

Quantum-Assisted Optimisation of Secret Key Rate Exchange for UAV Communications using a Rician Channel Model for Physical Layer Security

by

Piers Keegan

A thesis submitted to University College Dublin in partial fulfillment
of the requirements for the degree of
MEngSc in Electronic & Computer Engineering
in the College of Engineering & Architecture



School of Electrical & Electronic Engineering

Head of School: Professor Peter Kennedy
Supervisor: Assistant Professor Anshu Mukherjee

August 16, 2025

Declaration of Authorship

Student Name: Piers Keegan

Student Number: 24284403

Project Title: Quantum-Assisted Optimisation of Secret Key Rate Exchange for UAV Communications using a Rician Channel Model for Physical Layer Security

Supervisor: Dr. Anshu Mukherjee

Plagiarism: the unacknowledged inclusion of another person's writings or ideas or formally presented work (including essays, examinations, projects, laboratory presentations). The penalties associated with plagiarism designed to impose sanctions seriousness of University's commitment to academic integrity. Ensure that you have read the University's Briefing for Students on Academic Integrity and Plagiarism and the UCD Statement, Plagiarism Policy and Procedures, (<http://www.ucd.ie/registrar/>)

I declare that all of the following are true:

- 1) I fully understand the definition of plagiarism.
- 2) I have not plagiarised any part of this project and it is my original work.
- 3) All material in this report is my own work except where there is clear acknowledgement and appropriate reference to the work of others.

I acknowledge the contribution of the following post-graduate students / post-doctoral fellows / researchers or technicians to work detailed in this report: N/A.

Signed: *Piers Keegan*

Date: 15th August, 2025

Abstract

This thesis details the novel application of a quantum-classical hybrid deep reinforcement learning algorithm to maximise the secrecy rate of communication links between an unmanned aerial vehicle acting as an aerial base station and a legitimate ground user that is being subjected to eavesdropping. It is shown that the methodology outlined in this thesis successfully solves the joint optimisation problem's objective and its subproblems' objectives. The secrecy rate, data exchange rate, energy efficiency and UAV trajectory are shown to consistently converge as intended across a range of simulation episodes, demonstrating the effectiveness of the techniques used for solving joint optimisation problems applied to UAV-enabled wireless communications.

The thesis details the current state-of-the-art research that has been conducted on the topics of airborne wireless communications networks, optimisation of particular parameters related to aerial networking platforms, quantum computing techniques and how they have been applied to optimisation of airborne wireless communication platforms in a variety of ways. Different network architectures and platforms are considered and explained.

The scenario and simulation environment involving a single aerial base station providing coverage to a set of legitimate ground users with terrestrial eavesdroppers is detailed. A Rician channel model with a communications model employing non-orthogonal multiple accessing is described. The joint optimisation problem and its subproblems are then outlined and mathematically derived.

The system architecture is described at a high level, followed by a more detailed explanation of its constituent subsystems. The software engineering principles and Python implementation of the simulations is explained to convey how the simulations were run and the libraries and frameworks are useful for solving this problem are listed.

From there, the simulation parameters are listed. The results, displaying convergence and maximisation of the secrecy rate and the convergence of its subproblems are shown and discussed. It is noted that for deeper quantum circuits in use within the system, that the performance begins to degrade.

This thesis concludes by outlining the future work that can be carried out to extend this methodology and system architecture. Attention is also paid to work that could be conducted to simulate the system and solve the joint optimisation problem for different scenarios, networking platforms, quantum computing techniques and threat models. The impacts and benefits of the future work are explained in their respective subsections.

Contents

Abstract	ii
Contents	iii
Glossary	vi
List of Figures	viii
1 Introduction	1
2 Literature Review	3
2.1 Problem Area Contextualisation	3
2.2 Aerial Communications	4
2.2.1 UAV-HAPs Communications	4
2.2.2 UAVs as Aerial BSs	5
2.2.3 UAVs as Relays	6
2.2.4 UAVs as UEs	7
2.2.5 Ad-Hoc UAV Networks	8
2.3 Optimisation of UAV Communications	8
2.3.1 Security & Secrecy	9
2.3.2 Energy Efficiency	10
2.3.3 Resource Allocation	11
2.4 Quantum Computing Techniques	12
2.4.1 Quantum Annealing	12
2.4.2 Quantum Embedding & DRL	14

Contents

3 Methodology	19
3.1 System Model & Environment	19
3.1.1 UAV-BS	20
3.1.2 Legitimate GUs	20
3.1.3 Eavesdroppers	20
3.1.4 NOMA Communications Model	20
3.1.5 Rician Channel Model & Environmental Considerations	21
3.2 Mathematical Derivations & Problem Definition	22
3.2.1 Secrecy Rate Optimisation	22
3.2.2 Energy Efficiency Optimisation	24
3.2.3 Optimisation of UAV-GU Data Exchange Rate	26
3.2.4 Optimisation of UAV Trajectory	26
3.3 Algorithms & System Design	27
3.3.1 Design	27
3.3.2 Quantum Data Embedding & Parameter Encoding	28
3.3.3 Classical Computation of Gradient Descent	30
3.3.4 Reward Shaping	31
3.3.5 Memory Experience Replay	31
3.3.6 ANS Generation for Eavesdropping Links	32
3.4 Software Engineering & Implementation	33
3.4.1 Custom Gymnasium Environment	33
3.4.2 Top-Level Program	34
3.4.3 Simulated Quantum Circuits	34
3.5 Simulation Parameters & Chosen Scenario	34
4 Results	36
4.1 Optimised UAV Parameters	36
4.1.1 UAV Trajectory	36

Contents

4.1.2	Energy Efficiency	38
4.1.3	Data Exchange Rate	39
4.1.4	Secrecy Rate	40
4.2	DRL Performance	42
4.2.1	Local Loss	42
4.3	Quantum Actor-Critic Network	42
4.3.1	Impact of the Number of Layers on Performance	43
5	Conclusions	44
5.1	Achieved Objectives	44
5.2	Interpretation of Results	44
5.3	Future Work & Potential Improvements	44
5.3.1	Comparison with Classical DRL Algorithms	44
5.3.2	MARL Support	45
5.3.3	UAV-HAPs Network Architecture	45
5.3.4	Impact of Environmental Conditions	45
5.3.5	Threat Models	46
5.3.6	Alternative Quantum Computing Techniques	46
Bibliography		48
A1 Figures & Data		53
A1.1	Results for Increasing Numbers of Layers	53
A1.1.1	UAV Trajectory	53
A1.1.2	Data Exchange Rate	56

Glossary

4G 4th Generation Wireless Communication. 3

5G 5th Generation Wireless Communication. 2, 3

ANS Additive Noise Signature. iv, 2, 19, 20, 23, 24, 28, 32–34, 42

AQC Adiabatic Quantum Computation. 12

AWGN Additive White Gaussian Noise. 23

BASH Bourne Again Shell. 2, 33

BS Base Station. iii, iv, viii, 2, 4–7, 11, 17, 19–21, 23, 26, 31–33, 35, 36, 43, 45

CQM Constrained Quadratic Model. 13

CSCG Circularly Symmetric Complex Gaussian. 21

DFT Discrete Fourier Transform. 30

DQN Deep Q-Network. 28

DRL Deep Reinforcement Learning. iii, v, 2, 11, 14–16, 18, 27, 28, 31, 32, 35, 42, 44–46

FSPL Free-Space Pathloss. 21, 26

GU Ground User. iv, 1, 2, 4, 5, 7, 9, 11, 13, 14, 17, 19–23, 25–28, 31–33, 35, 36, 39, 40, 44, 45

HAP High-Altitude Platform. iii, v, viii, 3–5, 9–11, 45

HPC High Performance Computing. 33

IP Internet Protocol. 46

IS Importance-Sampling. 32

JIT Just-in-Time. 34

LAP Low-Altitude Platform. 4, 5

LoS Line of Sight. 1, 4, 7, 21, 23, 35, 45

LQ-DRL Layerwise Quantum Deep Reinforcement Learning. viii, 15, 16, 18, 27, 35, 43, 45

LU Legitimate User. viii, ix, 2, 10, 20, 23, 24, 31–37, 39–44, 46

MANET Mobile Ad Hoc Network. 8

MARL Multi-Agent Reinforcement Learning. v, 35, 45

MASR Minimum Average Secrecy Rate. 10, 22

MEC Mobile Edge Computing. viii, 10

MER Memory Experience Replay. 15, 28, 30, 31, 34

MIMO Multiple-Input Multiple-Output. 2, 20

MitM Man in the Middle. 2, 19, 20, 23, 32, 33, 42

NLoS Non-Line of Sight. 21, 23, 35

NOMA Non-Orthogonal Multiple Access. iv, 2, 3, 10, 16, 17, 20, 22, 25, 27

OFDM Orthogonal Frequency Division Multiplexing. 3, 10

OOP Object-Oriented Programming. 33, 45

PER Prioritised Experience Replay. viii, 31–35

PPO Proximal Policy Optimisation. 11

PUD Positive Unique Differences. 30

QoS Quality of Service. 40

QUBO Quadratic Unconstrained Binary Optimisation. 12, 13

SCA Successive Convex Approximation. 10

SIC Successive Interference Cancellation. 21, 23

SNR Signal to Noise Ratio. 20, 23, 24, 26, 42

TD Terminal Device. 5, 9, 10

TD-error Temporal Difference Error. 32

UAV Unmanned Aerial Vehicle. iii–v, viii, 1–11, 13–17, 19–28, 31–41, 43–46, 53

UE User Equipment. iii, viii, 4, 7

List of Figures

1.1	Applications & Technologies of UAVs for Wireless Communication	1
2.1	UAVs Acting as Network Infrastructure in a Disaster Scenario	3
2.2	Illustration of a UAV-HAP Network Model	5
2.3	UAV Acting as a Relay Between 2 BSs	6
2.4	UAV-UEs used in Various Systems	7
2.5	Ad Hoc Configuration of UAV Network	8
2.6	Illustration of a Secure UAV-HAP Network Model	9
2.7	UAV-HAP-Enabled System Model for Secure MEC	10
2.8	Quantum Annealing Based Algorithms for Optimisation of Clustering, Sub-Channel Assignment and Power Allocation	13
2.9	LQ-DRL Framework Proposed by Silviranti et. al (2025)	16
2.10	Ansatz Used for Layerwise Quantum Embedding Proposed by Silviranti et. al (2025)	17
3.1	Simplified Diagram of the System Architecture	19
3.2	System Block Diagram	27
3.3	Single-Layered Actor-Critic Network Ansatz	29
3.4	PER Sum Tree	32
4.1	Maximum, Minimum & Mean Distances to the LU Centroid Across 30 Episodes	36
4.2	Maximum, Minimum & Mean Distances to LU Centroid Per Timestep Across 30 Episodes	37
4.3	UAV Trajectory in Simulated Environment	37
4.4	Maximum, Minimum & Mean Energy Efficiency Per Timestep	38
4.5	Energy Efficiency Across 30 Episodes	38

4.6 Maximum, Minimum & Mean $R_{U,k}(t)$ Values in Mbps from t to T Across 30 Episodes	39
4.7 Maximum, Minimum & Mean $R_{U,k}(t)$ in Mbps Values Across 30 Episodes	40
4.8 Maximum, Minimum & Mean Secrecy Rates for all LUs for each Timestep Across 30 Episodes	41
4.9 Maximum, Minimum & Mean Secrecy Rates for all LUs Across 30 Episodes	41
4.10 Critic Loss for $M = 1$	42
A1.1 Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 1$	53
A1.2 Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 2$	54
A1.3 Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 3$	54
A1.4 Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 4$	55
A1.5 Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 5$	55
A1.6 Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 1$	56
A1.7 Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 2$	57
A1.8 Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 3$	57
A1.9 Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 4$	58
A1.10Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 5$	58

1 Introduction

The use of unmanned aerial vehicles UAVs in communications is a concept that has been researched in recent years but has not been implemented practically for widespread communications and networking. This highlights how recent the concept is of using UAVs in the context of communication networks as an integral part of a network architecture and the need for research to be conducted on this topic for future networks and for specialised applications such as search and rescue operations in regions that are lacking conventional network infrastructure, such as dense urban environments in the wake of natural disasters or remote regions, where it is otherwise not possible to establish a wireless communications network.

UAVs offer many benefits over conventional, terrestrial network infrastructure in scenarios such as those outlined in this thesis for their ability to be deployed for a critical mission where connectivity must be established rapidly and where the network infrastructure can adapt to the environment. Fig. 1.1 displays the technologies involved in UAV communications and has been adapted from [1]. Fig. 1.1 displays the wide range of technologies, enhancements and some applications of UAV-enabled networking, illustrating the many avenues for research into this topic that are being explored. This thesis aims to utilise many of these technologies, in particular for communications and networking, while utilising enhancements such as machine learning techniques, physical layer security and optimisation techniques.

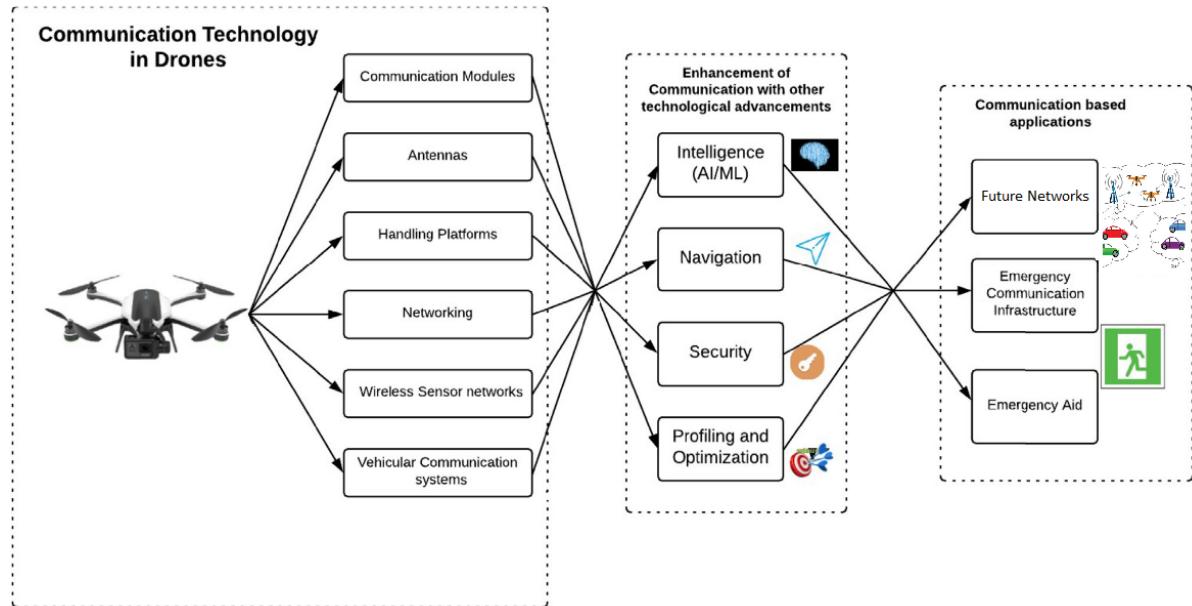


Figure 1.1: Applications & Technologies of UAVs for Wireless Communication

UAVs also can enhance the chance of dynamically establishing Line of Sight (LoS) connections with a Ground User (GU) in ways that are not possible with terrestrial and fixed network

1. Introduction

infrastructure. UAVs can be designed and deployed to serve different roles in wireless communications networks, such as acting as a Base Station (BS) or a relay.

To ensure efficient use of UAVs as aerial BSs is achieved, optimisation techniques and deep reinforcement learning (DRL) are employed to model the constraints on the system that are enforced upon an objective function. DRL is utilised to enable a UAV to adapt to a range of different scenarios rapidly.

Simulations applying DRL to this joint optimisation problem have been created using the Python programming language in which a set of legitimate ground users (GUs), or legitimate users (LUs), are provided with network coverage in a physically secure manner with the use of a UAV-BS while a set of eavesdroppers (referred to as "Eves") who are attempting to conduct Man in the Middle (MitM) attacks are detecting the UAV-LU signal that has been spiked with an Additive Noise Signature (ANS). Bourne Again Shell (BASH) scripting has been used for running the experiments and handling the storage of the output data from the Python simulations for post-processing and analysis. The LUs are able to filter out the ANS, whereas the Eve is not provided with the ANS information.

Due to the large volume of data and the large number of highly non-linear relationships between many different aspects of the environment, quantum computing techniques are utilised to offload some of the computation that otherwise would be quite challenging to compute classically. This is also done to explore the concept of integrating contemporary quantum computing capabilities into a large and complex joint optimisation problem as this is a very new and developing field of study.

This thesis aims to solve a joint optimisation problem for UAV-enabled wireless communications and networking with a particular focus on the physical layer security and secrecy of communications for a set of LUs. To solve this joint optimisation problem, many different aspects of the system must be considered and the subproblems to the secrecy rate optimisation problem must also be optimised. To do this, the data exchange rate, energy efficiency and UAV trajectory are treated as subproblems to the secrecy rate optimisation problem and they are also optimised.

As this is an exploration of future networking solutions, 5th Generation Wireless Communication (5G) and a Multiple-Input Multiple-Output (MIMO) system are considered for the wireless communications, with non-orthogonal multiple accessing (NOMA) methods being used for the multiple-accessing required for the number of GUs considered in the simulation environment.

2 Literature Review

2.1 Problem Area Contextualisation

Due to advancements in the processing power of embedded systems and developments in small-scale, low-cost UAVs, the topic of multi-UAV enabled networks has become a focus of research with aims to practical implementations in recent years. Other relevant developments in networking include the development of communication schemes such as Orthogonal Frequency Division Multiplexing (OFDM), which is a spectrally efficient modulation scheme for 4G as well as NOMA for 5G wireless communication standards.

In regions where there is no network infrastructure, aerial alternatives that make use of UAVs, High-Altitude Platforms (HAPs) or both UAVs and HAPs can be utilised to serve the purpose of traditional network infrastructure. Such environments may include regions where there is no conventional network infrastructure due to the infrastructure being damaged or destroyed as a result of a natural disaster or where the region is too remote with terrain that is too inhospitable to feasibly construct the network infrastructure required for modern connectivity and security demands. This application is illustrated in Fig. 2.1, which has been reproduced from [2].

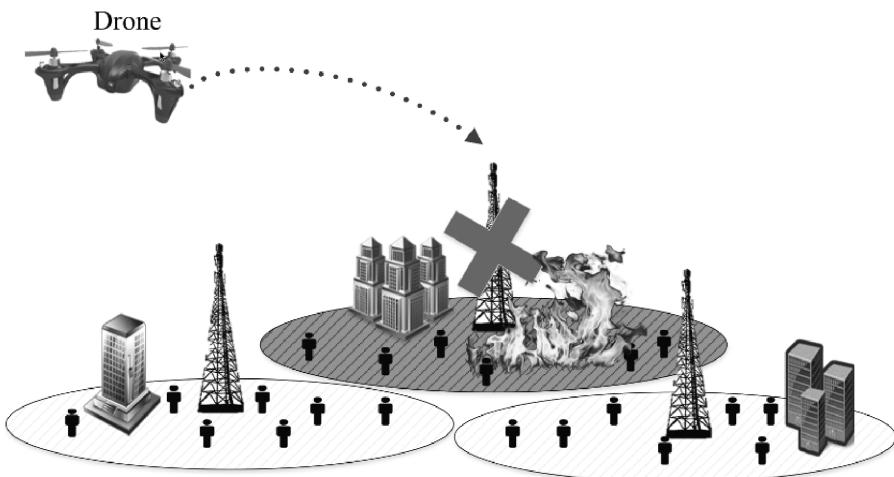


Figure 2.1: UAVs Acting as Network Infrastructure in a Disaster Scenario

Different approaches to this problem involving optimisation techniques will optimise a particular parameter as part of the system and treat others as constraints such that the best system performance with respect to the objective is achieved. This involves the design of algorithms based on closed-form mathematical derivations as well as numerical methods such as machine learning. Different system architectures are explored in the literature on this topic,

which include UAV-HAPs aerial networks [3–5], multi-UAV networks [6, 7] and HAP-enabled networks.

Optical communications exist and can be used effectively for secure wireless communications, however, the focus of this thesis and literature review is on radio communications as radio communications are more commonly used for wireless communications.

Both quantum [7–10] and classical optimisation techniques with applications to UAV-enabled networks have been explored in the literature.

2.2 Aerial Communications

Airborne networks provide some benefits over terrestrial ones. One major benefit of airborne networks involving the use of UAVs for communications is that the networking platforms can reach higher altitudes than standard radio towers and their position can be adjusted dynamically to optimise their Line of Sight (LoS) for air-to-air, air-to-ground and ground-to-air communications links. This is a substantial benefit over ground-based networks in regions where a clear LoS cannot be attained easily, such as a dense urban environment, woodlands, etc., where communications links are otherwise very difficult to establish without the use of airborne communications platforms such as UAVs [11].

UAVs can serve a variety of purposes as communications platforms. UAVs acting as a Low-Altitude Platform (LAP) oftentimes will have shorter missions and are more suitable for short-term, dynamic coverage, whereas HAPs are better utilised for longer-term missions [11].

UAVs can serve a variety of purposes in an airborne communications network, such as acting as BSs, relays, wireless User Equipment (UE) and others [2, 11].

UAVs can be used for jamming signals with a noise signature that is known by the friendly nodes in the network, such as GUs, HAPs, BSs, etc. and is unknown to eavesdroppers. These interfering signals can be used to mask the communication signals through a channel in a way that can be decoded by friendly actors in the network and serve as a challenge for eavesdroppers to be able to interpret the data being transmitted through an airborne network [3, 12].

2.2.1 UAV-HAPs Communications

Both LAP and HAP UAVs can be used together [3, 4, 13, 14] to create a network in which longer-term, more stable network coverage and communications are provided by the HAP portion of the system and the shorter-term, lower altitude communications are handled by the LAP portion of the system.

Some challenges involving this kind of airborne network architecture involve the multiple

degrees of freedom introduced by both HAP and LAP UAVs. This requires the use of optimisation algorithms being developed for various parameters involving both systems such that the entire system is performing as desired, i.e., with the most optimal outcomes depending on the focus of the network. Some examples of aspects of the system that require optimisation are the HAP/LAP deployment, energy efficiency of both HAPs and LAPs, resource allocation and secrecy [3]. An illustration of a UAV-HAP network reproduced from [15] is shown in Fig.

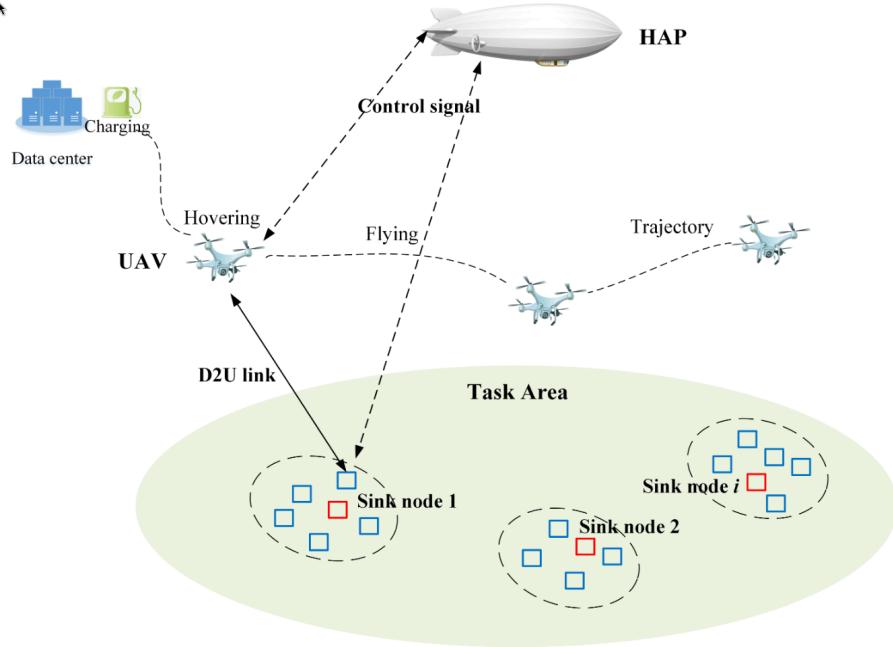


Figure 2.2: Illustration of a UAV-HAP Network Model

2.2, where UAVs and HAPs are integrated as an airborne network over a given task area to form device-to-UAV communications. These device-to-UAV communication links provide a means of communication between terrestrial terminal devices (TDs) or ground users (GUs) in sink nodes.

2.2.2 UAVs as Aerial BSs

In the case of a UAV serving as an aerial BS, the UAV itself is the provider of wireless communication services. For purposes such as this, the UAVs behave as low-altitude platforms (LAPs) and thus have a shorter mission time than HAPs or terrestrial BSs.

For network architectures like this, the deployment will constantly be changing and thus, typically requires some optimisation of the UAV deployment. This constantly changing positioning and deployment model also introduces a need for a channel model that can accurately describe this form of BS [2].

2.2.3 UAVs as Relays

UAVs can be used to extend the range of coverage for a terrestrial or airborne network by serving as network relays between BSs [2, 11, 13].

Fig. 2.3 is a diagram reproduced from [16] depicting a UAV serving as a relay between two terrestrial BSs. UAVs acting as relays for a network can be used to overcome challenges involv-

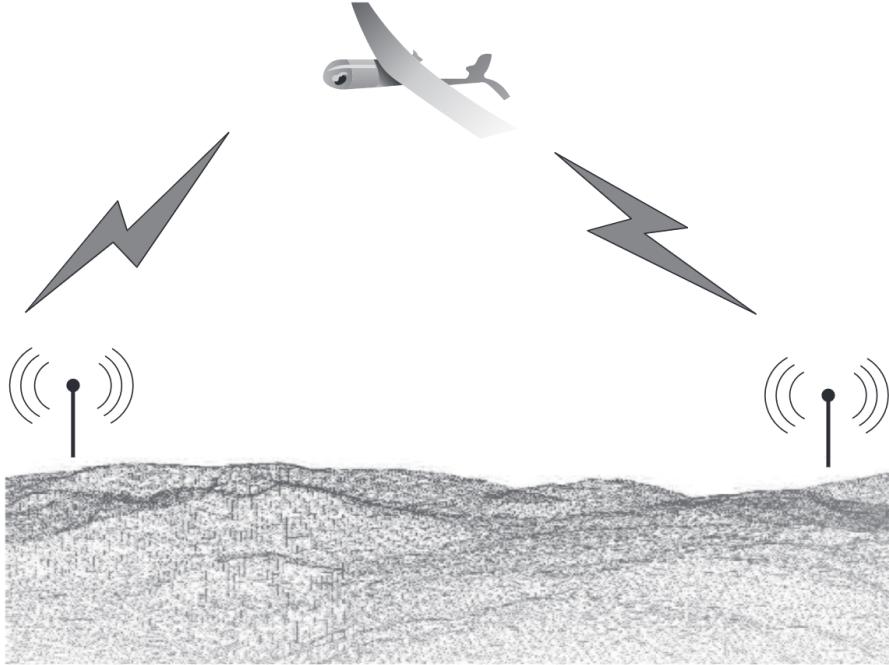


Figure 2.3: UAV Acting as a Relay Between 2 BSs

ing poor LoS between other links in a network due to their ability to overcome environmental obstacles dynamically in ways that terrestrial BSs cannot.

The use of UAVs as relays does present its own set of challenges, however. Some of these include the necessity to adapt existing relaying mechanisms or to devise novel relaying schemes. To ensure proper relaying is achieved, the information related to the control systems of the UAVs, such as their position, altitude, environmental conditions, available resources, etc. must be communicated and known by other UAVs that are directly communicating with it within the network [2]. Air-to-air links in particular have changing propagation environments, which will be changing frequently as the UAV relays are adjusting their positions and altitude as part of any given mission. Multi-hop UAV relays for air-to-air links require the use of dynamic routing algorithms based on the control and communications data from the UAVs in the network, which can be quite challenging to implement on small-scale embedded systems frequently used in small-scale UAVs [2].

2.2.4 UAVs as UEs

UAVs may act as user equipment (UE) to communicate with existing wireless communication networks, such as WiFi or cellular networks. A key challenge involving UAVs as wireless UEs is that existing terrestrial BSs have been optimised and designed to provide the best coverage to GUs rather than airborne UEs. This has been achieved by designing the main antennae lobes on such BSs such that they're pointing downwards towards GUs [2]. Furthermore, GUs and UAV UEs must be differentiated by the network operators to distinguish both classes of user, which has not been a commonly implemented mechanism in conventional networking technologies for UEs.

Some applications of cellular-connected UAV-UEs are presented in Fig. 2.4, which has been adapted from [17].

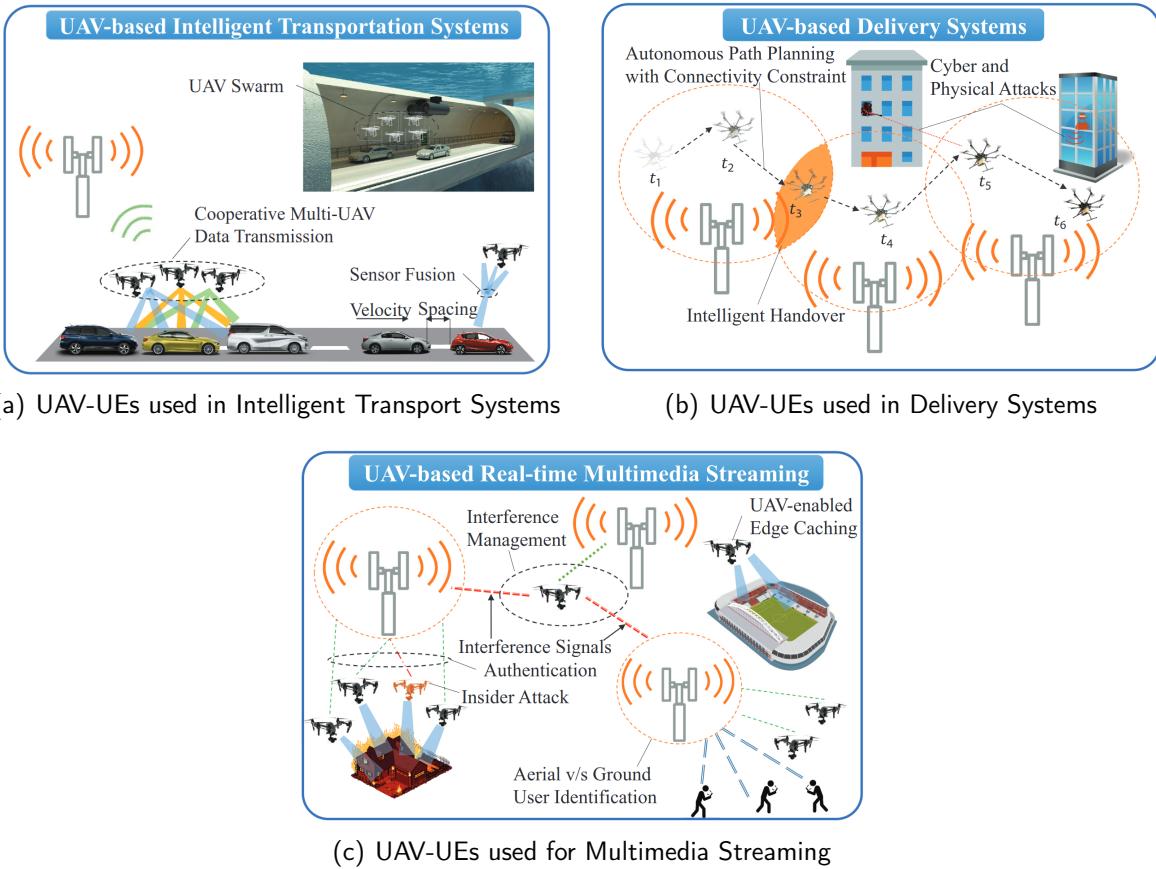


Figure 2.4: UAV-UEs used in Various Systems

Cellular-connected UAVs provide beyond LoS control, low latency, real-time communication, wide levels of coverage and robust security [17], which is advantaged over UAV UEs connected to a network over short-range communication schemes such as WiFi or Bluetooth.

2.2.5 Ad-Hoc UAV Networks

Mobile ad hoc networks (MANETs) are self-organising networks that are formed by mobile nodes [11, 16, 18]. This literature review focuses particularly on UAV-enabled MANETs, however, other forms of MANET can exist, such as ground vehicle-enabled MANETs [16].

UAV-enabled MANETs are multi-hoop networks that can be used for transmitting and receiving information over long distances. Fig. 2.5 depicts an ad-hoc configuration of a UAV-based network architecture reproduced from [16].

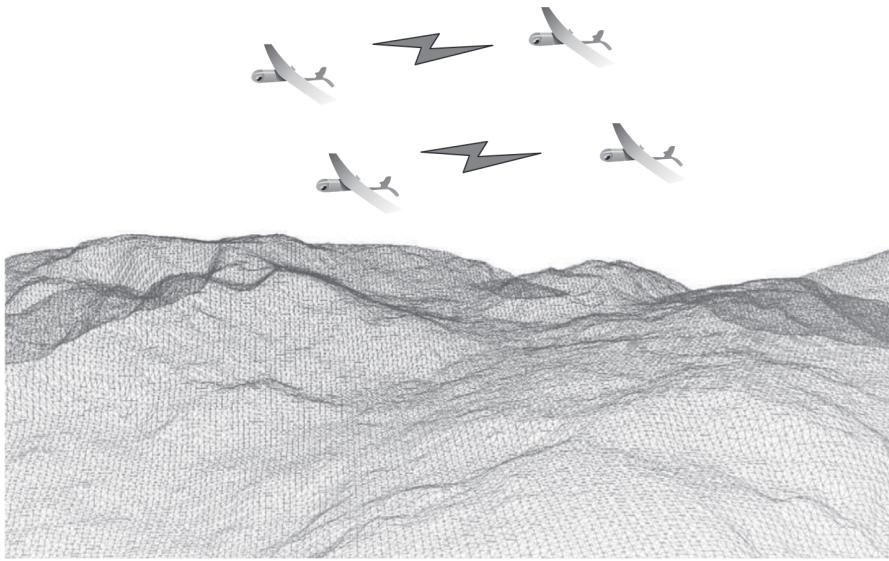


Figure 2.5: Ad Hoc Configuration of UAV Network

MANETs do not require the use of other network infrastructure, such as satellites or centralised servers to support the swarm of vehicles [11], however, it's also expected that there is some assistance from terrestrial control stations as part of the network architecture [16]. Each node in the UAV-enabled MANET acts as a relay, router and terminal.

2.3 Optimisation of UAV Communications

For optimal performance of an airborne network, optimisation techniques have been employed with a range of focuses for optimisation and constraints of a given system. The objective functions are typically mathematically derived and proven in the literature and algorithms are designed to implement them for the communications and control systems involved in airborne networks.

Different studies focus on various aspects of airborne networks, with emphases placed on particular objectives of the system subject to a variety of relevant constraints. Many of these studies are presented with a joint non-convex optimisation problem that cannot be solved efficiently or easily with a standalone algorithm and thus, piecewise approaches are taken to

optimise particular parameters of the system individually without violating the constraints of the overall problem. The main factors for optimisation that have been explored in the literature on this topic include energy efficiency [13, 15], UAV clustering [7, 15], communication secrecy [3, 6, 19], platform deployment [4], UAV trajectory [15], latency [10], resource allocation [4] and scheduling, to name a few.

The following subsections detail a set of focal points for optimisation and the constraints that are considered for aerial communications that have been documented to date.

2.3.1 Security & Secrecy

A key focus of many of these studies is the secrecy of communication and ensuring that the communications are secure between GUs, HAPs and LAPs [3, 6, 12, 14, 19]. In [3], HAPs are deployed for communication with GUs and UAVs are used for detection of eavesdroppers, who are referred to as "Eves" in the paper.

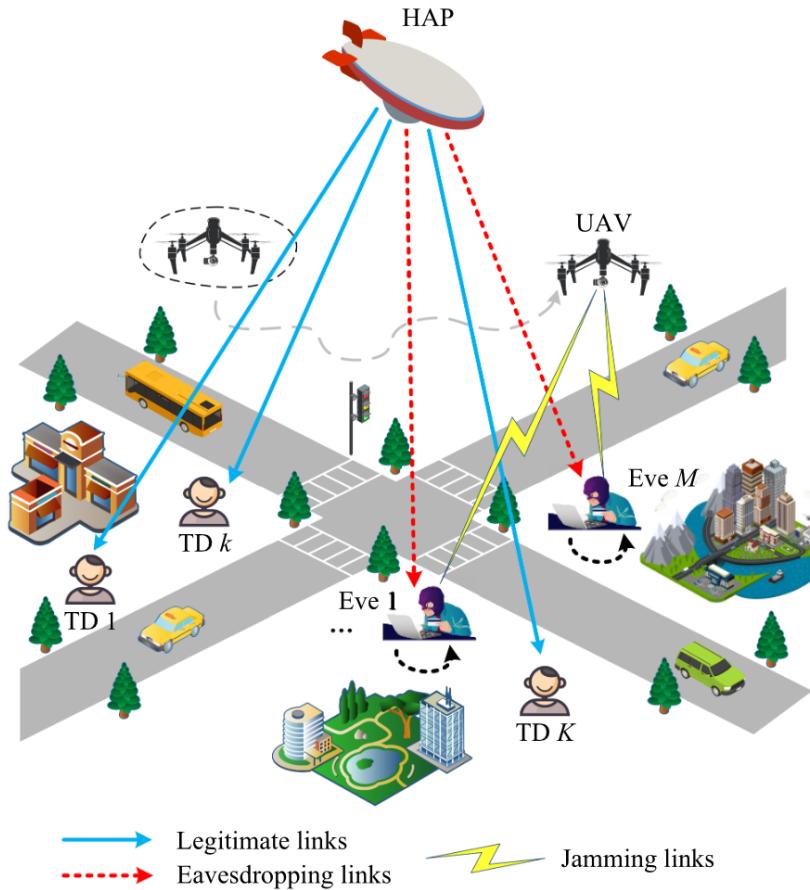


Figure 2.6: Illustration of a Secure UAV-HAP Network Model

An illustration of a secure UAV-HAP network reproduced from [3] is shown in Fig. 2.6, where UAVs are used as jammers to combat eavesdroppers and the HAP is used as the network provider for communication links between terrestrial TDs. The HAP network uses a UAV to

interfere with the signal decoding of multiple mobile eavesdroppers. The UAV's trajectory is dynamically adjusted and optimised based on the known locations of the eavesdroppers.

In this study, the system's orthogonal subcarrier scheduling among TDs, subcarrier power allocation, UAV trajectory and HAP deployment are jointly optimised to maximise the minimum average secrecy rate (MASR), subject to the constraints of UAV mobility and HAP power budget. This optimisation problem is a mixed-integer, non-convex optimisation problem, however, it is solved iteratively using successive convex approximation (SCA) algorithm. This is achieved by alternately optimising the UAV trajectory, power allocation and the subcarrier scheduling until convergence.

Another approach to ensuring secrecy of communication is detailed in [14], where the secrecy considerations are modelled as constraints in the system rather than objectives and the task computation latency, task scheduling and transmit power of legitimate users (LUs) are jointly optimised for secure mobile edge computing (MEC). The system model is shown in Fig. 2.7, which has been reproduced from [14].

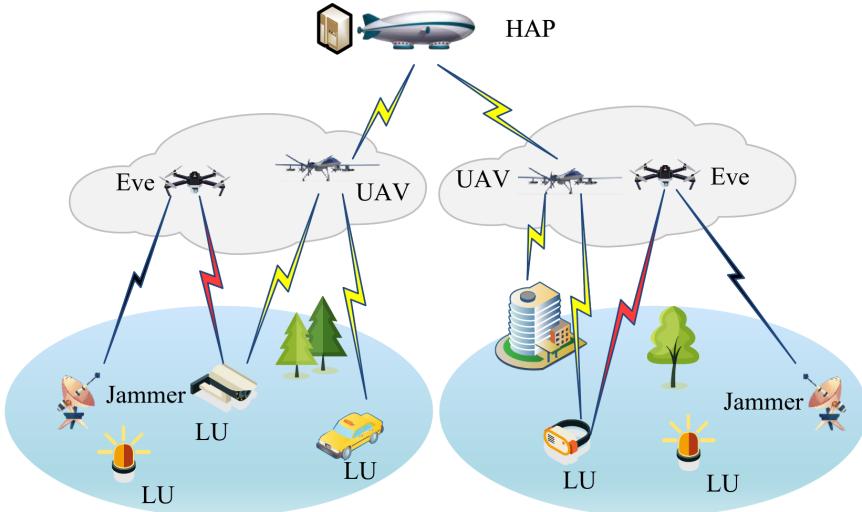


Figure 2.7: UAV-HAP-Enabled System Model for Secure MEC

An iterative approach is also taken for this proposed form of optimisation as in the case of [3], in which the joint non-convex optimisation problem is solved by individually optimising the various parameters such that the minimum secure offloading sum rate is maximised. NOMA is utilised for spectral efficiency over OFDM as it allows multiple users to transmit data over the same resource block, so it supports a greater level of spectral efficiency [14].

2.3.2 Energy Efficiency

Certain studies [8, 13, 15, 20] focus on maximising the energy efficiency of UAVs in airborne networks. The importance of optimising the energy efficiency of the airborne platforms can be viewed as a universal one for the optimisation of any given UAV mission for networking

due to the limited power capacity of LAPs, in particular that typically cannot generate any power while they're operational.

In [15], the trajectory of UAVs in the network is optimised jointly with the resource allocation for the UAVs to maximise the energy efficiency. A dynamic, self-adaptive clustering algorithm based on affinity propagation and a proximal policy optimisation (PPO) based deep reinforcement learning (DRL) algorithm are used to optimise the UAV clustering, trajectory planning and transmit power level of the sensing devices for providing device-to-UAV communication. In this study, HAPs are used as aerial BSs, whereas the UAVs are used for the device links and communication with the HAPs that are providing the networking coverage. The simulation of the algorithm uses a scenario involving network that consists of a single UAV and HAP-BS using the clustering and resource allocation algorithms to maximise the energy efficiency.

The approach taken in [13] involves the optimisation of the energy efficiency using the transmission power, UAV altitude and time allocation of HAPs and UAVs in the network. The architecture considered is a two-hop HAP-UAV-assisted wireless network that is serving a set of GUs.

The energy efficiency is maximised such that it does not violate the constraints of UAV altitude, UAV and HAP transmit power and the total operation time.

Three algorithms are utilised to meet this objective, algorithm 1 handles the global optimal UAV and HAP transmit power, algorithm 2 handles the optimal transmit power allocation for the UAVs and HAPs alternately and algorithm 3 directly handles the energy efficiency using the optimal transmit power and allocation for the UAVs and HAPs, thus algorithm's 3 complexity and efficacy is dependent on the performance and adoption of algorithms 1 and 2 [13].

2.3.3 Resource Allocation

The allocation of resources within the network is a focus for optimisation in airborne networks. In [4], a joint resource allocation problem is formulated and divided into constituent subproblems. The iterative algorithm proposed in [4] is sub-optimal, however, it can solve the resource allocation and HAP deployment optimisation problems efficiently and separately.

The resource allocation algorithm is based on the Gale-Shapely algorithm to produce a convergent solution to the two-sided matching game of the HAP deployment and resource allocation algorithms. Both algorithms are combined into a single algorithm that alternates between solving each of these problems iteratively, thus reducing it from a joint optimisation problem to two smaller subproblems for the algorithm to solve more efficiently.

2.4 Quantum Computing Techniques

In recent years, quantum computers have been developed in an attempt to compute problems that cannot be solved as effectively, or at all, using classical computers. Different approaches to create qubits that leverage different technologies such as ion trap qubits [21], superconducting qubits [22], semiconductor qubits [23] and photonic qubits [24].

Hybrid classical-quantum computer implementations have also been explored as feasible solutions to solving optimisation problems where some of the computation can be offloaded to one or the other means of computation depending on which method is more suitable to a portion of any given problem.

Quantum computing can be utilised to solve complex optimisation problems.

The two leading quantum computing paradigms are gate-based quantum computing and Adiabatic Quantum Computation (AQC) [25]. Gate-based quantum computing involves the application of a sequence of unitary quantum gates to a set of qubits. The states of the qubits collapse into a $|0\rangle$ or $|1\rangle$ state upon measurement.

AQC involves the preparation of a multi-qubit state as the ground state of a Hamiltonian operator. This Hamiltonian then has an adiabatic time evolution applied to it, which changes the Hamiltonian such that its ground state encodes the solution of the optimisation problem [25].

2.4.1 Quantum Annealing

Quantum annealing is a heuristic optimisation algorithm that can be used to solve combinatorial optimisation problems [25]. With the use of quantum annealing, an optimisation problem can be converted into Quadratic Unconstrained Binary Optimisation (QUBO) or Ising models [7]. The expression for the QUBO problem is shown in 2.1.

$$Obj(x, Q) = x^T \cdot Q \cdot x \quad (2.1)$$

Where x is the vector of N binary variables and Q is the symmetric matrix defining interaction terms between the variables, which is an upper diagonal matrix [7, 25]. In the case of the QUBO in [7], the goal is to minimise x in the objective function, so the QUBO is expressed as shown in 2.2.

$$\min_x H_{QUBO}(x) = x^T \cdot Q \cdot x \quad (2.2)$$

The QUBO model can be extended to represent the binary combinatorial optimisation problem with linear and quadratic terms and, in this case, an $N \times N$ binary vector in 2 dimensions is

considered. This model can be expressed as 2.3.

$$H_{QUBO}(x) = \sum_{i=1}^N Q_{ii}x_i + \sum_{i=1}^N \sum_{j=1}^N Q_{ij}x_i x_j \quad (2.3)$$

In [7], two quantum annealing-based algorithms are used for the optimisation of UAV clustering and for sub-channel assignment and power allocation. This algorithm is described pictorially in Fig. 2.8, which has been reproduced from [7].

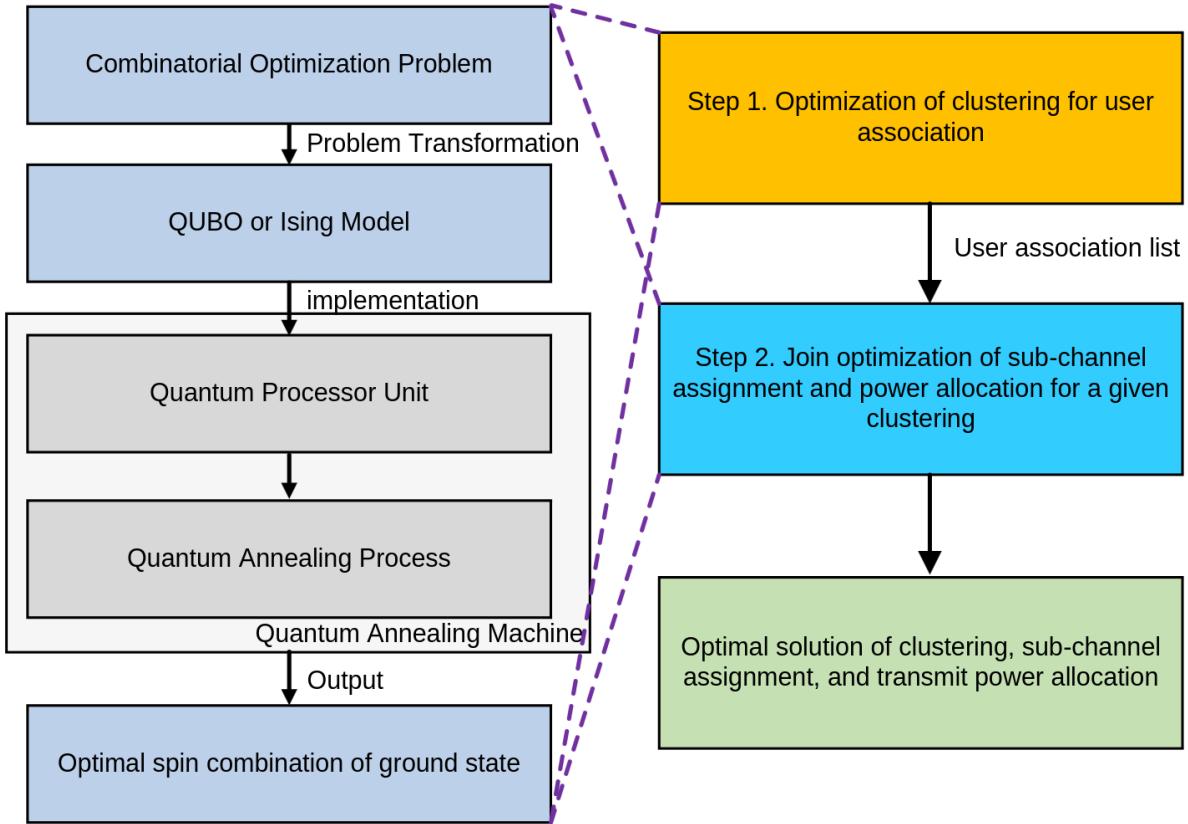


Figure 2.8: Quantum Annealing Based Algorithms for Optimisation of Clustering, Sub-Channel Assignment and Power Allocation

In [7], a quantum annealing-based clustering algorithm with a constrained quadratic model (CQM) is used to optimise a binary variable that indicates if a UAV is associated with a GU or not. The algorithm takes the locations of the GUs and UAVs and builds the CQM object. It then maps the cost function for optimising the UAV and GU clustering to a QUBO Hamiltonian that is energy-based for use with D-Wave's quantum annealing machine, runs the CQM sampler and then returns the optimal spin combination for the association of GUs with UAVs for clustering. This algorithm provides the clustering configuration for the UAVs for the set of GUs. The next algorithm handles the joint optimisation of the power allocation and sub-channel assignment for this clustering configuration that has been determined.

The proposed quantum annealing based algorithm using a D-Wave hybrid QUBO solver out-

performed classical K-means, simulated annealing [26] and steepest descent [27] algorithms in terms of maximisation of the sum rate, measured in Mbps. This increase in the quantum annealing based algorithms' performance also scaled better than the pre-existing algorithms with increases in the number of GUs. The simulation time was also faster and remained low for increasing numbers of UAVs where the simulation time of the other algorithms began to increase for increasing numbers of UAVs with the proposed scenarios presented in the paper.

The optimisation algorithms that were compared against the proposed algorithm were not specially designed for UAV clustering or airborne communications and were published in the 1980s and 1990s. While this does not infringe on the validity of these methodologies, comparisons against newer algorithms that have been specifically designed for this problem could reflect its efficacy more accurately.

2.4.2 Quantum Embedding & DRL

Quantum embedding is a process in which classical data can be stored and encoded into quantum states by passing a vector of data through a quantum circuit and storing the data upon measurement once this state has undergone time evolution through the circuit. The encoding operation acting on an input vector of classical data x with N elements can be mapped to a quantum state such that the input data is mapped to a quantum state as shown in 2.4.

$$S_x : \left\{ x_n \right\}_{n=1}^N \rightarrow \left\{ |\psi\rangle \right\}_{n=1}^N \quad (2.4)$$

Where $|\psi\rangle$ denotes the n^{th} quantum state [8]. Different encoding techniques can be used for quantum embedding [28, 29]. Basis encoding directly maps classical bits to qubits as shown in 2.5.

$$|x\rangle = |b_1\rangle \otimes |b_2\rangle \dots |b_n\rangle \quad (2.5)$$

Where $|b_i\rangle$ are the binary digits of x . Basis encoding requires many qubits to encode data as a result of it being a one-to-one mapping scheme, thus, it scales poorly for increasing dataset sizes.

Superposition encoding is a technique that involves encoding classical data as a superposition of multiple states at once and it represents a quantum state as a linear combination of basis states, illustrated by the example shown by 2.6.

$$|101\rangle \rightarrow \alpha |000\rangle + \beta |010\rangle + \gamma |001\rangle \quad (2.6)$$

Angle encoding encodes classical data in the relative phase between different basis states with the use of phase rotation gates, i.e., $R_X(\theta)$, $R_Y(\theta)$ and $R_Z(\theta)$. This encoding scheme requires n qubits for n data points being encoded into those qubits, which can lead to deep quantum

circuits for large values of n .

Amplitude encoding involves encoding classical information into the amplitudes of a quantum state and it can be expressed mathematically as shown in 2.7.

$$|x\rangle = \frac{x_1}{\sqrt{\sum_{i=1}^n x_i^2}} |0\rangle + \frac{x_2}{\sqrt{\sum_{i=1}^n x_i^2}} |1\rangle + \cdots + \frac{x_n}{\sqrt{\sum_{i=1}^n x_i^2}} |n-1\rangle \quad (2.7)$$

Amplitude encoding requires a smaller number of qubits compared to the other major encoding schemes, however, it also requires a novel preparation protocol and the use of quantum tomography to determine what the probability amplitudes are as a result of the encoding process [28, 29].

Generally, several layers of quantum encoding are required for the quantum embedding process. Different approaches can be taken to obtain the classical values after the embedding process in which all of the data is obtained once all of the quantum embedding layers have encoded all of the classical information into quantum states, however, in [8], measurements are taken after each layer to employ local loss training.

DRL is a branch of machine learning that combines the concepts of deep learning and reinforcement learning and this integration enables the agents interacting with the environment in DRL algorithms to be able to handle high-dimensional input spaces, such as the environmental, secrecy and energy considerations of a network of UAVs.

Generally, DRL algorithms work by using a learning agent to interact with a training environment by taking an action based on the observation of the current state. It then moves onto the next state. The optimal action is taken at discrete intervals under a particular policy to ensure the maximum reward for the network is attained and higher rewards are obtained over time. An episode refers to the single-chain agent interaction and the interaction experience from each episode is stored in a memory experience replay, which is used to update the policy until it converges into the optimal policy [8].

The memory experience replay (MER), shown in the LQ-DRL framework pictured in Fig. 2.9 stores the following parameters: action a taken by agent, observed state s of the system at time t , the reward r for the action taken and s' refers to the next state at time $t+1$.

The layerwise quantum deep reinforcement learning (LQ-DRL) framework refers to the integration of the layerwise quantum embedding with local loss training with the DRL-based actor-critic network.

The proposed framework used in [8] for the LQ-DRL methodology is displayed in Fig. 2.9.

It can be seen in Fig. 2.9 that a hybrid approach is taken for computing the optimal parameters for the DRL algorithm, i.e., part of the computation is performed using quantum gates and the other part is performed using classical computing methods, however, both portions of the

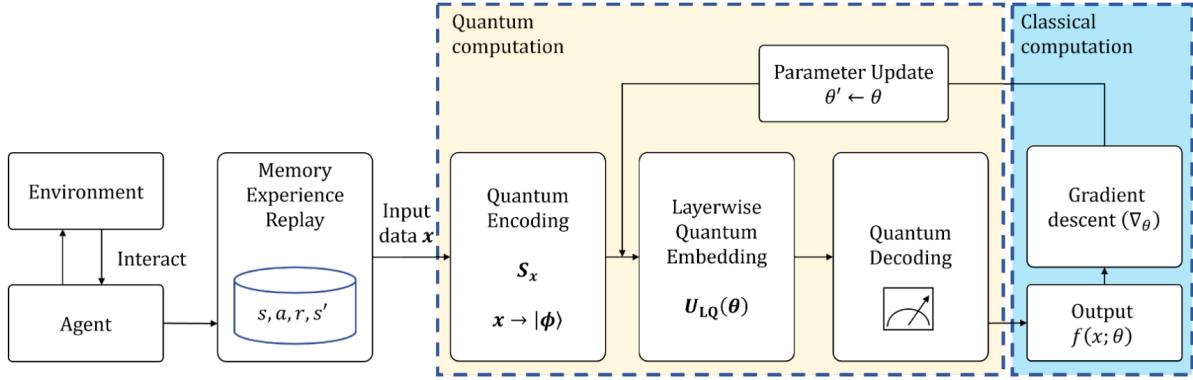


Figure 2.9: LQ-DRL Framework Proposed by Silviriante et. al (2025)

overall system rely on one another to perform DRL.

The study involved the use of a DRL that employs a deep neural network to enable the agent to learn in a high-dimensional, continuous space and not a discrete one, which is typical for DRL algorithms.

The optimal action policy is selected by the actor network and the critic network evaluates the quality of this selected policy by referring to the maximum Q-values at that given layer. The Q-values represent a value given a particular action and state at a given time. As they are directly related to the rewards given for particular actions, they must be updated over time as opposed to being kept the same from iteration to iteration [30].

The use of quantum embedding and measurement for each layer was performed for encoding all of the data into quantum states with the use of an ansatz for quantum embedding, i.e. a quantum circuit. There is a decoding operator that acts on the quantum state for measurement to collapse the quantum information into classical information, which then provides an input for a gradient descent optimisation algorithm that updates parameters for the next layer to be embedded and subsequently decoded. The decoding operation is then used to calculate the value of the local loss for each iteration of this process and each measurement operation is performed K_{shot} times. The local loss is a measure of the difference between the decoded output and the desired output averaged over N_{data} times, where N_{data} is the size of the encoded data.

The proposed ansatz for the layerwise quantum embedding in this study is shown in Fig. 2.10, which has been reproduced from [8].

The layerwise quantum embedding process is performed for the actor and critic networks.

The employed LQ-DRL scheme was used to jointly optimise the UAV trajectory planning, transmit power allocation and NOMA user grouping.

The scenario considered for evaluating the performance of the algorithm involved a single UAV

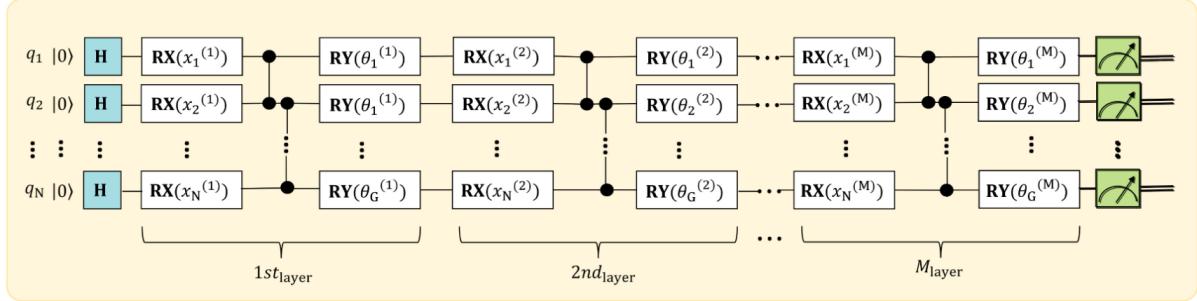


Figure 2.10: Ansatz Used for Layerwise Quantum Embedding Proposed by Silvianti et. al (2025)

in three-dimensional space, with full power E_{max} , acting as an aerial BS to provide downlink coverage serving K GUs that were distributed randomly over the horizontal plane, i.e. in two dimensions and moving at constant speeds. The authors formulated energy consumption models, NOMA channel grouping, transmission power models, a noise and interference model and subsequently formulated an objective function that aimed to maximise the energy efficiency of the UAV network subject to the following constraints:

- The total maximum transmit power of the UAV must not be exceeded by the total allocated power for the GUs in a NOMA group.
- The range of the allocated power coefficient for each user must not be exceeded and must fall between 0 and 1 .
- Higher transmission power must be allocated to GUs with a lower channel gain.
- The minimum threshold data rate must be exceeded by the achievable data rate of any GU within a given group at a particular time
- The UAV must remain within its maximum allowed co-ordinates in Cartesian co-ordinates.

The authors formulated a state space for the UAV in which the location, remaining energy to consume and the states of the GUs are considered with $(2K + 4)$ state space dimensions, where K denotes the number of GUs.

An action space was also formulated based on the objective problem of jointly optimising the UAV trajectory, NOMA user grouping and power allocation. Action a considered the following actions:

- **UAV Trajectory** controlled by the velocity of the UAV and related to it's maneuvre.
- **NOMA User Grouping** to address every possible GU clustering outcome and its objective is reward maximisation.
- **Dynamic Power Allocation** which orders the transmit power allocations based on the highest to lowest channel gains for all of the GUs.

The action space has 5 dimensions in total.

The reward r is determined based on the energy efficiency $\eta(t)$ of a given episode and this is the value that's assigned as a reward if the energy efficiency optimisation is met, otherwise a value of 0 is assigned for reward r .

The LQ-DRL algorithm used in [8] demonstrated that the LQ-DRL algorithm had higher effective dimensionality compared to a classical DRL approach. The layer loss was also shown to decrease in magnitude for each layer. The energy efficiency, critic and actor loss also decreased substantially, with improvements being demonstrated to occur for each layer.

3 Methodology

3.1 System Model & Environment

The proposed network architecture in use for this problem consists of a UAV-BS for providing network coverage to a set of legitimate GUs, in which physical layer security is upheld with the use of an ANS to mask the wireless signal from eavesdroppers attempting to perform MitM attacks on legitimate communications links. A diagrammatic representation of the system architecture is shown in Fig. 3.1, with eavesdroppers referred to as "Eves" and the UAV-BS shown to provide network coverage for a set of \mathcal{K} legitimate GUs.

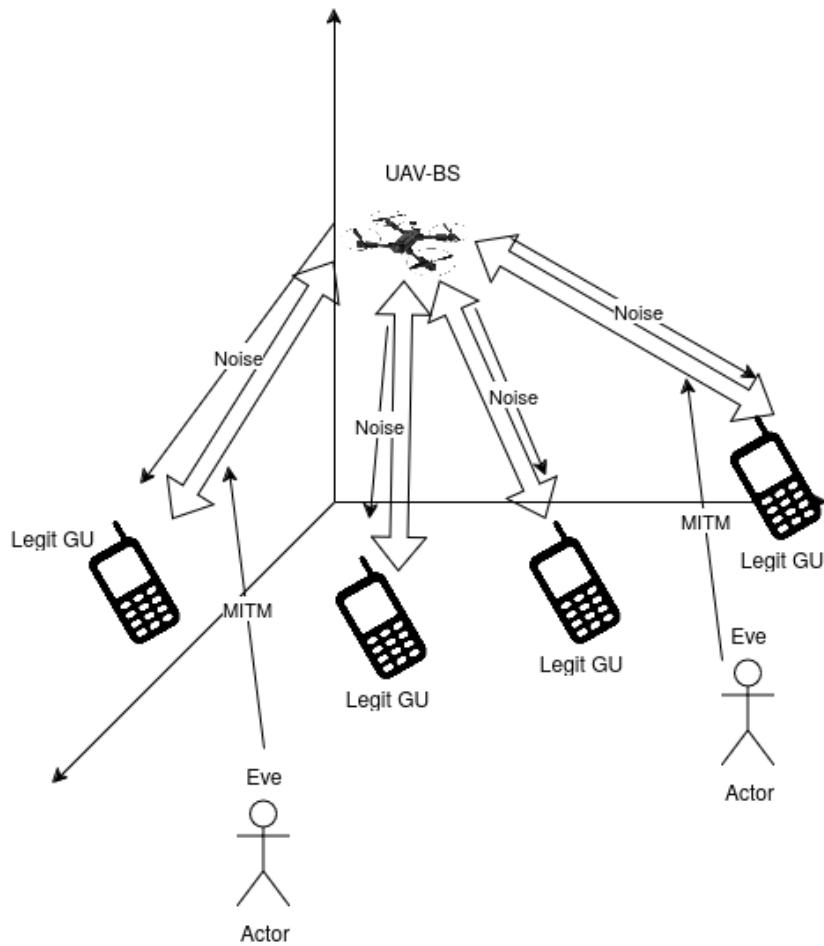


Figure 3.1: Simplified Diagram of the System Architecture

Within the system architecture, there is a set \mathcal{K} of K legitimate GUs, a set \mathcal{M} of M Eves and a set \mathcal{X} of X UAV-BSs.

3.1.1 UAV-BS

The purpose of the UAV-BSs are to establish and maintain UAV-GU communication links with legitimate GUs while ensuring those communications are physically secure and maintaining an acceptable secrecy rate. They provide network coverage to the LUs and their trajectory, data rate of exchange, energy efficiency and secrecy rate must be optimised to ensure the delivery of adequately secure and energy-efficient communication links for legitimate GUs. The mobility model for the UAV is shown in 3.1 [8].

$$v_{UAV}(t) = \delta V_{max}, \quad 0 < \delta \leq 1 \quad (3.1)$$

In this model, the UAV-BSs can only be connected to the legitimate GUs and must generate an ANS to combat eavesdropping from Eves based on their observed positions.

3.1.2 Legitimate GUs

The LUs within the system model are the GUs that the UAV-BS wishes to communicate with secretly. This involves the LU having to receive a noisy signal that contains the desired modulated data for the LU with an ANS that the LU is provided with information about to enable the LU's receiver to filter the ANS from the signal.

3.1.3 Eavesdroppers

The Eves within the system are eavesdroppers that wish to perform MitM attacks by detecting and storing the UAV-LU communications data. With the Eves in the environment, they have an eavesdropping rate $R_{E,k}(t), \forall k$ for each of the K LUs which describes how much data they are managing to intercept and store between the UAV-BS and any of the LUs. The UAV-BS must estimate the worst-possible value for $R_{E,k}(t)$ to inject ANS into the modulated waveform that minimises the SNR for all $E \in \mathcal{M}$ Eves.

3.1.4 NOMA Communications Model

A NOMA communication scheme has been chosen as it offers potentially higher data exchange rates and all GUs can be communicated with by the UAV-BS using a MIMO antenna array on the UAV without the need for division of frequency or time for multiple GUs to communicate with the UAV. The transmitted signal using NOMA is shown in 3.2 and the received signal is shown in 3.3 [8, 31].

$$x_n = \sum_{k=1}^K \sqrt{P_{k,n}^{Tx}} s_k \quad (3.2)$$

$$y_n = h_{k,n}(t)x_n + I_{k,n} \quad (3.3)$$

In 3.2, $P_{k,n}^{Tx}$ refers to the transmitted signal power and s_k refers to the modulated symbols being transmitted. In 3.3, $h_{k,n}$ is the channel gain and $I_{k,n}$ is the interference in the channel caused by the other GUs being communicated to be the UAV-BS. $I_{k,n}$ in 3.3 is removed from the received signal with the use of Successive Interference Cancellation (SIC).

3.1.5 Rician Channel Model & Environmental Considerations

The Rician channel model describes a communications channel in which the propagating waveform is made up of a dominant LoS path between the receiver and transmitter as well as non-dominant paths. The channel model can be characterised with the use of Rician K-factors and Ω -factors [32], shown in 3.5 and 3.6. Rician K-factors describe the ratio of the dominant signal power to the non-dominant signal power in a communication channel. The Ω -factors act as a scaling factor for the Rician distribution and is the sum of the dominant and scattered paths.

$$v(t) = C \cdot \cos(\omega_c t) + \sum_{n=1}^N r_n \cos(\omega_c t + f_n) \quad (3.4)$$

$$K = \frac{v^2}{2\sigma^2} \quad (3.5)$$

$$\Omega = v^2 + 2\sigma^2 \quad (3.6)$$

The K-factors are computed dynamically with the use of channel gain coefficients that are fixed parameters of the channel, A_1 and A_2 . The elevation angle, θ between the UAV-BS and a given GU is given by 3.7 [33].

$$\theta = \sin^{-1} \left(\frac{z_U}{d_U} \right) \quad (3.7)$$

$$K = A_1 e^{A_2 \theta} \quad (3.8)$$

$$g_U[k] = \sqrt{\frac{K}{1+K}} g + \sqrt{\frac{1}{K+1}} \tilde{g} \quad (3.9)$$

$$\zeta_{k,n}^{fs} = 20 \log_{10}(d_U[k]) + 20 \log_{10}(f_c) + 20 \log_{10} \left(\frac{4\pi}{c} \right) \quad (3.10)$$

$$h_U[k] = g_U[k] d_U[k]^{-\zeta_{k,n}} \quad (3.11)$$

In 3.9, g is the deterministic LoS component and \tilde{g} is the Non-Line of Sight (NLoS) random scattered components, which is represented as a Circularly Symmetric Complex Gaussian (CSCG). The Free-Space Pathloss (FSPL) exponent shown in 3.10 in dB, $\zeta_{k,n}$ and f_c refers to the carrier frequency of the waveform and c refers to the speed of light. The channel gain $h_U[k]$ shown in 3.11 is a function of 3.9, 3.10 and $d_U[k]$ [34].

3.2 Mathematical Derivations & Problem Definition

The desired outcome of the algorithm and overall system involving UAVs as networking platforms is that the UAVs acting in the environment can perform their task of providing optimal network coverage that must be energy efficient, secret and secure for legitimate GUs while also ensuring that eavesdropping GUs are prevented from eavesdropping on legitimate UAV-GU communications. The maximisation of the secrecy rate is a joint optimisation problem that is reliant on the maximisation of the energy efficiency, the maximisation of the data exchange rate, the optimisation of the data exchange rate and the minimisation of the UAV trajectory. The definitions and explanations for these optimisation problems are outlined in the following subsections.

3.2.1 Secrecy Rate Optimisation

The secrecy rate optimisation is derived from the NOMA data exchange rate optimisation, energy efficiency optimisation and the UAV trajectory optimisation.

The secrecy rate, with aspects adapted from [3], with the model for NOMA communications from [8], is derived from the subchannel bandwidth, which allocates a frequency band from the total carrier frequency bandwidth at a given time in which a portion of the total transmit power is allocated to each GU based on each of their respective channel gains, $h_{U,k}$.

$$\sum_{k=1}^K B_{sc} \leq BW_{f_c} \quad (3.12)$$

As shown in 3.12, the sum of the subchannel bandwidths must not exceed the total bandwidth of the carrier frequency f_c .

The position of the UAV must only change such that it cannot exceed the maximum step change, δ multiplied by the maximum velocity of the UAV, V_{max} .

Equation 3.13 shows the allocated transmit power cannot exceed the maximum allocated power in timeslot n .

$$\sum_{i=0}^I P_{U,i}[n] \leq P_{U,k}^{Tx}[n], \forall n \quad (3.13)$$

The distance between the UAV and GU, shown in 3.14 is factored into the channel gain along with the directional gain of the antennae on the UAV and the GU. This UAV channel gain $h_{U,k}$ for k GUs, shown in 3.15 is subsequently used in 3.16 for computing the data rate in the UAV-GU communications channel, which in turn, is used to compute the MASR or the secrecy rate of the communications.

$$d_{U,k} = \sqrt{(x_{U,k} - x_{GU,k})^2 + (y_{U,k} - y_{GU,k})^2 + (z_{U,k} - z_{GU,k})^2} \quad (3.14)$$

$$h_{U,k}[n] = g_{U,k}[n] d_{U,k}[n]^{-\zeta_{k,n}} \quad (3.15)$$

$$SNR : \gamma_{U,k,i}[n] = \frac{P_{U,i}[n] h_{U,k}[n]}{I_{k,n} + (\sigma_{LoS} + \sigma_{NLoS} + \sigma_{AWGN})^2} \quad (3.16)$$

The SNR is the ratio of the channel power by the channel gain to the sum of the AWGN, LoS and NLoS noise power values and $I_{k,n}$ is the interference from other users, which is cancelled with SIC. The rate of data exchange in the UAV-GU communication link is expressed in 3.17, which is a function of the subchannel bandwidth $B_{sc}[n]$ in slot $[n]$ and the SNR.

$$R_{U,m,i}[n] = B_{sc}[n] \log_2(1 + \gamma_{U,m,i}[n]), \forall i, m, n \quad (3.17)$$

The position of Eve m in timeslot n is shown in 3.18.

$$s_m[n] = \begin{pmatrix} x_m[n] & y_m[n] & z_m[n] \end{pmatrix}^T \quad (3.18)$$

The UAV-BS observes the state with all of the GUs, and those that have not been authenticated as LUs within the network are classed as Eves. Based on the proximity of the perceived Eves within the environment, the UAV-BS must calculate the eavesdropping rate that each Eve has for each LU as shown in 3.19.

$$R_{E,k,m}[n] = B_{sc}[n] \log_2(1 + \gamma_{E,k,n}[n]), \forall k, m, n \quad (3.19)$$

The eavesdropping rates with the highest values are used for the calculation of the ANS. With this secrecy scheme, each of the LUs are protected from the worst-case eavesdropping rate, i.e., the eavesdropping rate of whichever Eve has the greatest SNR to conduct their MitM attack. The worst-case eavesdropping rate is shown in 3.20 and the minimum secrecy rate is expressed in 3.21, being the absolute difference between 3.17 and 3.20 [3]. The secrecy sum rate is expressed in 3.22

$$\Phi = \max_{\mathcal{M}} \left\{ \max_{\max \gamma_{E,k,m}[n]} R_{E,k,m}[n] \right\}, \forall k, m, n \quad (3.20)$$

$$R_{k,i}^{sec}[n] = [R_{U,k,i}[n] - \Phi]^+, \forall i, k, n \quad (3.21)$$

$$\bar{R}_{k,i}^{sec}[n] = \frac{1}{N} \sum_{n=1}^N \sum_{i=1}^I R_{k,i}^{sec}[n], \forall k \quad (3.22)$$

The secrecy rate is to be maximised subject to the following constraints, the UAV must not exceed its range and altitude limits in three-dimensional Cartesian space (3.29), the power consumption must not exceed the maximum power consumption over time (3.24), the

allocated transmit power must be greater than or equal to 0 (3.25) and the change in position of the UAV must not exceed the step size δ multiplied by the maximum velocity of the UAV, V_{max} (3.28). The co-ordinate position of the UAV in Cartesian co-ordinates, $[x_U, y_U, z_U]$ is denoted as q_U in 3.28.

$$\max_{q_U, A, E, Q} \bar{R}_{k,i}^{sec}, s.t. \quad (3.23)$$

$$\sum_{i=1}^I P_{U,i} \leq P_{tot} \quad (3.24)$$

$$0 \leq P_{U,k}^{Tx} \leq P_{U_{tot}}^{Tx}, \forall k \quad (3.25)$$

$$\sum_{k=1}^K B_{sc,k}[n] \leq BW_{fc}, \forall k \quad (3.26)$$

$$E_{cons} \leq E_{max} \quad (3.27)$$

$$\|q_U[n+1] - q_U[n]\|^2 \leq (\delta V_{max})^2, \forall n \quad (3.28)$$

$$x_{min} \leq x_U \leq x_{max}, y_{min} \leq y_U \leq y_{max}, z_{min} \leq z_U \leq z_{max} \quad (3.29)$$

The subchannel allocation, UAV trajectory and energy efficiency must be maximised as part of this joint optimisation problem to ensure that the secrecy rate is maximised. The Eve SNR must also be minimised in order to minimise the eavesdropping rate, as described in 3.30 and 3.31. The maximisation of the UAV-LU data rate is also required to minimise the eavesdropping rate and the maximisation of the ANS, ψ .

$$\max_{\psi} \gamma_{E,m,k} \quad (3.30)$$

$$\min_{\min \gamma_{E,k,m}} R_{E,k,m}[n], \forall k, m, n \quad (3.31)$$

3.2.2 Energy Efficiency Optimisation

The energy consumption mathematical model has been adapted from [8] to describe the consumption of energy on board any given UAV. The model accounts for the energy consumed while travelling, hovering, by the avionics and the communication energy. The energy consumption at time t , $E_{cons}(t)$ can be expressed as shown in 3.32.

$$E_{cons}(t) = \sum_{n=1}^N \sum_{k=1}^{K_n} \left(\underbrace{\frac{\sum_{i=1}^I n_i g |q_U(t)|}{Kr}}_{\text{Travelling}} + \underbrace{\frac{((\sum_{i=1}^I n_i)g)^{\frac{3}{2}}}{\sqrt{2r\zeta\theta}}}_{\text{Hovering}} + \underbrace{\Lambda \frac{|q_U(t)|}{v(t)}}_{\text{Avionics}} + \underbrace{P_{k,n}^{Tx}(t) R_{k,n}(t)}_{\text{Communications}} \right) \quad (3.32)$$

In 3.32, n_i is the mass of the UAV frame and battery, g is acceleration due to gravity (9.81 ms^{-2}), $q_U(t)$ is the position of the UAV at time t , K is the lift-to-drag ratio, shown in 3.33, r denotes the number of UAV rotors, ζ refers to the air density, θ is the spinning blade of a single rotor, Λ is the avionics power and $v(t)$ is the velocity of the UAV at time t [8, 35].

$$K = \frac{L}{D} = \frac{\sum_{i=1}^2 n_i g v_U}{P} \quad (3.33)$$

$\sum_{i=1}^2 n_i$ is the mass of the UAV frame and battery, g is the acceleration due to gravity, v_U is the velocity of the UAV and P is the power required for the UAV [36].

As can be seen in the expression for the communications energy, the transmit power is included within the energy consumption model, which is factored into the secrecy rate in 3.23. The total energy efficiency can be expressed as shown in 3.34.

$$\eta(T) = \int_{t=1}^T \frac{\sum_{n=1}^N \sum_{k=1}^{K_n} R_{k,n}(t)}{E_{cons}(t)} dt \quad (3.34)$$

The energy efficiency must be maximised as part of the joint optimisation expressed in 3.23. The optimisation of 3.34 is detailed in 3.35 and the maximisation of $\eta(t)$ involves the optimisation of the user grouping, UAV position and power allocation on the UAV.

$$\max_{G_n \in N, q_U(t), P_{k,n}^{Tx}(t)} \eta(t), \text{s.t.} \quad (3.35)$$

$$\sum_{n=1}^N \sum_{k=1}^{K_n} P_{k,n}^{Tx} \leq P_{UAV}^{Tx} \quad (3.36)$$

$$0 < \sum_{k=1}^{K_n} a_{k,n} \leq 1 \quad (3.37)$$

$$a_{1,n} < \dots < a_{K_n,n} \quad (3.38)$$

$$R_{k,n}[n] \geq R_{min} \quad (3.39)$$

$$x_{min} \leq x_U \leq x_{max}, y_{min} \leq y_U \leq y_{max}, z_{min} \leq z_U \leq z_{max} \quad (3.40)$$

The constraints applied to 3.35 include the limit of the maximum transmit power in a NOMA group of GUs (3.36), the transmit power coefficient range for each GU (3.37), the fact that each transmit power coefficient must be ordered such that a higher coefficient is granted to users with a lower channel gain (3.38), the transmitted data rate must exceed the minimum threshold data rate (3.39) and the UAV must remain within the bounds of its range and altitude (3.40).

3.2.3 Optimisation of UAV-GU Data Exchange Rate

As the function of data rate exchange shown in 3.17 is a function of the SNR, which itself is a function of the channel gain, which is dependent on the distance between the UAV-BS and any given GU as a result of the FSPL, the UAV-GU data exchange rate can be optimised by minimising the distance between the UAV-BS and all K GUs. To achieve 3.41, the difference in distances between all GU to the UAV-BS must be minimised and the distance between the UAV-BS to the GU centroid must be minimised. The differences in distances between the UAV-BS and the GUs is expressed in 3.42. Minimising this distance as shown in 3.43 ensures that all GUs are receiving a level of network coverage that is above R_{min} . Minimising the distance between the UAV-BS helps to ensure that the UAV trajectory is optimised, as shown in 3.47, while also maximising $R_{U,k}[n]$, as expressed in 3.41. The GU centroid, $C_{GU}[n]$ is the average value between all of the GU positions.

$$\max R_{U,k}[n] \quad (3.41)$$

$$\lambda_{GU}[n] = \sum_{k=1}^K \sum_{j \neq k} |d_{U,k}[n] - d_{U,j}[n]| \quad (3.42)$$

$$\min \lambda_{GU}[n] \quad (3.43)$$

$$C_{GU}[n] = \frac{1}{K} \sum_{k=1}^K d_{U,k}[n] + z_{min} \quad (3.44)$$

$$\min C_{GU}[n] \quad (3.45)$$

The objective functions shown in 3.41, 3.43 and 3.45 are subject to the constraints shown in 3.28, 3.29, 3.36 and 3.39.

3.2.4 Optimisation of UAV Trajectory

The UAV trajectory can be expressed as 3.46.

$$c(t) = q_U[n+1] - q_U[n] \quad (3.46)$$

The trajectory is to be minimised such that the UAV can take the shortest path to serve its role for a set of K GUs and other UAVs.

$$\min q_U(t), s.t. \quad (3.47)$$

$$x_{min} \leq x_U \leq x_{max}, y_{min} \leq y_U \leq y_{max}, z_{min} \leq z_U \leq z_{max} \quad (3.48)$$

$$d_U, k \leq 100, \forall k \quad (3.49)$$

3.3 Algorithms & System Design

The proposed approach for solving the outlined optimisation problem leverages DRL to determine the optimal secrecy rate for UAV-GU communications such that $R_{U,k,n} \forall k, n$ over NOMA is optimised, the energy efficiency converges to higher values, the resource allocation is optimised and the UAV trajectories are optimised.

3.3.1 Design

The overall architecture of the system is illustrated in Fig. 3.2.

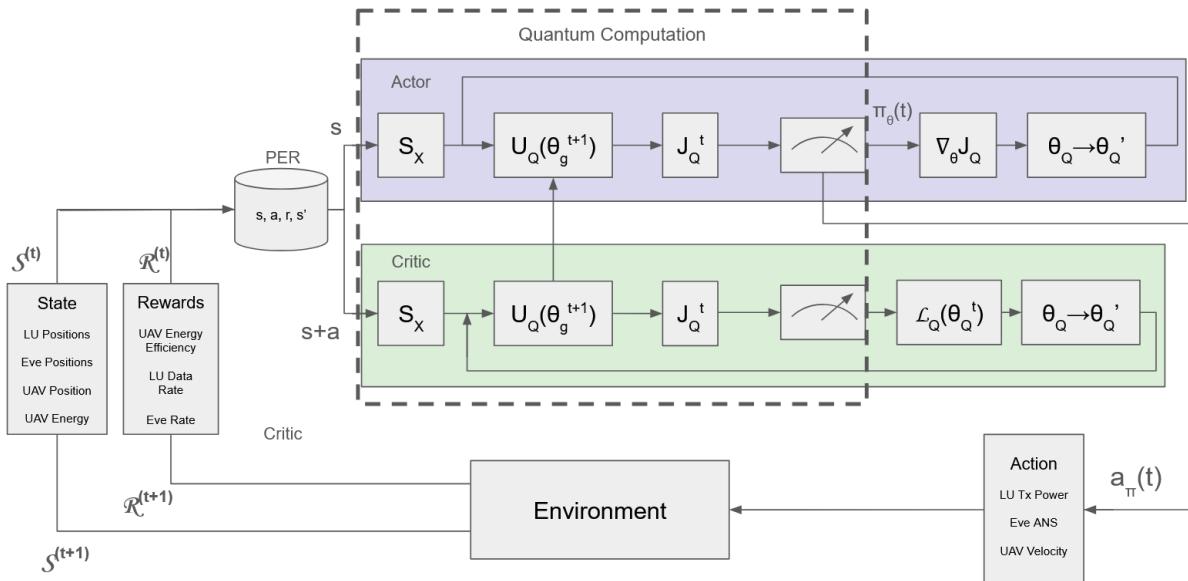


Figure 3.2: System Block Diagram

As shown in Fig. 3.2, the overall system is a quantum-classical hybrid approach to the joint optimisation problem outlined in the previous section of this chapter. This system architecture has been adapted from [8] and [37]. The design of the LQ-DRL algorithm employing layerwise quantum embedding involves the use of storing parameters of the system as quantum states using a quantum circuit, referred to as an ansatz. This process is referred to as quantum encoding. The input parameter data is stored in a vector \mathbf{x} of N elements, which are encoded into a set of quantum states $|\phi\rangle_{0,1,\dots,N-1}$. The encoding process is denoted as S_x . The UAV is an agent interacting with the environment to observe the state space s , determine an action a that is best in line with the desired policy π that yields a reward r and creates the updated state space s' . These parameters are embedded within the ansatz as N qubits, where N denotes the number of parameters required for the quantum data embedding. The value function of the selected policy π for a state s is shown in 3.50 and μ' is the learning discount rate [38].

$$V_\pi(s) = E \left(\sum_{t=1}^T \mu' r_t | s_0 = s \right) \quad (3.50)$$

The system has been designed as a Deep Q-Network (DQN). Q-learning involves evaluating the quality or value of a state/action pair by determining its Q-value $Q(s, a)$. The Q-value can be updated as shown in 3.51 [38].

$$Q^{update}(s, a) = Q^{old}(s, a) + \alpha \left(r_t + \mu \max_a Q(s_{t+1}, a) - Q^{old}(s_t, a_t) \right) \quad (3.51)$$

Q-learning is employed to essentially evaluate both 3.50 and the policy itself simultaneously. By maximising the Q-value, the quality of state/action pairs can influence the DRL algorithm's agent to perform more optimally as it learns.

The system contains a MER, which stores the states (s), actions (a), rewards (r) and the next state (\hat{s}). The experiences of the agent are stored and sampled as the agent learns to optimise its parameters and actions based on the observed state space which is sampled by the critic to evaluate the quality of an action taken by the actor network. The more experiences that the UAV has, the larger its set of experiences to sample and compare its current action from grows.

3.3.2 Quantum Data Embedding & Parameter Encoding

The ansatz in use for the actor and critic networks involves the encoding of data into the circuit that is measured and decoded at the output and passed to a gradient descent algorithm that employs the two-term parameter-shift rule to compute the quantum gradients. The actor network takes the observed state space at time t , which contains $2K + 4$ dimensions, where $K \in \mathcal{K}$, i.e., the number of GUs in the environment. The critic network takes both the state and action at time t as its inputs, which contains $(2K + 4) + 5$ dimensions as the action space contains 5 dimensions, the UAV transmit power, the ANSs for the Eves and the UAV velocity and trajectory.

The quantum gates in use within the quantum circuits are the Hamiltonian, x-rotation gate, y-rotation gate and the controlled Pauli-Z (denoted as CZ) gates, whose matrix representations are shown in 3.52, 3.53, 3.54 and 3.55.

$$H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (3.52)$$

$$RX(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -i \cdot \sin(\frac{\theta}{2}) \\ -i \cdot \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \quad (3.53)$$

$$RY(\theta) = \begin{pmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{pmatrix} \quad (3.54)$$

$$CZ = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & -1 \end{pmatrix} \quad (3.55)$$

The quantum embedding system design uses the ansatz presented in [8] and the quantum circuit in use for the actor and critic networks is presented in Fig. 3.3.

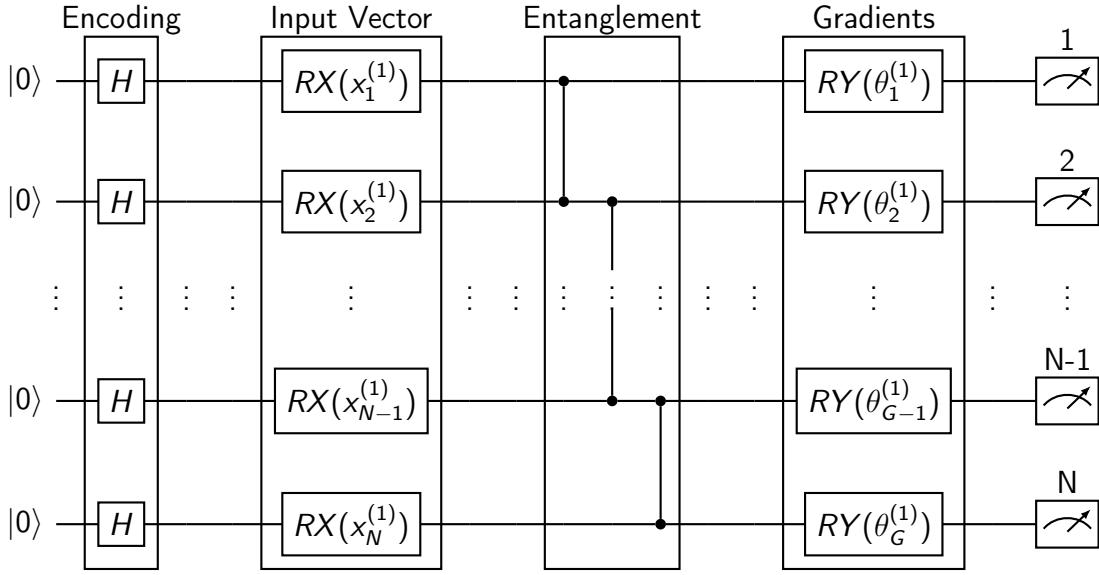


Figure 3.3: Single-Layered Actor-Critic Network Ansatz

The actor and critic quantum circuits can be represented as unitary operators, which evolve a Hermitian operator over time. The equation describing the ansatz design presented in Fig. 3.3 is shown in 3.56 [8].

$$U_{LQ}(\theta) = \bigotimes_{g=1}^G \bigotimes_{n=1}^N (S_{\theta_g}) \left(\prod_{n=1}^N CZ(\phi_2|\phi_1) \otimes \cdots \otimes CZ(\phi_N|\phi_{N-1}) \right) (S_{x_n}) H \quad (3.56)$$

The encoding of the input parameter data vector \mathbf{x} is shown in 3.57 and the encoding of the quantum gradients is shown in 3.58.

$$S_x : \bigotimes_{n=1}^N RX(x_n) \quad (3.57)$$

$$S_\theta : \bigotimes_{g=1}^G RY(\theta_g) \quad (3.58)$$

The decoding operation for the actor network is shown in 3.59, 3.60 and the operation is

reduced to 3.61, where K_{shots} is the number of measurements taken of the output [8].

$$J_Q = \langle 0 | U_Q^\dagger(x) H U_Q(x) | 0 \rangle \quad (3.59)$$

$$J_Q(\theta_g) = Z(|\phi_n\rangle) \quad (3.60)$$

$$y \leftarrow \frac{1}{K_{shots}} \sum_{k=1}^{K_{shots}} Z(|\phi\rangle) \quad (3.61)$$

As the actor's output state has to be measured in the computational basis, its basis is transformed into the Pauli Z-basis for measurement.

The critic outputs the loss calculation based on the generated Q-value for a given action a in state s based on target rewards sampled from the MER. This operation is shown in 3.62 [8].

$$\mathcal{L}_Q = \frac{1}{N_{data}} \sum_{n=1}^{N_{data}} (y_n - \hat{y}_n)^2 \quad (3.62)$$

The variable y_n refers to the actor output and \hat{y}_n refers to the target output. The difference between these values is taken and squared to ensure that the result is positive-valued. The critic is calculating a loss value based on the difference between an optimal target action and the action taken by the agent.

The θ parameters are updated for each timestep as shown in 3.63 [8].

$$\theta_g = \theta_g - \beta \nabla_{\theta_g} \mathcal{L}_Q(\theta_g) \quad (3.63)$$

$\theta_0, \theta_1 \dots \theta_{G-1}, \theta_G$ is updated using the gradient descent calculation of 3.62 multiplied by β , which is the learning rate for the critic [8].

3.3.3 Classical Computation of Gradient Descent

The two-term parameter shift algorithm is used to compute the gradient descent for computation of new parameters in the quantum circuit after every timestep. This approach involves modelling the expectation value of the measured output of a quantum circuit as a Fourier series using a finite set of Positive Unique Differences (PUD). The Fourier series has a finite number of terms and can be computed classically. The Discrete Fourier Transform (DFT) can be used to determine the coefficients of the Fourier series. The expectation value for a general gate $U(x) = e^{ixG}$ defined by a Hermitian operator G and parameterised by x can be expressed as shown in 3.64.

$$E(x) = a_0 + \sum_{l=1}^R \left[a_l \cos(\Omega_l x) + b_l \sin(\Omega_l x) \right] \quad (3.64)$$

The two-term parameter-shift rule for computing quantum gradients can then be expressed by 3.65 having introduced a shift parameter $s \in \mathbb{R}$, such that $\frac{\Omega s}{\pi} \notin \mathbb{N}$ [39].

$$\frac{dE(x)}{dx} = \frac{E(x+s) - E(x-s)}{2 \sin \Omega s} \Omega \quad (3.65)$$

3.3.4 Reward Shaping

The reward function is shown in 3.66. The computed energy efficiency for a given timestep is awarded on the condition that the minimum secrecy rate for all legitimate GUs has been matched or exceeded.

$$\mathcal{R}(t) = \begin{cases} \eta(t), & R_{U,k}^{sec}[n] \geq R_{min}^{sec}[n] \forall k, n \\ 0, & \text{otherwise} \end{cases} \quad (3.66)$$

The reward allocation is penalised if any of the constraints to the optimisation problems are violated. The more the difference in distances between the LUs is minimised, the agent receives incremental reward boosts to the reward. This also occurs when the UAV is within a proximal region of 30 m surrounding the LU centroid, which is defined as shown in 3.44. The reward is also penalised if it has violated any of the constraints outlined in the previous section of this chapter. If the UAV flies beyond the allocated flight area that it has been placed in, then its reward is penalised by 95% of the computed reward for that step to ensure that the UAV-BS agent learns to return to the flight zone, where it's able to receive rewards.

3.3.5 Memory Experience Replay

The MER is a feature of DRL algorithms in which the states, actions, rewards and updated states are stored and sampled from at random to set targets from previous experiences for the DRL algorithm to generate Q-values and learn via Q-learning.

With the basic MER, previous experiences are stored and sampled entirely at random to set target rewards for the agent to compare its actions against. This approach can lead to the generation of some poor Q-values as all experiences have an equal probability of being selected for Q-learning, including poor actions that led to poor experiences.

With a prioritised experience replay (PER), the experiences are stored in a sum-tree as shown in Fig. 3.4, in which the experiences with higher rewards are stored within the leaf nodes of the sum tree and the experiences with the lower rewards are stored in the roots. As there are more leaf nodes than root nodes in the sum tree, the distribution for sampling more desirable experiences for reinforcement learning and generating higher Q-values is increased by using the PER as poorer experiences and actions are selected more rarely than actions

and experiences that yielded greater rewards for the agent. The PER prioritises the replay sampling by Temporal Difference Error (TD-error).

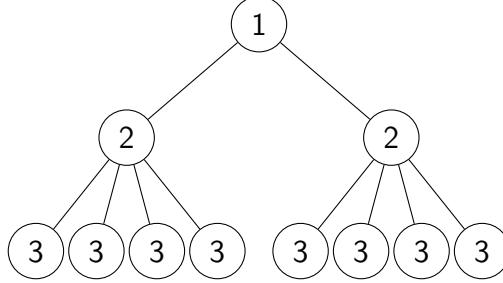


Figure 3.4: PER Sum Tree

Despite the distribution being skewed in favour of sampling experiences that yielded higher rewards, all experiences still have a non-zero chance of being sampled from the PER, ensuring that the DRL is not outright discounting any experiences. The stochastic sampling method in use with the PER is shown in 3.67 [40].

$$P(i) = \frac{p_i^\alpha}{\sum_k p_k^\alpha}, \quad p^\alpha > 0 \quad (3.67)$$

In 3.67, p_α is the priority of transition i . The exponent α determines how much prioritisation is used, with $\alpha = 0$ being the uniform sampling case. As the distribution is being changed based on the priority of a given state-wise transition and the system is reliant on expectation value of the update being the same as the expectation of its estimated value, bias is introduced to the sampling and must be corrected for. The bias is corrected for using Importance-Sampling (IS) weights, shown in 3.68. The IS weights fully compensate for the non-uniform sampling probabilities when $\beta = 1$ [40].

$$w_i = \left(\frac{1}{N} \frac{1}{P_i} \right)^\beta \quad (3.68)$$

3.3.6 ANS Generation for Eavesdropping Links

For a set of \mathcal{M} Eves attempting to perform MitM attacks on the set of \mathcal{K} LUs, the UAV-BS must dynamically adjust a noise signature for the legitimate GUs to filter out to maximise the secrecy rate. This is done to minimise the eavesdropping rate and in turn, maximise the secrecy rate. To achieve this, the UAV-BS introduces an ANS to the signals being transmitted and received in the LU communications channels based on the worst-case estimate of the eavesdropping rate. The ANS is generated by the UAV-BS, which observes the position of all of the non-authenticated GUs and calculates the eavesdropping rate for all of the perceived Eves for each LU. A scaling variable, ρ is calculated to amplify the noise signature ψ based on the proximity of any given Eve to each LU. For greater levels of proximity of an Eve, measured by $d_{E,k}$ relative to the maximum flight zone distance, the value of ρ increases, as shown in

3.69.

$$\rho = \left[\frac{d_{max,k} - d_{E,k}}{d_{max,k}} \right]^{-1} \quad (3.69)$$

This variable is then used to scale the ANS waveform as shown in 3.70.

$$\psi = \rho P_{U,k}^{Tx} \cos(f_n) \quad (3.70)$$

Where f_n is the frequency of the ANS waveform, which is also scaled by ρ as shown in 3.71.

$$f_n = \rho f_c \quad (3.71)$$

3.4 Software Engineering & Implementation

The simulation was written in the Python programming language and structured using the Object-Oriented Programming (OOP) programming paradigm. All code and scripts in use for the project besides the PER implementation were written from scratch. The key packages that were used to design the simulation were OpenAI's Gymnasium and PennyLane frameworks, made for deep reinforcement learning and quantum machine learning applications, respectively.

Git was used for software version control throughout the entire development of the project. Different features were tested on their own branches and eventually merged into the main branch once they were functioning as intended.

The initial experiments for testing particular features and ensuring that the code would run reliably were performed locally, however, as the computational load began to increase for longer and more comprehensive experiments, the Sonic High Performance Computing (HPC) cluster provided by the university was used with batch scripts, written in BASH, running the Python experiment script while writing the output log data to a text file for post-processing data analysis.

3.4.1 Custom Gymnasium Environment

UAVs, GUs and the Gymnasium environment are modelled as classes with particular attributes. Inheritance is utilised for different subclasses of GU, i.e., LUs and Eves and for different classes of UAVs so that the design is extensible to support different classes of UAV, such as UAV relays and separate UAVs for interfering with MitM attacks from Eves.

These classes were written to ensure that the design can be extended for other network architectures, however, in the final implementation, the UAV-BS handles the ANS generation.

All of the core computation involving the environment is contained within functions for com-

puting $\eta_{EE}(t)$, the channel dynamics, ANS generation, reward allocation, UAV movement, subchannel allocation for LUs and observation of the constraints is handled within the environment program.

There are also functions for accessing the results of these calculations in getter methods so that the top-level script can access this data for logging and visualisation purposes.

3.4.2 Top-Level Program

The overall simulation was designed with separate programs, each serving a particular purpose and all called upon using a top-level Python script in which the simulation parameters are instantiated, e.g., the capacity of the Memory Experience Replay (MER) or Prioritised Experience Replay (PER) buffer, the number of layers in the actor-critic network ansatz, the number of episodes to be run, the MER or PER random sampling batch size, target directories for the output data and the data visualisation.

The actor and critic loss, gradient optimisation and gradient descent calculations are also handled in this script, where the computed quantum gradients are updated in the quantum circuitry. The gradients are computed using the JAX interface and the Optax library is in use for the optimisation of the gradients.

The JAX interface was chosen for its Just-in-Time (JIT) compilation via OpenXLA, open-source machine learning compiler ecosystem. This script imports the custom Gymnasium environment, the MER, the PER and the quantum circuits for the actor and critic network.

3.4.3 Simulated Quantum Circuits

The quantum actor-critic network Python program contains the function definitions for the quantum circuits using the PennyLane library as well as the functions for visualisation the quantum circuits and the decoding operations for both the actor and critic networks. The critic class contains a method for evaluating the Q-value for the action taken by the actor, which is called for each timestep within the simulation. The JAX interface is specified for the quantum circuits and the qubit type is the PennyLane lightning qubit, which is contained within the PennyLane-Lightning library for faster computation and has a C++ backend as opposed to a Python one. The decoded outputs are vectorised within the top-level experiment script using JAX's NumPy-like arrays.

3.5 Simulation Parameters & Chosen Scenario

The values that were used in the simulations are listed in Table 3.1. These parameters were used to yield the results presented in the following chapter. The scenarios tested as part of

this research did not involve obstacles within the flight zone.

Parameter	Definition	Value
Ep	Number of Episodes	30
B	PER Sampling Batch Size	30
M	Number of Layers	1-5
BW_{f_c}	Bandwidth	1 MHz
K	Number of LUs	4
E	Number of Eves	2
X	Number of UAV-BSs	1
σ_{LoS}^2	LoS Noise	-100 dBm
σ_{NLoS}^2	LoS Noise	-80 dBm
R_{min}	Minimum Data Rate	9.5 Mbps
R_{min}^{sec}	Minimum Secrecy Rate	9.5 Mbps
$x_{max}, y_{max}, z_{max}$	Flight Area	150 m, 150 m, 122 m
$x_{min}, y_{min}, z_{min}$	Minimum UAV Co-ordinates	0 m, 0 m, 10 m
ζ	Air Density	1.225 kg m^{-3}
$\sum_{i=1}^2 n_i$	Mass of UAV Frame & Battery	1.46 kg
K	Lift-to-Drag Ratio	6.65
K_{shots}	Number of Quantum Measurements	1024
β_{actor}	Actor Learning Rate	0.01
β_{critic}	Critic Learning Rate	0.01
V_{max}	Maximum UAV Velocity	50 m s^{-1}
$P_{U,max}^{Tx}$	Maximum Transmit Power	30 dBm
A_1, A_2	Rician Channel Characteristics	4, 0.1
E_{max}	Total UAV Energy	50 kJ
r	Number of UAV Rotors	4
θ	Spinning Blade of 1 Rotor	0.0507 m^2
μ	Discount Factor	0.99

Table 3.1: Simulation Parameters

Multiple UAVs were tested along with larger numbers of GUs early in the development of this project, however, the problem was reduced to a single-agent DRL approach rather than immediately starting with a multi-agent reinforcement learning (MARL) approach. This is also a very computationally intensive system to simulate, so fewer of GUs and a single UAV-BS led to the simulations being faster and a greater ability to debug and improve the system throughout the development of the simulations. A classical DRL implementation was attempted early in the development of the project for comparative purposes, however, this proved to detract from the LQ-DRL implementation and was abandoned.

4 Results

4.1 Optimised UAV Parameters

The parameters that were optimised by maximisation and minimisation of values were the UAV energy efficiency $\eta_{EE}(T)$, the data exchange rate $R_{U,k}[n]$ between the UAV and LUs, the total secrecy rate $R_{U,k}^{sec}$ and the UAV trajectory $c(t)$, leading to a decrease in the distances between the UAV-BS and LUs.

4.1.1 UAV Trajectory

For $\eta_{EE}(t)$, $R_{U,k}(t)$ and $R_{U,k}^{sec}$ to be optimised, the UAV trajectory also had to be optimised to fairly and securely provide coverage to all of the LUs. This involved ensuring that distance to the GU centroid as well as a smaller difference in distance between the UAV and all of the LUs were minimised and that rewards were allocated to the agent for minimising these values from timestep to timestep.

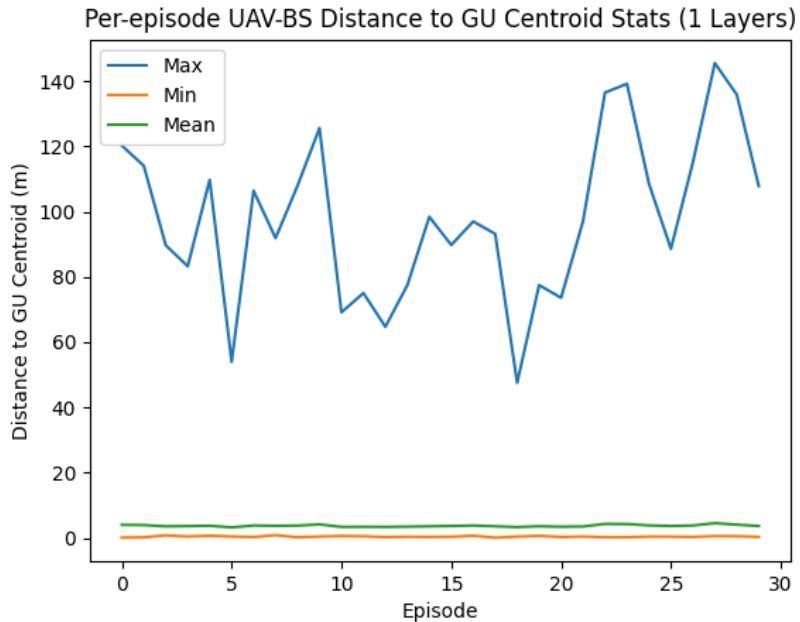


Figure 4.1: Maximum, Minimum & Mean Distances to the LU Centroid Across 30 Episodes

As can be seen in Fig. 4.1 and Fig. 4.2, the UAV converged towards the LU centroid (10 m above the average x, y co-ordinates for all LUs) from a number of randomly generated starting locations where it was initialised in each episode. This demonstrates that the UAV consistently learned to achieve this objective, regardless of where it started in the simulation.

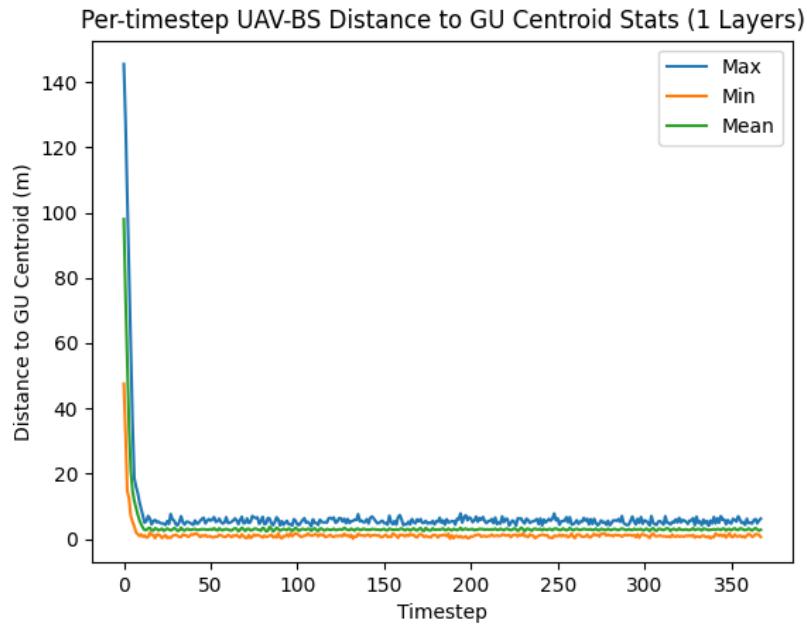


Figure 4.2: Maximum, Minimum & Mean Distances to LU Centroid Per Timestep Across 30 Episodes

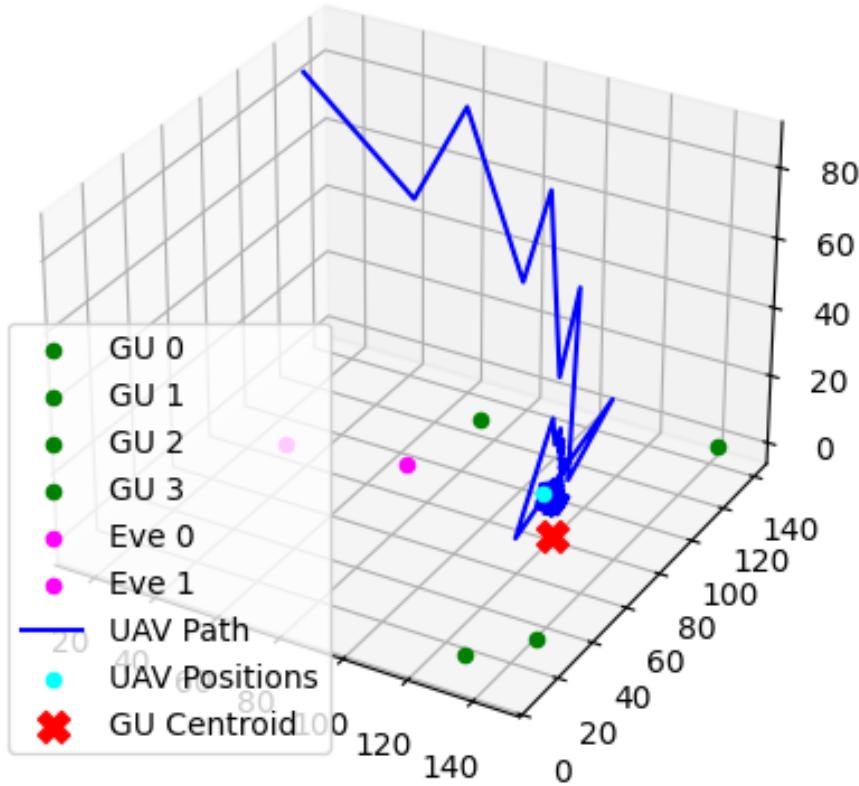


Figure 4.3: UAV Trajectory in Simulated Environment

As can be seen in Fig. 4.3, the UAV managed to learn, based on the observed state of the environment, to converge on the centroid of the LUs despite its initial position being on the far side of the flight area.

4.1.2 Energy Efficiency

The stabilisation and convergence of $\eta(t)$, i.e., the energy efficiency at timestep t and $\eta(T)$, i.e., the integral of the energy efficiency from $t = 0$ to $t = T$ is shown in Fig. 4.4 & 4.5, respectively. A mean value of 40kbps/Hz/J for $\eta(T)$ is maintained for all episodes in the simulation as the agent exhibits that it can and does effectively learn to optimise $\eta(t)$ early in the simulation with a single-layered ansatz in the actor and critic networks' quantum circuits.

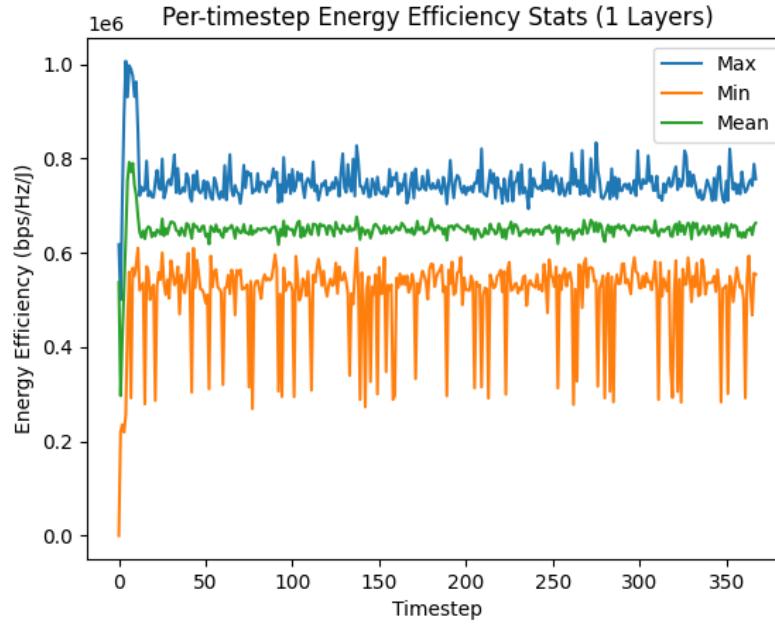


Figure 4.4: Maximum, Minimum & Mean Energy Efficiency Per Timestep

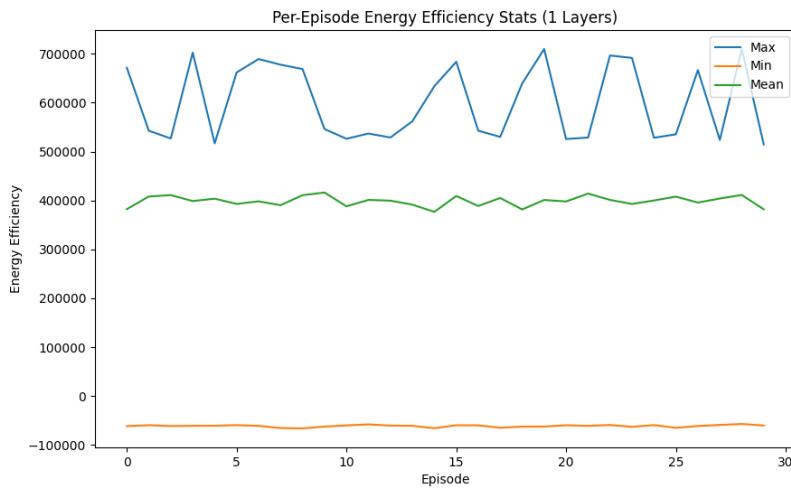


Figure 4.5: Energy Efficiency Across 30 Episodes

As shown in Fig. 4.4 and Fig. 4.5, $\eta_{EE}(t)$ converges consistently towards $\tilde{40}\text{kbps}/\text{Hz}/\text{J}$ across all of the 30 episodes that were run.

As the UAV trajectory has been optimised as well as the data exchange rate and the secrecy rate, the energy efficiency increases as the UAV converges towards a lower level of energy consumption as it ceases to move around the environment and is able to hover above the optimal location for secure communications with the LUs.

4.1.3 Data Exchange Rate

The data exchange rate $R_{U,k}(t)$ is used to shape the reward allocation along with penalties for violating the constraints of the joint optimisation problem. As rewards are dependent on the data exchange and secrecy rates for all LUs and $\eta(t)$ is a function of $R_{U,k}(t)$, the maximisation of $\eta(T)$ requires that $R_{U,k}(t)$ is maximised and E_{cons} is minimised.

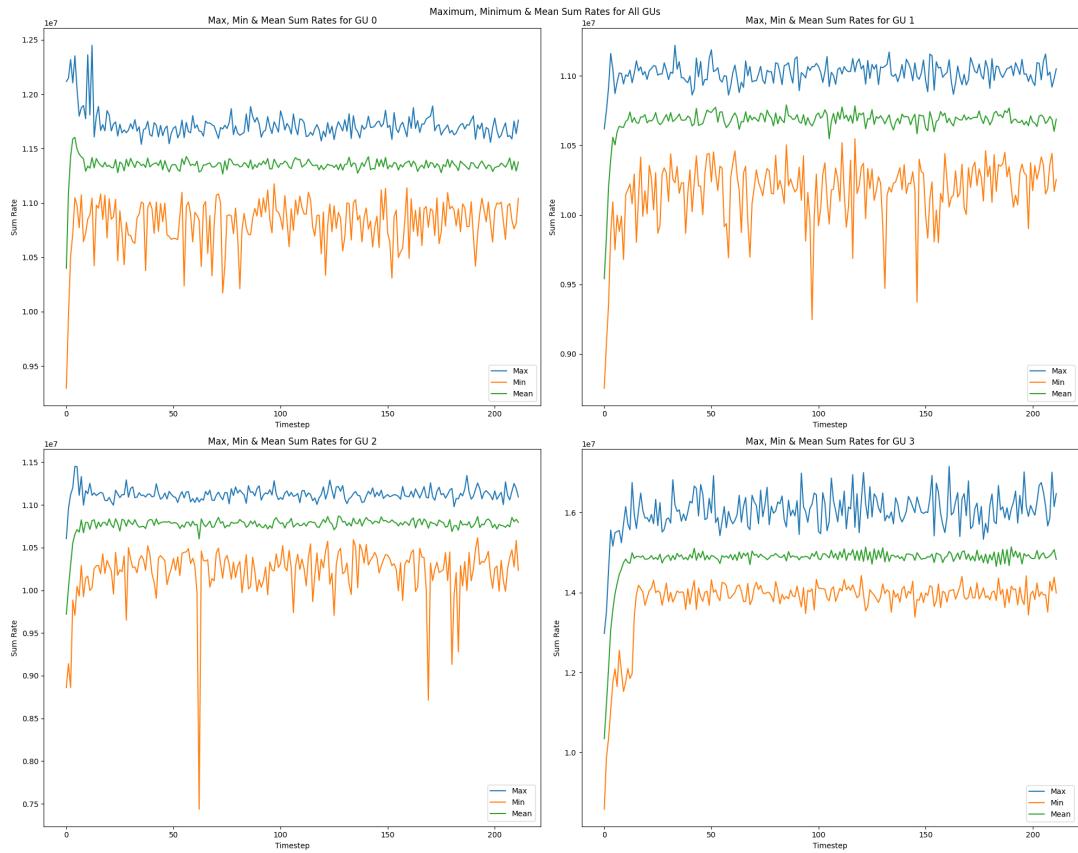


Figure 4.6: Maximum, Minimum & Mean $R_{U,k}(t)$ Values in Mbps from t to T Across 30 Episodes

As shown in Fig. 4.6, for all of the legitimate GUs in the simulated scenario, the average sum rate converges from 0 to above 10 Mbps for $R_{min} = 9.5 \text{ Mbps}$ consistently within a short period of time.

Fig. 4.7 displays the maximum, minimum and mean values for the data exchange rates across

30 episodes. Again, it can be seen that the agent effectively managed to learn to maximise $R_{U,k}(t)$ for all LUs such that the R_{min} threshold has been met for all LUs, ensuring fairness in the distribution of $R_{U,k}(t)$ for all \mathcal{K} LUs.

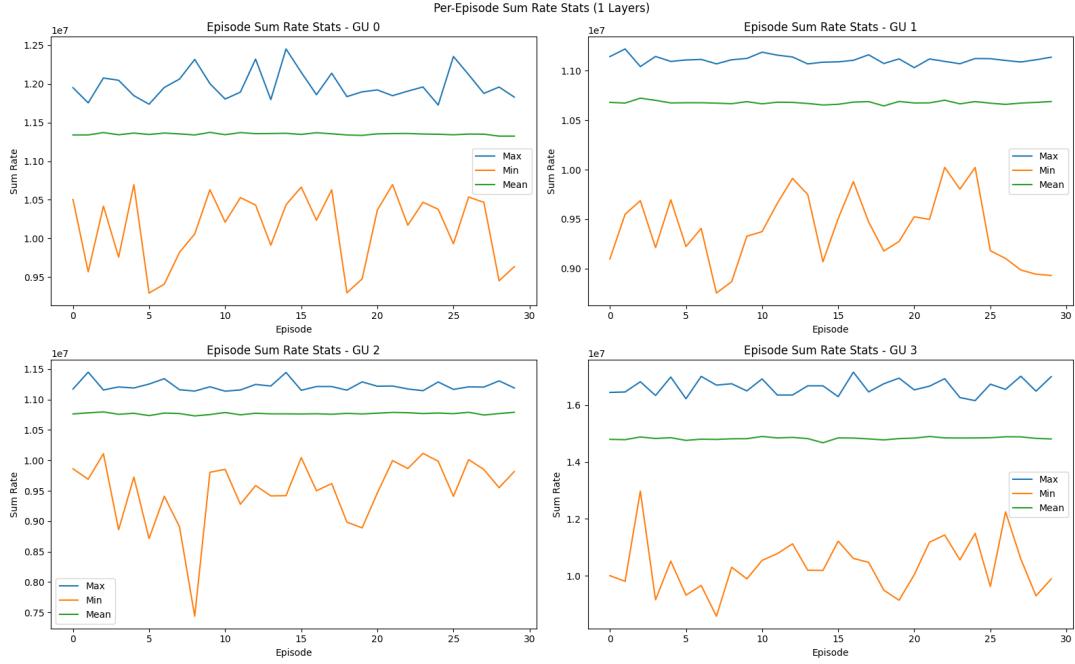


Figure 4.7: Maximum, Minimum & Mean $R_{U,k}(t)$ in Mbps Values Across 30 Episodes

This demonstrates that the agent effectively learns to maximise the data exchange rate as well as the fairness in data rates for all GUs such that all GUs receive an acceptable and consistent QoS. None of the LUs have a value for $R_{U,k}(t) < R_{min}$ upon convergence, thus adhering to the constraints of the optimisation of $R_{U,k}(t)$.

4.1.4 Secrecy Rate

The secrecy rate for the UAV-LU communications had to be above $R_{min}^{sec} = R_{min} = 9.5 \text{ Mbps}$. This was achieved with a single layer in the actor-critic quantum circuits and is shown in Fig. 4.8 and Fig. 4.9, respectively.

It can be seen that $R_{U,k}^{sec}$ converges to above R_{min}^{sec} consistently, which is in-line with the increase in $R_{U,k}$, as the increase in $R_{U,k}^{sec}$ is dependent on the increase of $R_{U,k}$. As shown in Fig. 4.8 and Fig. 4.9, the threshold for the minimum acceptable secrecy rate was reached for all of the legitimate GUs. No single LU was left for its average total secrecy rate $\bar{R}_{U,k}^{sec}$ to fall below R_{min}^{sec} .

Each LU had its secrecy rate converge to above 10 Mbps within the simulation to the point of the secrecy rate converging onto the value of the data exchange rate. This is a desirable outcome for this problem, demonstrating the efficacy of spiking the UAV-LU communications

4. Results

4.1. Optimised UAV Parameters

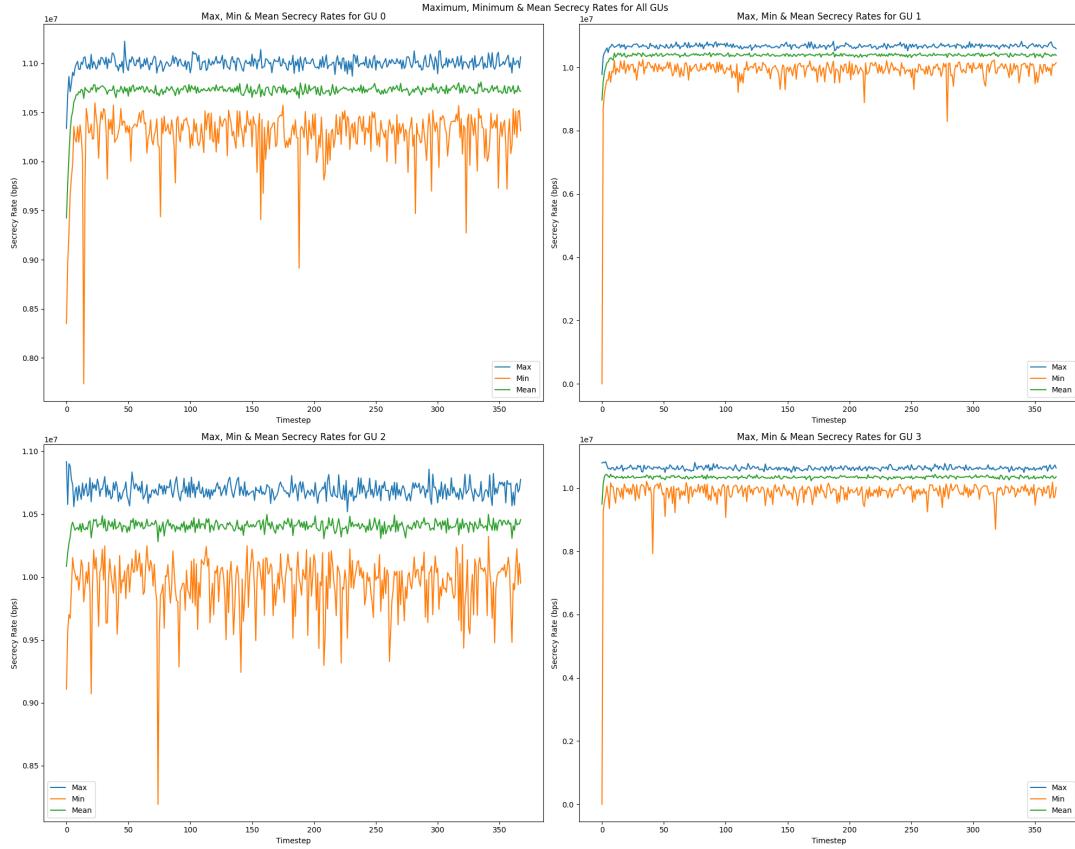


Figure 4.8: Maximum, Minimum & Mean Secrecy Rates for all LUs for each Timestep Across 30 Episodes

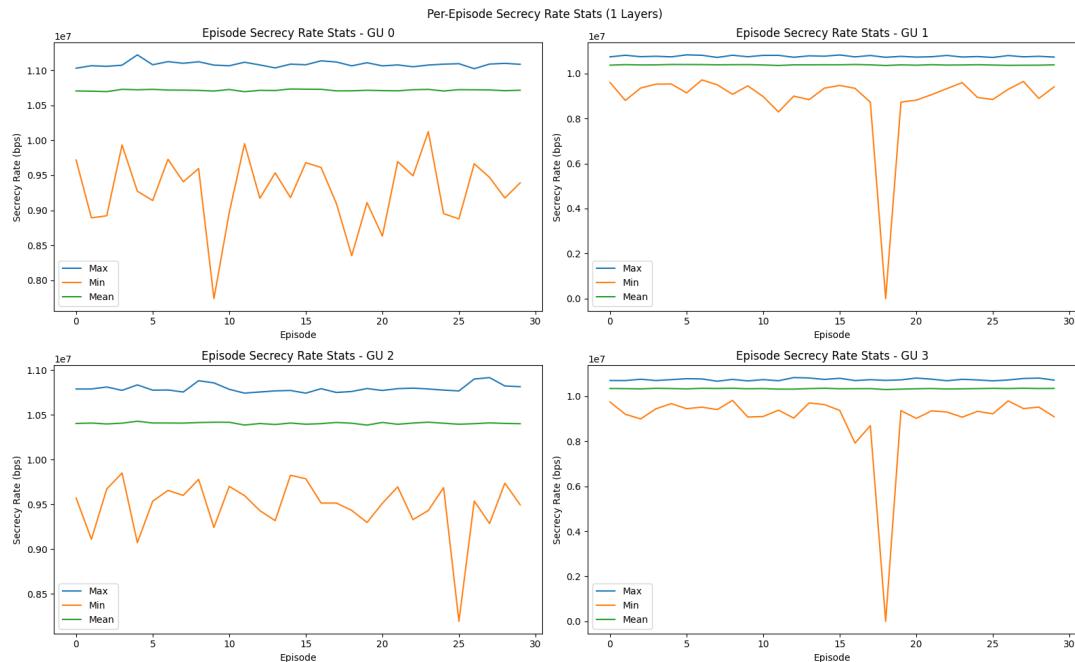


Figure 4.9: Maximum, Minimum & Mean Secrecy Rates for all LUs Across 30 Episodes

with an ANS. This demonstrates that the Eve links were so affected by the ANS that their SNR plummeted to a very low value, i.e., any form of useful information from the signal was overpowered entirely by the ANS. This shows that the eavesdropping rate has dropped, increasing the secrecy rate and minimised the capabilities for Eves to conduct MitM attacks on the LUs.

4.2 DRL Performance

4.2.1 Local Loss

The critic loss within the algorithm began to decrease and stabilise around a small set of values for a single layer, as shown in 4.10.

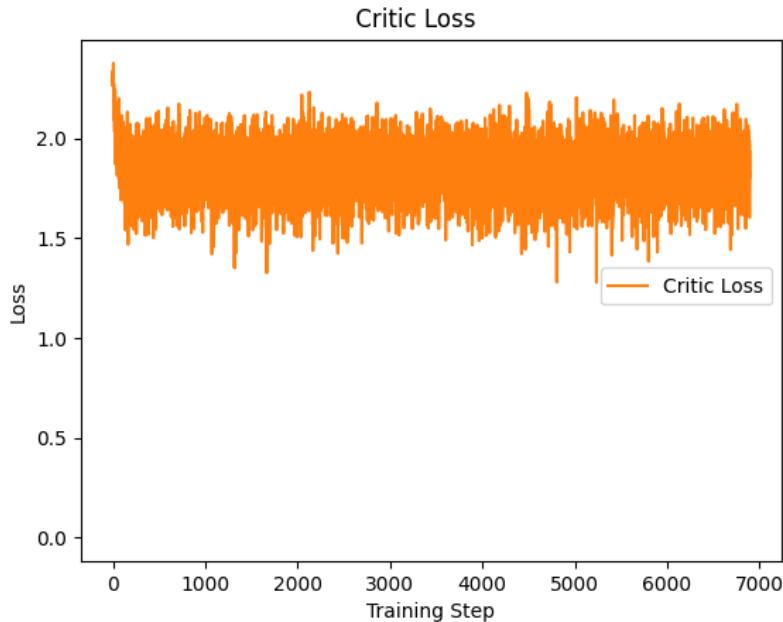


Figure 4.10: Critic Loss for $M = 1$

The results for the loss function from the critic network are quite noisy as a result of having to sample and decode the values K_{shot} times in a simulated open quantum system. The length of the quantum circuit is also quite large, with 17 inputs being fed into the critic network, leading to a noisy output prior to decoding, however, this did not appear to impact the results presented in the previous section in a negative way.

4.3 Quantum Actor-Critic Network

The actor-critic network quantum circuitry appeared to perform most optimally with a single layer within the ansatz rather than increasing the number of layers.

The quantum actor-critic network was tested with 1-5 layers within the system. This was performed to determine the impact on the number of layers and if the increase in the number of layers positively or negatively impacted the performance of the system beyond a particular number of layers.

4.3.1 Impact of the Number of Layers on Performance

It was found that for higher numbers of layers beyond a single layer within the ansatz that the performance of the LQ-DRL algorithm began to degrade and become far more unpredictable, leading to very noisy curves for the UAV trajectory, $\eta_{EE}(t)$, $R_{U,k}$ and $R_{U,k}^{\sec}$.

After 3 layers within the actor-critic ansatz circuits, it was observed that the results for the secrecy rate, data exchange rate, trajectory and energy efficiency all became far too noisy and unpredictable to be of any practical value.

This phenomenon may arise from the fact that the same values are being recomputed repeatedly despite only a single layer being useful, leading to increased levels of noise arising from the increased depth of the quantum circuits.

Figures with data for each timestep of each episode for each number of layers can be found in A1.

As shown in A1, the results are consistent and converge towards their optimal values for the UAV trajectory and the data exchange rate, however, for increasing numbers of layers, the UAV can either do the opposite of what its objective is for $M = 2$ and begins to become too noisy and unpredictable for 3-5 layers. In the case of $M = 2$, the fact that the data exchange rate and the distance between the UAV-BS and the LU centroid converged to the opposite of the desired outcome could be useful for providing some insights into the optimisation problem, e.g., it could serve as a means to model a dual optimisation problem, however, this has not been explored extensively as part of this thesis.

5 Conclusions

5.1 Achieved Objectives

The core objective of the joint optimisation problem outlined in this thesis was to maximise the secrecy rate of UAV-LU communications by optimising the UAV trajectory, data exchange rate and energy efficiency. The trajectory was minimised and the energy efficiency, data exchange rates and secrecy rates were all maximised and shown to converge consistently across a range 30 of episodes.

At the beginning of each episode, the UAV and all of the GUs are initialised in a random location within the environment. The quantum-assisted DRL algorithm and system outlined in this thesis demonstrated consistent convergence and an ability to rapidly adapt to any given environment that was tested.

This quantum/classical hybrid DRL algorithm and system has been applied to a novel optimisation problem of secrecy and physical layer security and the convergence of the secrecy rate and its subproblems demonstrate that it is an effective methodology for the optimisation problem outlined in this thesis.

5.2 Interpretation of Results

The results demonstrate that the algorithm and system design is effective for maximising the secrecy rate for UAV-LU communications. The outliers within the secrecy rate, the sum rate and the energy efficiency plots show that the UAV does explore its environment to avoid getting stuck within a local maximum or minimum, which is a problem faced by many optimisation and machine learning algorithms. This exploration balanced with a convergence towards the optimal values for the joint optimisation problem yields an optimal result from $t = 0$ to $t = T$ across all of the episodes that have been run as part of these experiments.

5.3 Future Work & Potential Improvements

5.3.1 Comparison with Classical DRL Algorithms

A comprehensive comparative study with a classical DRL implementation can illustrate the effectiveness and performance gains that can be achieved with the use of quantum computing being incorporated within the system and what the trade-offs are between entirely

classical algorithms and quantum-classical hybrid algorithms such as the one outlined in this thesis.

While this was attempted early in the development of the project, it proved to be too time-consuming to complete and detracted from the development of the LQ-DRL implementation. Development of an comparable scheme that only relies on classical DRL could be developed to determine how much of a benefit the LQ-DRL approach provided to the system.

5.3.2 MARL Support

The system has been designed using the OOP paradigm and has been designed to be extensible for more than one UAV agent, greater numbers of GUs and different environments. While this was not tested throughout the development of the code used for this thesis, the programs were designed to be modular with extensibility in mind for future work.

5.3.3 UAV-HAPs Network Architecture

The system could be expanded even further to include a HAP to act as a more stable BS, with UAVs acting as relays between the GUs and the HAP. Separate UAVs for jamming and interfering with Eve links could also be used within the system, with separate algorithms dedicated to identifying and sabotaging Eve links.

A HAP could be better suited to serve the purpose of a BS, with a stable connection to each of the UAVs which could afford to be more dynamic and adapt to a more rapidly changing environment with the HAP being treated as a networked anchor for these UAVs. Such a system could also allocate more dependence on the success of the network in a more distributed manner with a larger number of UAVs and a HAP providing the network coverage.

5.3.4 Impact of Environmental Conditions

Some future work that could be incorporated into the simulations of the system would be to test the algorithm in a variety of different scenarios. The system could be tested with environments containing different kinds of terrain, e.g., dense urban areas, clear rural plains, etc. with each kind of terrain providing benefits and drawbacks, such as obstacles for the UAV to avoid, new kinds of constraints to consider and so on. The different kinds of terrain could be compared for their effects on the Rician channel dynamics, the UAV trajectory optimisation as the probability of establishing a LoS connection may be impeded among many others. Buildings and materials such as concrete or conducting materials such as metals could affect the performance of the communications model.

Another aspect of the environmental conditions that could be tested could be different weather conditions. Rain and humid weather can have a negative effect on the propagation of com-

munications signals through the environment, introducing new scattering parameters and necessitating more techniques to overcome these difficulties. An environment with varying wind speed could also be tested in future to determine its effects on the manoeuvrability of the UAV, which could lead to the UAV having to consume more energy to combat this to complete its mission, thus impeding on the performance of the learning algorithm and potentially necessitating a mechanism within the system to handle the effects of wind on the UAV's performance.

5.3.5 Threat Models

Other cyber-attack and electronic warfare threat models could be considered for the system, with protocols for handling attacks such as domain or Internet Protocol (IP) address spoofing.

Another threat model that could be considered could be more aggressive threats, such as bad actors jamming or interfering with UAV-LU communications links. This could require the introduction of other wireless technologies and models to combat any form of interference from a bad actor or other forms of electronic warfare.

Further and more robust authentication models to differentiate between LUs and bad actors would have to be incorporated within the system to combat this, potentially requiring both higher-level digital and physical layer security being incorporated into the system.

Other means of classifying LUs and bad actors could also be incorporated to tighten the security of the system and secrecy of the communications even more.

5.3.6 Alternative Quantum Computing Techniques

Other quantum computing techniques that were explored in the literature review of this thesis, such as quantum annealing could be tested and experimented with this system model to solve the joint optimisation problem.

Quantum kernels could also be utilised for the process of identification or more robust authentication between LUs and Eves or other bad actors. A jamming and interference UAV could utilise a quantum kernel technique for classifying LUs and bad actors in a dynamic environment.

Another technique that could be attempted would be to use amplitude encoding instead of angle encoding for the quantum-assisted DRL algorithm. Amplitude encoding is a far more efficient method of quantum embedding as for 2^n data points to be embedded as inputs to a quantum circuit, only n qubits would be required, thus, exponentially decreasing the size of the quantum circuit. This technique could exponentially decrease the complexity of the algorithm while also allowing for more data to be embedded into an ansatz. The lower quantum volume

of such a circuit could also potentially increase the risk of errors induced by the environment in an open quantum system, i.e., on a practical quantum computer, however, the effects of errors, such as bit-flip or phase-flip errors in the quantum channel could have a proportionally greater effect on the performance of the system, however, to the author's knowledge, a system such as the one outlined in this thesis has not been implemented with the use of amplitude encoding at the time of writing and undertaking research for this thesis. To achieve this, a novel preparation scheme would have to be devised and quantum tomography would have to be employed to ensure that the amplitudes of the states correspond correctly to the data that has been embedded into the quantum circuit.

Bibliography

- [1] A. Sharma, P. Vanjani, N. Paliwal, C. M. Basnayaka, D. N. K. Jayakody, H.-C. Wang, and P. Muthuchidambaranathan, "Communication and networking technologies for uavs: A survey," *Journal of Network and Computer Applications*, vol. 168, p. 102739, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1084804520302137>
- [2] W. Saad, M. Bennis, M. Mozaffari, and X. Lin, *Wireless Communications and Networking for Unmanned Aerial Vehicles*, 1st ed. Cambridge University Press, Apr. 2020. [Online]. Available: <https://www.cambridge.org/core/product/identifier/9781108691017/type/book>
- [3] Y. Zhang, Z. Na, W. Lu, and C. Ji, "One Friendly UAV Against Multiple Mobile Eves in HAP Networks: A Novel Physical Layer Secure Communication Scheme," *IEEE Transactions on Vehicular Technology*, vol. 74, no. 1, pp. 1352–1364, Jan. 2025. [Online]. Available: <https://ieeexplore.ieee.org/document/10659178/>
- [4] A. Ji, X. Guo, R. Zhang, J. Wu, and X. Cheng, "Joint HAP deployment and resource allocation for HAP-UAV-terrestrial integrated networks," *IET Communications*, vol. 17, no. 1, pp. 86–97, Jan. 2023. [Online]. Available: <https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/cmu2.12512>
- [5] A. Azizi, M. A. Kishk, and A. Farhang, "Exploring the Impact of HAPS-RIS on UAV-Based Networks: a Novel Network Architecture," 2024, version Number: 2. [Online]. Available: <https://arxiv.org/abs/2409.17817>
- [6] G. Mu, "Security Efficiency Maximization for Multi-UAV–Aided Network With Mobile Edge Computing," *Frontiers in Computer Science*, vol. 3, p. 691854, Jun. 2021. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fcomp.2021.691854/full>
- [7] S.-G. Jeong, P. D. A. Duc, Q. V. Do, D.-I. Noh, N. X. Tung, T. V. Chien, Q.-V. Pham, M. Hasegawa, H. Sekiya, and W.-J. Hwang, "Quantum Annealing-Based Sum Rate Maximization for Multi-UAV-Aided Wireless Networks," *IEEE Internet of Things Journal*, pp. 1–1, 2025. [Online]. Available: <https://ieeexplore.ieee.org/document/10907925/>
- [8] Silvirianti, B. Narottama, and S. Y. Shin, "Layerwise Quantum Deep Reinforcement Learning for Joint Optimization of UAV Trajectory and Resource Allocation," *IEEE Internet of Things Journal*, vol. 11, no. 1, pp. 430–443, Jan. 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10153404/>

Bibliography

- [9] Y. Li, A. H. Aghvami, and D. Dong, "Intelligent Trajectory Planning in UAV-Mounted Wireless Networks: A Quantum-Inspired Reinforcement Learning Perspective," *IEEE Wireless Communications Letters*, vol. 10, no. 9, pp. 1994–1998, Sep. 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9456900/>
- [10] M. Saravanan and V. Pathmanaban, "Optimizing UAV Base Station Positioning through Quantum-Inspired Solution Workflow," in *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. Abu Dhabi, United Arab Emirates: IEEE, Apr. 2024, pp. 347–352. [Online]. Available: <https://ieeexplore.ieee.org/document/10621505/>
- [11] K. Namuduri, S. Chaumette, J. H. Kim, and J. P. G. Sterbenz, Eds., *UAV Networks and Communications*, 1st ed. Cambridge University Press, Nov. 2017. [Online]. Available: <https://www.cambridge.org/core/product/identifier/9781316335765/type/book>
- [12] P. Lohan, D. Mishra, J. Nguyen, and V. Gupta, "Secrecy-aware UAV position-aided jamming for practical eavesdropper localization models," *Vehicular Communications*, vol. 33, p. 100430, Jan. 2022. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S2214209621000991>
- [13] Y. Zhang and D. Mishra, "Energy-Efficient UAV-Relayed High-altitude Platform to Ground User Communication," in *2024 IEEE 35th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. Valencia, Spain: IEEE, Sep. 2024, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/10817374/>
- [14] X. Qin, Z. Na, X. Wu, M. Xiong, and B. Lin, "Secure Mobile Edge Computing for Integrated HAP and UAV Networks," in *2023 IEEE/CIC International Conference on Communications in China (ICCC)*. Dalian, China: IEEE, Aug. 2023, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/10233325/>
- [15] H. Zhao and Z. Chang, "Energy Efficient Trajectory Optimization and Resource Allocation for HAP-Assisted UAV Wireless Networks," in *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*. Kuala Lumpur, Malaysia: IEEE, Dec. 2023, pp. 3765–3770. [Online]. Available: <https://ieeexplore.ieee.org/document/10436986/>
- [16] K. Namuduri, Y. Wan, and M. Gomathisankaran, "Mobile ad hoc networks in the sky: state of the art, opportunities, and challenges."
- [17] U. Challita, A. Ferdowsi, M. Chen, and W. Saad, "Machine Learning for Wireless Connectivity and Security of Cellular-Connected UAVs," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 28–35, Feb. 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8641422/>

Bibliography

- [18] O. K. Sahingoz, "Mobile networking with UAVs: Opportunities and challenges," in *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*. Atlanta, GA, USA: IEEE, May 2013, pp. 933–941. [Online]. Available: <http://ieeexplore.ieee.org/document/6564779/>
- [19] L. Yan, C. Wang, and W. Zheng, "Secure efficiency maximization for UAV-assisted mobile edge computing networks," *Physical Communication*, vol. 51, p. 101568, Apr. 2022. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1874490721002718>
- [20] Silvirianti and S. Y. Shin, "Energy-Efficient Multidimensional Trajectory of UAV-Aided IoT Networks With Reinforcement Learning," *IEEE Internet of Things Journal*, vol. 9, no. 19, pp. 19 214–19 226, Oct. 2022. [Online]. Available: <https://ieeexplore.ieee.org/document/9750200/>
- [21] C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, "Trapped-ion quantum computing: Progress and challenges," *Applied Physics Reviews*, vol. 6, no. 2, May 2019. [Online]. Available: <http://dx.doi.org/10.1063/1.5088164>
- [22] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Applied Physics Reviews*, vol. 6, no. 2, Jun. 2019. [Online]. Available: <http://dx.doi.org/10.1063/1.5089550>
- [23] A. Chatterjee, P. Stevenson, S. De Franceschi, A. Morello, N. P. de Leon, and F. Kuemmeth, "Semicdonductor qubits in practice," *Nature Physics Reviews*, 2021. [Online]. Available: <https://www.nature.com/articles/s42254-021-00283-9#citeas>
- [24] P. Kok, W. J. Munro, K. Nemoto, T. C. Ralph, J. P. Dowling, and G. J. Milburn, "Linear optical quantum computing with photonic qubits," *Reviews of Modern Physics*, vol. 79, no. 1, p. 135–174, Jan. 2007. [Online]. Available: <http://dx.doi.org/10.1103/RevModPhys.79.135>
- [25] S. Yarkoni, E. Raponi, T. Bäck, and S. Schmitt, "Quantum Annealing for Industry Applications: Introduction and Review," *Reports on Progress in Physics*, vol. 85, no. 10, p. 104001, Oct. 2022, arXiv:2112.07491 [quant-ph]. [Online]. Available: <http://arxiv.org/abs/2112.07491>
- [26] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing."
- [27] R. Battiti, "First- and Second-Order Methods for Learning: Between Steepest Descent and Newton's Method," *Neural Computation*, vol. 4, no. 2, pp. 141–166, Mar. 1992. [Online]. Available: <https://direct.mit.edu/neco/article/4/2/141-166/5631>
- [28] N. Munikote, "Comparing Quantum Encoding Techniques," Oct. 2024, arXiv:2410.09121 [quant-ph]. [Online]. Available: <http://arxiv.org/abs/2410.09121>

Bibliography

- [29] M. Rath and H. Date, "Quantum Data Encoding: A Comparative Analysis of Classical-to-Quantum Mapping Techniques and Their Impact on Machine Learning Accuracy," Nov. 2023, arXiv:2311.10375 [quant-ph]. [Online]. Available: <http://arxiv.org/abs/2311.10375>
- [30] A. Skolik, S. Jerbi, and V. Dunjko, "Quantum agents in the Gym: a variational quantum algorithm for deep Q-learning," *Quantum*, vol. 6, p. 720, May 2022, arXiv:2103.15084 [quant-ph]. [Online]. Available: <http://arxiv.org/abs/2103.15084>
- [31] R. C. Kizilirmak, "Non-orthogonal multiple access (noma) for 5g networks," in *Towards 5G Wireless Networks*, H. K. Bizaki, Ed. Rijeka: IntechOpen, 2016, ch. 4. [Online]. Available: <https://doi.org/10.5772/66048>
- [32] S. O. Rice, "Statistical Properties of a Sine Wave Plus Random Noise," *Bell System Technical Journal*, vol. 27, no. 1, pp. 109–157, Jan. 1948. [Online]. Available: <https://ieeexplore.ieee.org/document/6767541>
- [33] C. You and R. Zhang, "3D Trajectory Optimization in Rician Fading for UAV-Enabled Data Harvesting," *IEEE Transactions on Wireless Communications*, vol. 18, no. 6, pp. 3192–3207, Jun. 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8698468/>
- [34] D. Li, G. Huang, and Y. Meng, "Joint Trajectory and Power Optimization for UAV Relay Network over Time-Varying Rician Channel," in *2024 4th International Conference on Neural Networks, Information and Communication (NNICE)*. Guangzhou, China: IEEE, Jan. 2024, pp. 1776–1781. [Online]. Available: <https://ieeexplore.ieee.org/document/10498163/>
- [35] J. Zhang, J. F. Campbell, D. C. Sweeney li, and A. C. Hupman, "Energy consumption models for delivery drones: A comparison and assessment," *Transportation Research Part D: Transport and Environment*, vol. 90, p. 102668, Jan. 2021, publisher: Elsevier BV. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1361920920308531>
- [36] B. Theys and J. D. Schutter, "Forward flight tests of a quadcopter unmanned aerial vehicle with various spherical body diameters," *International Journal of Micro Air Vehicles*, vol. 12, p. 1756829320923565, 2020. [Online]. Available: <https://doi.org/10.1177/1756829320923565>
- [37] Silvirianti, B. Narottama, and S. Y. Shin, "UAV Coverage Path Planning With Quantum-Based Recurrent Deep Deterministic Policy Gradient," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 7424–7429, May 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10374227/>

Bibliography

- [38] K. Murphy, "Reinforcement learning: An overview," 2025. [Online]. Available: <https://arxiv.org/abs/2412.05265>
- [39] D. Wierichs, J. Izaac, C. Wang, and C. Y.-Y. Lin, "General parameter-shift rules for quantum gradients," *Quantum*, vol. 6, p. 677, Mar. 2022, arXiv:2107.12390 [quant-ph]. [Online]. Available: <http://arxiv.org/abs/2107.12390>
- [40] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized Experience Replay," Feb. 2016, arXiv:1511.05952 [cs]. [Online]. Available: <http://arxiv.org/abs/1511.05952>

A1 Figures & Data

A1.1 Results for Increasing Numbers of Layers

A1.1.1 UAV Trajectory

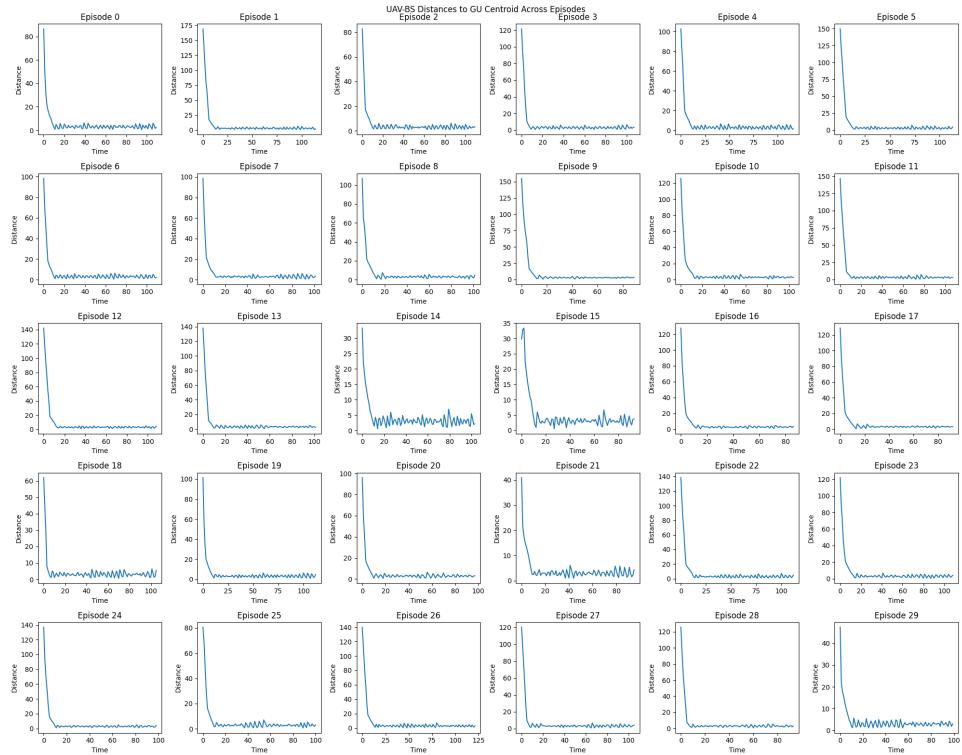


Figure A1.1: Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 1$

A1. Figures & Data

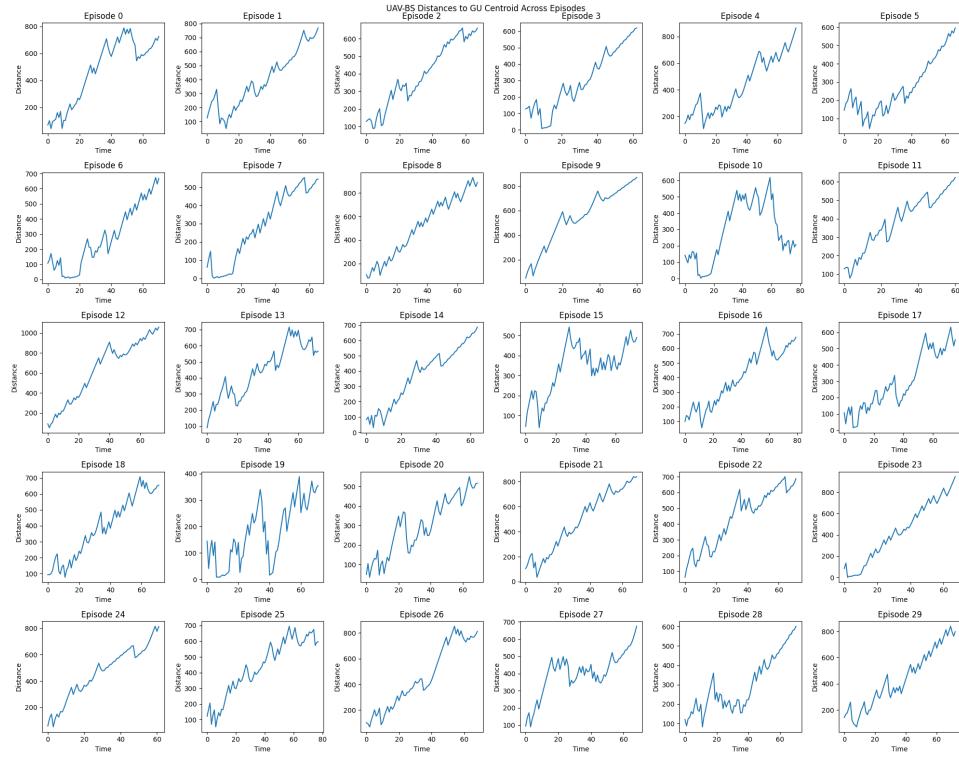


Figure A1.2: Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 2$

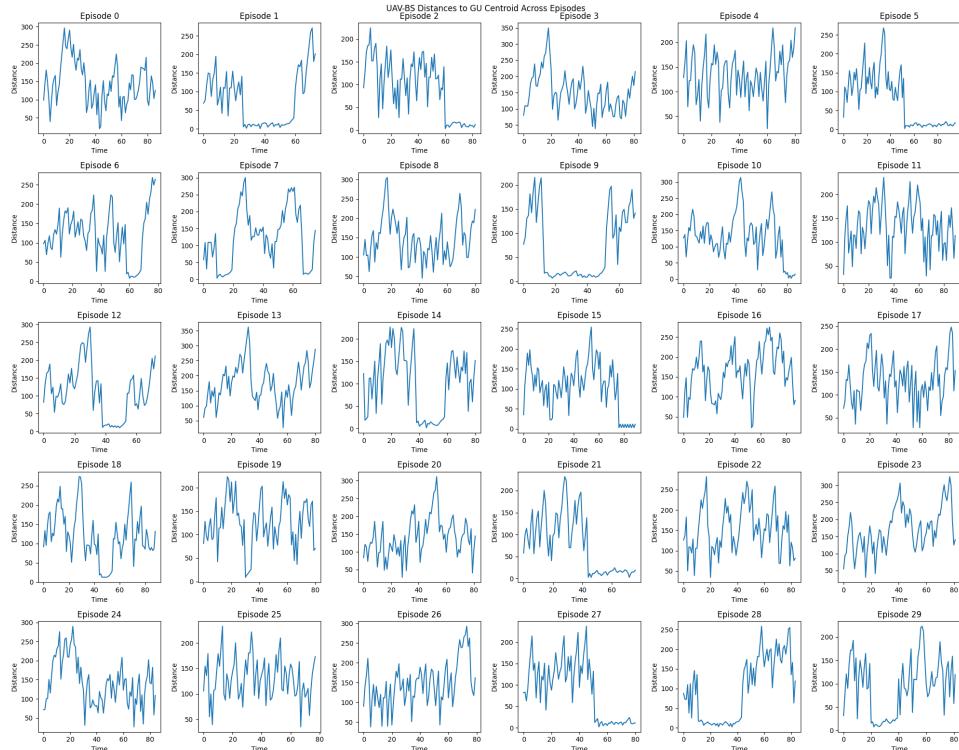


Figure A1.3: Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 3$

A1. Figures & Data

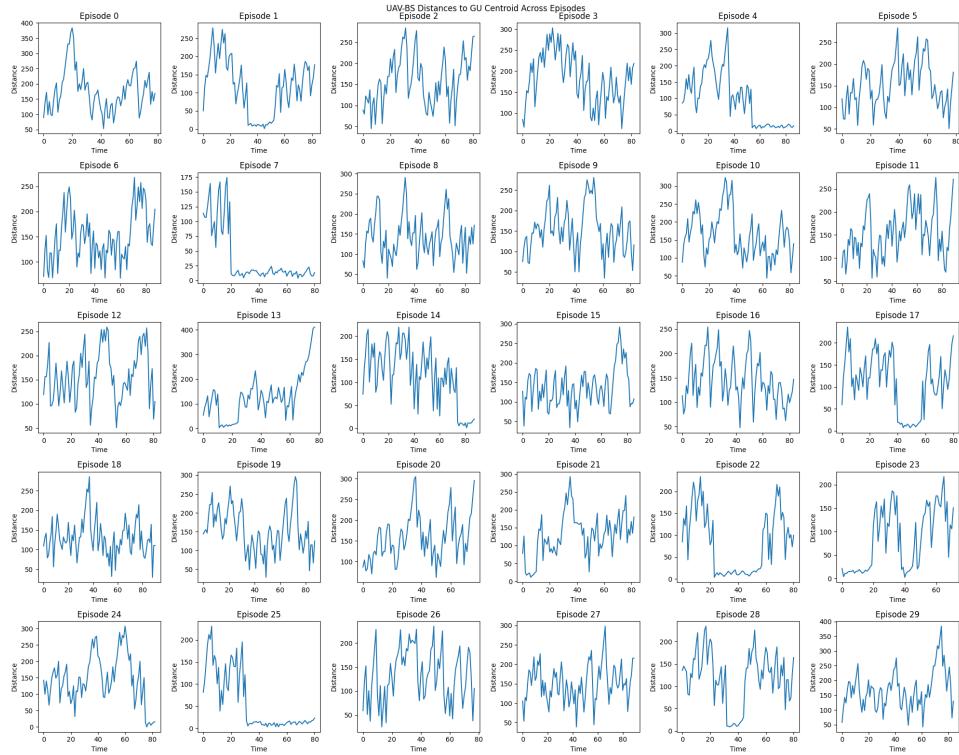


Figure A1.4: Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 4$

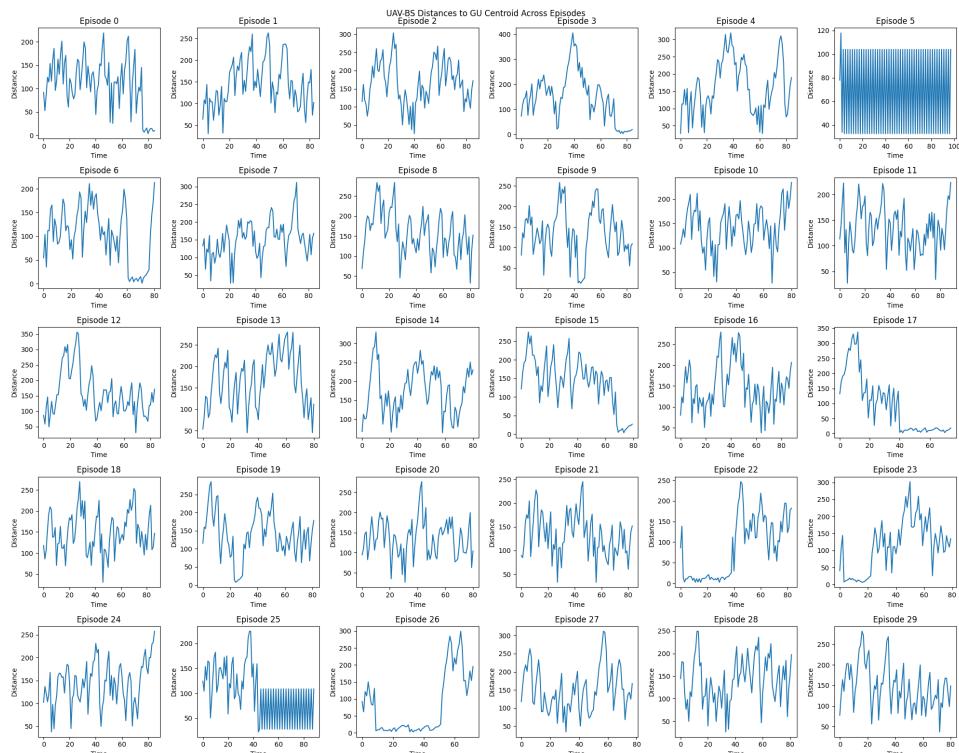


Figure A1.5: Max, Min & Mean Distances to Centroid Across 30 Episodes for $M = 5$

A1.1.2 Data Exchange Rate

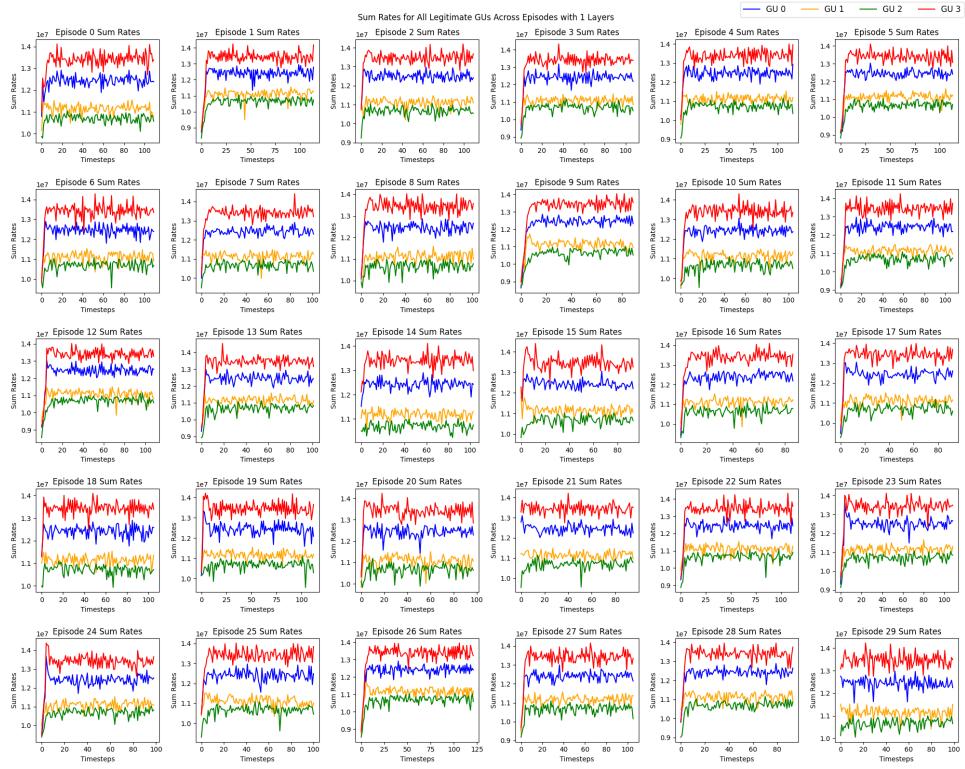


Figure A1.6: Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 1$

A1. Figures & Data

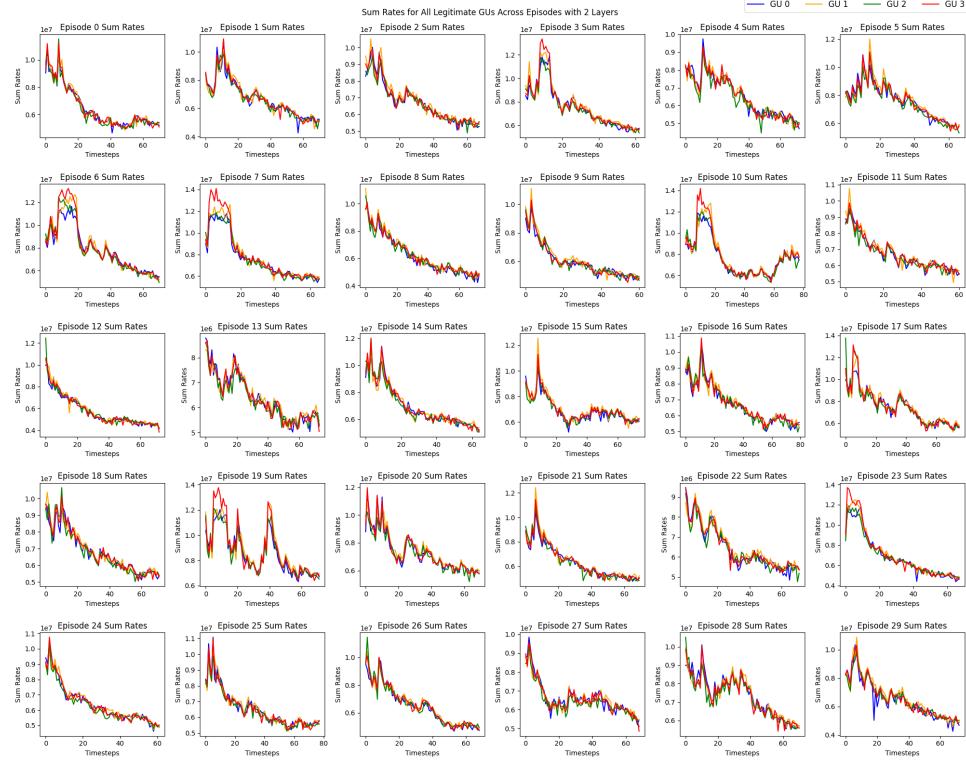


Figure A1.7: Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 2$

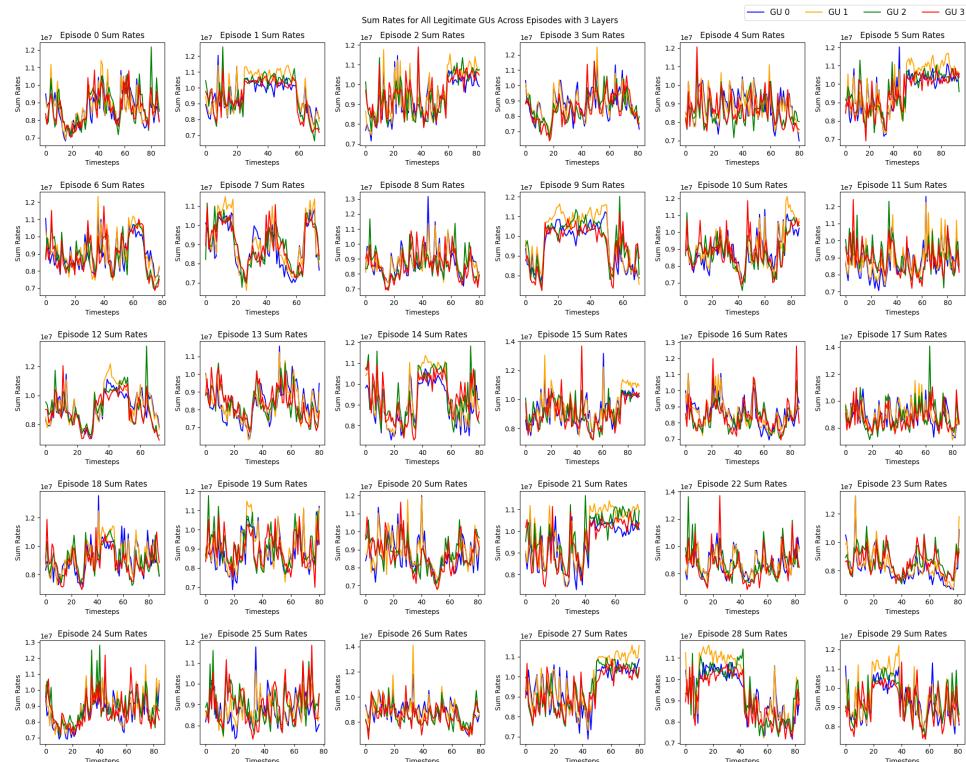


Figure A1.8: Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 3$

A1. Figures & Data

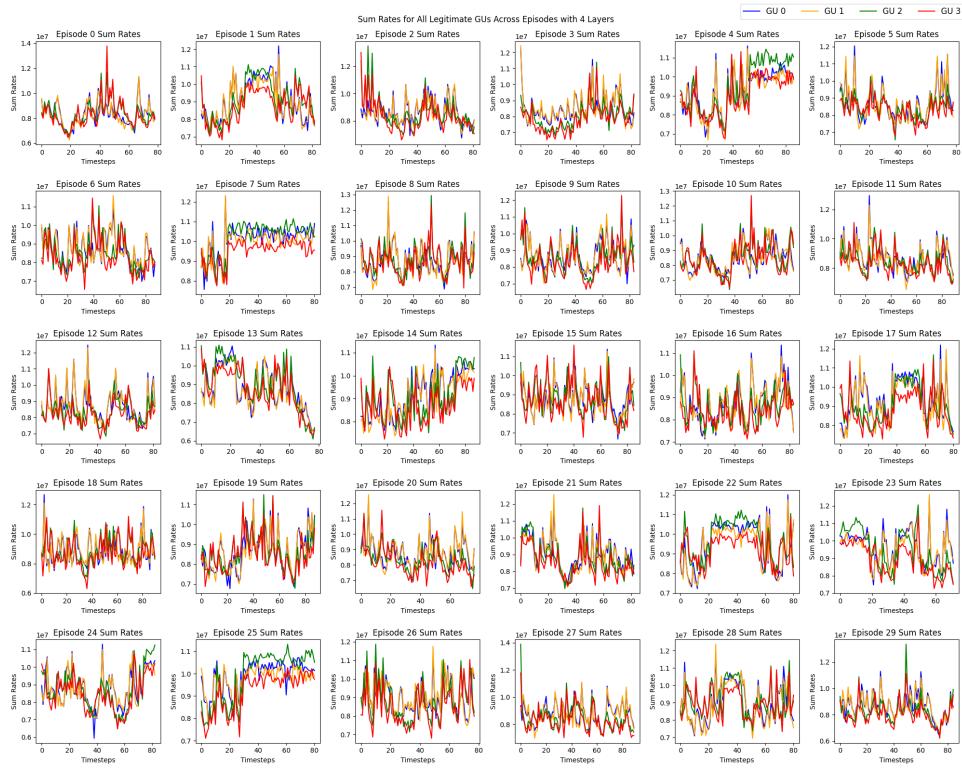


Figure A1.9: Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 4$

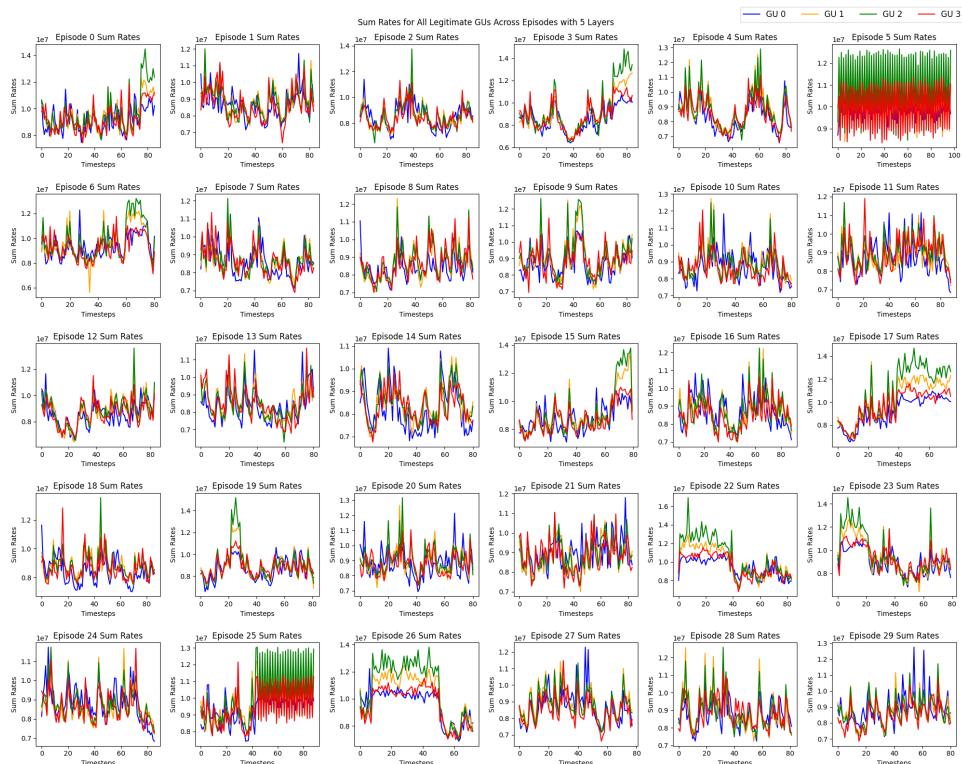


Figure A1.10: Max, Min & Mean Data Exchange Rates Across 30 Episodes for $M = 5$