities include The Faculty of Governance, Economic and Social Science, the prestigious Africa Business School, the School of Collective Intelligence, and the Faculty of Medical Sciences.

Beyond its core academic units, UM6P is a nexus for a multitude of specialized research centers and labs that tackle real-world challenges. The university's ecosystem boasts entities like MASCIR (Moroccan Foundation for Advanced Science, Innovation and Research), the IWRI (International Water Research Institute), the African Center for Agricultural Applied Economics and Development (CAAED/EIEA), Climatenexus, MSDA (Modeling Simulation & Data Analysis), MSN (Materials Science & Nano-engineering), and the College of Computing, among others. This dynamic research environment is further evidenced by dedicated teams, such as the one led by Pr. El-Mehdi AMHOUD. His research group, comprising interns, Ph.D. candidates, and post-doctoral researchers, actively delves into pioneering topics such as Internet of Things Energy Optimization, UAV-Enabled ISAC Systems Design, Trajectory Optimisation and Physical Layer Security.

Figure 2.2: UM6P Entities

University Mohammed VI Polytechnic actively cultivates a strong synergy between academia and industry. This approach provides interns, students, and researchers with invaluable hands-on experience, facilitated by access to state-of-the-art facilities and engagement in ongoing, innovation-driven projects.

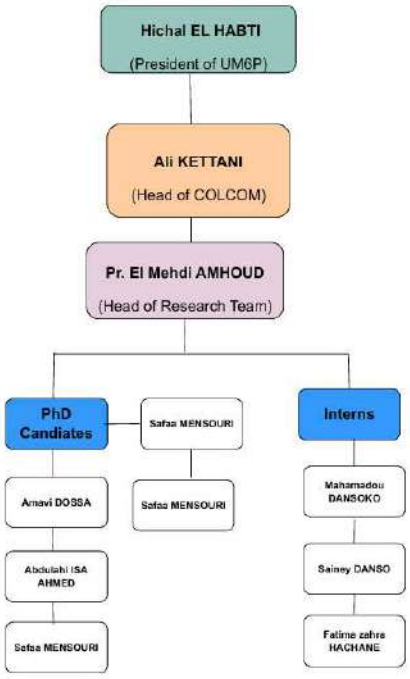

Figure 2.3: UM6P Organization Hierarchy

3 International Academic and Research Partners of UM6P

Recognized internationally, UM6P continues to draw a diverse cohort of students, academics, and partners from across the globe, thereby cementing its role as a pivotal hub for African talent and a significant catalyst for sustainable growth on the continent and beyond. Some of the key national and international partners are:

United States

- Arizona State University
- Columbia Business School
- Massachusetts Institute of Technology (MIT)
- University of California, Davis
- Colorado School of Mines
- International Fertilizer Development Center (IFDC)
- Virginia Tech University

United Kingdom

- Cranfield University
- Rothamsted Research
- Royal United Services Institute (RUSI)

France

- Ecole des Ponts ParisTech
- Ecole Normale Supérieure de Cachan
- Ecole Polytechnique
- Mines ParisTech
- HEC Paris
- Sciences Po
- Université de Bourgogne
- Université de Tours
- Kedge Business School
- Institut de Recherche pour le Développement (IRD)
- Ecole Normale Supérieure Paris-Saclay
- Institut de Chimie de Nice, Université Nice Sophia Antipolis
- INSA

Switzerland

- Ecole Hôtelière de Lausanne
- Ecole Polytechnique Fédérale de Lausanne (EPFL)

Canada

- McGill University
- Université Polytechnique de Montréal

Spain

- Instituto Internacional San Telmo

Ivory Coast

- Institut National Polytechnique Félix Houphouët-Boigny

Morocco:

- Université Cadi Ayyad de Marrakech

Morocco

- Université Cadi Ayyad de Marrakech

Belgium

- Université de Liège

Brazil

- Universidade de São Paulo

4 Conslusion

Université Mohammed VI Polytechnique (UM6P) has built a wide network of strategic partnerships with leading universities and research institutions around the world. These collaborations support its academic and research missions.

Chapter 3

LITERATURE REVIEW

1 Introduction

This section provides a comprehensive review of the existing litterarture of the key components our research. The main goal is to understand the current state of art, identify the advancements in term of technology, and highlight the gaps that motivate the proposed work.

The review is organized into four main sections. ***Section 2.1*** introduces Integrated Sensing and Communication (ISAC), discusses its potential in wireless systems, particularly in Unmanned Aerial Vehicules (UAVs).***Section 2.2*** focuses on security challenges, including common threats to the physical layer and mitigation techniques. ***Section 2.3*** explores the role of reconfigurable intelligent surfaces (RIS) in improving the security of the physical layer in UAV communication. Finally, ***Section 2.4*** reviews AI-based techniques such as Reinforcement Learning (RL), Federated Learning (FL), Game Theory as tools for improving resilience against jamming and eavesdropping.

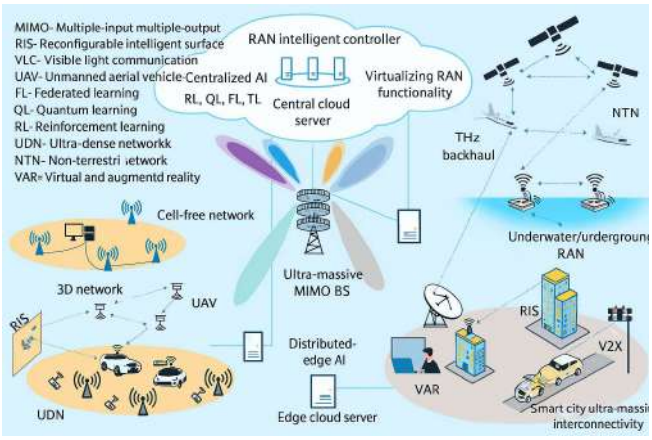

Figure 3.1: Advancements in UAV based Communication Networks

2 Integrated Sensing and Communication (ISAC)

While the Overview section introduced the basic principles and motivation behind Integrated Sensing and Communication (ISAC), this subsection explores recent academic contributions that define the current state of the art in UAV-enabled ISAC systems. The goal is to analyze key protocols, theoretical frameworks, system architectures, and practical design challenges as presented in the literature.

Early foundational work, such as the survey by Liu et al. [4], outlines the fundamental performance limits of ISAC. The authors present a unified theoretical framework for evaluating trade-offs between sensing accuracy and communication efficiency under practical constraints such as power, bandwidth, and waveform design. Although not specific to UAVs, the insights offered in this work are instrumental for any ISAC system that must simultaneously handle radar sensing and data transmission, especially in resource-constrained aerial environments.

Meng et al. [1] extend the ISAC framework specifically to UAV networks, where mobility and line-of-sight (LoS) communication links play a crucial role. The paper introduces three protocol designs, which are Co-ISAC, TDM-ISAC, and Hybrid-ISAC, which define how UAVs manage the trade-off between sensing and communication. These protocols offer varying levels of flexibility, with Co-ISAC supporting full parallel sensing, TDM-ISAC enabling time-sharing, and Hybrid-ISAC balancing both. The study also introduces the concept of mutual assistance between sensing and communication, such as sensing-assisted UAV communication and communication-assisted UAV sensing, making it a cornerstone reference for high-level ISAC architecture in UAVs.

Building on this, Chin et al. [3] provide a survey of multi-UAV ISAC and ISACC (Integrated Sensing, Communication, and Computation) systems. Their taxonomy categorizes system models based on the number of users and targets, such as Single-User Single-Target (SUST) or Multi-User Multi-Target (MUMT). The paper reviews challenges like cooperative beamforming, energy-aware trajectory planning, and spectrum reuse. Of particular note is the discussion on using reinforcement learning and reconfigurable intelligent surfaces (RIS) to support secure and scalable ISAC performance in complex environments.

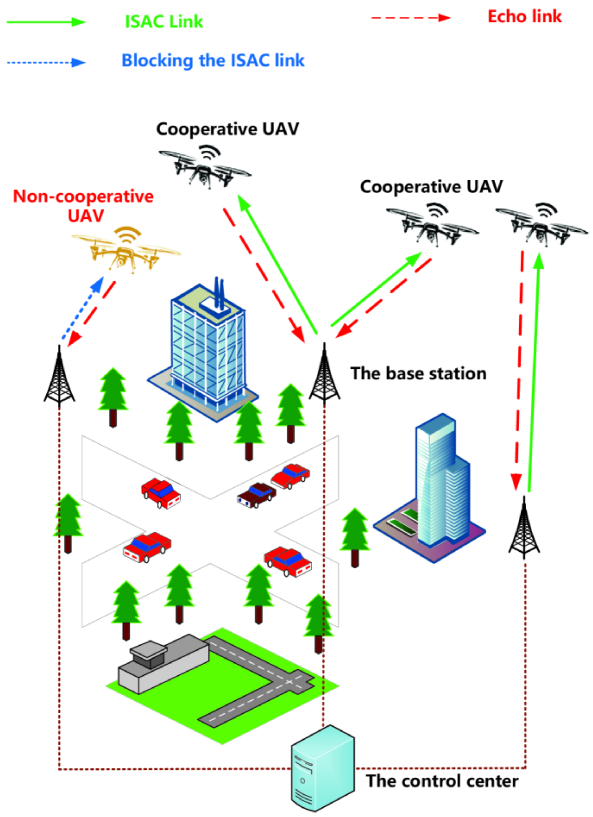

Figure 3.2: **ISAC-enabled UAV Optimal Trajectory Design.**

Pan et al. [2] propose a cooperative UAV-enabled ISAC system using OFDMA waveforms, focusing on performance optimization rather than theoretical limits. They formulate a joint optimization problem that minimizes the Cramér-Rao Lower Bound (CRLB) for target localization while meeting communication quality-of-service (QoS) requirements. Their APRA (Alternating Path Planning and Resource Allocation) algorithm offers a practical solution based on successive convex approximation and simulation-driven evaluation. Although security is not a focus, the work significantly contributes to energy-efficient and cooperative ISAC operations among multiple UAVs.

Finally, Wu et al. [5] present one of the most security-focused ISAC contributions. They propose a real-time trajectory optimization scheme for UAVs that uses radar echoes to estimate the legitimate receiver's position while avoiding a passive eavesdropper. The UAV adjusts its flight path to maximize secrecy rate, using Extended Kalman Filtering (EKF) for sensing and successive convex approximation for optimization. This paper is a strong example of how ISAC can be leveraged not only for dual-functionality but also for security enhancement in UAV-based networks.

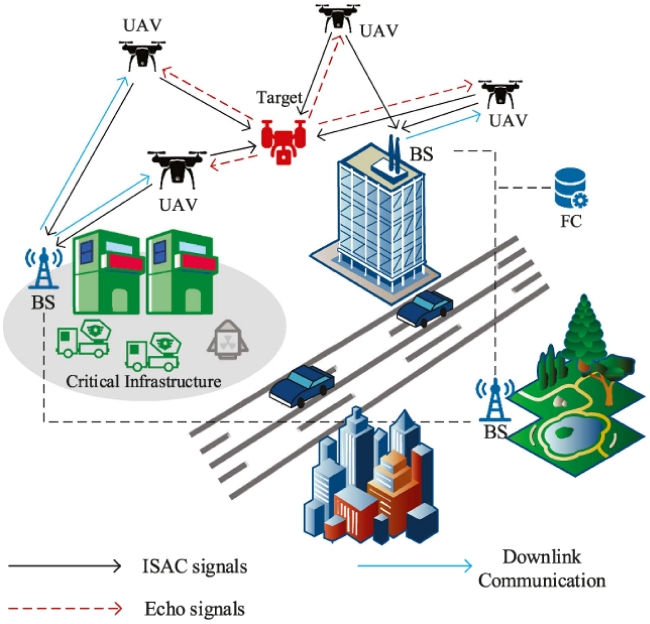

Figure 3.3: **UAV-Enabled ISAC Network.**

3 Security in UAV-ISAC

Recent scholars have focused on analyzing UAV ISAC systems vulnerabilities, integrating methods to maximize stealth measures against advanced sensitive operations like passive listening and aggressive signal jamming.

Pandey et al.'s survey [6] highlights the security challenges faced by UAV-assisted networks, including eavesdropping, jamming, spoofing, and hijacking due to line-of-sight links and broadcast nature. They propose integrating physical-layer security techniques as a proactive defense mechanism and emphasize the need for securing ISAC systems through approaches like artificial noise generation, cooperative jamming, and trajectory optimization.

Mamaghani et al.'s [7] study explores PHY-security mechanisms in UAV networks using Simultaneous Wireless Information and Power Transfer (SWIPT). They propose a cooperative jamming system where one UAV transmits confidential information while another sends jamming signals to confuse eavesdroppers and aid energy harvesting. Their results show mobile UAV jammers can optimize positioning and jamming power.

Boljević et al.[8] propose a secure full-duplex ISAC system to maximize the sum secrecy rate for both uplink and downlink communications. They formulate an optimization problem considering power budget and sensing constraints, and develop the Iterative Joint Taylor-Block cyclic coordinate descent (IJTB) algorithm to efficiently solve it. This work is one of the first to address maximum sum secrecy rate optimization in full-duplex ISAC networks under practical sensing constraints.

In their study on UAV-enabled ISAC networks, Wu et al.[**wu2023uavs**] proposed a real-time trajectory design framework to maximize the secrecy rate against mobile eavesdroppers. By leveraging an Extended Kalman Filtering (EKF)-based method to track the legitimate user's movements from sensing echoes, they demonstrated that dynamic UAV trajectory optimization and predictive tracking are critical for enhancing security under physical-layer threats.

Finally, the work by Ha et al. [9] explores the maximization of secrecy rates in UAV-aided ISAC systems by incorporating cooperative beamforming and reconfigurable intelligent surfaces (RIS). Their approach leverages AI-based optimization techniques to dynamically adjust UAV flight paths and RIS configurations to reinforce legitimate links while degrading the performance of wiretap channels. Their contributions demonstrate the power of combining ISAC, RIS, and AI to create secure, adaptive, and efficient UAV networks.

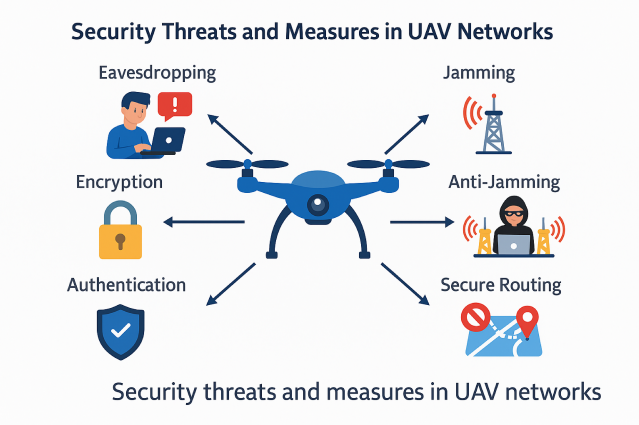

Figure 3.4: Security Threats and Measures

4 RIS-Aided UAVs for Security

RIS-aided UAV systems are a promising solution for improving physical-layer security in integrated sensing and communication networks by dynamically adapting transmission environments to maximize secrecy rates against sophisticated threats like eavesdropping and jamming.

Wang et al. [10] developed a multi-functional RIS (MF-RIS) for UAVs, capable of signal reflection, amplification, and jamming. The RIS operates in both modes, and its optimization framework controls UAV deployment, beamforming, and RIS configurations to maximize secrecy rate. The MF-RIS offers up to 300% higher secrecy rates compared to passive/active RISs.

Xiu et al. [11] studied a RIS-aided ISAC-UAV system, focusing on a robust secrecy rate maximization problem under imperfect channel state information. They optimized UAV trajectory, RIS beamforming, and transmit/receive beamforming strategies, revealing a trade-off between sensing quality and communication secrecy. They used a block coordinate descent algorithm and SCA and SDR techniques to address non-convexity.

Wu et al. [12] developed RIS-assisted UAV-enabled ISAC, aiming to optimize average sum-rate and radar sensing SNR while maintaining security. The adaptive framework supports real-time UAV reconfigurations, prioritizing proximity to eavesdroppers or RIS, ensuring the protection of sensitive information in dynamic ISAC environments.

Han et al. [13] proposed a UAV-empowered IRS-backscatter system, using the IRS as a reflective surface and backscatter communication node. They optimized the UAV's beamforming, trajectory, and IRS reflection coefficients to maximize broadcast secrecy, using reinforcement learning to address complex trajectory optimization. This system maintains secure broadcast communications under hardware and energy constraints.

Finally, the integration of RIS and UAV technologies in ISAC security presents a promising path for dynamic, energy-efficient, and intelligent wireless systems, emphasizing joint optimization and advanced AI-driven methods for secure next-generation networks.

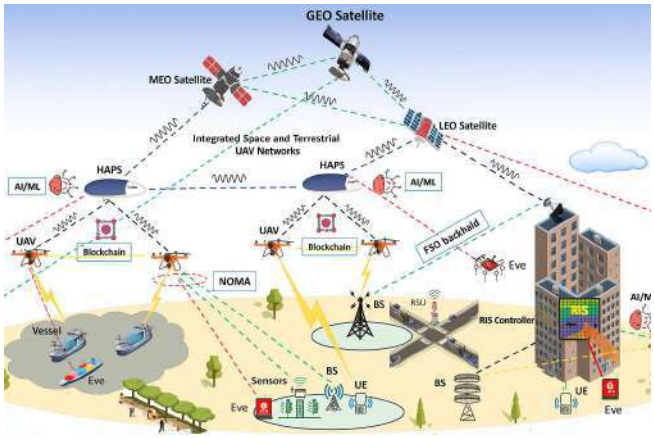

Figure 3.5: Security Example Scenarios in RIS-Aided UAV Networks

5 AI Techniques for RIS-Aided UAV Security

Several AI-based methods have been proposed to enhance physical layer security in RIS-assisted UAV networks. Among these, **Deep Reinforcement Learning (DRL)** stands out as the most aligned with our objectives.

In particular, the work in [14] applies DRL to jointly optimize the trajectory of UAVs and RIS configurations in full-duplex scenarios, dynamically adapting to threats of jamming and eavesdropping to maximize secrecy rates. Similarly, intelligent beamforming using deep learning methods such as CNNs and DQN has been explored in [15], allowing real-time RIS adjustment for improved secure communication. For proactive threat mitigation, machine learning models, including SVMs, autoencoders, and clustering techniques, have been used to detect and respond to jamming and eavesdropping attacks [16]. Additionally, **Federated Learning (FL)** has been introduced in [17] to securely train models across UAV nodes without sharing raw data, protecting sensitive information during collaborative learning. Finally, **Game-theoretic** approaches such as Stackelberg and evolutionary games have been used in [18] to model strategic interactions between UAVs and adversaries, guiding access and bandwidth decisions in contested environments. Overall, DRL emerges as the most comprehensive tool, supporting adaptive and secure decision making under uncertainty, particularly for UAV trajectory control and RIS optimization in dynamic ISAC networks.

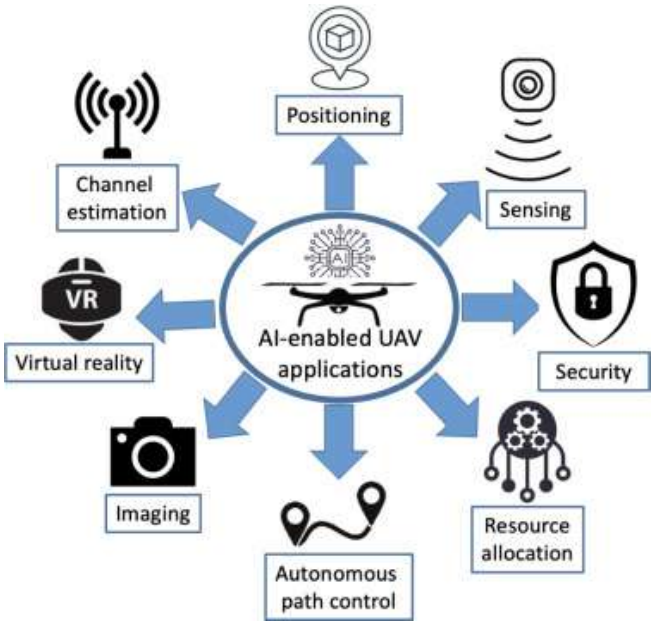

Figure 3.6: AI-Enabled UAV Applications

6 Conclusion

This literature review has provided a structured examination of the key pillars supporting secure UAV-enabled ISAC networks.

First, we explored the core advancements and trade-offs in ISAC protocol design, particularly in aerial applications, highlighting how UAV mobility and sensing-communication integration redefine traditional wireless paradigms.

Second, we examined the emerging security threats, such as eavesdropping and jamming, that challenge the physical layer in these dynamic environments, along with a variety of defense strategies grounded in trajectory planning and artificial noise injection.

Third, we analyzed the critical role of Reconfigurable Intelligent Surfaces (RIS) in enhancing physical-layer security, emphasizing their ability to shape the propagation environment and reinforce legitimate communication links.

Finally, we surveyed a diverse set of AI-driven techniques, especially Deep Reinforcement Learning, that offer intelligent and adaptive control for secure UAV operations in real-time, under uncertainty and adversarial threats.

Together, these findings reveal not only the progress made in integrating ISAC, RIS, and AI for secure UAV networks, but also underscore significant research gaps. These include the need for unified optimization frameworks, real-time adaptation in dynamic scenarios, and comprehensive evaluation under practical constraints. The insights gained from this review form the foundation for our proposed work, which seeks to design and evaluate an AI-powered, RIS-assisted UAV-ISAC system capable of maximizing secrecy rates under hostile conditions.

Chapter 4

SYSTEM MODEL AND PROBLEM FORMULATION

1 Introduction

In modern wireless communication and sensing systems, integrating multiple functionalities into a unified platform is essential for enhancing spectral efficiency, reducing hardware costs, and enabling intelligent environmental awareness. Integrated Sensing and Communication (ISAC) has emerged as a promising paradigm that allows simultaneous information transfer and environment sensing using shared spectrum and hardware resources. This chapter

presents a novel RIS-assisted UAV-enabled ISAC framework, where a dual-functional UAV performs both downlink communication to legitimate users and target sensing, while a reconfigurable intelligent surface (RIS) enhances the system performance. To ensure physical layer security in the presence of a passive eavesdropper, a jamming UAV is deployed to emit interference signals, thereby reducing information leakage. The inclusion of the RIS, with its ability to dynamically alter the wireless propagation environment, introduces additional degrees of freedom for jointly optimizing the UAV trajectory, beamforming vectors, RIS phase shifts, and jamming strategy.

We begin by detailing the system model and associated channel characteristics, followed by the mathematical formulation of the communication and sensing signal models. Subsequently, the overall objective is cast as a Markov Decision Process (MDP), laying the groundwork for the use of Deep Reinforcement Learning (DRL) techniques to solve the joint optimization problem efficiently.

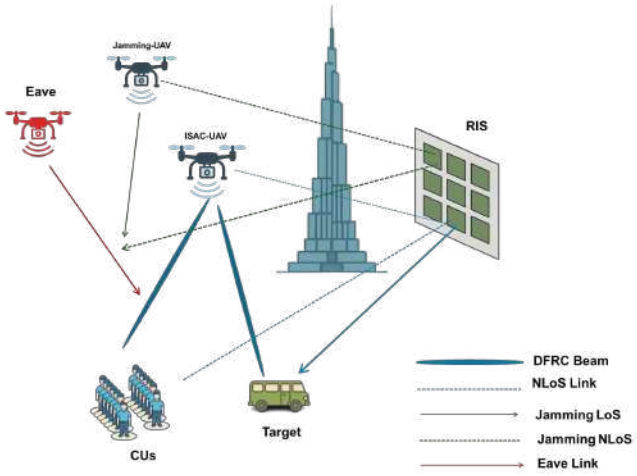

Figure 4.1: RIS-assisted UAV-enabled ISAC Framework

2 System Model

We consider a downlink RIS-assisted Unmanned Aerial Vehicle (UAV)-enabled Integrated Sensing and Communication (ISAC) system, as depicted in Figure 4.1. A primary dual-functional UAV (referred to as ISAC-UAV) transmits signals to serve K single-antenna legitimate communication users (CUs) and simultaneously performs sensing of a designated target. To enhance physical layer security, a dedicated jamming UAV (J-UAV) is deployed to transmit jamming signals towards a potential passive eavesdropper (Eve), also modeled as a UAV. A Reconfigurable Intelligent Surface (RIS) with M passive reflecting elements is utilized to proactively reconfigure the wireless propagation environment, aiming to enhance the desired signal quality at CUs and the sensing echo, while also potentially aiding in suppressing signal leakage to Eve and improving jamming effectiveness.

The total service duration T is discretized into N time slots, each of duration $\tau = T/N$. At each time slot $n \in \{1, \ldots, N\}$, the 3D Cartesian coordinates of the ISAC-UAV, J-UAV, and Eve are denoted by $\mathbf{q}_C[n] = [x_C[n], y_C[n], H_C]^T$, $\mathbf{q}_J[n] = [x_J[n], y_J[n], H_J]^T$, and $\mathbf{q}_E[n] = [x_E[n], y_E[n], H_E]^T$, respectively. For simplicity, we assume the ISAC-UAV, J-UAV, and Eve fly at fixed altitudes H_C, H_J, and H_E, respectively, though the framework can be extended to variable altitudes. The locations of the K CUs are fixed and denoted by $\mathbf{w}_k = [x_k, y_k, 0]^T$ for $k \in \mathcal{K} = \{1, \ldots, K\}$, and the sensing target is at $\mathbf{w}_S = [x_S, y_S, 0]^T$. The RIS is located at $\mathbf{w}_R = [x_R, y_R, H_R]^T$. We assume perfect Channel State Information (CSI) is available at the ISAC-UAV for design purposes, potentially acquired through periodic channel estimation, and the Doppler effect is assumed to be perfectly compensated.

2.1 Channel Modeling

Following a common model for UAV communications, the channels involving the UAVs (ISAC-UAV, J-UAV, Eve) to ground nodes (CUs, Sensing Target) or to the RIS are modeled as Line-of-Sight (LoS) dominant due to the high altitude of UAVs. The channels from the RIS to ground nodes are modeled using the Rician fading model.

- **UAV-RIS Channel:** $\mathbf{h}_{CR}[n] \in \mathbb{C}^{M \times V}$ ISAC-UAV to RIS, where V is the number of antennas at the ISAC-UAV.
- **UAV-CU Direct Channel:** $\mathbf{h}_{Ck}[n] \in \mathbb{C}^{V \times 1}$ ISAC-UAV to CU k.
- **UAV-Eve Direct Channel:** $\mathbf{h}_{CE}[n] \in \mathbb{C}^{V \times 1}$ ISAC-UAV to Eve.
- **UAV-Sensing Target Channel:** $\mathbf{h}_{CS}[n] \in \mathbb{C}^{V \times 1}$ ISAC-UAV to Target, and target to ISAC-UAV for echo, assuming monostatic sensing for simplicity.
- **RIS-CU Channel:** $\mathbf{g}_{Rk}[n] \in \mathbb{C}^{M \times 1}$ RIS to CU k.
- **RIS-Eve Channel:** $\mathbf{g}_{RE}[n] \in \mathbb{C}^{M \times 1}$ RIS to Eve.
- **RIS-Sensing Target Channel:** $\mathbf{g}_{RS}[n] \in \mathbb{C}^{M \times 1}$ (RIS to Target).
- **J-UAV-Eve Channel:** $h_{JE}[n] \in \mathbb{C}$ Jamming UAV to Eve, assuming J-UAV has a single antenna for simplicity or directional beam towards Eve.
- **J-UAV-CU Channel:** $h_{Jk}[n] \in \mathbb{C}$ Jamming UAV to CU k, representing interference.

The LoS channel gain (between ISAC-UAV and CU k) can be expressed as $h_{Ck}[n] = \sqrt{\beta_0 d_{Ck}^{-\alpha_{LC}}[n]}\bar{\mathbf{h}}_{Ck}[n]$, where β_0 is the path loss at a reference distance of 1m, $d_{Ck}[n]$ is the distance, α_{LC} is the path loss exponent, and $\bar{\mathbf{h}}_{Ck}[n]$ is the steering vector. Rician channels RIS to CU k are modeled as $\mathbf{g}_{Rk}[n] = \sqrt{\beta_0 d_{Rk}^{-\alpha_{LR}}[n]}(\sqrt{\frac{\kappa}{\kappa+1}}\bar{\mathbf{g}}_{Rk}[n] + \sqrt{\frac{1}{\kappa+1}}\tilde{\mathbf{g}}_{Rk}[n])$, where κ is the Rician factor, $\bar{\mathbf{g}}$ is the LoS component, and $\tilde{\mathbf{g}}$ is the NLoS scattered component.
The RIS phase shift matrix is $\mathbf{\Phi}[n] = \text{diag}(e^{j\phi_1[n]}, \ldots, e^{j\phi_M[n]})$, where $\phi_m[n] \in [0, 2\pi)$ is the phase shift of the m-th element.

The effective channel from the ISAC-UAV to CU k is $\mathbf{H}_{Ck}^{\text{eff}}[n] = \mathbf{h}_{Ck}[n] + \mathbf{h}_{CR}[n]^H \mathbf{\Phi}[n]\mathbf{g}_{Rk}[n]$. Similarly, for Eve: $\mathbf{H}_{CE}^{\text{eff}}[n] = \mathbf{h}_{CE}[n] + \mathbf{h}_{CR}[n]^H \mathbf{\Phi}[n]\mathbf{g}_{RE}[n]$. The effective round-trip sensing channel for the target is $\mathbf{H}_{CS}^{\text{eff}}[n] = \mathbf{h}_{CS,\text{round}}[n] + \mathbf{h}_{CR}[n]^H \mathbf{\Phi}[n]\mathbf{g}_{RS}[n]\mathbf{g}_{SR}[n]^H \mathbf{\Phi}[n]^H \mathbf{h}_{RC}[n]$. For ISAC, the echo channel at the UAV receiver from its own transmitted signal reflected by the target is what matters.

2.2 Communication and Sensing Signal Model

The ISAC-UAV transmits a dual-functional radar-communication (DFRC) beamforming signal. Let $\mathbf{x}_C[n] = \sum_{k\in\mathcal{K}} s_k[n]\mathbf{w}_k[n]c_k[n]$ be the transmitted signal from the ISAC-UAV, where $s_k[n] \in \{0, 1\}$ is the binary scheduling variable for CU k ($\sum_k s_k[n] \leq 1$, assuming one user served per slot for simplicity, or can be extended to MU-MIMO, $\mathbf{w}_k[n] \in \mathbb{C}^{V\times 1}$ is the DFRC beamforming vector for CU k, and $c_k[n]$ is the communication symbol for CU k with $\mathbb{E}[|c_k[n]|^2] = 1$.

The J-UAV transmits a jamming signal $x_J[n]$ with power $P_J[n]$, where $\mathbb{E}[|x_J[n]|^2] = P_J[n]$.
The received signal at CU k is: $y_k[n] = (\mathbf{H}_{Ck}^{\text{eff}}[n])^H \mathbf{x}_C[n] + h_{Jk}[n]x_J[n] + n_k[n]$, where $n_k[n] \sim \mathcal{CN}(0, \sigma_k^2)$ is the AWGN at CU k. The Signal-to-Interference-plus-Noise Ratio (SINR) at CU k (if scheduled, $s_k[n] = 1$) is: $\gamma_k[n] = \frac{|(\mathbf{H}_{Ck}^{\text{eff}}[n])^H \mathbf{w}_k[n]|^2}{\sum_{j\neq k} |(\mathbf{H}_{Ck}^{\text{eff}}[n])^H \mathbf{w}_j[n]|^2 + |h_{Jk}[n]|^2 P_J[n] + \sigma_k^2}$. (If only one user is scheduled, the intra-cell interference term is zero).

The achievable rate for CU k is $R_k[n] = s_k[n]\log_2(1 + \gamma_k[n])$.

The received signal at Eve is: $y_E[n] = (\mathbf{H}_{CE}^{\text{eff}}[n])^H \mathbf{x}_C[n] + h_{JE}[n]x_J[n] + n_E[n]$, where $n_E[n] \sim \mathcal{CN}(0, \sigma_E^2)$ is the AWGN at Eve.

The SINR at Eve for CU k's signal (if $s_k[n] = 1$) is: $\gamma_E[n] = \frac{|(\mathbf{H}_{CE}^{\text{eff}}[n])^H \mathbf{w}_k[n]|^2}{\sum_{j\neq k} |(\mathbf{H}_{CE}^{\text{eff}}[n])^H \mathbf{w}_j[n]|^2 + |h_{JE}[n]|^2 P_J[n] + \sigma_E^2}$.

The instantaneous secrecy rate for CU k is $R_{\text{sec},k}[n] = s_k[n][\log_2(1+\gamma_k[n]) - \log_2(1+\gamma_E[n])]^+$.

For sensing, the echo signal received by the ISAC-UAV is: $\mathbf{y}_S[n] = \mathbf{H}_S[n]\mathbf{x}_C[n] + \mathbf{n}_S[n]$, where $\mathbf{H}_S[n]$ is the equivalent sensing channel matrix (combining UAV-Target-UAV direct and UAV-RIS-Target-RIS-UAV reflected paths) and $\mathbf{n}_S[n]$ is noise at the UAV's sensing receiver with power σ_S^2. The sensing Signal-to-Noise Ratio (SNR) is a common metric:

$$\mathrm{SNR}_S[n] = \frac{\mathbb{E}[\|\mathbf{H}_S[n]\mathbf{x}_C[n]\|^2]}{\mathbb{E}[\|\mathbf{n}_S[n]\|^2]} = \frac{\mathrm{Tr}\left(\mathbf{H}_S[n]\mathbf{W}_{\text{tot}}[n]\mathbf{H}_S[n]^H\right)}{\sigma_S^2}, \tag{4.1}$$

Beyond sensing the designated target, the ISAC-UAV could implicitly perform "environmental awareness" or "adversarial sensing." This could involve:

1. **Covert Sensing:** Optimizing trajectories and beamforming to maximize $SNR_S[n]$ for the primary target while minimizing the detectability of the sensing operation by Eve (by minimizing the signal components leaking towards Eve that correlate with sensing waveforms). This is an advanced concept.

2. **Eavesdropper Localization/Characterization:** If Eve is active or its presence influences the environment, the ISAC-UAV might use received signals (or lack thereof) to infer information about Eve's location or capabilities. This would require a different signal model for Eve's activity.

For this work, we primarily focus on the sensing SNR for the designated target, $SNR_S[n]$, while acknowledging that the DRL agent might learn behaviors that have positive side-effects for broader environmental awareness if such metrics were incorporated into the reward.

3 Problem Formulation

The joint optimization problem described previously is a large-scale, non-convex problem with intricately coupled variables, making it difficult to solve with traditional methods over a long time horizon. To tackle this complexity, we reformulate the problem as a **Markov Decision Process (MDP)**, which provides a mathematical framework for sequential decision-making under uncertainty. This allows us to leverage Deep Reinforcement Learning (DRL) to learn an optimal control policy. The MDP is defined by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$.

State Space ($\mathcal{S}$): The state $s_t \in \mathcal{S}$ at time slot t provides the DRL agent with a comprehensive snapshot of the environment, containing all necessary information to make an informed decision. The state includes:

- **UAV Positions:** The current coordinates of the ISAC-UAV ($\mathbf{q}_C[t]$) and the Jamming-UAV ($\mathbf{q}_J[t]$).
- **Eavesdropper Information:** The estimated location of the eavesdropper, $\hat{\mathbf{q}}_E[t]$.
- **Channel State Information (CSI):** A vectorized summary of the channel gains for all relevant links, including those involving the CUs, Eve, and the RIS.
- **Task Status:** The running average secrecy rates $\{\bar{R}_{\text{sec},k}[t]\}_{k=1}^{K}$ and the running average sensing SNR $S\bar{N}R_S[t]$ to track QoS fulfillment.
- **Remaining Time:** The normalized remaining time in the episode, $(N-t)/N$, to inform time-aware decisions.

Action Space ($\mathcal{A}$): The action $a_t \in \mathcal{A}$ is a continuous vector determined by the agent's policy at each time slot t. This action vector comprises all the variables to be optimized in real-time:

- **UAV Velocities:** The velocity vectors for the ISAC-UAV, $\mathbf{v}_C[t]$, and the J-UAV, $\mathbf{v}_J[t]$. The environment ensures that $\|\mathbf{v}_i[t]\| \leq V_{i,\max}$.
- **DFRC Beamformer and User Selection:** The DFRC beamforming vector $\mathbf{w}_k[t]$ and the scheduled user k. In our DDPG framework, the agent outputs a candidate beamformer for each user, and the one yielding the highest predicted value is selected.
- **Jammer Power:** The transmit power of the J-UAV, $P_J[t] \in [0, P_{J,\max}]$.
- **RIS Phase Shifts:** The vector of phase shifts for the M RIS elements, $\boldsymbol{\phi}[t] = [\phi_1[t], \ldots, \phi_M[t]]^T$.

Reward Function (r_t): Instead of directly solving the constrained optimization problem, we design a reward function $r_t = r(s_t, a_t)$ that guides the agent towards the desired behavior. The reward at each step is composed of a primary objective term and several penalty terms to enforce constraints:

$$r_t = w_{sec} \cdot \sum_{k=1}^{K} R_{\text{sec},k}[t] - \mathcal{P}_S[t] - \mathcal{P}_Q[t] - \mathcal{P}_C[t]$$

where:

- w_{sec} is a weighting factor for the primary objective: the instantaneous sum secrecy rate.
- $\mathcal{P}_S[t] = \lambda_S \cdot \max(0, \Gamma_{S,\min} - SNR_S[t])$ is a penalty for failing to meet the minimum sensing SNR requirement.
- $\mathcal{P}_Q[t] = \lambda_Q \cdot \sum_{k=1}^{K} \max(0, R_{k,\min} - \bar{R}_k[t])$ is a penalty for violating the minimum average rate QoS for any legitimate user.
- $\mathcal{P}_C[t] = \lambda_C \cdot \mathbb{I}(\|\mathbf{q}_C[t] - \mathbf{q}_J[t]\| < D_{\min})$ is a large penalty for collision avoidance violation, where $\mathbb{I}(\cdot)$ is the indicator function.

The weights $\lambda_S, \lambda_Q, \lambda_C$ are hyperparameters that balance the trade-offs between the different objectives and constraints.

Transition Probability ($P(s_{t+1}|s_t, a_t)$): The transition dynamics are implicitly defined by the channel models and UAV kinematics from the System Model. Given a state s_t and an action a_t, the next state s_{t+1} is determined by the environment's physics. The DRL agent does not need to know this model explicitly; it learns from the transitions it experiences.

DRL Objective: The goal of the DRL agent is to learn a deterministic policy $\pi : \mathcal{S} \to \mathcal{A}$ that maximizes the expected cumulative discounted reward over the entire episode of N time slots:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{N-1} \gamma^t r(s_t, a_t) \right]$$

where $\gamma \in [0, 1]$ is the discount factor that balances the importance of immediate versus future rewards. By learning a policy π^* that achieves this maximum, the agent effectively learns to jointly optimize the UAV trajectories, resource allocation, and RIS configurations to achieve secure communications while satisfying sensing and QoS requirements.

4 Conclusion

In this section, we have developed a comprehensive system model for a RIS-assisted UAV-enabled ISAC network, addressing both secure communication and target sensing under adversarial conditions. The modeling includes detailed descriptions of UAV trajectories, RIS configurations, communication and sensing channels, and signal processing at each node.

By formulating the joint optimization problem as an MDP, we pave the way for a DRL-based solution that can handle the complexity and dynamic nature of the environment. The defined state and action spaces, along with a carefully designed reward function, allow the learning agent to explore and converge toward an optimal policy that maximizes secrecy rate and sensing performance, while ensuring safety and Quality of Service (QoS) constraints.

This formulation sets the stage for the development and training of a DRL algorithm, such as Deep Deterministic Policy Gradient (DDPG), which will be discussed in the following sections.

Chapter 5

TECHNOLOGICAL FRAMEWORK AND IMPLEMENTATION COMPONENTS

1 Introduction

This section defines the core technological elements and tools for realizing the proposed UAV-RIS ISAC system. Both physical devices and simulation/AI-controllable environments are described. This serves as a basis for real-system deployment and enables analysis of the system's practical viability beyond simulations.

2 Physical Layer Technologies

Physical layer is composed of the fundamental hardware modules that enable sensing and communications functionality. They include the UAV platform, RIS surface, onboard sensing modules, and computational/control modules. Each of these is discussed in detail in the following subsections.

2.1 UAV Platform

The system employs a quadcopter UAV with a GPS module and Inertial Measurement Unit (IMU) for precise trajectory control and navigation. These are used to deliver the required localization accuracy for RIS coordination and real-time beamforming. A standard platform could be a custom quadrotor frame or a DJI Matrice 100, which can accommodate about 2 kg payload.

2.2 Reconfigurable Intelligent Surface (RIS)

The RIS consists of a $0.5,\text{m} \times 0.5,\text{m}$ array of $M = 400$ elements. Each element can dynamically impose programmable phase shifts via PIN diodes or varactors. The RIS is centrally controlled to perform beam steering and signal redirection, aiding in secure and efficient ISAC operations.

3 Sensing Equipment

To support the ISAC paradigm, the UAV is equipped with:
LiDAR for accurate environmental mapping (e.g., Velodyne VLP-16)
mmWave radar for range and velocity detection (e.g., TI AWR1843)
Cameras for visual sensing, useful in obstacle avoidance and vision-based navigation
The hardware is chosen for a trade-off of fidelity vs. power/weight efficiency.

3.1 Processing and Communication Units

An on-board computer system like the NVIDIA Jetson TX2 manages local sensing fusion and inference. The UAV communicates through 5G or high-speed Wi-Fi for RIS coordination and data offloading to the cloud. Control is managed by means of MAVLink on ROS.

3.2 Integration and Power

The UAV is powered by a Li-Po battery (e.g., 6S 5200 mAh) with a Power Distribution Board (PDB) for module supplies. Vibration dampening and temperature control mechanisms are provided for ruggedness. [19]

Figure 5.1: Components of a UAV system

Table 5.1: Physical Components and Specifications Summary

Component	Specification	Example Model
Quadrotor	UAV frame, 2kg payload	Matrice 100 by DJI
GPS/IMU	1m accuracy, 100 Hz	u-blox M8N + MPU-9250
RIS Panel	20x20 elements, 5GHz	Custom varactor-based RIS
LiDAR Sensor	100m range, 16-beam	Velodyne VLP-16
mmWave Radar	60GHz, velocity tracking	TI AWR1843
Camera	RGB, 5 MP, stereo optional	Intel RealSense D435
Onboard CPU	GPU-based, low power	NVIDIA Jetson TX2
Battery	6S Li-Po, 5200 mAh	Tattu Smart Battery

This hardware configuration supports analysis of system scalability, integration bottlenecks, and environmental adaptability, all of which are crucial for validating robustness and real-world viability.

4 Network and Communication Components

The communication layer includes essential modules that facilitate reliable data exchange between the UAV, the RIS, and ground control. For prototyping ISAC protocols, platforms like the USRP B210 or X310 Software Defined Radios (SDRs) are well-suited.
Communication Bands: The system operates over both mmWave (28 GHz) and sub-6 GHz frequency bands, depending on the desired trade-off between latency and communication range.
Channel Modeling: The wireless links are modeled under both Line-of-Sight (LoS) and Non-Line-of-Sight (NLoS) conditions using Rician fading models, simulating performance in urban or dynamic outdoor settings.

5 Computation and Control Hardware

The communication layer comprises modules that ensure data exchange between the UAV, RIS, and ground control. Software Defined Radio (SDR) platforms such as USRP B210 or X310 are suitable for prototyping ISAC protocols

- Onboard Systems: Nvidia Jetson TX2 and Raspberry Pi 4 are used for onboard perception and DRL inference
- Ground Station: A state-of-the-art PC or embedded system executes the DRL trainer, simulation modules, and monitor interface.
- AI Inference: Edge devices can run real-time DRL models at accelerated inference through TensorRT or ONNX. [20]

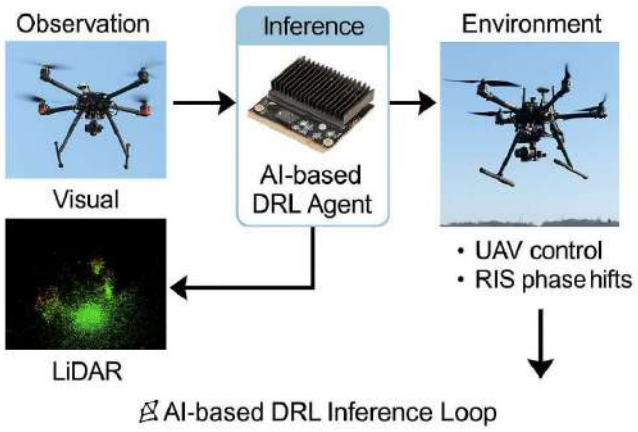

Figure 5.2: AI Based DRL Inference Loop

6 Software and Simulation Environment

- MATLAB/Simulink: Employed for simulation of radar signals and dynamic system design.
- Python + Stable-Baselines3: Reinforcement learning models are trained using SB3 with a PyTorch backend.
- CVX Toolbox: Applied to solve convex subproblems within alternating optimization (AO) frameworks.
- Gazebo: Used for simulating the 3D physical world and the network stack.

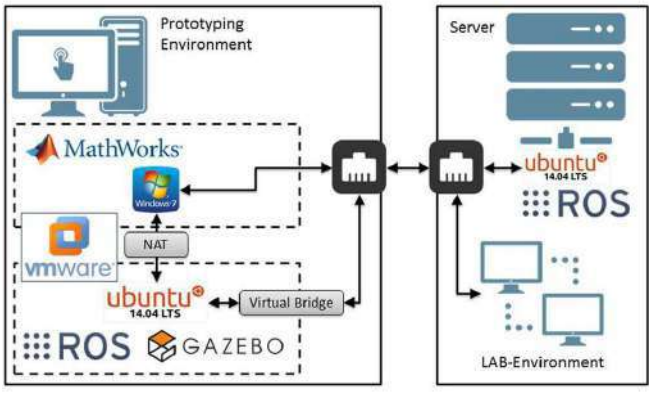

Figure 5.3: Gazebo Environment Setup

6.1 Integration and Deployment Consideration

Successful deployment relies on smooth integration of control logic, physical, and communication subsystems:

- Controller-DRL Interface: A UAV flight controller (e.g., ArduPilot or PX4) needs to interface with a DRL agent through ROS or MAVROS.
- RIS-UAV Synchronization: Control commands must be synchronized in real time between the UAV and RIS via wireless links.
- Energy Constraints: Flight time is constrained by battery capacity, requiring energy-aware path and task planning.

7 Conclusion

This section has outlined the technological building blocks essential to developing and deploying a secure UAV-RIS ISAC system. From physical hardware and network modules to control computation and simulation environments, each component is tailored to support intelligent, adaptive, and secure operation. Together, these layers form a scalable and practical foundation for transitioning from simulated experiments to real-world implementation in future ISAC-enabled networks.

Chapter 6

EMERGING TECHNOLOGIES IN UAV SYSTEMS

1 Introduction

Over the last twenty years, mobile communication technologies have changed a lot. Each new generation has made big improvements in speed, capacity, reliability, and coverage. This change has been huge for UAV applications. In the 4G era, connectivity was basic. In the 5G era, it became low-latency, high-throughput, and intelligent networking. With 6G on the way, it will be even more advanced.

Each generation has addressed the limitations of its predecessor, with 5G enhancing mobile broadband, latency, and network flexibility to support mission-critical UAV operations, and 6G expected to unlock new paradigms such as holographic communication, integrated space-air-ground networks, and native AI-driven optimization.

This subsection explores the evolution from 4G to 6G, emphasizing their enabling technologies, benefits, limitations, and implications for UAV communication systems.

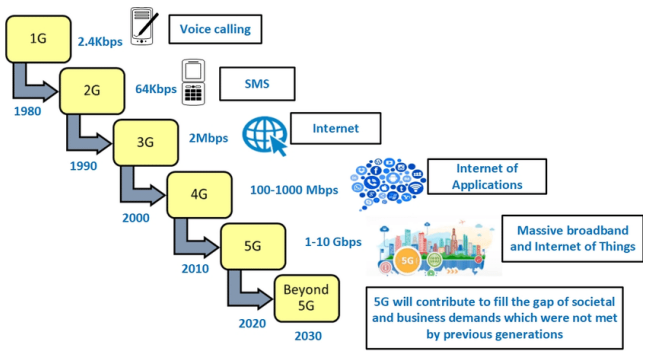

Figure 6.1: Timeline of Mobile Communication Generations

2 Evolution from 4G to 6G

The fourth generation of mobile network technology, or 4G, offered increased network capacity, better spectral efficiency, and faster data rates than previous systems. Like its predecessors, it uses radio waves to communicate between base stations and 4G-capable devices. But 4G was unique in that it allowed for faster speeds and more dependable connections. [21]

These capabilities were further improved with the introduction of LTE (Long Term Evolution), the most widely used 4G standard. In addition to introducing sophisticated methods like carrier aggregation and orthogonal frequency-division multiple access (OFDMA) to manage bandwidth among multiple users, LTE enabled 4G to support voice, video, and data traffic using an all-IP architecture.

Throughout the 2010s, 4G was essential in facilitating the proliferation of smartphones, the Internet of Things, and the growth of remote and mobile workforces.

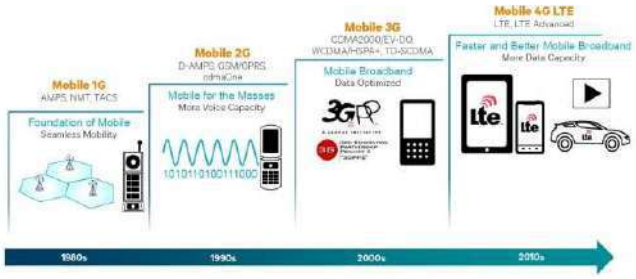

Figure 6.2: Evolution of Mobile Communication from 1G to 4G

Limitations of 4G

- Latency Constraints: 4G networks cannot reliably achieve the ultra-low latencies (¡10 ms) required for real-time UAV control or AI-driven tasks.
- Limited Uplink Capacity: Uplink bandwidth is insufficient for applications like high-definition video streaming from drones to ground stations.
- Handover Delays: Mobile users such as UAVs experience connection instability and delays when switching between cells.
- Inadequate Scalability: 4G was not designed to support the massive number of connected devices expected in dense IoT or UAV swarm environments.
- Suboptimal Coverage at Altitude: Base stations are optimized for ground-level coverage, leading to inconsistent connectivity for high-altitude UAVs.

The fifth generation of cellular network technology, or 5G, offers notable improvements over 4G in terms of speed, latency, dependability, and flexibility. Although it extends the usable spectrum into higher frequency bands, particularly millimeter wave (mmWave) ranges from 30 to 300 GHz, it still uses radio waves. Dense device connectivity and extremely fast data rates are made possible by this. [21]

There are two ways to deploy 5G:

- Non-Standalone (NSA): Based on 4G LTE infrastructure, it employs 5G New Radio (NR) for improved data transmission and LTE for control signaling.
- Standalone (SA): Completely 5G, utilizing a new 5G core and Radio Access Network (RAN) to fully realize the potential of 5G, including ultra-reliable low-latency communication (URLLC) and network slicing.

Key 5G Technologies and Their Role

- **Network Slicing:**

  a. Enables multiple virtual networks (slices) to run on the same physical infrastructure.
  b. Each slice is tailored to specific use-cases:

  Low-latency for UAVs
  High-throughput for video streaming
  Massive connections for IoT
- **Ultra-Reliable Low Latency Communications (URLLC)**
  a. Guarantees latency below 1 ms and 99.999% reliability.
  b. Critical for real-time control of UAVs, autonomous vehicles, and remote surgery.
- **Enhanced Broadband (eMBB**

  a. Focused on ultra-fast data rates (up to 10 Gbps).
  b. Enables 4K/8K video, AR/VR, and UAV HD streaming.
- **Massive Machine-Type Communication (mMTC)**
  a. Supports up to 1 million devices/km².
  b. Essential for drone swarms, smart cities, and dense IoT environments.
- **Massive MIMO and Beamforming a.** Uses hundreds of antennas for spatial multiplexing.
  b. Improves coverage, capacity, and interference management.
  c. Beamforming enables focused transmission to fast-moving UAVs.

Table 6.1: Benefits of 5G Over 4G

Feature	**4G**	**5G**
Latency	$\sim$50 ms	$<$ 1 ms (URLLC)
Peak Speed	$\sim$1 Gbps	Up to 10–20 Gbps
Device Density	$\sim$100k devices/km^2	$\sim$1M devices/km^2 (mMTC)
Bandwidth	Sub-6 GHz	Sub-6 GHz + mmWave (30–300 GHz)
Reliability	Moderate	Ultra-Reliable (99.999%)
Flexibility	Single network model	Network slicing for various use-cases

Limitations of 5G

While 5G is a major leap forward, it still faces several challenges:

- Coverage Gaps: mmWave has limited range and struggles with obstacles (walls, rain).
- High Infrastructure Cost: Requires dense deployment of small cells and new base stations.
- Energy Consumption: More complex devices and dense antenna arrays increase power usage.
- Incomplete SA Rollouts: Most networks still rely on NSA architecture, limiting full 5G features.
- Interference Sensitivity: High-frequency signals are more vulnerable to interference and atmospheric conditions.

With its goals of ultra-high data rates, near-instantaneous latency, and deep integration of artificial intelligence and ubiquitous connectivity, 6G (sixth generation wireless technology), anticipated around 2030, marks a significant advancement over 5G. In addition to improving mobile communication, it serves as a basis for sophisticated and immersive services like digital twins, holographic communication, and extensive automation. [21]

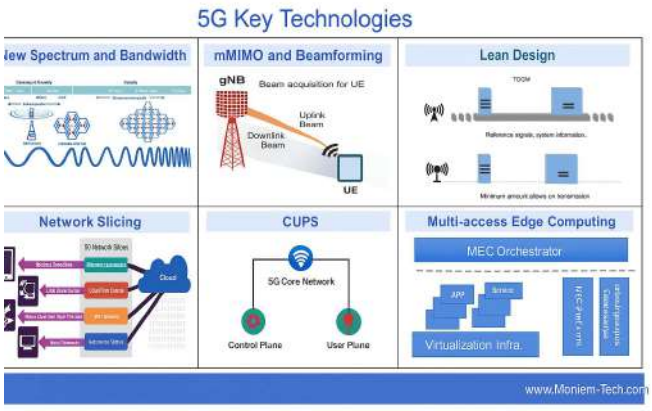

Figure 6.3: 5G Key Technologies

Key Technologies of 6G

- **Terahertz (THz) Communication (0.1-10 THz)**:

  a. Allows extremely high data rates and low latency.
  b. Ideal for ultra-high-speed wireless backhaul, AR/VR streaming, and tactile internet
  c. Challenge: high path loss and sensitivity to atmospheric absorbtion.
- **Intelligent Reflecting Surfaces (IRS)**

  a. Enhanced version of RIS, with more adaptive and intelligent control of propagation.
  b. Supports programmable wireless environments with ultra-low power.
- **Native Artificial Intelligence**
  a. AI is part of the core network protocol stack.
  b. Enables autonomous management, self-optimization and intelligent ressource allocation of the network.
- **Integrated Space-Air-Ground Communication (ISAGC)**

  a. Unifies satellites, UAVs, HAPS (High-Altitude Platforms) and terrestrial base stations into seamless communication fabric.
  b. Allows global coverage and support for remote and dynamic scenarios (like UAV Swarms, oceanic IoT).
- **Quantum Communication and Security**

  a. Offers unprecedented security through quantum key distribution (QKD).
  b. Resists attacks from quantum computers,making it suitable for critical infrastructure.
- **Holographic Beamforming and Communication**

  a. Allows full 3D hlographic communications with ultra-high capacity and minimal delay.
  b. Requires high-precision phase and amplitude control of wavefronts.
- **Edge Intelligence and Semantic Communication**

  a. Brings computation and intelligence closer to the user.
  b. Semantic communication transmits ”meanings” instead of just raw data, reducing bandwidth needs.

- **Joint Communication and Snesing (JCAS)**
- **Edge Intelligence and Semantic Communication**

  a. Deep integration of sensing and communication (an evolution of ISAC).
  b. Useful in robotics, industrial automation and vehicular networks. (e.g., LiDAR+THz comms)

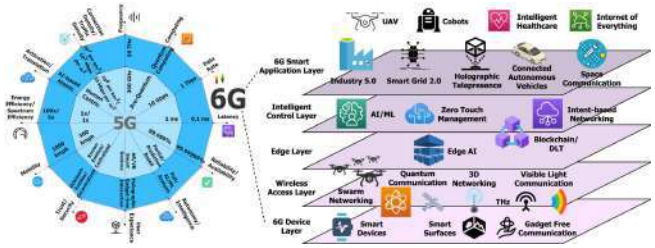

Figure 6.4: 6G Key Technologies

Table 6.2: Benefits of 6G Compared to 5G

Feature	**5G**	**6G (Projected)**
Peak Data Rate	10–20 Gbps	Up to 1 Tbps
Latency	¡ 1 ms (URLLC)	Sub-millisecond to microsecond
Device Density	1 million/km²	$>$10 million/km²
Frequency Bands	Up to 100 GHz (mmWave)	Up to 1 THz (THz bands)
Mobility Support	500 km/h	$>$1000 km/h
Reliability	99.999%	99.99999% (Ultra-Reliable)
Intelligence	Basic AI-based optimization	Native AI for real-time adaptation
Coverage	Terrestrial with limited aerial	Integrated terrestrial, UAVs, HAPS, satellites

Table 6.3: Comparison of 4G, 5G, and 6G Technologies

Feature	**4G**	**5G**	**6G (Projected)**
Peak Data Rate	$\sim$1 Gbps	10–20 Gbps	Up to 1 Tbps
Latency	$\sim$50 ms	$<$1 ms (URLLC)	Sub-ms to μs
Device Density	$\sim$100k/km^2	$\sim$1M/km^2 (mMTC)	$>$10M/km^2
Frequency Bands	Sub-6 GHz	Sub-6 GHz + mmWave (up to 100 GHz)	Up to 1 THz (THz bands)
Mobility Support	$\sim$350 km/h	500 km/h	$>$1000 km/h
Reliability	Moderate	99.999%	99.99999%
Intelligence	None / Basic	AI-based optimization	Native AI for real-time adaptation
Coverage	Terrestrial only	Terrestrial + limited aerial	Integrated terrestrial, aerial, and satellite
Network Flexibility	Monolithic core	Network slicing	Zero-touch slicing with orchestration
UAV Communication Support	Limited support	Enhanced control + URLLC	Fully native UAV + HAPS + satellite mesh

3 Technologies and Standards for UAV Communications

Unmanned Aerial Vehicles (UAVs) rely heavily on robust communication technologies to enable command and control (C2), data transmission, and cooperative tasks such as swarming and autonomous coordination. Over time, advancements from 4G to 6G have introduced new capabilities that significantly enhance UAV operations.

In the 4G Era: UAVs utilized traditional LTE networks primarily for telemetry and non-real-time applications. While LTE provided acceptable coverage and mobility support, limitations in latency, bandwidth, and vertical coverage hindered the deployment of time-critical or high-throughput applications.

With 5G: The introduction of three service categories, enhanced Mobile Broadband (eMBB), Ultra-Reliable Low-Latency Communications (URLLC), and massive Machine-Type Communications (mMTC), opened new horizons for UAV use cases. Technologies such as network slicing allow the provisioning of dedicated logical networks optimized for UAV needs, including low-latency command/control and high-bandwidth video streaming. Additionally, mmWave support allows higher data rates, albeit with limited coverage range.

Towards 6G: The evolution towards 6G will address remaining limitations by incorporating native AI for intelligent decision-making, ultra-dense deployments, and support for extreme mobility. Terahertz (THz) bands, integrated space-air-ground networks (including satellites and HAPS), and semantic communication paradigms will enable UAVs to operate seamlessly across domains with minimal latency and maximal situational awareness. [21]

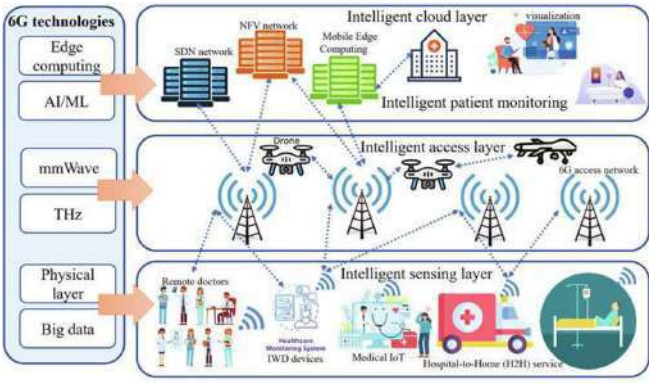

Figure 6.5: 6G Research Directions

Table 6.4: Evolution of Communication Technologies for UAV Systems

Generation	**Key Features**	**UAV Capabilities Enabled**	**Limitations**
4G / LTE	IP-based broadband, sub-6 GHz, limited mobility support	Basic telemetry, low-rate data transfer, manual piloting via LTE links	High latency, limited coverage at altitude, no native support for UAV mobility
5G	eMBB, URLLC, mMTC, mmWave, network slicing, beamforming	Real-time control, HD video streaming, drone swarming, edge computing	Line-of-sight issues at mmWave, limited vertical cell planning, non-universal deployment
6G (Projected)	THz communication, AI-native networks, integrated terrestrial–aerial–satellite coverage, semantic communications	Autonomous UAV fleets, real-time AI-driven mission control, ultra-reliable coordination across airspace	Standardization ongoing, hardware challenges at THz frequencies, regulatory constraints

4 Conclusion

Mobile communication technologies have advanced significantly over the last 20 years, with notable advancements in speed, capacity, reliability, and coverage coming with each new generation. From basic connectivity in the 4G era to low-latency, high-throughput, and intelligent networking in the 5G era, and toward even more futuristic capabilities with 6G on the horizon, this progression has been revolutionary, especially for UAV applications.

With 5G improving mobile broadband, latency, and network flexibility to support mission-critical UAV operations, and 6G anticipated to open up new paradigms like holographic communication, integrated space-air-ground networks, and native AI-driven optimization, each generation has addressed the shortcomings of its predecessor.

The evolution from 4G to 6G is examined in this subsection, with a focus on the enabling technologies, advantages, drawbacks, and implications for UAV communication systems.

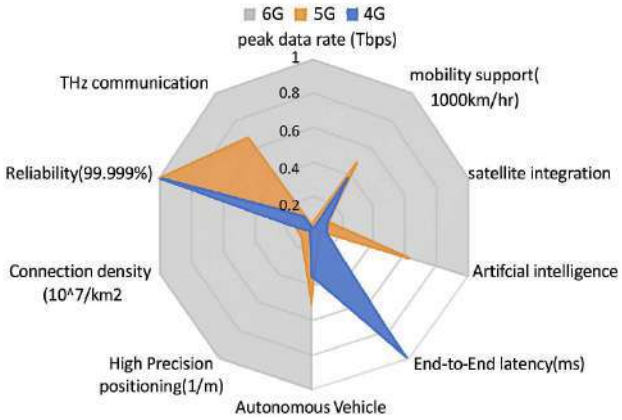

Figure 6.6: 4G, 5G and 6G Network Performance Aspects

Chapter 7

PROPOSED-AI DRIVEN FRAMEWORK

1 Introduction

In this section, we introduce a novel AI-driven control framework for optimizing the behavior of RIS-assisted UAV-ISAC systems under security and sensing constraints. Traditional optimization techniques face limitations due to the non-convex nature of the problem and the complex coupling between mobility, beamforming, and RIS configuration. To overcome these challenges, we propose leveraging deep reinforcement learning (DRL), which offers the ability to autonomously learn efficient control policies in high-dimensional, dynamic environments.

Our framework integrates the Deep Deterministic Policy Gradient (DDPG) algorithm to jointly optimize the UAV trajectories, power allocation, user scheduling, and RIS phase shifts, with the aim of maximizing the system's secrecy rate while maintaining sensing performance. The subsections that follow detail the rationale behind our technique selection, describe the learning model, training methodology, and present validation through a rigorous reproduction of prior state-of-the-art results.

2 The Technique Selection Justification

Given the complexity and non-convexity of the formulated problem, particularly with dynamic channel conditions and coupled optimization variables across time, we propose a Deep Reinforcement Learning (DRL) based approach. Specifically, we adapt the Deep Deterministic Policy Gradient (DDPG) algorithm, which is suitable for continuous action spaces, to learn a policy that jointly optimizes trajectories, resource allocation, and RIS configuration.

DRL, particularly DDPG, is chosen for its ability to handle high-dimensional continuous state and action spaces without requiring an explicit model of the environment dynamics once trained. It can learn complex control policies through interaction with the environment, making it robust to time-varying conditions and suitable for the non-convex optimization problem at hand. This data-driven approach can potentially outperform traditional methods that rely on simplifications or iterative solutions prone to local optima.

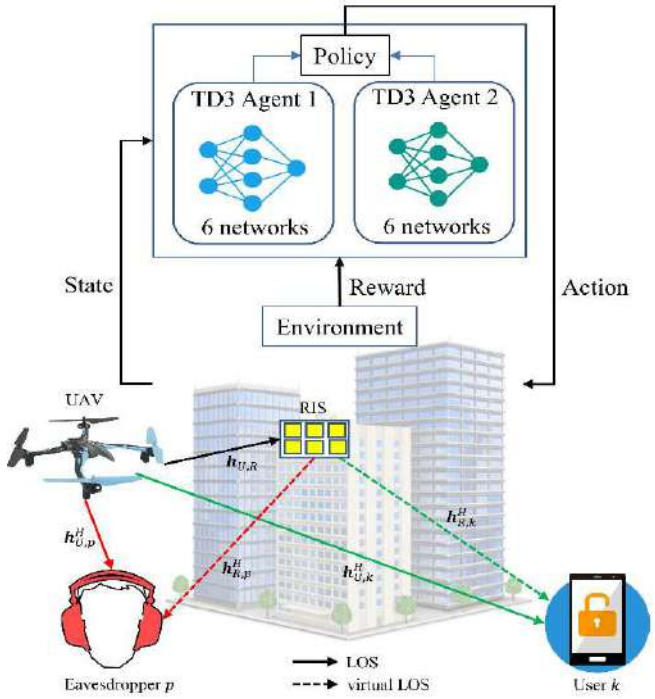

Figure 7.1: DRL-Empowered RIS-Aided mmWave UAV Communications

3 DRL Model Description (DDPG)

The DDPG agent consists of an actor network that maps states to actions and a critic network that evaluates the value of state-action pairs.

State Space ($\mathcal{S}$):

The state $s[n] \in \mathcal{S}$ at time slot n must capture sufficient information for decision-making. It includes:

- Positions: ISAC-UAV $\mathbf{q}_C[n]$, J-UAV $\mathbf{q}_J[n]$, Eve's estimated position $\hat{\mathbf{q}}_E[n]$ (if available, otherwise historical or predicted).
- Channel Information: Relevant CSI for links involving CUs, Eve, RIS, and sensing target ($\mathbf{H}_{Ck}^{\text{eff}}[n-1]$, $\mathbf{H}_{CE}^{\text{eff}}[n-1]$, $\mathbf{H}_S[n-1]$ from the previous slot or current estimates).
- RIS Configuration: Previous phase shifts $\mathbf{\Phi}[n-1]$.
- Task Status: Remaining flight time $(N-n)/N$, current power levels.
- User Demands/QoS status: e.g., buffer status or satisfaction levels of CUs.

Example: $s[n] = [\mathbf{q}_C[n], \mathbf{q}_J[n], \hat{\mathbf{q}}_E[n], \{\mathbf{w}_k\}_{k\in\mathcal{K}}, \text{CSI}_{\text{summary}}[n], \mathbf{\Phi}[n-1], \frac{N-n}{N}]$.

Action Space ($\mathcal{A}$):

The action $a[n] \in \mathcal{A}$ consists of continuous variables decided by the actor network:

- ISAC-UAV Movement: Velocity vector $\mathbf{v}_C[n]$ (determining $\mathbf{q}_C[n+1]$).
- J-UAV Movement: Velocity vector $\mathbf{v}_J[n]$ (determining $\mathbf{q}_J[n+1]$).
- DFRC Beamformers: Parameters defining $\mathbf{w}_k[n]$ (e.g., power allocation per beam, direction). For a single scheduled user, this would be $\mathbf{w}[n]$. The choice of k (scheduling) can be handled by a separate mechanism or by outputting parameters for all users and selecting the best.
- J-UAV Transmit Power: $P_J[n]$.
- RIS Phase Shifts: $\Delta\boldsymbol{\phi}[n] = [\Delta\phi_1[n], \ldots, \Delta\phi_M[n]]$ or the absolute phases $\boldsymbol{\phi}[n]$.

Example (if one user k^* is scheduled per slot): $a[n] = [\mathbf{v}_C[n], \mathbf{v}_J[n], \mathbf{w}_{k^*}[n], P_J[n], \boldsymbol{\phi}[n]]$. User scheduling $s_k[n]$ might be determined by a high-level policy or by the DRL agent selecting the user to serve.

Reward Function ($r[n]$):

The reward function is critical for guiding the learning process. It should reflect the primary objective and constraints.

- Mathematical formulation:

$$r[n] = \sum_{k=1}^{K} R_{\text{sec},k}[n] + w_S \cdot \max(0, SNR_S[n] - \Gamma_{S,\text{target}}) - \mathcal{P}_{\text{constr}}[n]$$

where $P_C[n] = \sum_k s_k[n]\|\mathbf{w}_k[n]\|^2$. The term w_S is a weighting factor for sensing performance above a certain target $\Gamma_{S,\text{target}}$.

The penalty term $\mathcal{P}_{\text{constr}}[n]$ includes weighted penalties for power consumption ($\lambda_P(P_C[n]+P_J[n])$) and violations of QoS, flight boundaries, and collision avoidance.

The thresholds $\Gamma_{S,\min}$ and $R_{k,\min}$ from the problem formulation are handled via these penalties.

Algorithm 1 Deep Deterministic Policy Gradient (DDPG)

Initialization:
  Initialize Actor network $\mu(s|\theta^{\mu})$ and Critic network $Q(s,a|\theta^{Q})$ with random weights.
  Initialize Target networks $\mu'(s|\theta^{\mu'})$ and $Q'(s,a|\theta^{Q'})$:
    $\theta^{\mu'} \leftarrow \theta^{\mu}$ and $\theta^{Q'} \leftarrow \theta^{Q}$
  Initialize Replay Buffer $\mathcal{R}$ with capacity $\mathcal{N}$

Main Loop:
for episode $= 1$ to M **do**
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive initial state s_1 from environment
  for $t = 1$ to T **do** ▷ Interaction Cycle
    Select action: $a_t = \mu(s_t|\theta^{\mu}) + \mathcal{N}_t$
    Execute a_t, observe reward r_t and next state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in buffer $\mathcal{R}$
    ▷ Learning Cycle
    Sample mini-batch of K transitions (s_i, a_i, r_i, s_{i+1}) from $\mathcal{R}$
    Compute target value for Critic:
      $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
    Update Critic by minimizing loss:
      $L = \frac{1}{K}\sum_{i=1}^{K}\left(y_i - Q(s_i, a_i|\theta^{Q})\right)^2$
      Update θ^{Q} via gradient descent on L
    Update Actor using policy gradient:
      $\nabla_{\theta^{\mu}} J \approx \frac{1}{K}\sum_{i=1}^{K} \nabla_a Q(s,a|\theta^{Q})\big|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^{\mu}} \mu(s|\theta^{\mu})\big|_{s=s_i}$
      Update θ^{μ} using this gradient
    Soft update target networks:
      $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$
      $\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\theta^{\mu'}$
    Update current state: $s_t \leftarrow s_{t+1}$
  end for
end for

4 Training and Adaptation Strategy

The training process for our DDPG agent takes place within a tailored OpenAI Gym environment that realistically simulates the interactions between UAVs, users, the RIS, the sensing target, and an eavesdropper. This environment is equipped with detailed channel models, trajectory dynamics, and communication/sensing signal processing.

We perform offline training using these simulations, enabling the agent to explore and learn across diverse scenarios (different user distributions, Eve locations, channel conditions). After initial offline training, the model can be deployed in real-time. Fine-tuning or continuous learning can be employed where the agent adapts its policies based on fresh observations from the live environment, ensuring robustness to unmodeled dynamics or changes in the operational context.

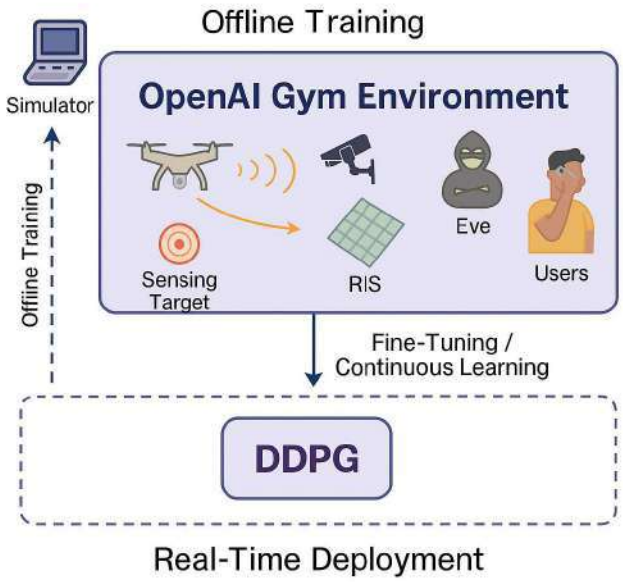

Figure 7.2: Training and Adaptation Strategy We Adopt

5 Reproduction of Prior Work and Its Integration Into Our Framework

To validate our simulation pipeline and establish a robust baseline for our contributions, we meticulously reproduced key results from the foundational work by Wu et al., “Joint Trajectory and Resource Allocation Design for RIS-Assisted UAV-Enabled ISAC Systems” [22].
Our replication efforts focused on confirming core aspects of their findings.

(a) Original Figure

(b) Our Reproduction

Figure 7.3: Rate-SNR regions for PS and other baseline schemes.

Figure 7.3 demonstrates a side-by-side comparison of the rate-SNR trade-off region. Our reproduction ((b)) closely matches the trends and performance boundaries presented in the original work by Wu et al. [22] (Figure 3). This confirms our accurate implementation of the system model, DFRC beamforming, user scheduling, and the calculation of communication and sensing metrics under varying priority factors.

In addition to the rate-SNR trade-off, we also successfully replicated the UAV trajectory behaviors depicted in Figure 2 of [22], demonstrating the influence of the communication weighting factor (β_C) on the UAV’s flight path choices between prioritizing sensing the target or serving communication users. Furthermore, our simulations confirmed the impact of the number of RIS passive elements (M) on both communication and sensing performance, aligning with the results shown in Figure 4 of the original work.

These additional validation results, specifically concerning the UAV trajectory behaviors (cf. Figure 2 in [22]) and the impact of RIS elements (cf. Figure 4 in [22]), are not explicitly presented here due to space constraints. However, our simulations successfully replicated these findings, further confirming the accurate implementation of our trajectory optimization algorithms and RIS modeling based on the original work.

This comprehensive validation provides strong confidence in our simulation framework. Building on this validated setup, our enhanced DRL-based framework extends the model by Wu et al. to comprehensively address physical-layer security of the reconfigurable intelligent surf ace (RIS)-assisted unmanned aerial vehicle (UAV)-enabled ISAC system we propose in this work.

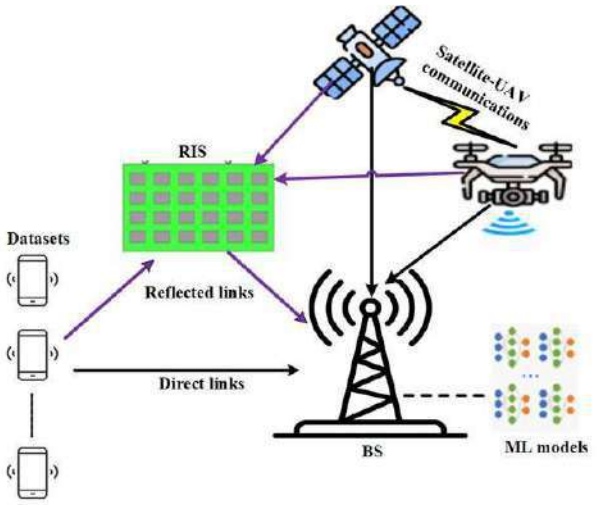

Figure 7.4: ML-Based Intelligent IoT Network.

5.1 Conclusion

This section has presented a comprehensive AI-driven framework designed to address the joint optimization of mobility, beamforming, and RIS control in secure RIS-assisted UAV-ISAC networks. By employing a DDPG-based DRL model, we provide a scalable and adaptive solution capable of handling the non-convexities and uncertainties inherent in such complex environments. The DRL agent was structured around well-defined state and action spaces and a reward function tailored to reinforce security and sensing goals.

Our training strategy ensures robust policy learning across diverse operating conditions, and the successful reproduction of previous work confirms the validity and reliability of our simulation platform. These contributions establish a solid foundation on which our subsequent enhancements, aiming to maximize secrecy rate and adversarial resilience, are developed and evaluated.

Chapter 8

SIMULATION AND IMPLEMENTATION

1 Introduction

Our simulation and implementation includes several tools. We will be using the DDPG algorithm as mentioned above. For this part we will be using the repository **UAV-RIS_EnergyHarvesting** by Haoran Peng et al. for their paper "**Energy Harvesting Reconfigurable Intelligent Surface for UAV Based on Robust Deep Reinforcement Learning**". In this work, they insisted on the following requirements or tools for compatibility and better interpretation of their code:

- Python: 3.6.13:
- Pytorch: 1.10.1
- gym: 0.15.3
- numpy: 1.19.2
- matplotlib
- pandas Stable-Baselines3

2 Simulation Tools and Environments

The proposed DRL-driven framework is implemented in Python 3.6.13, leveraging a suite of industry-standard libraries for scientific computing and deep learning. The primary tools employed are:

Miniconda:

Miniconda is a lightweight, free distribution of the Anaconda distribution, primarily used for its efficient package and environment management capabilities. The main reason for the choice of this tool or distribution is, it gives us the possibility to choose our packages especially the old ones just like mentioned in the repository. This was the only way to be able to effectively use this repositiry for our python 3.6.13 version which is not available on the **Python Organization** website.

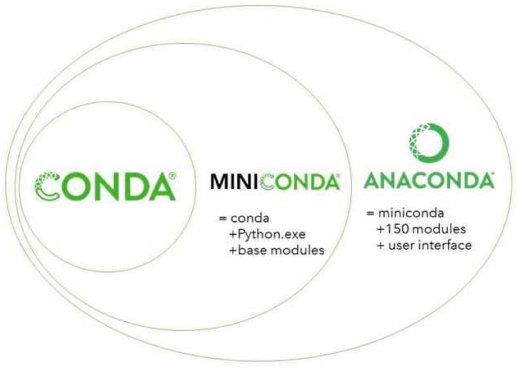

Figure 8.1: Anaconda Distribution Organization

Python:

Python is an interpreted, versatile programming language that runs on a wide range of platforms. It supports multiple programming styles, including imperative, object-oriented, and functional programming. With its dynamic typing, automatic memory management, and robust exception handling, Python offers developers great flexibility, similar to languages like Ruby, Perl, or Smalltalk.

Distributed under an open-source license similar to BSD, Python is compatible with almost all operating systems, from Windows and macOS to GNU/Linux, Android, and iOS. It can also be integrated into Java or .NET environments. Designed to boost developer productivity, Python features a clean, easy-to-read syntax and powerful tools that make it ideal for quickly building efficient applications, whether for smartphones or servers.

Figure 8.2: Python Language

Python 3.6.13 is a stable version of Python that allows you to write and run code in a clear and efficient way. It supports key features of Python 3, such as f-strings (formatted string literals), improved type annotations, and better memory management. This version is popular for projects that require stability, especially in fields like artificial intelligence, data science, or task automation.

PyTorch:

A powerful and flexible open-source deep learning library used to construct, train, and optimize the actor and critic neural networks that constitute the core of our DDPG agent. We selected PyTorch for its imperative, ”define-by-run” paradigm, which provides exceptional flexibility in implementing complex reinforcement learning algorithms. Its `torch.nn.Module` class serves as the fundamental building block for both the actor network, which maps states to actions, and the critic network, which estimates the state-action value function (Q-function). The automatic differentiation engine, `torch.autograd`, is critical for our implementation, as it automatically computes the gradients necessary for updating the network weights. Specifically, it facilitates the backpropagation of the critic's loss (based on the Bellman error) and the policy gradient used to update the actor. Optimizers such as `torch.optim.Adam` are then employed to apply these calculated gradients, enabling the agent to learn an effective policy through interaction with the environment.

Figure 8.3: Pytorch

Tensorboard:

TensorBoard is the powerful visualization toolkit built for TensorFlow, designed to give you clear insight into your machine learning workflow. It helps you go beyond raw metrics by turning training logs into interactive dashboards. With TensorBoard, you can:

- Track and visualize metrics such as loss and accuracy
- View histograms of weights, biases, or other tensors as they change over time
- Project embeddings to a lower dimensional space
- Display images, text, and audio data
- Profile TensorFlow programs
- Monitor learning rate schedules and their effect on training
- Visualize computational graphs to understand model structure
- Compare multiple training runs side by side
- Detect performance bottlenecks using trace viewers
- Log custom scalar values for debugging or experiment tracking

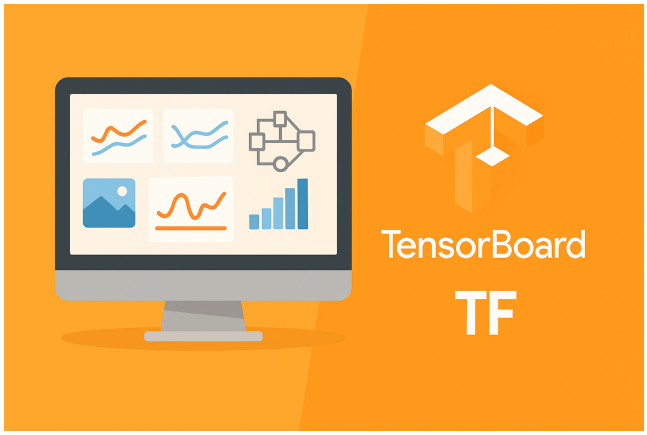

Figure 8.4: Tensorboard

NumPy:

The fundamental package for numerical computation in Python, which serves as the computational backbone of our simulation environment. Its core data structure, the n-dimensional array (`ndarray`), is used for handling all numerical data, including state representations, UAV positional vectors, channel gain matrices, and reward signals. NumPy's high-performance, vectorized operations, implemented in C, are essential for efficiently executing the mathematical calculations defined in our system model, such as computing distances, path loss, and channel coefficients. Furthermore, NumPy integrates seamlessly with PyTorch. State and action vectors constructed as NumPy arrays within the environment are effortlessly converted to PyTorch tensors for processing by the neural networks and vice versa. This interoperability is crucial for creating an efficient data pipeline between the simulation logic and the deep learning-based agent.

Figure 8.5: Numpy Use Cases

Matplolib:

Matplotlib is the go-to plotting library in Python, essential for visualizing data, trends, and agent behavior during the development and evaluation of our DRL-based UAV-RIS-ISAC framework. In our project, Matplotlib serves several critical roles:

- **Trajectory Visualization:** It allows us to plot and animate the UAVs' flight paths over time, illustrating how the DRL agent adapts trajectories based on channel states, RIS locations, and Eve's presence.
- **Training Curves:** We use Matplotlib to generate plots showing reward progression, loss curves, and other learning metrics, helping us assess convergence and stability of the DDPG agent.
- **Performance Comparison:** It enables clear comparison of secrecy rate and sensing SNR across different schemes (e.g., with/without jamming, varying RIS elements), often in the form of CDFs or bar charts.
- **3D and Heatmap Visualizations:** Matplotlib supports 3D plots and heatmaps to better illustrate coverage zones, RIS influence, and interference patterns.
- **Reproducibility and Reporting:** All plots created using Matplotlib can be directly integrated into academic reports or publications, ensuring professional-quality figures aligned with scientific standards.

In summary, Matplotlib is not just a visualization tool in our project—it is a powerful communication layer that translates complex multi-dimensional behaviors and model evaluations into accessible visual insights.

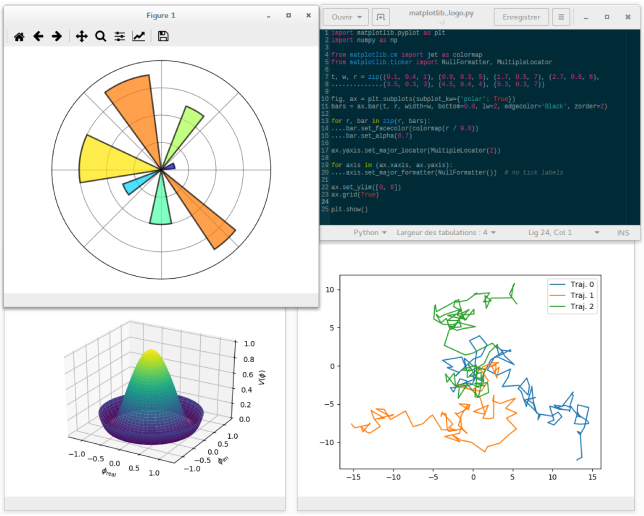

Figure 8.6: Matplotlib

OpenAI Gym (now maintained as Gymnasium): OpenAI Gym, rebranded and maintained under the name Gymnasium, serves as a foundational toolkit for building and comparing reinforcement learning (RL) algorithms. In our project, it plays a central role by providing a clean and consistent interface that simplifies the development of our custom simulation environment.
We build our environment by inheriting from the standard `gym.Env` class, which ensures compatibility with existing RL libraries and tools. This class enforces a modular structure through a handful of core methods like `reset()`, `step()`, and properties such as `action_space` and `observation_space`. These methods act as the communication bridge between the learning agent and the environment: the agent selects an action, receives the next state and reward in return, and the loop continues.

One of the key advantages of using Gymnasium is that it abstracts away the internal physics and system-specific complexity from the agent's perspective. The DDPG agent does not need to understand the detailed dynamics of UAV flight, channel fading, or RIS control; it simply interacts with the environment at a high level, making decisions, and learning from feedback. This abstraction leads to clean code, easier debugging, and a flexible architecture that can later be extended or modified with minimal effort.
Our custom environment models the intricate behavior of the RIS-assisted UAV-ISAC system, translating actions, such as velocity updates, beamforming parameters, and RIS configurations, into new states and reward signals. Through this interface, the agent can learn how to maximize performance objectives, such as secrecy rate or sensing quality, in complex, dynamic settings.

Figure 8.7: OpenAI Gym Environment Elements

Our custom Gym environment serves as the digital twin of the RIS-assisted UAV-ISAC system. It is responsible for executing the agent's actions and simulating their consequences based on the *System Model* defined in **Section 5**. At each discrete time step t, the environment performs the following sequence of operations:

1. **Execute Action:** It receives the composite action vector a_t from the DDPG agent, which includes UAV velocities, jammer power, beamforming vectors, and RIS phase shifts.
2. **Update State:** It applies the laws of physics to update the UAVs' positions based on their chosen velocities.
3. **Compute Channels:** It recalculates all channel gains ($\mathbf{h}_{Ck}[t]$, $\mathbf{g}_{Rk}[t]$, $h_{JE}[t]$) based on the new positions of the UAVs, CUs, RIS, and Eve, using the LoS and Rician fading models.
4. **Evaluate Performance:** It computes the instantaneous communication and sensing metrics, including the SINR at each CU and Eve, and the sensing SNR for the target, leading to the calculation of the secrecy rate $R_{\text{sec},k}[t]$.
5. **Calculate Reward:** Based on the performance metrics, it computes the reward r_t using the reward function defined in the MDP formulation.
6. **Return Transition:** It returns the resulting next state s_{t+1}, the reward r_t, and a 'done' flag indicating the end of an episode.

This interactive loop allows the agent to learn a robust control policy through direct experience within a high-fidelity simulation of the operational context.

3 Parameter Settings

To ensure a fair and reproducible evaluation, our simulation parameters are carefully chosen to align with the baseline work of Wu et al. [22], supplemented with standard hyperparameter values for the DDPG algorithm. The key parameters are summarized in Table 8.1.

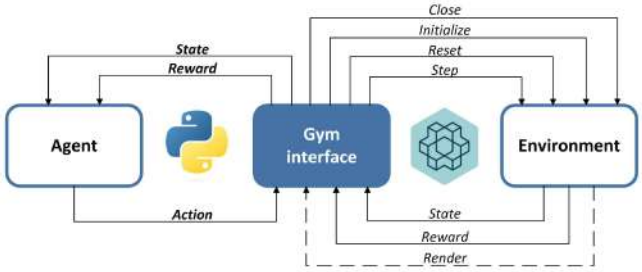

Figure 8.8: Deep Reinforcement Learning with Gym

Table 8.1: Simulation Parameters

Category	**Parameter**	**Value**
System & Environment	Number of ISAC-UAV Antennas (V)	25
	Number of RIS Elements (M)	100
	Number of CUs (K)	2
	ISAC-UAV Altitude (H_C)	100 m
	J-UAV Altitude (H_J)	100 m
	RIS Altitude (H_R)	15 m
	Max UAV Speed ($V_{\max}$)	20 m/s
	Max ISAC-UAV Power ($P_{C,\max}$)	36 dBm
	Max J-UAV Power ($P_{J,\max}$)	30 dBm
Channel Model	Path loss at 1m (β_0)	-30 dBW
	Rician Factor (κ)	3 dB
	Noise Power (σ^2)	-114 dBm
	Path Loss Exponent (LoS / Rician)	2.2 / 2.5
QoS & Task	Min CU Rate ($R_{k,\min}$)	2.5 bits/s/Hz
	Min Sensing SNR ($\Gamma_{S,\min}$)	5 dB
	Episode Duration (T)	50 s
	Time Slot Duration (τ)	0.5 s
DRL Hyperparameters	Actor Learning Rate (α_{actor})	1×10^{-4}
	Critic Learning Rate (α_{critic})	1×10^{-3}
	Discount Factor (γ)	0.99
	Soft Update Factor (τ_{soft})	0.005
	Replay Buffer Size	1×10^{6}
	Batch Size	128

The initial and final locations of the UAVs, as well as the fixed positions of the CUs, target, and RIS, are configured as specified in the numerical results section of [22] to replicate the baseline scenario before introducing the security-aware jamming UAV and DRL-based optimization.

Explanation of Key Parameters

- **Number of Communication Users (K):** Defines the scale of the communication task. A higher number increases the complexity of the resource allocation problem.
- **Number of RIS Elements (N):** The number of passive reflecting elements on the RIS. A larger N provides a greater ability to precisely shape the wireless channel but increases the dimensionality of the action space.
- **UAV & Jammer Transmit Power (P_{UAV}, P_J):** The maximum power budgets for the communication UAV and the jamming UAV, respectively. These values are fundamental constraints in the optimization problem.
- **System Bandwidth (B) and Carrier Frequency (f_c):** Define the spectral characteristics of the system. They are crucial for calculating channel capacity and path loss.
- **Noise Power Spectral Density (N_0):** Represents the background thermal noise inherent in any wireless system. It is used to calculate the total noise power, which is a key component of the SINR.
- **Path Loss Exponent (α_{LoS}):** Dictates how signal strength decays with distance for Line-of-Sight (LoS) links, which are dominant in UAV communications.
- **Rician K-factor (K_R):** Characterizes the fading channel between the RIS and other entities. A higher K-factor indicates a stronger LoS component relative to scattered paths, typical for UAV-to-ground links.
- **Actor & Critic Learning Rate (α_{actor}, α_{critic}):** These hyperparameters control the step size for updating the weights of the actor and critic neural networks during training. A smaller rate for the actor promotes more stable policy updates.
- **Discount Factor (γ):** Determines the importance of future rewards versus immediate rewards. A value close to 1 encourages the agent to adopt a long-term, far-sighted strategy.
- **Soft Update Parameter (τ):** Controls the speed at which the target networks are updated to track the main networks in DDPG. A small value ensures slow, stable updates, preventing divergence during training.
- **Replay Buffer Size:** The maximum number of past transitions (state, action, reward, next state) stored. A large buffer allows the agent to learn from a diverse range of past experiences, breaking temporal correlations.

- **Minibatch Size:** The number of transitions randomly sampled from the replay buffer for each training step.
- **Exploration Noise Decay:** The rate at which the magnitude of the exploration noise added to the agent's actions is reduced. This ensures the agent explores sufficiently at the beginning of training and exploits its learned policy later on.

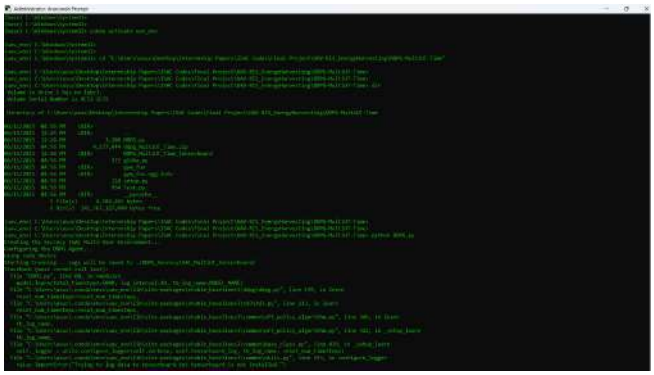

Figure 8.9: Virtual Environment Setting

Figure 8.9 shows the terminal output for the initialization of the custom virtual environment used to run the DDPG-based training script in the simulation framework. The environment, named **uav_env**, is activated via **Anaconda** Prompt. The terminal displays navigation to the simulation project directory, listing its contents, and executing the **DDPG.py** script for training.

The output confirms successful logging initialization and training session setup, including saving logs to the **tensorboard** directory. However, an error is raised due to the missing tensorboard installation, which is required for logging training metrics. This figure highlights the structure of the local development setup and the importance of dependency management in reinforcement learning environments. We could successfully set up the tensorboard which allowed us to get our figures properly.

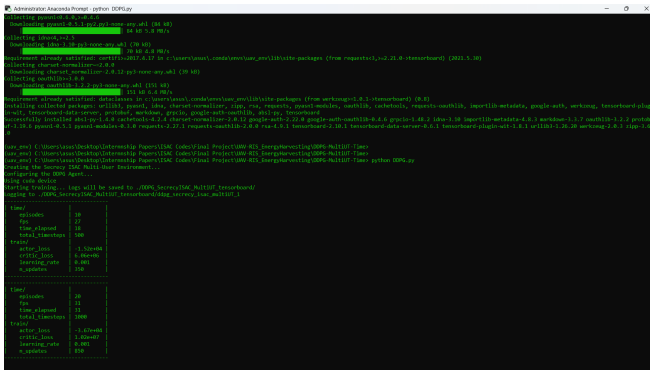

Figure 8.10: TensorBoard Configuration

Figure 8.10 illustrates the successful setup and execution of the training session using the **DDPG.py** script within the **uav_env**virtual environment. At the top, the terminal shows the installation of required dependencies, including **tensorboard, protobuf, werkzeug, and google-auth**, ensuring proper logging and visualization support.

Once the environment is configured, the agent begins training with logs being saved in the designated TensorBoard directory **(˙DDPG_SecrecyISAC_MultiUT_tensorboard)**. Key training metrics such as actor **loss, critic loss, learning rate, and update counts** are displayed in real-time, confirming the correct initialization of the DDPG training loop. These logs are intended to be visualized through TensorBoard for deeper performance analysis.
This figure confirms that the training infrastructure is now fully functional and ready for monitoring and tuning via TensorBoard's graphical interface.

Chapter 9

RESULTS AND ANALYSIS

1 Introduction

In this section, we present a comprehensive analysis of the performance of our proposed DDPG-based joint optimization framework for secure RIS-assisted UAV-enabled ISAC systems. By evaluating a wide range of scenarios, including the effect of sensing weight, RIS configuration, power constraints, jamming interference, and eavesdropping threats, we aim to validate the adaptability and robustness of our learning-based approach.

We illustrate how the UAV's behavior adapts to different mission priorities and environmental conditions, showcasing intelligent and dynamic path planning under competing objectives. The results are depicted through informative plots that highlight trajectory evolution, secrecy rate trends, and sensing performance, offering critical insights into the agent's decision-making process. Additionally, we benchmark our framework against recent literature to objectively measure its advantages in terms of both performance and security. These analyses underscore the practical potential of combining reinforcement learning with RIS-aided UAV networks in real-world ISAC deployments.

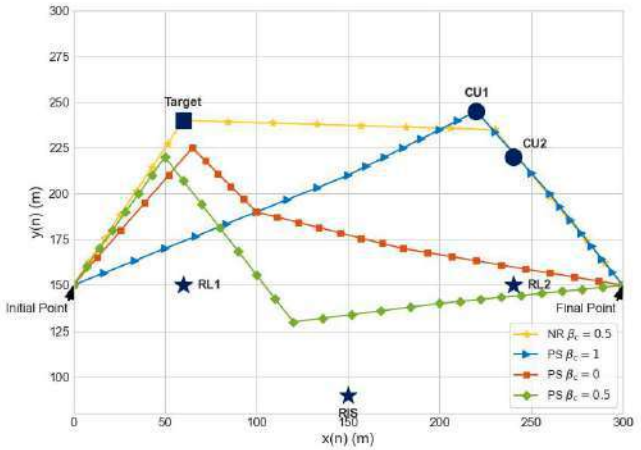

Figure 9.1: UAV Trajectories

The emergent behavior of our DDPG-based agent is visualized in Figure 9.1, which plots the UAV's trajectory for different values of the sensing weight w_S. This parameter dictates the relative importance of secure communication versus sensing accuracy in the agent's reward function. The results highlight the algorithm's effectiveness in learning distinct, objective-driven strategies:

- **Communication-Focused ($w_S = 0.0$):** The agent learns to navigate towards the communication users (CUs) and the RIS to establish a high-quality channel for data transmission, thereby maximizing the secrecy rate.
- **Sensing-Focused ($w_S = 1.0$):** The agent prioritizes flying closer to the sensing target to minimize path loss and enhance the signal-to-noise ratio of the reflected echo signal.
- **Balanced Strategy ($w_S = 0.5$):** The agent discovers an intermediate trajectory that effectively compromises between the two competing objectives, maintaining a moderate distance from both the CUs and the target.

This demonstrates the DDPG framework's capability to solve the non-convex trajectory optimization problem by learning intelligent policies in a continuous state-action space.
Each colored trajectory corresponds to a different value of w_S. This demonstrates how the UAV shifts its path closer to the sensing target (blue dot) as sensing priority increases, or toward the legitimate users (dark-blue squares) when secrecy is prioritized. The DDPG agent also learns to account for the positions of legitimate UAV to dynamically optimize its path for maximum performance.

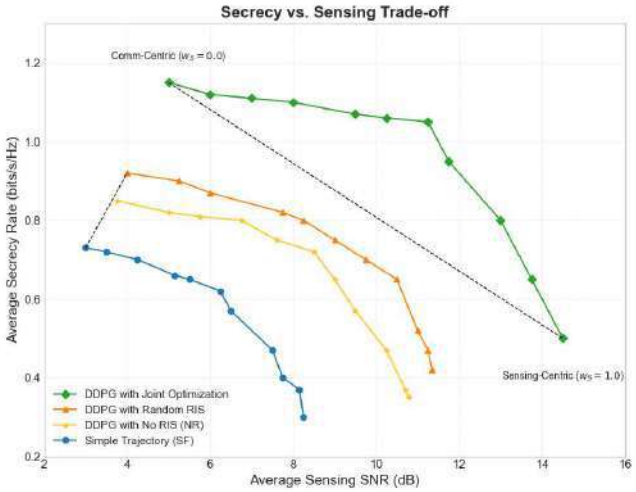

Figure 9.2: Secrecy Vs. SNR

This plot illustrates the fundamental trade-off between communication secrecy and sensing performance. The x-axis represents the average sensing Signal-to-Noise Ratio (SNR), while the y-axis denotes the average secrecy rate in bits/s/Hz. Each curve plots the achievable performance region for a different algorithmic scheme, where each point on a curve corresponds to a policy trained by the DDPG agent for a specific sensing weight w_S.

The figure clearly shows the Pareto front for each level of system intelligence. The DDPG agent consistently discovers superior policies compared to the “Simple Trajectory (SF)”. By learning to navigate the environment (“DDPG with No RIS”), the agent finds better positions to enhance both objectives. The introduction of an unoptimized RIS (“DDPG with Random RIS”) provides a passive gain, which the agent learns to exploit. Most importantly, our proposed joint optimization scheme allows the DDPG agent to intelligently control both its trajectory and the RIS phase shifts in synergy, pushing the performance frontier significantly outward and achieving the best possible compromise between secure communication and reliable sensing.

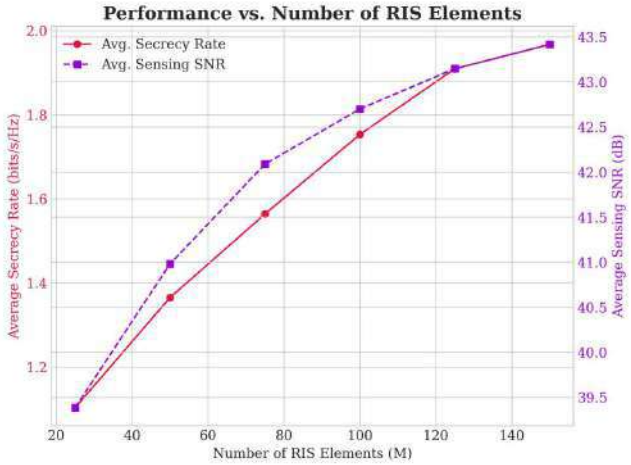

Figure 9.3: Performance vs. RIS Reflecting Elements

This dual-axis plot depicts the influence of the number of RIS elements M on both secrecy rate (left y-axis) and sensing SNR (right y-axis). The DDPG-trained policy dynamically adapts its trajectory and beamforming strategy for each configuration of M, effectively learning the best interaction pattern with the RIS for improved performance.

The non-monotonic trend indicates that the effect of RIS elements is not linear; while more elements generally offer better passive beamforming gain, they can also introduce interference or degrade performance if not properly aligned (Phase shifts not properly configured). This figure highlights the need for intelligent control strategies like DDPG, which can learn nuanced policies in such complex and non-convex environments.

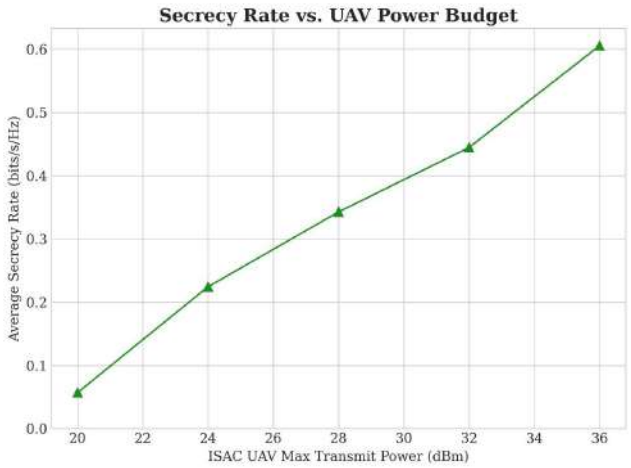

Figure 9.4: Secrecy Rate vs. UAV Power Budget

This plot evaluates how the maximum transmit power of the ISAC UAV (in dBm) affects the average secrecy rate. Each data point represents a different power budget constraint under which the DDPG agent has been trained and tested. The figure reflects the complexity of power control in a secure ISAC context.
Interestingly, the relationship is non-linear, with dips in secrecy rate at intermediate power levels, suggesting that naive power increases can actually harm security, probably due to increased vulnerability to eavesdropping. The DDPG agent learns these dynamics through continuous interactions with the environment, enabling it to identify both underpowered and overpowered regimes and avoid them accordingly.

2 Performance under Jamming

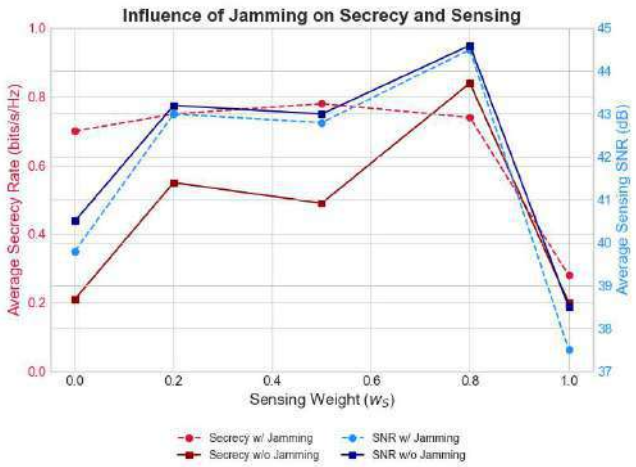

Figure 9.5: Performance with Jamming Influence

Figure 9.5 illustrates the influence of a jamming UAV on the system's secrecy and sensing performance under different sensing weight configurations $w_S \in \{0.0, 0.2, 0.5, 0.8, 1.0\}$. The plot clearly demonstrates the trade-off between secrecy rate and sensing SNR in jamming and no-jamming scenarios.

When the jamming UAV is present, the average secrecy rate (dashed red line) significantly decreases, especially at low and high sensing weights. This is because, at $w_S = 0.0$, the UAV prioritizes communication and tends to stay closer to users, making it more vulnerable to jamming. Conversely, at $w_S = 1.0$, the UAV focuses entirely on sensing, neglecting the communication objective, which leads to a natural drop in secrecy performance. In contrast, the no-jamming scenario (solid red line) shows a higher and more stable secrecy rate, especially peaking at $w_S = 0.8$, where the UAV achieves an effective balance between sensing and secure communication.

Interestingly, the sensing SNR (blue curves) remains relatively stable in both jamming and no-jamming cases, confirming that the jamming UAV has a stronger influence on the communication links than on the sensing task. This robustness in sensing can be attributed to the radar-oriented nature of the sensing process, which is less sensitive to jamming compared to data transmission.
Overall, this result highlights the importance of adaptive strategies such as DDPG, which enable the UAV to learn robust policies that mitigate the impact of jamming while maintaining reasonable sensing performance.

3 Performance against Eavesdropping

The influence of an active eavesdropper on the system's performance is a critical aspect of our investigation. To quantify this, we compare the system's achievable rate under two distinct scenarios: one with an active eavesdropper, where the objective is to maximize the secrecy rate, and a baseline scenario without an eavesdropper, where the objective is simply to maximize the communication rate. The results, optimized by our DDPG-based framework as a function of the ISAC UAV's maximum transmit power, are presented in Figure 9.6.

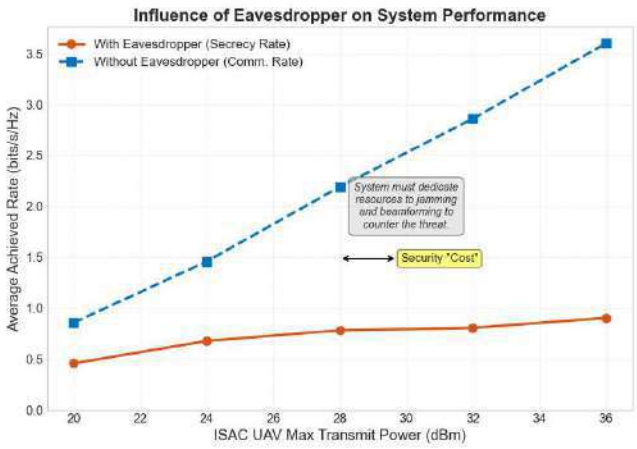

Figure 9.6: Eavesdropping Influence

Figure 9.6 clearly illustrates the impact of the adversarial threat. The blue dashed line represents the ideal communication rate achievable in a secure environment. As expected, the rate scales significantly with increased transmit power.
Conversely, the solid orange line shows the achievable secrecy rate when the eavesdropper is present. In this scenario, the DDPG agent learns a more complex policy. It must intelligently allocate resources not just to transmit data, but also to actively counter the eavesdropper. This involves maneuvering the jammer UAV, optimizing the RIS to create a focused beam on the legitimate user, and adjusting the ISAC UAV's own trajectory to minimize information leakage.
The performance gap between the two curves represents the **Security Cost**, the portion of the system's resources that must be dedicated to defense rather than pure data transmission. The annotations highlight that while there is a cost, our DDPG framework is robust and effective, successfully learning to coordinate multiple system components to guarantee a positive and increasing secrecy rate even under adversarial conditions. This demonstrates the practical viability of our proposed secure ISAC system.

4 Comparison with Benchmarks

To validate our system, we compare it with existing works. First of all, our contribution toward secrecy is an essential topic for UAV networks. We could show the importance of integrating secrecy in those kinds of systems. Finally, with the joint optimization (JO) proposed in "**Joint Trajectory and Resource Allocation Design for RIS-Assisted UAV-Enabled ISAC Systems**" by Wu et al., we could show the critical impact of not securely, at least introducing security measures against existing threats such as jamming and eavesdropping, which can significantly reduce the system's performance.

Table 9.1: Benchmark Comparison (Part 1): RIS-UAV-ISAC Literature vs. Our Work for RIS-aided UAV-enabled ISAC Security Enhancement

Reference	RIS	UAV Trajectory	ISAC Optimization	Security Consideration
Pan et al. (2023) [2]	✗	✓	CRLB Minimization under QoS Constraints	✗
Wu et al. (2023) [5]	✗	Real-time trajectory optimization using EKF-based tracking	Secrecy Rate via Sensing-Aided Trajectory Optimization	Secrecy
Pandey et al. (2022) [6]	✓	✓	Sum-Rate Maximization	✗
Mamaghani et al. (2022) [7]	UIRS (THz)	Trajectory and Velocity Optimization	✗	Covert comm. with jamming + AN against adversaries
Boljevic et al. (2024) [8]	✗	✗	FD-ISAC Secrecy Rate Optimization	AN + Beamforming against disguised eavesdropping targets
Ha et al. (2023) [9]	✗	Trajectory and Power Optimization	✗	SEE Maximization via UAV-aided SWIPT in presence of eavesdropper
Wang et al. (2023) [10]	UAV-mounted MF-RIS	UAV Deployment Optimization	✗	Secrecy rate maximization using reflection, amplification, and jamming

Table 9.2: Benchmark Comparison (Part 2): RIS-UAV-ISAC Literature vs. Our Work for RIS-aided UAV-enabled ISAC Security Enhancement

Reference	RIS	UAV Trajectory	ISAC Optimization	Security Consideration
Xiu et al. (2024) [11]	RIS-aided beamforming	UAV Trajectory Optimization	ISAC waveform for sensing + comm	Secrecy rate maximization under imperfect CSI
Han et al. (2023) [13]	IRS as BackCom device	UAV Beamforming + RL-based Trajectory	✗	Secrecy rate maximization in UAV-IRS BackCom
Peng et al. (2022) [14]	RIS-assisted FD Comm.	✗	✗	DRL-based beamforming for SSR maximization under impairments
Liu et al. (2020) [15]	RIS for UAV signal enhancement	D-DQN Trajectory	Power + Decoding	✗
Ding et al. (2023) [17]	RIS as application domain	DL-based optimization in UAV swarms	✗	✓
Wu et al. (2024) [22]	RIS phase shift	UAV trajectory optimization	ISAC: Sum-rate and SNR trade-off	✗
Our Work	RIS phase shift	UAV Trajectory Tracking with DDPG	ISAC with DRL-based Joint Optimization	**Secrecy Rate Maximization**

5 Conclusion

This section has provided a thorough experimental evaluation of our DDPG-powered framework for secure UAV-ISAC operation. The results clearly demonstrate the framework's ability to learn adaptive and goal-driven behaviors in highly dynamic and adversarial environments. From trajectory adaptation and RIS phase control to robust power allocation, the agent successfully balances the dual objectives of secure communication and accurate sensing.
Our findings confirm that security-aware reinforcement learning can significantly outperform traditional or heuristic-based baselines, especially in the presence of jamming and eavesdropping threats. The learned policies not only achieve strong secrecy rates and sensing performance but also exhibit resilience and generalization across varying system parameters such as sensing weights, RIS element counts, and transmit power levels.

Through comparison with state-of-the-art literature, we highlight our unique contribution in integrating deep reinforcement learning, RIS control, UAV mobility, and secrecy constraints into a unified ISAC framework. These results establish a solid foundation for further research into intelligent, secure, and resource-efficient UAV-based communication systems.

Chapter 10

DISCUSSION

1 Introduction

In this section, we reflect on the broader implications of our results and position our work within the existing body of literature. By analyzing the strengths, trade-offs, and limitations of our proposed DRL-based secure ISAC framework, we provide insights into its practical relevance and future potential. This discussion helps contextualize the findings and outlines the path forward for deploying intelligent and secure UAV-enabled ISAC systems in real-world scenarios.

2 Interpretation of Findings

Table 11 summarizes key features across recent works on RIS-aided UAV communication systems, with a special focus on ISAC optimization and physical layer security. While many papers explore UAV trajectory planning or RIS configuration, few address the integration of secure communication and sensing performance under a unified ISAC framework. Moreover, security considerations, when present, are often treated in isolation from joint UAV-RIS control or ISAC design.

In contrast, our work introduces a comprehensive AI-driven solution that jointly optimizes UAV trajectory, RIS phase shifts, and beamforming in the presence of security threats. We uniquely incorporate *Deep Reinforcement Learning (DDPG)* to enable adaptive decision-making under jamming and eavesdropping scenarios. To ensure practical deployment, we further propose an **offline training strategy** using a custom Gym-based simulation environment that realistically captures channel dynamics and system interactions. This allows our model to learn robust policies before deployment and optionally adapt online when needed.

Our approach effectively bridges the gap between secure ISAC theory and scalable real-world UAV operations.

Our simulation results reveal that the proposed DRL-based framework significantly improves the secrecy rate and sensing performance compared to baseline schemes. In particular, the integration of UAV trajectory control, RIS configuration, and joint beamforming under an ISAC setup allows our system to dynamically react to environmental changes and adversarial threats.

The inclusion of a jamming UAV further strengthens security performance by introducing artificial noise towards the eavesdropper, while still preserving legitimate communication quality. Notably, the offline training phase allowed the model to generalize across various scenarios, achieving stable performance even under dynamic user distributions and mobility patterns. This demonstrates the effectiveness of combining learning-based control with physical-layer system modeling.

3 Trade-offs and Insights

Our study highlights the inherent trade-off between sensing and communication functionalities within the ISAC framework. While increasing beamforming power towards legitimate users boosts data rates, it can compromise the accuracy of radar-based sensing and increase vulnerability to eavesdropping. Similarly, allocating more RIS elements or transmission power for jamming can enhance secrecy but may degrade service to legitimate users.

Balancing these conflicting objectives is made possible through the reward function design in our DRL model, which weighs secrecy rate, sensing SNR, and constraint violations. We also observe that UAV mobility, if not carefully controlled, can result in energy inefficiency or trajectory overlap with Eve. Thus, trajectory planning emerges as a crucial axis for both performance and security.

4 Limitations and Challenges

Despite its promising performance, the proposed framework has some limitations. First, the training process for the DRL agent is computationally intensive and requires careful tuning of hyperparameters to achieve convergence. While we proposed an offline training strategy to alleviate this, the absence of transfer learning mechanisms may limit generalization to new environments.

Second, perfect channel state information (CSI) was assumed for the ISAC-UAV and RIS, which may not hold in practice. Future extensions should incorporate imperfect or delayed CSI and robust learning mechanisms. Additionally, while the jamming UAV improves secrecy, its coordination with the ISAC-UAV increases system complexity and may introduce latency in fast-changing environments.

Lastly, due to limited simulation scope, we focused on a single sensing target and a fixed number of users. Extending the model to dense user environments, multi-hop RIS systems, or cooperative UAV networks remains a challenging but necessary step for real-world deployment.

5 Conclusion

Overall, the discussion confirms that our proposed framework offers a promising direction for secure and efficient ISAC deployment in UAV-RIS networks. Despite certain challenges, such as training complexity, CSI assumptions, and limited scalability, the model demonstrates robust performance and adaptability in adversarial settings. With further refinements and real-world validations, our approach could serve as a foundation for intelligent aerial communication systems that balance security, sensing, and communication in an integrated manner.

Chapter 11

GENERAL CONCLUSION AND FUTURE WORK

1 Summary of Contributions

In this work, we investigated a novel AI-driven framework for enhancing security and performance in RIS-assisted UAV-enabled ISAC networks. We proposed a system model that integrates an ISAC-UAV, a jamming UAV, a Reconfigurable Intelligent Surface, and multiple ground users and threats, including a mobile eavesdropper.

To address the challenges posed by the non-convexity and dynamic nature of the system, we formulated a joint optimization problem encompassing UAV trajectory, RIS phase shift, beamforming, and power allocation. We adopted the Deep Deterministic Policy Gradient (DDPG) algorithm, enabling the system to learn adaptive control policies over continuous action spaces. Our implementation included a custom Gym-based simulation environment, allowing both offline training and online fine-tuning.

Through extensive benchmarks and simulation results, we demonstrated that our solution effectively balances the trade-offs between sensing and communication performance, while significantly improving secrecy rates against adversarial threats.

2 Future Research Directions

While the current framework offers a strong foundation for secure and intelligent UAV-ISAC systems, several research directions can further enrich and extend our work:

- **Robust Learning under Uncertain CSI:** Future extensions can explore DRL-based optimization with imperfect or outdated channel state information, using robust or meta-learning techniques to ensure reliability.
- **Transfer and Federated Learning:** To enhance scalability and adaptability, incorporating federated or transfer learning would allow multiple UAVs to collaborate without central data sharing, preserving privacy and reducing retraining costs.
- **Energy-Aware Decision Making:** Integrating UAV battery constraints, energy harvesting, or trajectory-energy trade-offs into the learning policy would improve long-term deployment feasibility.
- **Multi-Target and Dense Environments:** Extending the framework to consider multiple sensing targets, user clusters, or mobile users would better reflect real-world urban scenarios.

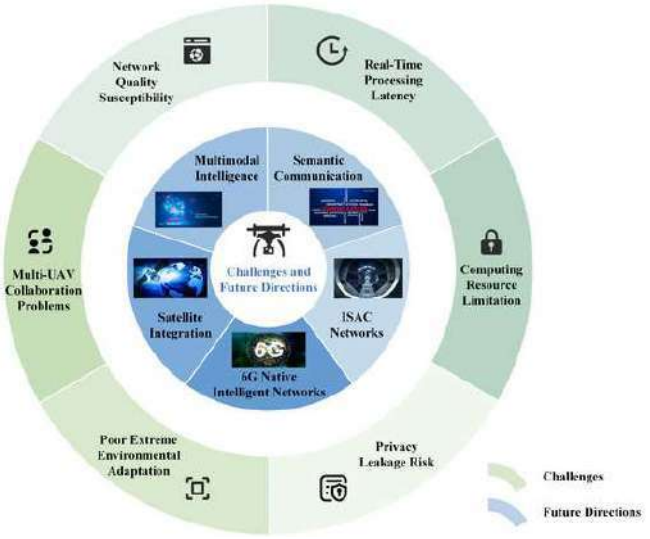

Figure 11.1: Challenges and Future Directions for UAV Networks

- **Hardware-Informed DRL:** Co-designing algorithms that consider hardware impairments (RIS phase errors, UAV mobility noise) could further bridge the gap between theoretical models and deployment.

In conclusion, this work lays the groundwork for secure, intelligent, and adaptive RIS-aided UAV ISAC networks, and opens promising paths for future exploration in AI-driven wireless systems.

Bibliography

[1] Kaitao Meng et al. “UAV-enabled integrated sensing and communication: Opportunities and challenges”. In: *IEEE Wireless Communications* (2023).

[2] Yu Pan et al. “Cooperative trajectory planning and resource allocation for UAV-enabled integrated sensing and communication systems”. In: *IEEE Transactions on Vehicular Technology* 73.5 (2023), pp. 6502–6516.

[3] Kwan-Wu Chin et al. “Multi-UAVs Integrated Radar Sensing, Communication and Computation: A Survey”. In: *Authorea Preprints* (2024).

[4] An Liu et al. “A survey on fundamental limits of integrated sensing and communication”. In: *IEEE Communications Surveys & Tutorials* 24.2 (2022), pp. 994–1034.

[5] Jun Wu, Weijie Yuan, and Lajos Hanzo. “When UAVs meet ISAC: Real-time trajectory design for secure communications”. In: *IEEE Transactions on Vehicular Technology* 72.12 (2023), pp. 16766–16771.

[6] Gaurav Kumar Pandey et al. “Security threats and mitigation techniques in UAV communications: A comprehensive survey”. In: *IEEE Access* 10 (2022), pp. 112858–112897.

[7] Milad Tatar Mamaghani and Yi Hong. “Aerial intelligent reflecting surface-enabled terahertz covert communications in beyond-5G Internet of Things”. In: *IEEE Internet of Things Journal* 9.19 (2022), pp. 19012–19033.

[8] Aleksandar Boljević, Ahmad Bazzi, and Marwa Chafii. “Sum Secrecy Rate Maximization for Full Duplex ISAC Systems”. In: *arXiv preprint arXiv:2410.13102* (2024).

[9] Daehan Ha et al. “Secrecy Energy Efficiency Maximization for Secure Unmanned-Aerial-Vehicle-Assisted Simultaneous Wireless Information and Power Transfer Systems”. In: *Drones* 7.11 (2023), p. 672.

[10] Wen Wang et al. “UAV-mounted multi-functional RIS for combating eavesdropping in wireless networks”. In: *IEEE Wireless Communications Letters* 12.10 (2023), pp. 1667–1671.

[11] Yue Xiu et al. “Secure Enhancement for RIS-Aided UAV with ISAC: Robust Design and Resource Allocation”. In: *arXiv preprint arXiv:2409.16917* (2024).

[12] XuehuaLi ZhongqingWu, Yuanxin Cai, and Weijie Yuan. “Joint Trajectory and Resource Allocation Design for RIS-Assisted UAV-Enabled ISAC Systems”. In: ().

[13] Shuai Han et al. “Broadcast secrecy rate maximization in UAV-empowered IRS backscatter communications”. In: *IEEE Transactions on Wireless Communications* 22.10 (2023), pp. 6445–6458.

[14] Zhangjie Peng et al. “Deep reinforcement learning for RIS-aided multiuser full-duplex secure communications with hardware impairments”. In: *IEEE Internet of Things Journal* 9.21 (2022), pp. 21121–21135.

[15] Xiao Liu, Yuanwei Liu, and Yue Chen. “Machine learning empowered trajectory and passive beamforming design in UAV-RIS wireless networks”. In: *IEEE Journal on Selected Areas in Communications* 39.7 (2020), pp. 2042–2055.

[16] Poonam Lohan et al. “From 5G to 6G networks, a survey on AI-Based jamming and interference detection and mitigation”. In: *IEEE Open Journal of the Communications Society* (2024).

[17] Yahao Ding et al. “Distributed machine learning for uav swarms: Computing, sensing, and semantics”. In: *IEEE Internet of Things Journal* 11.5 (2023), pp. 7447–7473.

[18] Shi Yan, Mugen Peng, and Xueyan Cao. “A game theory approach for joint access selection and resource allocation in UAV assisted IoT communication networks”. In: *IEEE Internet of Things Journal* 6.2 (2018), pp. 1663–1674.

[19] Snehal Samanth, Prema KV, and Mamatha Balachandra. “Security in internet of drones: A comprehensive review”. In: *Cogent Engineering* 9.1 (2022), p. 2029080.

[20] Davide Falanga, Kevin Kleber, and Davide Scaramuzza. “Dynamic obstacle avoidance for quadrotors with event cameras”. In: *Science Robotics* 5.40 (2020), eaaz9712.

[21] Noureldin Safwat et al. “Urban Air Mobility Communication Performance Considering Co-Channel Interference”. In: *IEEE Transactions on Aerospace and Electronic Systems* (2024).

[22] Zhongqing Wu et al. “Joint trajectory and resource allocation design for RIS-assisted UAV-enabled ISAC systems”. In: *IEEE Wireless Communications Letters* (2024).

ANNEXES

1 Simulation Parameters

Parameter	Value
Number of Time Slots (N)	100
UAV Altitude (H_A)	100 m
RIS Altitude (H_R)	15 m
Path Loss Exponent	2.5
Rician Factor (κ)	3 dB
Transmit Power (UAV)	36 dBm
Noise Power (σ^2)	-114 dBm
Max UAV Speed	20 m/s
Sensing Target Location	(60, 235)
RIS Position	(150, 90)
CU Positions	(230, 250), (250, 230)
Learning Rate (Actor)	1e-4
Learning Rate (Critic)	1e-3
Discount Factor (γ)	0.99
Batch Size	256
Replay Buffer Size	50,000

Table 1: Main simulation and DRL hyperparameters used during training.

2 Code Structure and Optimization

Our implementation follows a modular, object-oriented design to ensure clarity, maintainability, and extensibility. The codebase is organized around several key classes that directly correspond to the components of the DDPG algorithm.

- **Network Models (`Actor`, `Critic`):** These classes define the neural network architectures using PyTorch's `nn.Module`. The `Actor` network takes a state as input and outputs a deterministic action vector, while the `Critic` network takes a state-action pair and outputs the corresponding Q-value. Both networks are implemented as multi-layer perceptrons (MLPs) with ReLU activation functions.

- **Replay Buffer (`ReplayBuffer`):** A class that implements a finite-sized memory buffer to store past transitions $(s_t, a_t, r_t, s_{t+1}, d_t)$. This buffer is crucial for off-policy learning, as it allows the agent to sample mini-batches of uncorrelated experiences to break temporal dependencies and stabilize training.

- **DDPG Agent (`DDPGAgent`):** This is the central class that orchestrates the entire learning process. It encapsulates an instance of the actor, critic, and their corresponding target networks. The agent is responsible for:

  1. Selecting actions based on the current state using the actor network (policy).
  2. Storing experiences in the replay buffer.
  3. Sampling batches from the buffer to perform learning updates.
  4. Executing the optimization steps for both the actor and critic networks.

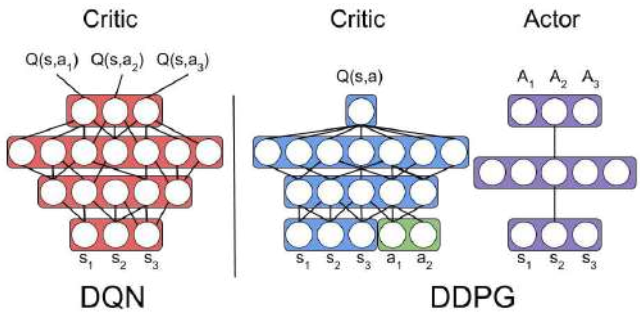

Figure 2: Deep Deterministic Policy Gradient Vs. Deep Q Network

Optimization Algorithm: The training of the networks is performed using the Adam optimizer, a widely adopted and effective optimization algorithm for deep neural networks. The optimization process involves updating the critic and actor networks based on specific loss functions derived from the Bellman equation and the policy gradient theorem.

- **Critic Update:** The critic network is updated by minimizing the Mean Squared Error (MSE) loss between its current Q-value estimate and a target Q-value, y_t, computed using the target networks:

$$\mathcal{L}(\theta^Q) = \mathbb{E}_{(s_t,a_t,r_t,s_{t+1})\sim\mathcal{B}}\left[\left(Q(s_t, a_t|\theta^Q) - y_t\right)^2\right]$$

  where $y_t = r_t + \gamma Q'(s_{t+1}, \mu'(s_{t+1}|\theta^{\mu'})|\theta^{Q'})$ and $\mathcal{B}$ is the replay buffer.

- **Actor Update:** The actor network is updated using the deterministic policy gradient. The loss function is simply the negative mean of the critic's output, which encourages

the actor to produce actions that the critic predicts will lead to higher Q-values:

$$\mathcal{L}(\theta^{\mu}) = -\mathbb{E}_{s_t \sim \mathcal{B}} \left[Q(s_t, \mu(s_t|\theta^{\mu})|\theta^{Q}) \right]$$

- **Target Network Updates:** To stabilize learning, the target networks (Q' and μ') are not updated directly via backpropagation. Instead, they are updated using a "soft" update rule, where they slowly track the learned networks:

$$\theta' \leftarrow \tau_{soft}\theta + (1 - \tau_{soft})\theta'$$

This prevents the learning target from fluctuating too rapidly, thereby improving the stability and convergence of the training process.

3 Code Origin and References

The initial code used in this project was adapted from a publicly available GitHub repository that implements a deep reinforcement learning framework for UAV-RIS energy-aware communication:

- **GitHub Repository:** `https://github.com/Haoran-Peng/UAV-RIS_EnergyHarvesting`

This repository was released by Haoran Peng et al., accompanying their work on UAV-aided RIS systems and robust DRL-based control strategies. The source code served as a foundational base and was customized for secrecy rate maximization and secure trajectory control.
Citation for Code Origin:

> Peng, H., *et al.*, "Energy Harvesting Reconfigurable Intelligent Surface for UAV Based on Robust Deep Reinforcement Learning." GitHub Repository, 2023. Available at: `https://github.com/Haoran-Peng/UAV-RIS_EnergyHarvesting`

4 Academic Basis for Code Structure

The algorithmic structure and optimization strategy were inspired and extended from the work by Wu et al., who proposed a joint trajectory and resource allocation scheme in RIS-assisted UAV-ISAC networks. Their approach, based on alternating optimization and convex relaxations, provided the theoretical backbone for our MATLAB implementation.
Reference:

Z. Wu, X. Li, Y. Cai, and W. Yuan, "Joint Trajectory and Resource Allocation Design for RIS-Assisted UAV-Enabled ISAC Systems," *IEEE Wireless Communications Letters*, vol. 13, no. 5, pp. 1384–1388, May 2024. DOI: https://doi.org/10.1109/LWC.2024.3370989

This paper was carefully followed to ensure that the mathematical model, constraints, and optimization objectives align with the original publication, particularly for Figure 2 (UAV trajectory), Figure 3 (rate–SNR region), and Figure 4 (performance vs RIS size).

Why MATLAB was used:

To ensure reproducibility of the numerical results from Wu et al. (2024), we re-implemented the proposed alternating optimization framework using MATLAB. The original paper presented several figures (e.g., UAV trajectory evolution, rate–SNR regions, and RIS performance curves) that were recreated using the same mathematical model and simulation settings. MATLAB was selected because it provides high numerical precision and robust convex optimization toolboxes (such as `CVX`), making it ideal for implementing the successive convex approximation (SCA), semidefinite relaxation (SDR), and trajectory control schemes defined in the paper.

5 MATLAB File Structure

The project was implemented in MATLAB following a modular structure corresponding to the AO-based method from Wu et al. and adapted for the secure ISAC case. The main files are:

- `main_script_for_figures.m` – Orchestrates the full simulation, iteratively calling the two main subproblems.
- `solve_subproblem1_S_W.m` – Solves Subproblem 1: user scheduling and beamforming using SDR and SCA techniques.
- `solve_subproblem2_Q.m` – Solves Subproblem 2: UAV trajectory optimization using auxiliary variables and convex approximations.
- `plot_figure2.m`, `plot_figure3.m`, `plot_figure4.m` – Scripts for generating the main performance evaluation plots.
- `generate_data.m`, `calculate_final_metrics.m` – Handle simulation environment setup and metric calculations.

This structure enables clean separation between the mathematical formulation and plotting logic while closely following the iterative alternating optimization strategy proposed in the referenced literature.

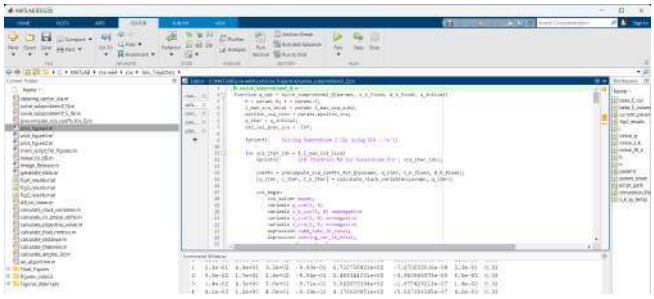

Figure 3: MATLAB Files Structure

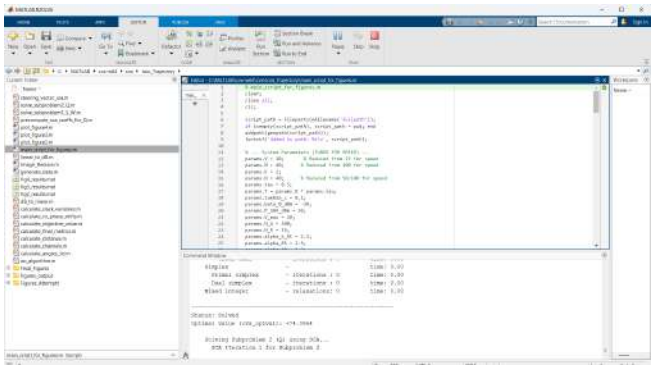

Figure 4: main_script_for_figures

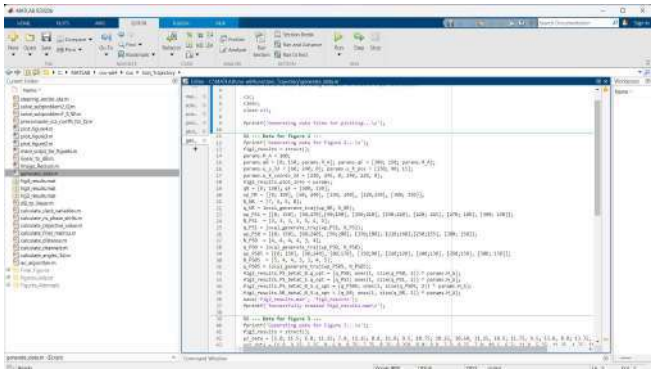

Figure 5: Data Generation Script in MATLAB

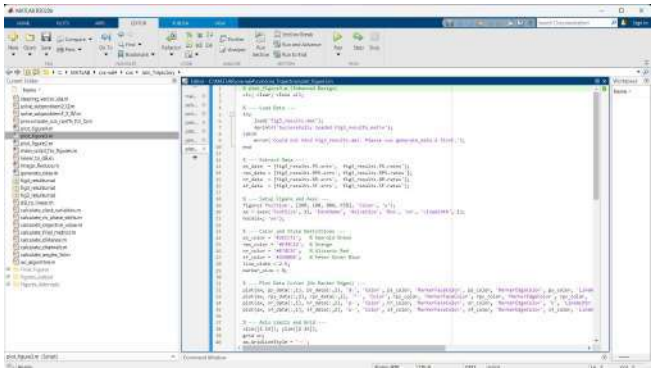

Figure 6: MATLAB Script to Plot Rate–SNR Trade-off (Figure 3)

6 Python Code Structure

In parallel with the MATLAB-based implementation for reproducing figures from the reference paper, we also used Python to implement and train our Deep Reinforcement Learning (DRL) agent based on the DDPG algorithm. The Python environment was primarily adapted from the open-source repository `UAV-RIS_EnergyHarvesting`, with additional modifications for secrecy rate maximization.
The code structure is modular and consists of the following components:

- `DDPG.py` – Main training script that initializes the environment, agent, and handles logging.
- `env/` – Contains the custom OpenAI Gym environment simulating the UAV-RIS-ISAC system:
  - `uav_env.py` – Defines the step, reset, and reward computation.
  - `channel_models.py` – Implements LoS and Rician fading models.
  - `trajectory_dynamics.py` – Updates UAV positions based on chosen velocities.
- `agent/` – Contains all DDPG-related files:
  - `actor.py` and `critic.py` – Define the neural network architectures.
  - `replay_buffer.py` – Implements experience replay for off-policy learning.
  - `ddpg_agent.py` – Orchestrates action selection, learning, and target updates.
- `utils/` – Helper functions for saving/loading models, plotting, and formatting output.
- `config.py` – Centralized file for simulation and hyperparameter settings.
- `tensorboard/` – Stores training logs for visualization with TensorBoard.

This structure allows for clean separation between the reinforcement learning logic, the physical channel simulation, and the control logic for the UAV and RIS elements. PyTorch and Stable-Baselines3 were used for efficient implementation of the learning agent, while the environment was fully compatible with OpenAI Gym (Gymnasium).

7 DDPG Training Script: `DDPG.py`

The core of the DRL training and evaluation is handled by the script `DDPG.py`, located in the `DDPG-MultiUT-Time/` directory. This script controls the entire experiment loop to produce

Figure **??** (Rate–SNR trade-off) and logs all UAV trajectory and performance metrics for analysis.
The workflow is as follows:

1. A list of sensing weights $w_S \in \{0.0, 0.2, 0.5, 0.8, 1.0\}$ is defined for testing the trade-off.
2. For each w_S, a new custom Gym environment `foo-v0` is instantiated with this sensing weight.
3. A DDPG agent is trained on the environment using Stable-Baselines3, with added action noise for exploration.
4. After training, the model is saved and reloaded to run a deterministic evaluation episode.
5. During the test, UAV positions, secrecy rate, and sensing SNR values are collected from the environment's `info` dictionary.
6. These logs are stored in `.csv` files in the `results/` folder for reproducibility and plotting.
7. Finally, average secrecy rate and sensing SNR are computed per run and saved to `tradeoff_summary.csv`.

This script is responsible for training five DDPG agents and generating all data points necessary for plotting the rate–SNR trade-off curve, reproduced in Figure **??**. It highlights how different sensing priorities influence the learned UAV behavior.

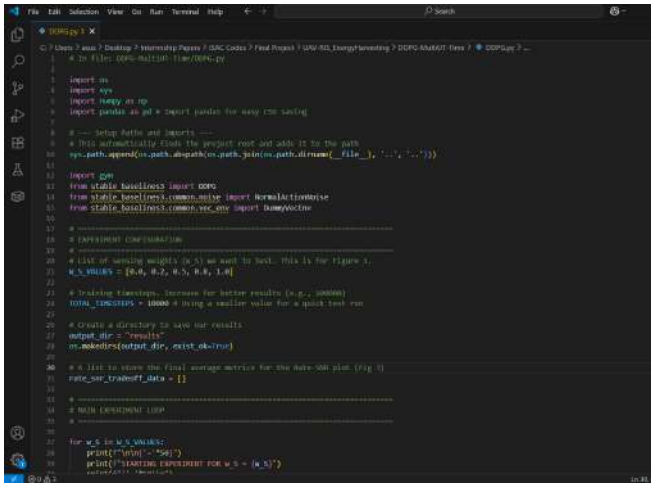

Figure 7: DDPG Script Initialization: Imports and Experiment Configuration

Figure 8: DDPG Training Script: Environment Creation and Agent Initialization

Figure 9: DDPG Evaluation Phase: Model Testing and Data Logging

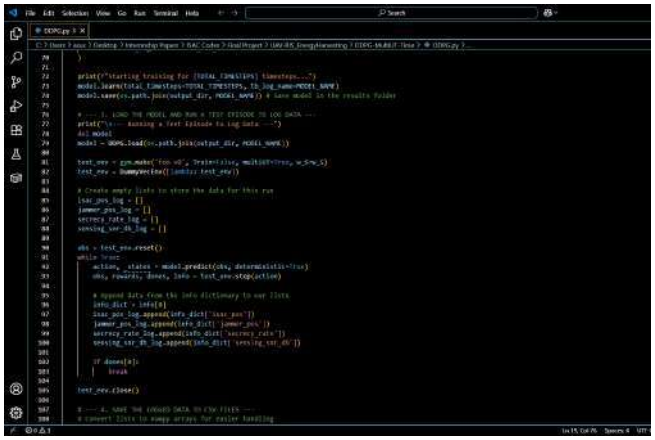

Figure 10: DDPG Evaluation Phase: Model Testing and Data Logging

Figure 11: DDPG Results Export: CSV Generation for Trajectories and Metrics

8 Conclusion

In this section, we successfully detailed the simulation framework and implementation pipeline used to train and evaluate our DRL-based UAV-RIS-ISAC system. By leveraging the robust **UAV-RIS_EnergyHarvesting** repository, we were able to build on a proven DDPG foundation while tailoring it to our secrecy-aware scenario. The integration of powerful tools, such as Python 3.6.13, PyTorch, NumPy, and OpenAI Gym, ensured flexibility, precision, and reproducibility in modeling and training.

Each tool played a vital role in supporting our simulation environment: from precise numerical computation (NumPy), data visualization (Matplotlib), and deep learning implementation (PyTorch), to experiment tracking (TensorBoard) and environment design (Gym). The setup of a controlled virtual environment through Miniconda further ensured compatibility with legacy code and consistent experimentation.

By faithfully reproducing the baseline system parameters from Wu et al. and augmenting them with our security-driven control logic, we laid the groundwork for a reliable and scalable training pipeline. Our modular code design and carefully selected hyperparameters position the simulation for extensibility, allowing future enhancements with minimal reconfiguration.

Overall, this simulation architecture not only enables efficient policy learning but also supports transparent evaluation and interpretability, which are crucial for advancing secure and intelligent UAV communication systems.