Trajectory Scheduling of Multi-UAV System in Infrastructure Deficient Environment

A Thesis Submitted

in Partial Fulfilment of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

by

Amanjot Kaur

(2015CSZ0001)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY ROPAR

Dec, 2023

Dedicated to
My Parents,
My Husband
and
My Daughter

Declaration of Originality

I hereby declare that the work that is being presented in the thesis entitled **Trajectory Scheduling of Multi-UAV System in Infrastructure Deficient Environment** has been solely authored by me. It presents the result of my independent investigation/research conducted during the period from Jan 2016 to Dec 2023 under the supervision of Dr. Shashi Shekhar Jha, Assistant Professor in the Department of Computer Science & Engineering at the Indian Institute of Technology Ropar and Prof. Jiong Jin from School of Science, Computing and Engineering Technologies, Swinburne University of Technology, Melbourne, Australia. To the best of my knowledge, it is an original work, both in terms of research content and narrative, and has not been submitted or accepted elsewhere, in part or in full, for the award of any degree, diploma, fellowship, associateship, or similar title of any university or institution. Further, due credit has been attributed to the relevant state-of-the-art and collaborations (if any) with appropriate citations and acknowledgments, in line with established ethical norms and practices. I also declare that any idea/data/fact/source stated in my thesis has not been fabricated/ falsified/ misrepresented. All the principles of academic honesty and integrity have been followed. I fully understand that if the thesis is found to be unoriginal, fabricated, or plagiarized, the Institute reserves the right to withdraw the thesis from its archive and revoke the associated Degree conferred. Additionally, the Institute also reserves the right to appraise all concerned sections of society of the matter for their information and necessary action (if any). If accepted, I hereby consent for my thesis to be available online in the Institute's Open Access repository, inter-library loan, and the title & abstract to be made available to outside organizations.

Amanjot Kaur
Signature

Name: Amanjot Kaur
Entry Number: 2015CSZ0001
Program: PhD
Department: Computer Science & Engineering
Indian Institute of Technology Ropar
Rupnagar, Punjab 140001

Date: 21/12/23

Acknowledgement

I am deeply thankful for the support and guidance I've received. I would like to express my profound appreciation to my thesis advisor, Dr. Shashi Shekhar Jha and Prof. Jiong Jin, for their constant encouragement and invaluable guidance throughout my research endeavors.

I also want to extend my heartfelt thanks to my Doctoral committee members, Dr. Suman, Dr. V. Gunturi, Dr. Apurv Mudgal, and Dr. Nitin Auluck. Their insightful feedback, constructive critiques, and thoughtful recommendations have been pivotal in steering both the direction and success of my research journey.

My sincere thanks go to my peers and colleagues for their companionship and support during this challenging yet rewarding journey.

I am deeply grateful for the unwavering support of all my family members and in-laws throughout this journey. Their encouragement has been invaluable to me.

Sincerely,

Amanjot Kaur

Certificate

This is to certify that the thesis entitled **Trajectory Scheduling of Multi-UAV System in Infrastructure Deficient Environment**, submitted by **Amanjot Kaur** for the award of the degree of **Doctor of Philosophy** of Indian Institute of Technology Ropar, is a record of bonafide research work carried out under my guidance and supervision. To the best of my knowledge and belief, the work presented in this thesis is original and has not been submitted, either in part or full, for the award of any other degree, diploma, fellowship, associateship or similar title of any university or institution. In my (our) opinion, the thesis has reached the standard fulfilling the requirements of the regulations relating to the Degree.

Signature of the Supervisor(s)
Dr. Shashi Shekhar Jha
Department of Computer Science & Engineering
Indian Institute of Technology Ropar
Rupnagar, Punjab 140001

Signature of the Supervisor(s)
Prof. Jiong Jin
School of Science, Computing and Engineering Technologies
Swinburne University of Technology, Melbourne, Australia

Lay Summary

Our research is centered around the use of drones, or Unmanned Aerial Vehicles (UAVs), in places where the infrastructure isn't well-developed, like in remote or underdeveloped regions. The study explores how UAVs can be effectively used for collecting and sharing information in such challenging environments. The main idea is to develop a system where one type of UAVs, called the Access UAV, manages and coordinates the activities of other UAVs that are equipped with cameras and a limited range of communication. These camera-equipped UAVs, referred to as Inspection-UAVs, are responsible for collecting visual data from various locations and then sending this information to Base Station via Access UAV for processing and storage. The Access UAV plays a critical role in deciding the trajectory scheduling of the Inspection-UAVs, ensuring they collect data efficiently from the right places. The approach aims to make sure that all areas of interest are covered fairly and that the UAVs operate in an energy-efficient manner, which is important for long-duration missions. Additionally, the research includes a strategy to keep the data collection process smooth and data queues of the network stable, ensuring that all information is gathered and transmitted without delays or backlogs.

Overall, the work presents hierarchical multi-UAV network that learns and adapts, ensuring effective and efficient data gathering and communication in challenging environments.

Abstract

The use of unmanned aerial vehicles (UAVs) is rapidly growing in research, particularly for surveillance and communication in areas without developed infrastructure. Their versatility allows for a wide range of applications, including remote sensing, traffic monitoring, and target tracking. By deploying a network of multiple UAVs, extensive areas can be covered efficiently, enabling synchronized operations that are both quick and cost-effective. This is especially crucial for real-time monitoring tasks where consistent and reliable communication is key to maintaining high-quality service.

However, when it comes to real-time monitoring tasks, uninterrupted and reliable communication channels become crucial to maintain a high Quality of Service (QoS). This continuous connectivity is essential for the effective and seamless functioning of UAV-based systems, especially in scenarios that demand constant and accurate data transmission. This thesis introduces a multi-UAV system designed for efficient data collection in resource-limited settings. The multi-UAV system is comprised of two types of UAVs: the Access UAV (A_UAV) and Inspection-UAVs (I_UAVs). These UAVs differ in terms of their operational capabilities and maneuverability in the environment. The A_UAV serves as a central access platform, coordinating the data collection efforts of I_UAVs, each equipped with a visual sensor for capturing and relaying data to the cloud. This system is engineered to optimize the trajectory of both A_UAV and I_UAV, ensuring data is collected from designated points in a decentralized fashion.

For optimizing the trajectories of the UAVs, this thesis introduces the Distance and Access Latency Aware Trajectory (DLAT) optimization specifically for the A_UAVs. This optimization method plays a crucial role in balancing the trajectory planning with the need to minimize the consumption of total system energy for end to end data offloading from I_UAVs to the base stations. In addition, a Lyapunov-based online optimization strategy is employed to ensure the stability of the system, particularly focusing on the average queue backlogs that is critical for dynamic data collection. To facilitate effective coordination between the I_UAV and A_UAV, the system incorporates a message-based mechanism. This aspect is essential for ensuring that data collection and transmission are synchronized and efficient.

Further, the thesis delves in the aspect of Age-of-Information (AoI) of the data being collected. A Deep Reinforcement Learning (DRL) framework-based model is conceived utilizing an actor-critic deep network for learning the optimal policy for the A_UAVs to minimize the AoI of the data. The AoI problem is mapped to the Markov Decision Processes (MDP) with a curated reward function to solve trajectory scheduling for the A_UAV. RL provides a robust framework for modeling the decision-making process, considering the stochastic nature of UAV environments and various parameters of the state space such as location, battery levels, and environmental factors. Experiments are performed against multiple baselines with different parameter settings and multiple seeds. The proposed approaches in this thesis have shown improved performances against the available baselines and the methods prevalent in the literature.

Keywords: Deep Reinforcement Learning (DRL); Age of Information (AoI); Data Collection; Unmanned Aerial Vehicles (UAVs); Access Schedule; Multi-UAV System;

List of Publications

Journal

- Kaur A, Jha SS, Jin J, Ghaderi H. Optimizing Trajectory and Dynamic Data Offloading Using a UAV Access Platform. *IoT* 2022 Nov 24;3(4):473-92.
- A. Kaur and S. S. Jha. Age-of-Information based Multi-UAV Trajectories using Deep Reinforcement Learning. *IETE Technical Review* May 2024;1–13.

Symposium

- Kaur A, Multi-UAV based Data collection and relay using Intelligent Edge Device *IBM-IEEE AI Compute Symposium (AICS2021)*

Contents

List of Figures

List of Algorithms

List of Symbols

T Set of time slots

N Set of I_UAVs deployed to collect images

$Q_i(t)$ The queue of the i^{th} I_UAV in time slot t

$\mathcal{S}_i(t)$ The position of the I_UAV in time slot t

$\mathcal{S}_{access}(t)$ The position of A_UAV in time slot t

L(t) The queue length of A_UAV server in time slot t

$d_i^{off}(t)$ The amount of data offloaded to A_UAV by the I_UAV in time slot t

$A_i(t)$ The amount of data bits arrived at i^{th} I_UAV in time slot t

$d_{access}^{off}(t)$ The amount of data offloaded by the A_UAV in time slot t

$p_i(t)$ Transmission power of i^{th} I_UAV in time slot t

$P_{access}(t)$ Transmission power of A_UAV in time slot t

τ Transmission power of A_UAV in time slot t

v(t) Transmission power of A_UAV in time slot t

e_o, e_1 Environmental constants

$E_i^{comm}(t)$ Transmission energy of I_UAV

$E_{access}^{transition}(t)$ Transition energy of the A_UAV in time slot t

$E_{access}^{comm}(t)$ Transmission energy of A_UAV in time slot t

V The tradeoff parameter between transition and transmission energy

g_0 Path loss constant

ϕ The tradeoff parameter between transition and transmission energy

ζ Channel power gain

ψ_i 3D coordinates of the PoI

N_0 Power spectral density of noise

θ Elevation angle of A_UAV

$x_i(t)$ Elevation angle of A_UAV

σ standard deviation of normal distribution used for data generation at each PoI

ϱ the maximum distance between two consecutive PoIs in the trajectory of I_UAVs

μ mean value of normal distribution used for data generation at each PoI

Q_{max} max buffer of I_UAV

τ_{search} search time to find the exact location of I_UAVs

τ_{comm} time allotted for data transmission

τ_{trans} time taken to transit from one location to another

l_i tuple with location and data information

$B_i(t)$ the battery of i^{th} I_UAV

$B_{access}(t)$ the battery of A_UAV

$r_i^{penalty}(t)$ reward penalty of i^{th} I_UAV

$r_i(t)$ reward of i^{th} I_UAV

$r_{access}(t)$ reward of A_UAV at time slot t

$CI_i(t)$ crowd density of i^{th} I_UAV at time slot t

$RL_i^{avg}(t)$ average access latency of PoIs in the cluster of i^{th} I_UAV at time slot t.

List of Abbreviations

Terms	**Abbreviations**
UAV	Unmanned Aerial Vehicle
AoI	Age of Information
DLAT	Distance and Latency Aware Trajectory
HDLAT	Hybrid Distance and Latency Aware Trajectory
DAT	Distance Aware Trajectory
MDP	Markov Decision Process
PoI	Point of Interest
DQN	Deep Q-Network
RR	Round Robin
A2C	Advantage Actor-Critic
DDPG	Deep Deterministic Policy Gradient
MADDPG	Multi-agent Deep Deterministic Policy Gradient
MaxAF	Maximal AoI First
MinDF	Minimum Distance First
GA	Genetic Algorithm
PSO	Particle Swarm Optimization

1| Introduction

In the past decade, a dramatic rise in the use of Unmanned Aerial Vehicles (UAVs) across various sectors has been observed. Their ability to adapt to and meet the complex requirements of modern tasks has clearly demonstrated their value and utility, making UAVs a key solution in a rapidly evolving market. Unmanned Aerial Vehicles (UAVs) are now coming up in all sorts of innovative roles across different fields such as [1], payload delivery [2], precision agriculture [3] and search and rescue operations [4]. Also, the focus is growing on automation, sensing technologies, and information exchange in the latest technical solutions deployed in various scenarios. UAV-based applications are proving reliable options in industries such as construction, mining, agriculture, and logistics, especially for monitoring operations and managing resource utilization. UAV-based solutions are particularly helpful as they can be easily deployed for data collection from large infrastructure-deficient environments [5, 6]. Additionally, using UAVs that are either autonomous or semi-autonomous can streamline various evaluations, such as tracking project progress, checking resources and safety, and spotting environmental risks.

In stochastic scenarios, the deployment of multiple Unmanned Aerial Vehicles (UAVs) based solutions offers improved capabilities for gathering information and surveillance. The requirement for uninterrupted communication is particularly inevitable in scenarios such as crowd monitoring, where timely and accurate information is required to ensure public safety and security. Additionally, in applications like remote sensing, continuous communication ensures that the gathered data is transmitted in real-time, allowing for immediate analysis and decision-making [7, 8, 9]

All multi-UAV applications require the coordination of agents in the field. The coordination among the UAVs depends on their location, capabilities, and other constraints. Similarly, trajectory scheduling of UAVs should be such that they can collectively achieve the system objective. Although integrating a multi-UAV based visual sensing and monitoring system has many benefits, developing such a system is challenging. A few of those challenges along with financial budget limitations that restricts the number of deployed UAVs, are as follows:

- limited battery of UAVs limits the observation span
- restricted on-board processing ability of UAVs makes online offloading of data necessary
- limited ability to connect for data gathering and offloading tasks in infrastructure-deficient environments that require smart trajectory planning.

The task of collecting data using UAVs is computationally intensive and limited on-board computation available with UAVs is a hurdle in the deployment of such solutions. To collect and/or process data within large sites with poor infrastructure, such as monitoring the progress of complex construction sites with limited battery adds another challenge. Many applications mentioned in existing literature often assume the existence of a continuous and reliable communication infrastructure. However, this assumption proves to be less reasonable, particularly in critical situations such as disaster response or emergencies. In these scenarios, the reliance on persistent communication infrastructure is a weak assumption. In such scenarios, UAV relay networks become crucial, as they provide essential connectivity in instances where direct access to the Base Station is unavailable [5, 6].

In surveillance applications, the effectiveness of UAV trajectory scheduling is critically evaluated based on the nature of the data being collected. This evaluation primarily revolves around two pivotal metrics: access latency and Age of Information (AoI). Access latency refers to the time taken for data to be collected and transmitted to the relevant endpoints or decision-making centers. This metric is crucial in time-sensitive scenarios where rapid data collection and processing are vital, such as in emergency response or real-time security monitoring. On the other hand, AoI [10] defines how fresh or up-to-date the received information is. In surveillance operations, having the most current data is often as critical as the speed of its acquisition. The AoI is defined as the time elapsed since the last piece of data was gathered, emphasizing the need for constant updates and ensuring that decision-makers have access to the latest information.

Both access latency and AoI play important role in the trajectory planning of UAVs in surveillance missions. The flight paths and schedules of UAVs must be carefully designed to minimize delays in data transmission (access latency) while regularly updating the collected data (minimizing AoI). This necessitates advanced algorithms and strategic planning in UAV operations, focusing on optimizing routes for quick data acquisition and timely updates, thus ensuring the overall efficacy and responsiveness of the surveillance system.

The multi-UAV coordination requires UAVs should make decisions autonomously. In autonomous decision-making and coordinating multiple UAVs, Reinforcement Learning (RL) could help find optimal policies [11, 12]. It helps these systems to quickly adapt to unseen environments. This is important for UAVs working in stochastic environments. They can learn to make decisions under uncertain conditions by exploring different actions and updating their policies based on outcomes. RL models can generalize their learned policies to new, unseen environments. This ability to generalize allows RL agents to apply knowledge gained in one context to similar but previously unseen scenarios. RL can be extended to handle multi-agent systems as well, where multiple learning agents interact with each other. This is beneficial in scenarios where coordination, competition, or cooperation among multiple entities is required.

1.1 Motivation, Objectives And Scope

Monitoring events or activities in an area can be challenging, especially when it comes to Points of Interest (PoIs) that are inaccessible to ground-based monitoring systems. It's often hard to predict how these situations will develop without proper observation tools.

It's really important to keep a close eye on events as they unfold and respond in the right way. In these cases, we need a system that can track the event's progress and keep the information current. Take surveillance in construction sites, for instance. To properly monitor how the work is progressing, we need data from areas that are hard to reach. This is where a system that can be quickly set up and adjusted becomes essential, especially in places lacking the necessary infrastructure. The same goes for managing crowds; understanding and following the dynamic changes in a crowd can be quite challenging.

Multi-UAV systems are typically tailored to tackle challenges specific to particular research areas within a chosen environment. In these systems, UAVs work together and exchange data to gather insights about specific elements of that environment. The collaboration of UAVs, regardless of their quantity or the size of the Area of Interest (AOI), leads to a more comprehensive understanding of the environment. This is achieved through coordinated efforts, where each UAV contributes a piece of the larger puzzle. By pooling their resources and capabilities, these UAVs can cover more ground, collect diverse data, and offer a richer, more detailed view than a single UAV could. This collaborative approach not only enhances data quality but also improves efficiency in tasks such as mapping, surveillance, and environmental monitoring, making it a valuable strategy in various fields. Coordination in multi-UAV systems often involves sharing observations, a key operation in many recent studies. These studies typically focus on the challenge of limited information, reflecting real-world scenarios where UAVs face communication constraints like limited range or bandwidth. In the study by Wan et al.[13], the researchers created a hierarchical Mobile Edge Computing (MEC) system. This system focuses on the online optimization of computational resources and employs reinforcement learning for trajectory optimization of multiple UAV Base Stations (UAV-BSs) tasked with data collection from a network of static sensors. In a separate study by Zhan et al. [14], the aim was to minimize the energy consumption and optimize the trajectory of a UAV, with a particular focus on reducing the task's completion time. The deployment of multi-UAV solutions for applications such as surveillance often relies on the assumption of a persistent communication infrastructure. However, this is not universally available, presenting significant challenges in coordinating multiple UAVs where connectivity is sporadic and bandwidth is limited. This lack of ubiquitous communication becomes particularly problematic given the constraints of UAV battery life.

Furthermore, an additional issue arises with buffer overflow in UAVs during data collection and offloading tasks. Limited onboard processing capabilities, coupled with the necessity to share bandwidth for data transfer, can lead to the overall system becoming unstable.

This situation is exacerbated by varying data traffic volumes and the continuous movement of UAVs, which complicates the task of stabilizing the system predictably.

Moreover, ensuring efficient energy management and optimizing flight paths in such unpredictable environments becomes a complex problem. The need to maintain a consistent data collection process and minimize the AoI of the system without persistent communication adds another layer of complexity. As UAVs operate in these dynamic conditions, they must also navigate physical obstacles and environmental factors, requiring advanced algorithms for autonomous decision-making and real-time adaptations.

Collectively, these challenges underscore the need for advanced research and development in the field of multi-UAV technology, particularly focusing on enhanced coordination, robust communication strategies, and adaptive system design to operate effectively in infrastructure-deficient environments. The challenges and solutions surrounding network and trajectory scheduling for UAV-based Mobile Edge Computing (MEC) systems are extensively explored. A significant area of application for these systems is crowd flow detection during large gatherings, where fixed cameras fall short due to their limited field of view. In this context, the deployment of UAVs equipped with visual sensors offers a dynamic solution for collecting visual data, as discussed by [15]. UAVs provide flexibility and broader coverage, making them ideal for monitoring large or crowded areas.

However, the use of multi-UAV based solutions in such scenarios has its own challenges. The primary constraints include the limited battery life and communication range of UAVs, which can significantly impact their operational efficiency. To mitigate these challenges, the literature suggests the adoption of UAV relayed networks. This approach enables extended coverage and enhanced data transmission capabilities, allowing UAVs to operate effectively over larger areas and for extended periods. Moreover, in applications like real-time image analysis for crowd management, the freshness of data is paramount. Delayed or outdated information can make such applications ineffective. Hence, AoI metric becomes crucial in evaluating the performance of these time-sensitive systems. AoI provides a measure of data timeliness, ensuring that the information used for decision-making is as current as possible.

In this thesis, we proposed two two-level heterogeneous multi-UAV framework. the system consists of Inspection UAVs (I_UAVs) which collect visual data and a single UAV Access Platform (A_UAV). The data is collected from dynamic sensors (I_UAVs) which send data to the Base Station (BS) in an environment where limited connectivity is present.

This thesis primarily focuses on the following research gaps:

- Investigate the Role of Multi-UAV Coordination in Resource-Deficient Environments: To explore and understand the significance of coordinating multiple UAVs in environments where resources are limited. This involves studying how such coordination impacts the effectiveness and efficiency of UAV operations in challenging settings.

- Develop a Trajectory Scheduling Framework for UAVs: To create a comprehensive framework for UAV trajectory scheduling. This framework will focus on enabling

UAVs to efficiently transfer data from dynamic sensors and include the optimization of multi-level queues within the UAV network, ensuring effective data management and transmission.

- Establish a Deep Reinforcement Learning-Based Coordination Framework for UAVs: To develop a framework based on Deep Reinforcement Learning (DRL) that facilitates coordination among UAVs. This approach focuses on achieving multiple goals for a complex system. It aims to reduce energy use, lower the AoI to ensure the latest data, and improve the data access latency. All of this will improve the overall efficiency of the multi-UAV system.

Based on the above research gaps, we identified the following research objectives.

1. The first objective is to develop to create algorithms that enable multi-UAV systems to function effectively in areas with limited infrastructure. This involves designing trajectory schedules for Access UAVs (A_UAVs) to efficiently collect data from dynamic sensors, which are deployed as Inspection UAVs (I_UAVs), especially considering their limited communication range.

2. The second objective focuses on developing a coordination framework for multiple UAVs operating in resource-deficient environments. A key aspect of this framework is to account for the limited buffer capacity of UAVs, ensuring optimal data handling and communication efficiency within the UAV network.

3. The third objective is to integrate a component that enables A_UAV to estimate the location of I_UAVs without the help of a central entity in an infrastructure-deficient environment.

4. The fourth objective is to implement a decentralized approach for scheduling the trajectories of A_UAVs in unpredictable, stochastic environments. This objective also includes minimizing critical operational parameters such as the AoI of the data and the overall energy consumption, ensuring an efficient and effective operational strategy.

5. The last objective is to implement a decentralized approach for scheduling the trajectories of I_UAVs in unpredictable, stochastic environments. The objective is to minimize the access latency of Points of Interest (PoIs), and energy consumption of UAVs.

The first three research objectives are addressed in Chapter 3, while the last two research objectives are covered in Chapter 4.
This thesis aims to demonstrate the effective trajectory scheduling in multi-UAV networks in resource-deficient environments. For each of the objectives mentioned above, the proposed solutions are highlighted in the research work, as outlined below:

1. An ILP-Based Trajectory Scheduling Method: We model the two level hierarchical multi-UAV system. designed and implemented a trajectory scheduling strategy for UAVs using Integer Linear Programming (ILP). This method would focus on UAVs deployed as dynamic sensors, with the primary objectives being to efficiently relay collected data, minimize system energy usage, and reduce access latency. This solution is to address the first objective.

2. Optimize Multi-Level Queue Management in Multi-UAV Networks: We employ a Lyapunov-based online optimization approach for managing multi-level queues within a multi-UAV network. The aim is to effectively handle the limited buffer capacities of UAVs, ensuring optimal data management and processing within the network. Additionally, we devised a technique that estimates the candidate locations of I_UAVs where A_UAV can look for them and collect the data. This solution addresses the second and third objectives.

3. Autonomous UAV Trajectory Scheduling Using DRL is designed for multi-UAV systems. We initially introduced a Markov Decision Process (MDP) formulation for the hierarchical structure of UAVs. This process involves separate encoding of state and action information for both I_UAVs and A_UAVs. As a result, the fourth objective, focusing on the use of RL algorithms for UAV trajectory scheduling, is accomplished through this MDP formulation.

4. Our final objective addresses the enhancement of trajectory scheduling of UAVs based on the Age of Information (AoI). We adjusted the reward system to optimize the overall system objective for both groups of UAVs. Each group of UAVs independently develops their policies, without the need for a centralized coordinating entity.

1.2 Summary of The Contributions

In this section, we provide a brief overview of each of the proposed solutions mentioned earlier.

1.2.1 Optimizing Trajectory and Dynamic Data Offloading using a UAV Access Platform

In our study, we introduce a heterogeneous multi-UAV system designed specifically for dynamic data collection in environments lacking robust infrastructure, as depicted in Figure 1.1. This system is composed of a single UAV access platform, termed as the Access UAV (A_UAV), and multiple Inspection UAVs (I_UAVs) The primary objective of this arrangement is to optimize the trajectory scheduling of the A_UAV to significantly reduce both energy consumption and access latency associated with the dynamic sensors deployed as I_UAVs. The system comprises heterogeneous UAVs, including a group of N Inspection UAVs (I_UAVs) and a single UAV Access Platform (A_UAV). The I_UAVs, which are smaller and more agile compared to the A_UAV, collect visual data

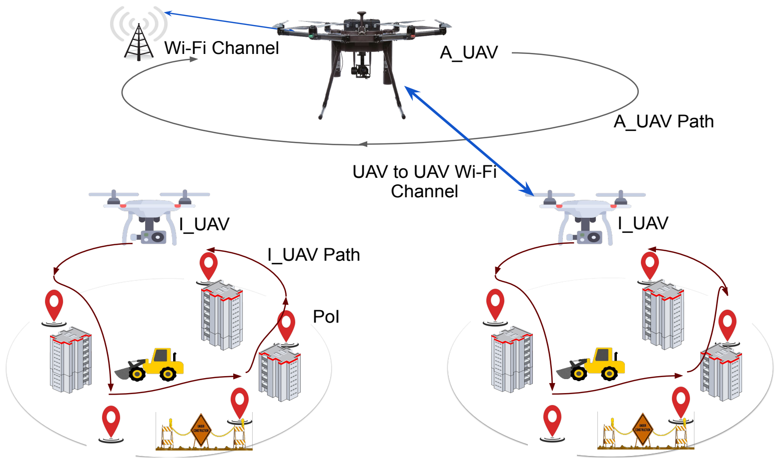

Figure 1.1: Overview of Multi-UAV Hierarchical Network

from a set of k Points of Interest (PoIs) represented as l_i. Operating in infrastructure-less environments with limited Access Points (APs) for cloud connectivity, the I_UAVs face challenges in direct data transfer to the cloud due to their limited connectivity range. The larger A_UAV, with higher computational capabilities, coordinates with the I_UAVs to collect data. It maintains a constant height, confining its trajectory to a horizontal plane. The A_UAV collects data from the dynamically moving I_UAVs and relays it to the cloud.

The primary objective of optimizing the Access UAV's (A_UAV) trajectory is to efficiently manage data collection and offloading from multiple Inspection UAVs (I_UAVs) in infrastructure-deficient environments, impacting energy consumption and access latency. By planning the A_UAV path strategically, unnecessary movements are minimized, conserving energy through Distance and Access Latency Aware Trajectory $DLAT$ optimization, which selects I_UAVs based on proximity. This optimization also reduces access latency by scheduling A_UAV visits efficiently, preventing I_UAV data queue overflows and enabling real-time trajectory adjustments to maintain low latency. In addition to trajectory optimization, our approach incorporates online optimization techniques to tackle system instabilities, particularly those arising from queue backlogs. This aspect of our work builds upon the foundational concepts introduced in [16, 17], further enhancing the system's efficiency and stability. The system involves a form of multilevel queue. The queue of data for each I_UAVs $Q_i(t)$ has its own queue where it stores the data it gathers from the points of interest (PoIs). This represents one level of the queue system. The queue of Data for the A_UAV (L(t)) has its queue where it accepts data from the selected I_UAVs in each time slot. This queue represents another level of the queue system. So, in this system, there are multiple levels of queues: one for each I_UAVs and another for the A_UAV. The collected data moves between these queues based on the system's operation and the offloading process.

Incorporating online optimization techniques, such as Lyapunov-based online optimization, is crucial for managing system instabilities caused by queue backlogs in multi-UAV systems. This adaptability is essential for maintaining the stability and

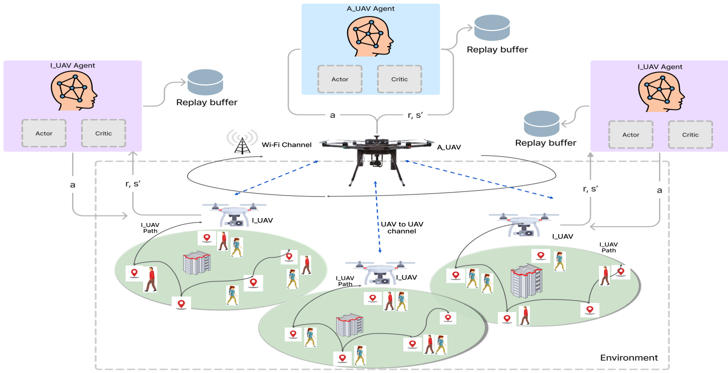

Figure 1.2: System Setup for Reinforcement Learning based Trajectory Scheduling

efficiency of UAV operations in infrastructure-deficient environments.

A crucial feature of our proposed system is the independent operation of the set of I_UAVs and A_UAV without any dependency on a central coordinating entity. To ensure effective coordination among these UAVs, we have implemented a message-based estimation of candidate locations of I_UAVs. This mechanism enables UAVs to share and update information, ensuring synchronized and efficient operations despite the lack of a centralized control system. We develop two ILP-Based algorithms for trajectory scheduling: the Distance and Latency Aware Trajectory (DLAT) and the Hybrid Distance and Latency Aware Trajectory (HDLAT), specifically for A_UAVs. The performance of these proposed algorithms is evaluated against baseline models, with focus on optimizing energy usage, covering more points, among others. Additionally, Lyapunov based framework ensures system stability, particularly in managing the multilevel queue of the system when all UAVs have limited buffer size. Overall, our work represents a significant advancement in multi-UAV coordination, addressing critical aspects like trajectory optimization, energy conservation, and autonomous coordination, which are vital for effective multiple UAV-based operations in various scenarios.

1.2.2 Age-of-Information based Multi-UAV Trajectories using Deep Reinforcement Learning

This research centers on utilizing UAVs to enhance the efficiency of crowd-monitoring systems by reducing the AoI. It introduces a network of multiple UAVs, where some are equipped with visual sensors (I_UAVs) and have limited transmission abilities, while one serves as a data relay (A_UAV) to a Base Station (BS). The focus is on planning the flight paths of these UAVs to significantly impact the system's AoI. Key to this is decentralized planning for each UAV's trajectory, coupled with optimizing their energy use. The project stands out by minimizing AoI and access latency in a UAV network without relying on a centralized controller for coordinating UAV movements. Challenges arise from limited communication ranges and the use of UAVs as dynamic data-generating sensors. This study is among the first to explore two-level decentralized trajectory planning, aiming to optimize energy use, AoI, and access latency in a UAV network using DRL. The system setup is shown in Figure 1.2. We employ algorithms such as Deep Deterministic Policy Gradients (DDPG) and Advantage Actor-Critic (A2C) methods. We validate our proposed approach to baseline strategies based on heuristics and meta-heuristic techniques. Various metrics such as Average AoI, average access latency, average cumulative reward, and total operational time among others are analyzed to quantify the performance of our proposed approach. The primary goal is to reduce both the AoI and energy consumption in a hierarchical multi-UAV network, especially in areas without direct access to a Base Station.

1.3 Outline of The Thesis

This section provides an overview of the overall organization of the thesis in five chapters. The brief description of each chapter of the thesis is as follows:

- **CHAPTER 1:** It presents an introduction to multi-UAVs, background on reinforcement learning, and provides a brief overview of its algorithms. It explores the motivation behind trajectory scheduling and multi-UAV coordination for infrastructure-deficient environments, particularly for the application of area surveillance.

- **CHAPTER 2:** This chapter offers a detailed review of existing literature related to trajectory scheduling in multi-UAVs, encompassing a wide range of applications. It also delves into the challenges and limitations found within these studies, underscoring the gaps in current research.

- **CHAPTER 3:** In this chapter, a strategy based on Integer Linear Programming (ILP), referred to as DLAT and HDLAT, is introduced for the trajectory scheduling of UAVs. This approach is specifically designed to consider factors such as access latency and the optimization of multiple queues within UAV networks. The proposed strategy aims to enhance the efficiency and effectiveness of UAV trajectory planning by integrating these critical operational parameters.

- **CHAPTER 4:** In Chapter 4, the focus is on area surveillance, where a solution based on Deep Reinforcement Learning (DRL) is proposed for coordinating multiple UAVs. This approach introduces a two-level decentralized system for scheduling UAV trajectories. The design is tailored to optimize several key objectives within the system, including battery life, AoI, and average access latency. This multi-objective optimization ensures efficient and effective use of UAV resources while maintaining high-quality surveillance.

- **CHAPTER 5 :** This chapter serves as the conclusion of the thesis, encapsulating the significant contributions made through the research. It summarizes the key findings and challenges in designing trajectories of UAVs. Furthermore, the chapter also offers a perspective on potential future avenues for research, suggesting how the groundwork laid by this thesis could be expanded and explored further in subsequent studies

2| Literature Review

The purpose of this chapter is to discuss the multi-UAV coordination problem and work done in literature in this direction. In this chapter, we will discuss contributions on different algorithms for UAV trajectory scheduling. This literature review chapter aims to provide a comprehensive overview of multi-UAV coordination techniques with challenges, and their applications in various scenarios. We will highlight the approaches already proposed in the literature and directions for further research.

2.1 Multi-UAV Trajectory Scheduling

Various studies have emphasized that trajectory planning of UAVs is an integral component of the UAV-based inspection and monitoring applications [18, 19, 20]. In [21], the authors presented the reconstruction of a 3D model and highlighted the importance of UAV trajectories for computer vision techniques to reconstruct the 3D structure accurately. In [22], the authors discussed how MEC can be divided into different architectures based on the role of UAVs, which could be users, computing entities, or data relay entities. The UAV-enabled MEC system is commonly employed in different scenarios to improve user experience and service availability or to increase the system's efficiency. The trajectory optimization of UAVs is an integral part of such MEC systems as it affects the energy consumption of the system and the service schedule of static or dynamic sensors. UAVs could be deployed to relay data further or provide partial computing to improve the system's overall quality of service (QoS). In [23], multiple UAVs were deployed for data relay tasks from mobile devices to the BS. The overall objective was to minimize the energy consumption of mobile devices by jointly optimizing the task scheduling and UAV trajectories in resource-constrained environments. Using a different approach, [24] proposed a single UAV-mounted cloudlet to serve a set of mobile users.

The overall framework minimizes the energy consumption of mobile users while optimizing the trajectory of the UAV-mounted cloudlet. The work of Xu et al. [25] also considered the multi-UAV based computing framework to minimize the latency of mobile device data relay task either by on-board computing or relaying to BS. In [26], a hierarchical multi-coalition UAV MEC network was discussed where the resource-constrained UAVs could offload task to other UAVs with high computational resources to improve the overall system efficiency. However, the authors did not consider the queue optimization, dynamic access of UAVs and challenges of an infrastructure-deficient environment as modeled in our work. In [27], authors focused on minimizing the weighted sum of energy consumption of UAV-enabled MEC system. They performed joint optimization of computation resource scheduling, bandwidth allocation to user equipment (UEs), and trajectory optimization of UAV-based edge servers with static ground sensors. The advantage of using multiple UAVs in the MEC system is further studied in the work of Diao et al. [28], where the effects of joint optimization of trajectories of multiple UAVs to improve the system metrics were considered. However, the dynamic evolution of the data queues of the UAV-based MEC system could alleviate the problem of queue stability and data offloading.

The authors in [29] addressed the stability issues with a Lyapunov-based joint resource optimization of bandwidth usage, processing power consumption, and transmission power. Similarly, the work [30] focus on UAV association and Lyapunov optimization to meet the system objective by dividing the solving the problem for each time slot. The Zhang et al. [14] presented a complex system within a dynamic environment that involves joint optimization of the computation resources of the multiple mobile users, UAV-BS, and trajectory optimization. The authors in [31] discussed a UAV-assisted mobile edge computing framework that jointly addressed energy minimization, trajectory optimization, CPU frequency and offloading schedule. In [32], author considered the completion time of the task along with the energy minimization and trajectory optimization of a UAV. One significant difference between our work and those reviewed in the literature is the estimation of the location of dynamic sensors (i.e. I_UAV). This problem brings another challenge of coordination among I_UAVs and A_UAV in the absence of ubiquitous connectivity with a limited battery.

The literature also discusses the network scheduling problem along with trajectory scheduling for UAV-based MEC. In [13], authors developed a hierarchical MEC system considering online optimization of computational resources and reinforcement learning-based trajectory optimization of multiple UAV-BSs for collecting data from a set of static sensors. In [33], a sense and send transmission protocol was proposed using multiple UAVs in a cellular network using an iterative trajectory sensing and scheduling algorithm. However, this approach does not consider the distributed and multi-layer interaction of UAVs to collect and offload data with limited connectivity. In [34], the authors employed reinforcement learning for sensing and sending information using a decentralized setup for multiple UAVs, however, their work did not consider the multi-layer UAV network with limited connectivity. As apparent from the literature, resource scheduling in multi-UAV based solutions is a challenging task, particularly in an infrastructure-deficient environment with limited connectivity. The dynamic deployment of mobile UAVs either to collect data or relay data to the cloud could mitigate the issues of progress tracking and job monitoring in industrial settings and aid in the performance of project deliveries. In this thesis, we propose a solution for end-to-end data offloading in large infrastructure-deficient environments using a hierarchical multi-UAV system. Based on the above literature the identified research objectives could be easily inferred as following

1. The first objective focus to create an algorithms that enable multi-UAV systems to function efficiently in areas with limited infrastructure. This involves designing trajectory schedules for Access UAVs (A_UAVs) to efficiently collect data from dynamic Inspection UAVs (I_UAVs) with limited communication range. The work of [13], emphasize the importance of trajectory planning for UAV-based inspection

and monitoring applications. However, their work focuses on static sensors. In contrast, our research addresses the challenges of a two-level UAV network, where dynamic sensors (I_UAVs) are used, adding complexity and requiring more advanced coordination strategies.

2. The second objective aims on designing a coordination framework for multiple UAVs operating in infrastructure environments. A key part of this framework is to manage the limited buffer capacity of UAVs, making sure they handle data and communicate efficiently within the network. The work of [13] employs Lyapunov optimization, but it does not consider a two-level UAV system, which is another research gap.

3. The third objective is to design a framework that allows A_UAVs to estimate the location of I_UAVs on their own, without relying on a central system, especially in areas with poor infrastructure. The literature on multi-UAV systems does not address decentralized control as mentioned in our work setup, highlighting a significant research gap.

2.2 AoI and Reinforcement Learning for Trajectory Scheduling

When monitoring large events of different types, stationary cameras fall short because they can't cover a wide area. Thus, to track crowd movement, it's necessary to use flexible options like UAVs equipped with cameras to gather visual information. Yet, the restricted battery life and communication distance of UAVs create challenges that call for the use of networks where UAVs relay information to each other [15]. Additionally, real-time image analysis mandates up-to-date data, and minimizing delay is crucial for such applications. Therefore, AoI metric proves to be a viable option for evaluating the performance of time-sensitive solutions. In our research, our emphasis has been on crowd management within an infrastructure-less environment, particularly in scenarios involving unplanned massive gatherings in a given area.

The AoI [10] is a metric that measures the time elapsed between the generation of data and its arrival at its destination through the network. It is particularly useful in evaluating the performance of time-sensitive applications, such as monitoring and control applications. A higher AoI implies a diminished value of information for the node generating the data at the receiving end. The AoI has been extensively studied in the literature for various applications, such as data collection or relay in sensor networks [35], surveillance [36], data routing in wireless networks and UAV-aided vehicular networks [37] and others. Yates et al. [38] conducted a survey that examined the evaluation of AoI across diverse systems. This encompassed single-source to multiple server systems, multiple sources to a single server system, multiple sources to multiple server systems, as well as variations in AoI dependent on the specific system characteristics.

Nowadays, Deep Reinforcement Learning (DRL) based solutions for the cyber-physical system are well studied in literature owing to the real environment being dynamic and stochastic [39]

Our work involves an AoI and access latency minimization approach in a multi-UAV network, distinguishing it from the work of Biplav et al. [40]. Notably, our system lacks a centralized controller to coordinate the trajectories of I_UAVs and A_UAVs.

In the literature, various studies are conducted on multi-UAV trajectory scheduling based minimization of AoI and energy of the system. In the work [41], the primary objective is to collect real-time data for crowd monitoring while optimizing the system's energy and ensuring efficient area coverage using a multi-UAV system. Another approach, proposed by Chaudary et al. [42], focuses on a centralized two-level AoI minimization approach using DRL for an IoT sensor network. Another work [43] presents the method that employs two UAVs: one for data collection and the other for device charging. In this work DRL-based techniques are utilized while designing UAV trajectory scheduling to optimize both average AoI and energy consumption. On the other hand, [44] deploys multiple UAV-BSs as edge servers to collect data from a distributed network of sensors. This study assumes the persistent connectivity between the UAV-BSs and the cloud, as well as a connection to the sensors.

The field of crowd monitoring using UAVs has been extensively studied in recent years. In [45], a group of UAVs is deployed to monitor crowd dynamics. The UAV agents in this work interact securely using a blockchain framework, achieving the system's objectives. Another work [46], uses genetic algorithms for crowd monitoring using multi-UAV system.

Based on the literature, we have identified two key research objectives for our second work: The first is to create a decentralized method for scheduling the paths of A_UAVs in unpredictable environments, focusing on reducing the Age of Information (AoI) and overall energy use to ensure efficiency. Secondly, develop a similar decentralized method for I_UAVs, aiming to reduce the access latency for Points of Interest (PoIs) and the energy consumption of the UAVs.

2.3 Introduction to Various Metrics

Based on the literature survey, various metrics have been identified and utilized in Chapters 3 and 4.

- Average Access Latency: This parameter measures the average time delay experienced by an I_UAV when trying to access the PoI. It is the time elapsed between the last access of PoI and the current time slot.

- Total Time Slots: his parameter refers to the total number of discrete time intervals (slots) considered in the system for trajectory scheduling.

- Average AoI (Age of Information): The Age of Information is a metric that measures the freshness of the information available at the A_UAV. The average AoI is the average age of data from the time they are generated at I_UAV to the time they are successfully received A_UAV.

- Average Crowd Density: This parameter measures the average number of people present in a specific area covered by the network. It reflects the level of number of people at particular PoI.

- Cumulative UAV Reward: This parameter is used in reinforcement learning (RL). The cumulative reward represents the total accumulated score earned by agents over a period, which could be based on various factors.

- Queue Length: Measures the average or maximum queue size at each UAV over time. This metric is crucial for assessing the risk of buffer overflow and the effectiveness of queue management strategies.

- Energy Consumption: Total energy consumed by the UAVs, during their flight and operation. This includes energy used for movement, hovering, and data transmission.

- System Stability: Measured by the variability or constancy of the queue lengths over time. A stable system exhibits minimal fluctuations in queue sizes.

- Total Flight Time: The amount of time UAVs spend in operation during a mission. This metric is linked to energy consumption but focuses on operational efficiency and mission duration.

- Coverage: Refers to the number of Points of Interest (PoIs) visited or covered during a mission.

2.4 Chapter Summary

This chapter investigates the vast array of research on autonomous Unmanned Aerial Vehicles (UAVs), focusing particularly on the coordination of multi-UAV systems for trajectory scheduling. The main focus is to study different methods for managing UAV trajectories in systems that need to balance several goals simultaneously. We explore a wide range of algorithms for multi-UAV coordination, spanning from heuristic and metaheuristic to rule-based and reinforcement learning (RL) techniques. Specifically, we look into the works related to optimizing the Age of Information (AoI), minimizing access latency, and enhancing energy efficiency, especially in the context of surveillance applications. Additionally, we identify significant research gaps, such as infrastructure deficiencies and the absence of reliable communication frameworks, which pose challenges to the effective deployment of multi UAV solutions. Another research gap is the absence of decentralized coordination in multi-UAV based autonomous solutions.

3| Optimizing Trajectory and Dynamic Data Offloading using a UAV Access Platform

This chapter introduces a hierarchical multi-UAV system designed for dynamic data collection in areas with sparse infrastructure. The system consists of a single Access UAV (A_UAV) and several Inspection UAVs (I_UAVs). The main objective of this system is to optimize the trajectory scheduling of the A_UAV. This optimization aims to decrease energy consumption and lower the time it takes to access data while ensuring efficient coordination with the I_UAVs. Additionally, our study explores the application of an online optimization framework, which is crucial for managing and controlling the backlog in a multi-level queue system.
This chapter is organised as follows: Section 3.1 presents the proposed multi-UAV framework and the system model. The overall system objective is discussed in Section 3.2. Sections 3.3 and 3.4 discuss the access latency aware trajectory optimization and Lyapunov based system stability, respectively. Finally the sections 3.5 and 3.6 discuss the experiments and results.

3.1 System Model

This section presents the key components of the proposed multi-UAV framework. The system consists of heterogeneous UAVs, including a set of **N** Inspection UAVs (I_UAVs) and a single UAV Access Platform (A_UAV). I_UAVs are smaller in size and more agile. They collect visual data from a set of **k** Point of Interests (PoIs) denoted as l_i. Because the framework considers infrastructure-less environments, limited Access Points (APs) available for cloud connectivity. Further, I_UAVs possess a limited connectivity range, making it difficult to transfer data directly to the cloud. A_UAV, which is larger in size and possesses higher computational capabilities, coordinates with the I_UAVs to collect data. We assume that the A_UAV always maintains a constant height, thus its trajectory lies in a horizontal plane. Figure 1.1 shows a high level overview of the system under consideration with I_UAVs tasked to collect data from the PoIs, whereas the A_UAV collects data from the dynamically moving I_UAVs and relay it to the cloud.

3.1.1 Communication Channel

The communication between I_UAV and A_UAV (A2A channel) has a limited range and capacity. This work assumes that the achievable data transmission rate of the $i^{th} I\_UAV$ in a given time slot as $d_i^{off}(t)$. The communication channel between I_UAVs

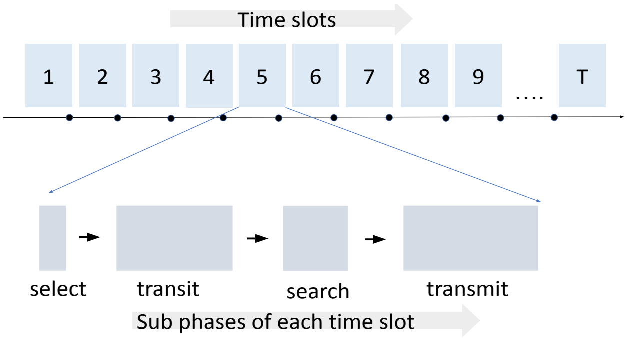

Figure 3.1: Division of a timeslot with different functions of A_UAV

and A_UAV involves both line-of-sight (LoS) and non-line-of-sight (NLoS) links as PoIs can be distributed vertically and longitudinally. Furthermore, the shadowing effect is also considered due to obstructions caused by buildings and other structures in the surroundings [47, 48]. The path loss of a link is given as follows:

$$L_\alpha = L_\alpha(r_0) + 10\phi \log(\frac{r^{'}}{r_0}) + X_\sigma \tag{3.1}$$

where X_σ is a shadowing factor that is indirectly proportional to the altitude of the PoI, $\alpha \in \{LoS, NLoS\}$ and ϕ is the path loss exponent. The probability of LoS link, (P_{LoS}), depends on the angle of elevation and environmental constraints (e_o and e_1) as given in Equation (3.2):

$$P_{LoS} = \frac{1}{1 + e_o.exp(-e_1[\theta - e_o])} \tag{3.2}$$

The average path-loss is calculated as:

$$L = P_{LoS}.L_{LoS} + (1 - P_{LoS}).L_{NLoS} \tag{3.3}$$

In this work, we have assumed Wi-Fi technology without a fixed access point for emergency or infrastructure deficient scenarios [49]. The network of I_UAVs and A_UAV provides connectivity to send collected data from PoIs to the cloud.

3.1.2 Data Gathering Process

Each PoI (l_j) is a tuple $(< d_j, O_j >)$, where d_j specifies the amount of data (e.g., images) to be collected and O_j denotes the 3D coordinates of the PoI. The sequence of PoIs to be visited is provided to the I_UAVs and the same is also shared with the A_UAV. During the traversal along the sequence of PoIs, if the buffer of any of the I_UAVs overflows then that I_UAV waits at the same PoI until its data is offloaded.

In order to gather and offload data, the A_UAV communicates with a single I_UAV in a time slot. Let us denote the data gathered by each of the I_UAVs in a time slot t by $A_i(t)$. Let $Q_i(t)$ be the queue of the $i^{th} I\_UAV$ and $d_i^{off}(t)$ denotes the amount of data offloaded to the A_UAV by the $i^{th} I\_UAV$ in time slot t. The recursive equation to update the $Q_i(t)$ is as follows:

$$Q_i(t+1) = max\{Q_i(t) - d_i^{off}(t), 0\} + A_i(t) \tag{3.4}$$

Let $L(t)$ be the queue of the A_UAV where A_UAV accepts the data from the selected I_UAV in the time slot t. The following equation updates $L(t)$ recursively:

$$L(t+1) = max\{L(t) + d_i^{off}(t)) - d_{access}^{off}(t), 0\} \tag{3.5}$$

where $d_{access}^{off}(t)$ is the amount of data offloaded to the cloud by the A_UAV in time slot t. Figure 3.1 shows the different functions performed by an A_UAV in a single time-slot. The decision function takes negligible time to decide on the next I_UAV for data gathering, followed by the transition function where A_UAV takes τ_{trans} time to move near the next possible location to connect with the chosen I_UAV. The search function (τ_{search}) estimates the location of the selected I_UAV based on the queue and position estimation algorithm given in Algorithm 1. The bound on the maximum time required to estimate the position of I_UAVs is discussed in Section 3.3.1. Finally, the data transmission function establishes the successful communication with the I_UAV (if it is not shadowed). The sequence of the functions mentioned above is repeated for every time slot. The next section describes the objective of the system and formulates it as an optimization problem.

3.2 System Objective

In the proposed framework, the offloading of data happens at two stages - 1) from I_UAV to A_UAV and 2) from A_UAV to the cloud. Our main focus is to achieve end-to-end data offloading to the cloud by minimizing the total energy consumption of the whole system (E_{sys}) given as:

$$E_{sys}(t) = E_{access}^{trans}(t) + E_{access}^{comm}(t) + E_{access}^{hover}(t) + \left(\sum_{i=1}^{N}(E_i^{comm}(t))\right) \tag{3.6}$$

where $E_{access}^{trans}(t)$ is the transition energy of the A_UAV, $E_{access}^{comm}(t)$ is the transmission of the A_UAV, $E_{access}^{hover}(t)$ is the hovering energy of A_UAV and $E_i^{comm}(t)$ is the transmission energy of the i^{th} I_UAV. The following subsections discuss the details of calculating each component of energy consumption in Equation (3.6).

3.2.1 Transition energy of A_UAV

The transition energy of A_UAV refers to the energy consumed when moving from one location to another [50, 18, 51] which is given as:

$$E_{access}^{trans} = \kappa \cdot ||vel(t)||^2 \tau_{trans} \tag{3.7}$$

where κ is a constant that depends on the total mass of the A_UAV, $vel(t)$ is the velocity of A_UAV and τ_{trans} is the time taken to transit from one location to another.

3.2.2 Transmission energy of A_UAV

A_UAV offloads data to the cloud via a wireless channel [52]. The transmission energy consumed to transmit the data to the cloud is given as:

$$E_{access}^{comm}(t) = (2^{\frac{d_{access}^{off}(t)}{W \cdot \tau}} - 1) \cdot \frac{N_0 W}{\zeta} \cdot \tau_{comm} \tag{3.8}$$

where τ_{comm} is the time allotted for data transmission. Other parameters such as $d_{access}^{off}(t)$, W, τ, N_0, ζ are defined in the List of Notations.

3.2.3 Hovering energy of A_UAV

A_UAV hovers above the PoI to collect the data. The hovering energy consumed to collect the data is given as:

$$E_{access}^{hover}(t) = P_{hover} \cdot \tau_{hover} \tag{3.9}$$

where, P_{hover} is the power consumed while hovering per unit time and τ_{hover} is the time for hovering.

3.2.4 Transmission Energy of I_UAVs

The energy consumed for offloading the $d_i^{off}(t)$ data bits at time slot t from the selected I_UAV to the A_UAV using the Air to Air channel of bandwidth W Hz is given similarly to Equation (4.3) as:

$$E_i^{comm}(t) = (2^{\frac{d_i^{off}(t)}{W \cdot \tau}} - 1) \cdot \frac{N_0 W}{\zeta} \cdot \tau \tag{3.10}$$

The wireless (Air to Air) channel power gain (ζ) from I_UAV to A_UAV can be given as:

$$\zeta = g_0 \cdot (\frac{r_0}{r'})^{\phi} \tag{3.11}$$

where g_0 is the path loss constant, r_0 is the reference distance, r' is the distance between the UAVs, ϕ is the path loss exponent and τ is the time. Given the system's energy consumption, our goal is to find the optimal settings to minimize the expected cumulative energy across the time horizon. The decision variables in every time slot t that affect the total system's energy are given by the set $\pi(t) = \{p_i(t), P_{access}(t), S_{access}(t)\}$ corresponding to the transmission energies of the I_UAVs & A_UAV, and the transition energy of A_UAV, respectively. Moreover, the channel information for the data offloading task is not deterministic and varies in the environment, hence the amount of data arrived at the A_UAV becomes stochastic which depends on the channel characteristics and the position of the selected I_UAV. Further, this framework does not consider the energy consumed for the movement of I_UAVs as the PoIs are predefined and the I_UAVs follow a predetermined trajectory consuming constant energy. The overall optimization model for the stable system performance is formulated as:

$$\textbf{P1} \quad \min_{\pi(t)} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[E_{sys}(t)] \tag{3.12}$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \tag{3.13}$$

$$P_{access}(t) \leq P^{max}, \quad \forall t \tag{3.14}$$

$$||S_{access}(t) - S_i(t)|| \leq v_{max}\tau, \quad \forall i, \forall t \tag{3.15}$$

$$\sum_{i=1}^{N} \frac{RL_i(t)(1 - x_i(t))}{N} \leq RL_{max}, \quad \forall i, \forall t \tag{3.16}$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i, \forall t \tag{3.17}$$

$$d_i^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta p^{max}(t)}{N_o W}), \quad \forall i, \forall t \tag{3.18}$$

$$d_{access}^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta P^{max}(t)}{N_o W}), \quad \forall t \tag{3.19}$$

$$\lim_{T\to\infty} \frac{\mathbb{E}[Q_i(t)]}{T} = 0, \quad \forall i, \forall t \tag{3.20}$$

$$\lim_{T\to\infty} \frac{\mathbb{E}[L(t)]}{T} = 0, \quad \forall t \tag{3.21}$$

Constraints (3.13) and (3.14) define the maximum transmission power of I_UAVs and A_UAV, respectively. Constraint (3.15) limits the maximum transition energy of A_UAV for every transition and Constraint (3.16) limits the time that has elapsed since the last access of i^{th} I_UAV to be less than RL_{max}. Additionally, the constraints (3.17), (3.18) and (3.19) bound the number of transmitted bits. Constraints (3.20) and (3.21) establish the rate stability of all the system queues (I_UAVs and A_UAV). Next, the model to optimize the trajectory of the A_UAV with respect to the trajectories of I_UAVs is discussed.

3.3 Distance and Latency Aware Trajectory (DLAT) Optimization

Flexible and dynamic trajectory planning of A_UAV is crucial to applications where terrestrial communication infrastructure is missing. As already mentioned, the position of I_UAVs changes in every time-slot since they move through different PoIs to collect data. The A_UAV's trajectory needs to be planned so that it can connect and access an I_UAV in a time-slot before the I_UAV's queue overflows. Whenever an I_UAV's queue gets full, it does not move to its next designated PoI. Instead, it sojourns at the same PoI until it can offload its data to the A_UAV and free up the queue space. In order to choose one of the I_UAVs to gather data, the A_UAV would require the real-time information about the queues of all I_UAVs in each time-slot. This information is not available *a priori* due to the dynamic nature of the system queues. We use a message passing based approach for estimating the queues of I_UAVs to make a selection. Further, the trajectory of the A_UAV must be optimized to consume minimal energy.

The trajectory optimization model of A_UAV optimizes the trade-off between the transition energy of A_UAV and the access latencies of all I_UAVs. In addition, this access latency based data offloading generates an access fair schedule for the I_UAVs to offload their data to the A_UAV. The access latency ($RL_i(t)$) of the i^{th} I_UAV in the time-slot t is the difference between the time of its last access by the A_UAV and the current time-slot. The distance and latency aware trajectory optimization of A_UAV is formulated as:

$$\mathbf{P2} \min_{S_{access}(t)} \sum_{t=1}^{T} \sum_{i=1}^{N} ||S_{access}(t+1) - S_{access}(t)||^2 - Vp_i(t) \tag{3.22}$$

s.t.

$$||S_{access}(t) - S_i(t)|| \leq v_{max}\tau, \quad \forall i, \forall t \tag{3.23}$$

$$\sum_{i=1}^{N} \frac{RL_i(t)(1 - x_i(t))}{N} \leq RL_{max}, \quad \forall i, \forall t \tag{3.24}$$

$$\sum_{i=1}^{N} (x_i(t) \cdot Q_i(t)) \geq 0, \quad \forall i, \forall t \tag{3.25}$$

$$\sum_{i=1}^{N} x_i(t) = 1, \quad \forall i, \forall t \tag{3.26}$$

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \tag{3.27}$$

$$x_i(t) \in \{0, 1\}, \quad \forall i, \forall t \tag{3.28}$$

where the first constraint (3.23) signifies that the distance travelled within a time-slot is limited by the maximum velocity. Constraint (3.24) limits the time that has elapsed since the last access of i^{th} I_UAV to be less than RL_{max}. The constraint in (3.25) selects the I_UAV which has data to offload whereas (3.26) enforces the selection of only one of the I_UAVs in a time-slot. The selected I_UAV transmission power should be bounded as given in (3.27).

3.3.1 Estimating Position and Queue Length

The exact position and queue length of I_UAVs is not known to the A_UAV *a priori*. The A_UAV maintains the last access statistics of each I_UAV using status messages. The track of status messages received over time helps in computing the position (l_i) and queue length ($Q_i(t)$) of I_UAVs in a time-slot. The status message comprises of the remaining queue size at the time of access and the data to be collected at the current PoI. Moreover, the pre-computed trajectory of each I_UAV provides the set of PoIs to be visited by each I_UAV. Algorithm 1 describes the procedure to estimate the queue length of each I_UAV in every time-slot.

The algorithm operates through a series of key steps. Initially, each I_UAV sends status messages to the A_UAV when it comes within range for data offloading. These messages include the I_UAV last known position and the remaining queue size, which are crucial for the A_UAV to update its knowledge about each I_UAV state. The A_UAV uses the data from these status messages to estimate both the position and the queue length of I_UAV over time. This estimation is done using a recursive update method, predicting the current state based on the previous state and the known behavior of the UAVs.

To calculate candidate positions, the algorithm starts with the last known position and queue size of an I_UAV . It considers the data collection rate and buffer capacity, which influence how quickly an I_UAV's queue fills up and when and where it might need to pause for offloading. The time elapsed since the last update is used to project the current queue size and position. If the I_UAV is expected to have filled its buffer, it might not have moved beyond a certain point, constraining its current possible locations to a smaller area.

As the I_UAV collects data while moving from one Point of Interest (PoI) to another, if its buffer reaches capacity, it must wait for the A_UAV to offload data before moving to the next PoI. This waiting time and the buffer overflow potential are key in predicting the I_UAV's movement. Using minimum and maximum data collection scenarios, the A_UAV calculates the possible range of positions (candidate positions) for each I_UAV. This range is defined by how far the I_UAV could have traveled given its data collection rate and buffer size.

Finally, with candidate positions mapped, the A_UAV plans its trajectory to efficiently meet and offload data from the I_UAVs. The path chosen by the A_UAV aims to minimize energy consumption while considering the urgency of data offloading for I_UAVs with nearly full buffers. This algorithm involves a complex interplay of data prediction, resource management, and real-time adjustment to operational constraints, all of which are critical for maintaining system efficiency and functionality in a dynamic and unpredictable environment.

3.3.2 Estimation of Search Time Bound

A_UAV estimates the location of I_UAVs in each time slot using the last access statistics. It could search the set of candidate locations to locate the precise location of selected I_UAVs, which contributes to the search time. The bound on the search time depends on the data generation rate and the maximum buffer of I_UAVs as derived below.

Lemma 1: The search time τ_{search} to locate the exact location of I_UAV with max

Algorithm 1: Estimated position and queue length of I_UAV

Input: $last\_access\_timeslot$, $cur\_timeslot$, $last\_accessed\_position$, $last\_accessed\_buffer$, $data\_left\_at\_last\_accessed\_position$, $Modes$, D_{min}, D_{max}
Output: $Q_{i,max}, Q_{i,min}$ ψ_{max}, ψ_{min}
Initialization:
  $time\_elapsed \leftarrow$ cur_timeslot$-$last_access_timeslot
  $l_i = last\_accessed\_positionofI\_UAVs$
  $\psi_{min} = l_i$
  $\psi_{max} = l_i$
  $Q_i = last\_accessed\_bufferI\_UAVs$
  $curr\_location\_data = data\ left\ at\ last\_accessed\_position\ of\ I\_UAVs$
  $Modes = Min\ or\ Max$;
$D \leftarrow D_{min}$;
if $Modes == Max$ **then**
  $D \leftarrow D_{max}$;
end
for I_UAVs **do**
  for $Modes$ **do**
    $j \leftarrow 0$
    while $j \leq time\_elapsed$ **do**
      if $Q_i \leq Q_{max}$ **then**
        if $curr\_location\_data\ is\ not\ collected$ **then**
          $l_i = last\_accessed\_position$
          $Q_i = last\_accessed\_buffer + data\_at\_current\_loc$
          $j \leftarrow j + 1$;
        else
          $i \leftarrow i + 1$
          $l_i =$ next_position
          $Q_i =$ last_accessed_buffer$+D_{min}$ or D_{max}
          $j \leftarrow j + 1$ **if** $Mode == Min$ **then**
            $Q_{i,min} = Q_i$
            $\psi_{min} = \psi_{min} \cup l_i$
          else
            $Q_{i,max} = Q_i$
            $\psi_{max} = \psi_{max} \cup l_i$
          end
        end
      else
        $l_i =$ last_accessed_position
        $Q_i =$ last_accessed_buffer
        break
      end
    end
  end
end
return $Q_{i,max}$, ψ_{max}, $Q_{i,min}$, ψ_{min}

buffer size Q_{max} is given as:

$$\tau_{search}(|\psi|) \leq \frac{1}{3}\frac{\sigma \cdot Q_{max} \cdot \varrho}{v_{max}(\mu^2 - \sigma^2)} \tag{3.29}$$

where ϱ is the maximum distance between two consecutive PoIs in the possible set of locations to be searched and $|\psi|$ is the number of candidate locations for I_UAV and data generation process at each PoI follows the normal distribution $D \sim \mathcal{N}(\mu, \sigma^2)$

Proof : The time taken to find the location of I_UAV depends on the travel distance to cover the candidate PoI locations as given in Equation (3.30).

$$\tau_{search}(|\psi|) \geq |\psi| \cdot \frac{\varrho}{v_{max}} \tag{3.30}$$

By generality,

$$|\psi_{min}| \geq |\psi_{max}| \tag{3.31}$$

where $\psi_{min} = \{l_i, ..., l_{i,min}\}$ is the set of locations visited when each location has minimum data D_{min} to be collected whereas $\psi_{max} = \{l_i, ..., l_{i,max}\}$ is the set of locations when maximum data D_{max} is present at each location. As the memory of each I_UAV is bounded by Q_{max}, it covers less number of locations for ψ_{max} as shown in Equation (3.31). Similarly, the data collected in both scenarios will be the same as the maximum memory size is fixed. The candidate locations are defined as the locations starting at $l_{i,max}$ and ending at $l_{i,min}$. Intuitively, the number of candidate locations $|\psi| = |\psi_{min}| - |\psi_{max}|$.

$$\begin{aligned} |\psi_{min}|.D_{min} &= |\psi_{max}| \cdot D_{max} \\ |\psi_{min}| &= |\psi_{max}| \cdot \frac{D_{max}}{D_{min}} \\ |\psi_{min}| - |\psi_{max}| &= \frac{(D_{max} - D_{min})}{D_{min}} \cdot |\psi_{max}| \\ |\psi_{min}| - |\psi_{max}| &= \frac{(D_{max} - D_{min})}{D_{min}} \cdot \frac{Q_{max}}{D_{max}} \end{aligned} \tag{3.32}$$

From the above derivation, the locations in the search trajectory are influenced by data rate and maximum limit of memory size for I_UAVs. The upper and lower limit of normally distributed data is given as $D_{max} = \mu + \sigma$ and $D_{min} = \mu - \sigma$ respectively. Thus Equation (3.30) can be written as

$$\tau_{search}(|\psi|) \leq \tfrac{1}{3}\tfrac{\sigma \cdot Q_{max} \cdot \varrho}{v_{max}(\mu^2 - \sigma^2)} \tag{3.33}$$

■

To calculate the upper bound for Equation (3.33), ϱ is the distance between consecutive PoIs which could be calculated from the pre-calculated trajectory of I_UAVs based on shortest path.

3.4 Energy Aware Data Offloading

The model presented in **P1** in Section 3.2 is a stochastic optimization problem as the arrival of data in the system queue is stochastic. Using the online Lyapunov optimization algorithm, we solve the stochastic optimization in **P1** and jointly stabilize all queues by finding the optimal policy to access each I_UAV in each time-slot. The quadratic Lyapunov function, as given in Equation (3.36) associates a scalar measure to the queues of the system. Further, the stability of the system is maintained by guaranteed mean rate stability of the evolving queues as given in Equations (3.34) and (3.35).

$$\lim_{T\to\infty} \frac{\mathbb{E}[Q_i(t)]}{T} = 0, \forall i \tag{3.34}$$

$$\lim_{T\to\infty} \frac{\mathbb{E}[L(t)]}{T} = 0 \tag{3.35}$$

$$Z(v(t)) = \frac{1}{2}\left[\sum_{i=1}^{N} Q_i(t)^2 + L(t)^2\right] \tag{3.36}$$

where $v(t) = [\{Q_i(t)\}_{i=1}^{N}, L(t)]$ consists of all system queues at a time t and $Z(.)$ is quadratic Lyapunov function of system queues.

The **Lyapunov drift** corresponding to the above function is given as:

$$\triangle Z(v(t)) = \mathbb{E}[(z(v(t+1)) - z(v(t)))] \tag{3.37}$$

The Lyapunov drift plus a penalty function is minimized to stabilize the queue backlog of the system is given as:

$$\triangle DP(t) = \triangle Z(v(t)) + V \cdot \mathbb{E}[E_{sys}(t)] \tag{3.38}$$

where V is a positive constant that controls the trade-off between Lyapunov drift and the expected system energy. A high value of parameter V signifies more weight on minimizing the energy of the system at the cost of a high queue backlog. Therefore, V acts as a trade-off parameter between system energy and queue backlog.

An upper bound on $\triangle Z(v(t))$ can be derived as follows, (for details see [16])

$$\triangle Z(v(t)) \leq \mathbb{E}[-\sum_{i=1}^{N} Q_i(t) \cdot d_i^{off}(t)] + \quad \mathbb{E}[L(t) \cdot (-d_{access}^{off}(t))] + C \tag{3.39}$$

where C is a deterministic constant.

As a result, the upper bound of the drift plus penalty function becomes

$$\triangle DP(t) \leq C - \mathbb{E}[\sum_{i=1}^{N} Q_i(t) \cdot d_i^{off}(t)] - \mathbb{E}[L(t) \cdot d_{access}^{off}(t)] + V \cdot \mathbb{E}[E_{system}(t)|v(t)] \tag{3.40}$$

Hence, the original formulation **P1** can be reduced to **P3** which bounds the system's drift to keep the system stable as follows:

$$\textbf{P3} \quad \min_{p_i(t),P_{access}(t)} \mathbb{E}\left[-\sum_{i=1}^{N} Q_i(t)_i^{off}(t)]\right] - \mathbb{E}[L(t)_{access}^{off}(t)] + V \cdot \mathbb{E}[E_{sys}(t)] \tag{3.41}$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i, \forall t \tag{3.42}$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i, \forall t \tag{3.43}$$

$$d_i^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta p^{max}(t)}{N_o W}), \forall i, \forall t \tag{3.44}$$

$$P_{access}(t) \leq P^{max}, \quad \forall t \tag{3.45}$$

$$d_{access}^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta P^{max}(t)}{N_o W}), \forall t \tag{3.46}$$

As can be observed, the constraints in **P3** is a subset of the constraints in **P1**. To further simplify the solution of the optimization formulation, we reformulate **P3** as two separate sub-problems provided the positions of A_UAV and I_UAV are fixed in a given time slot t. The Lyapunov-based online optimization is optimal for a stochastic system to derive the overall optimal solution [53].

3.4.1 Optimization of Transmission Energies of I_UAVs

The first sub-problem deals with the optimization of parameters related to the I_UAV. The variables $S_{access}(t)$, i.e., the position of A_UAV and the offloaded data $d_i^{off}(t)$ of the selected i^{th} I_UAV are coupled in a particular time-slot. The fixed position of A_UAV decouples these variables. In the optimization model **P 3.1**, the transmission energy is optimized for a single time-slot (t) given the position of A_UAV:

$$\textbf{P 3.1} \quad \min_{p_i(t)} - \sum_{i=1}^{N} Q_i(t) \cdot d_i^{off}(t) + V \cdot \tau_{comm} \cdot \sum_{i=1}^{N} p_i(t) \tag{3.47}$$

s.t.

$$p_i(t) \leq p^{max}, \quad \forall i \tag{3.48}$$

$$d_i^{off}(t) \leq Q_i(t), \quad \forall i \tag{3.49}$$

$$d_i^{off}(t) \leq W\tau_{comm} \log_2(1 + \frac{\zeta p^{max}(t)}{N_o W}), \forall i \tag{3.50}$$

The objective function in **P 3.1** is a convex function. The first & second constraints are linear and the third constraint is upper bounded by a concave function. As a result, the stationary point of the objective function is found to be: $p_i^*(t) = min\{max\{(\frac{Q_i(t)W}{V} - \frac{N_o W}{\zeta}), 0\}, p^{max}\}$.

3.4.2 Optimization of Transmission Energy of A_UAV

The second sub-problem deals with the optimization of the A_UAV parameters for the amount of data offloaded to the cloud at time t. The updated optimization model is given as:

$$\textbf{P 3.2} \min_{P_{access}(t)} -L(t) \cdot d_{access}^{off}(t) + V \cdot \tau_{comm} \cdot P_{access}(t) \tag{3.51}$$

s.t.

$$P_{access}(t) \leq P^{max} \tag{3.52}$$

$$d_{access}^{off}(t) \leq L(t) \tag{3.53}$$

$$d_{access}^{off}(t) \leq W\tau_{comm}\log_2(1 + \frac{\zeta P^{max}(t)}{N_o W}) \tag{3.54}$$

The model **P 3.2** has a convex optimization objective subject to convex constraints to solve for the optimal transmission power of the A_UAV. The stationary point of the optimization model is $P_{access}(t) = min\{max\{(\frac{L(t)W}{V} - \frac{N_0 W}{\zeta}), 0\}, P^{max}\}$.
Thus, the derived stationary points of the optimization model using the Lyapunov optimization framework are calculated in every time-step to optimize the A_UAV trajectory and data-offloading tasks. The overall proposed solution approach is presented in Algorithm 2. Next, we discuss the experimentation setup for evaluating the proposed solution.

Algorithm 2: Proposed Solution Approach for Trajectory Scheduling in the System

Input: Trajectories of all I_UAVs, List of Points of Interest (PoIs) l_i, Time horizon T
Output: Scheduled trajectories for A_UAV, Data collection plan Initialize: Trajectories of all I_UAV_i and list of PoIs l_i.
Time: $t = 0$ While$t \leq T$
    Compute and offload $d_{access}^{off}(t)$ as using P 3.2
    Update $L(t)$
    Using Algorithm 1 estimate the $\{Q_i(t)\}_{i=1}^{N}$and $\{S_i(t)\}_{i=1}^{N}$
    Select the i^{th} I_UAV to collect data using P2
    Compute $d_i^{off}(t)$ for $i^{th}I\_UAV$ using P 3.1 to offload data to A_UAV
    Update $Q_i(t)$
t = t+1

3.5 Experimentation

In this section, we present the simulation setup to validate the efficacy of our proposed Distance and Latency Aware Trajectory Optimization with Lyapunov based energy-aware data offloading followed by results and discussions. The simulation parameters are listed in Table 3.1.

Table 3.1: List of Simulation Parameters

Parameters	Values
Channel Bandwidth	1 MHz
κ	1
Noise Power for I_UAV	10^{-13}
Noise Power for A_UAV	10^{-20}
The path-loss constant g_0	10^{-4}
The path loss exponent θ	2 to 4
Memory capacity of I_UAV (Mem_{max})	10^5bits

We have considered a 600 x 600 square meter area with PoIs spread along the region in disjoint clusters and at heights ranging from 70 to 80 meters above the ground. All experiments are conducted for at least 30 times and the average of results are plotted. We sample 150 PoI locations uniformly randomly in three disjoint clusters. From a practical point of view of a multi-UAV system, we consider a system of three I_UAVs with one A_UAV in all the simulation experiments. Each I_UAV is assigned to a cluster where I_UAVs randomly chooses a starting location within the cluster. The sequence of PoIs to be visited by each I_UAV is generated using the shortest path algorithm. Before proceeding to the next PoI, an I_UAV collects all the data ($A_i(t)$) from that PoI. In the data collection process, an I_UAV may sojourn at the same PoI across multiple time-slots until all the data ($A_i(t)$) of PoI is collected. For each PoI, the amount of data

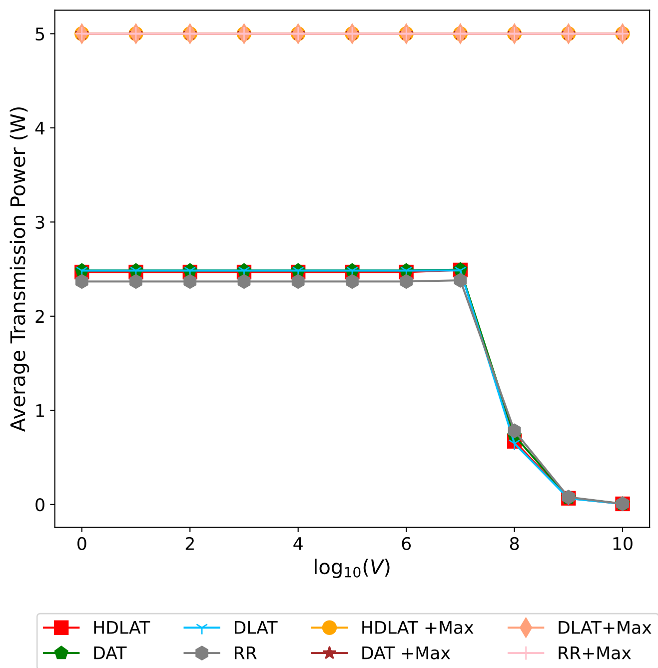

Figure 3.2: Analysis of Average Transmission Power of A_UAV.

to be collected is modeled as a Gaussian distribution with a mean of 150 Kb and variance of 50 Kb. The A_UAV gets partial information about the data generated at each location so

it could not accurately estimate the location of I_UAV in the next time slot; as a result, it has to search for candidate locations to access the selected I_UAVs as discussed in Section 3.3.1. The trade-off parameter V ranges from 10 to 10^{10}. The length of each time slot(τ) is 25 seconds divided into different sub slots as shown in Figure 3.1. The selection of I_UAV is assumed to take negligible time whereas transition may take up to 20 sec. The search and transmit function takes total of 5 seconds. The maximum transmission power for A_UAV and I_UAV are 5W and 2W, respectively [17]. The other simulation parameters are listed in Table 3.1.
The system's performance can be assessed using several key metrics. These metrics help in quantifying the effectiveness and efficiency of the proposed methods under various operational scenarios. The following metrics are used in our work to evaluate performance.

- Queue Length: Measures the average or maximum queue size at each UAV over time. This metric is crucial for assessing the risk of buffer overflow and the effectiveness of queue management strategies.
- Access Latency: The time gap between when data is ready to be offloaded and when it is offloaded.
- Total Flight Time: The amount of time UAVs spend in operation during a mission. This metric is linked to energy consumption but focuses on operational efficiency and mission duration.
- Energy consumption: Total energy consumed by the UAVs, particularly the A_UAV, during their flight and operation. This includes energy used for movement and hovering energy.
- Total Number Of PoIs: Refers to the number of Points of Interest (PoIs) visited or covered during a mission.

To validate the performance of our proposed approach, we compared our proposed approach with a set of baseline approaches on two broad categories of optimization parameters viz. Trajectory planning and Data offloading. We consider the following baseline approaches:

- Distance Aware Trajectory planning (DAT): In this approach, the A_UAV selects to access an I_UAV based on the shortest distance from the current location in each time slot.
- Round Robin based Trajectory planning (RR): In this approach, the A_UAV accesses I_UAVs in sequential order in each time slot.
- Maximum Transmission Power (MAX) data offloading: In each time slot, the A_UAV and the I_UAV operate at the maximum transmission power to offload data.

The proposed approaches are as follows:

- Distance and Latency Aware Trajectory Optimization (DLAT): In this approach, the A_UAV selects the I_UAV based on the minimum distance, with maximum bits to offload and access latency constraint as given in the trajectory optimization problem.

- Hybrid Approach for Trajectory Scheduling (HDLAT): In this approach, the A_UAV selects the I_UAV based on the minimum distance and access latency constraint as given in the trajectory optimization problem up to a certain threshold of battery, i.e., 75% of the total battery. Beyond the threshold, the scheduling algorithm switches to the DAT strategy (proposed approach).

- Lyapunov Optimization for data offloading: In each time slot, the A_UAV and the I_UAV calculate the optimal value of transmission energy using the Lyapunov Optimization.

Experiments were conducted by taking a combination of one of the approaches from both the categories: 1) DAT + MAX, 2) DLAT + MAX, 3) RR + MAX, 4) HDLAT + MAX, 5) DAT + Lyapunov, 6) RR + Lyapunov 7) DLAT + Lyapunov (proposed approach) and 7) HDLAT + Lyapunov

To validate our proposed approach, we compared it with a set of baseline approaches based on two broad categories of optimization parameters: Trajectory Planning and Data Offloading. The following combinations were used in our experiments:

- DAT + MAX: Distance Aware Trajectory planning combined with Maximum Transmission Power data offloading.

- DLAT + MAX: Distance and Latency Aware Trajectory Optimization combined with Maximum Transmission Power data offloading.

- RR + MAX: Round Robin based Trajectory planning combined with Maximum Transmission Power data offloading.

- HDLAT + MAX: Hybrid Approach for Trajectory Scheduling combined with Maximum Transmission Power data offloading

- DAT + Lyapunov: Distance Aware Trajectory planning combined with Lyapunov Optimization for data offloading.

- RR + Lyapunov: Round Robin based Trajectory planning combined with Lyapunov Optimization for data offloading.

- DLAT + Lyapunov: Distance and Latency Aware Trajectory Optimization combined with Lyapunov Optimization for data offloading.

- HDLAT + Lyapunov: Hybrid Approach for Trajectory Scheduling combined with Lyapunov Optimization for data offloading.

These combinations help in evaluating the performance of the proposed approach under various optimization strategies.

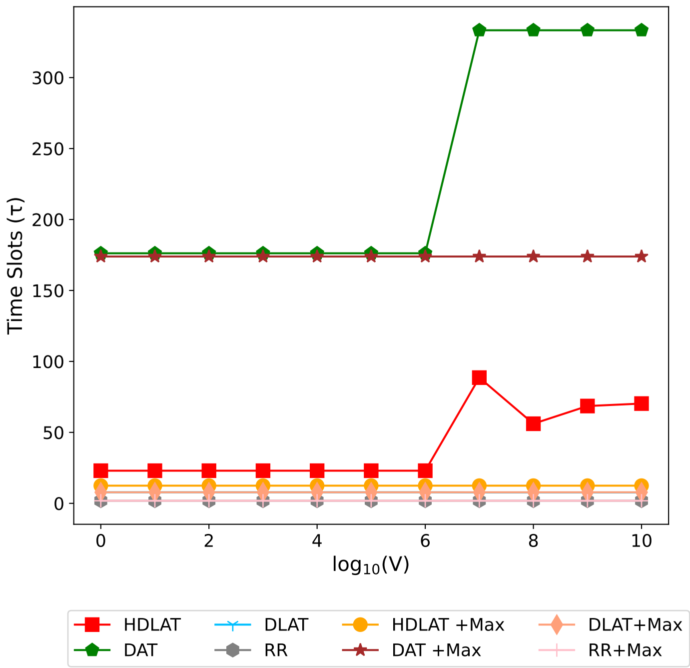

Figure 3.3: Analysis of Average Access Latency of A_UAV

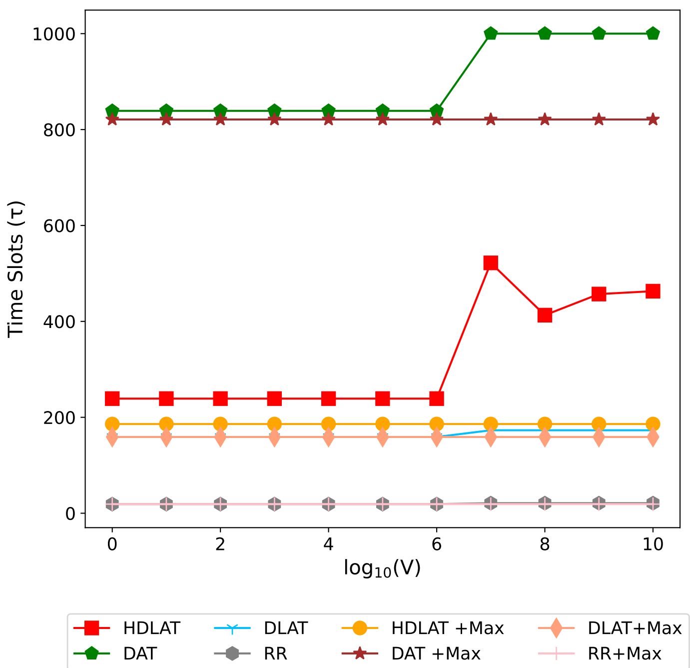

Figure 3.4: Analysis of Total Flight Time of A_UAV

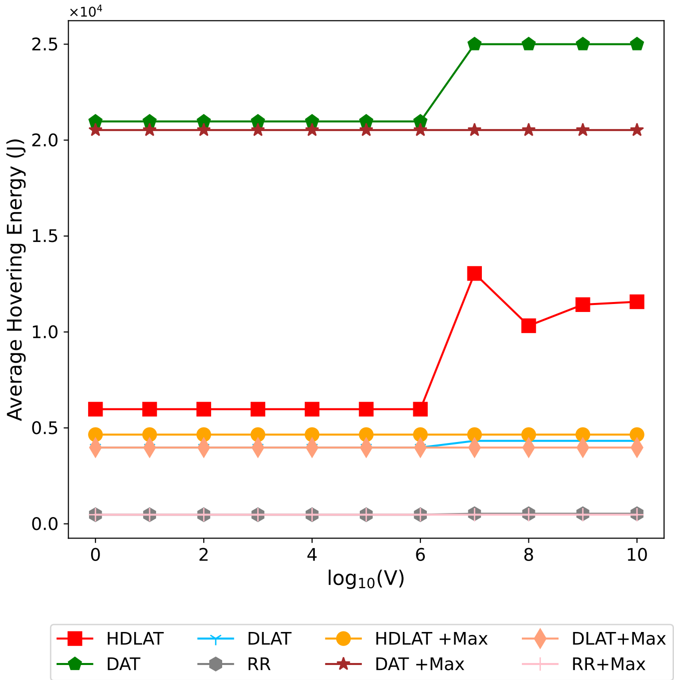

Figure 3.5: Analysis of Hovering Energy of A_UAV

3.6 Results and Discussions

In this section, we discuss the comparative performances of our proposed approaches with baseline approaches.

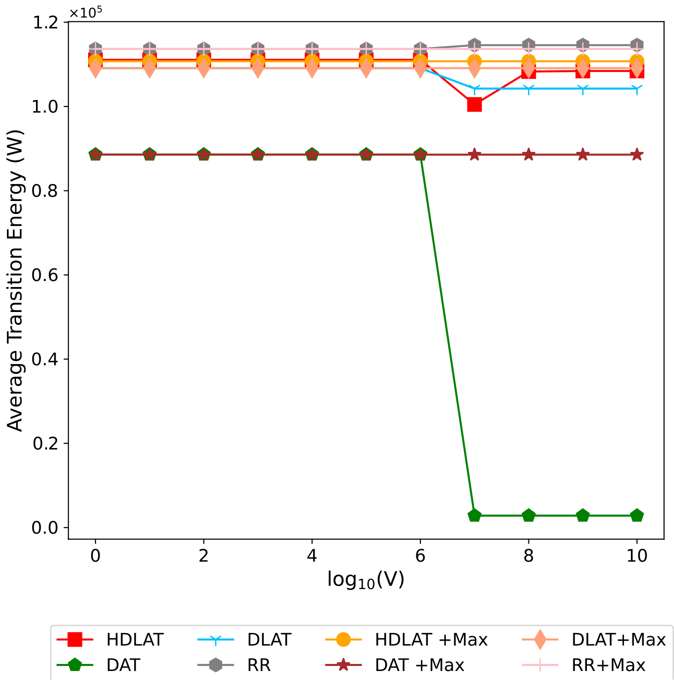

Figure 3.6: Analysis of Transition Energy of A_UAV

3.6.1 Transmission Power and Average Buffer Size

Figure 3.2 depicts the effect of the value of parameter V with respect to the transmission power of the A_UAV. It is evident from Figure 3.2 that all combinations with max power consumption for data offloading always consume the maximum energy, making the average transmission power consumption the same across different values of V. For baseline and proposed approach with Lyapunov-based data offloading, a drop in the energy consumption can be observed for log(V) values beyond 7, because large V forces the system to consume less energy, consequently less data is transmitted to A_UAV. It can be

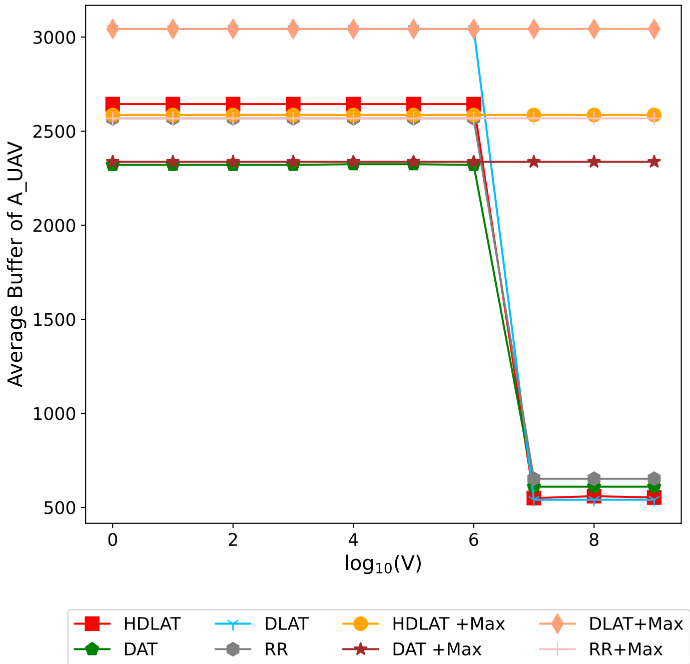

Figure 3.7: Analysis of Average Buffer of A_UAV

observed in Figure 3.8 that the average buffer length of I_UAVs increases in line with the rise of V after hitting the inflection point. The value of V between 6 and 7 could maintain the queue buffer and consume less transmission energy for I_UAVs. Similarly, A_UAV has fewer data transmitted from the I_UAVs for higher values of V, which would decrease the total data collection or transmission further by A_UAV as shown in Figure 3.7.

3.6.2 Hovering energy of A_UAV

The plot in Figure 3.5 depicts the impact of trade off parameter V on the hovering energy of A_UAV. The total hovering energy starts increasing after the inflection point because A_UAV takes more time slots to collect the same amount of data from I_UAVs. As a result, the energy consumption of A_UAV significantly increases for DAT and HDLAT, whereas it remains constant for DLAT (total flying time is less). Similarly, Figure 3.6 shows the evolution of total transition energy with V. It is interesting to observe that for DAT baseline A_UAV stays in the field for a longer time. As a result, transition energy

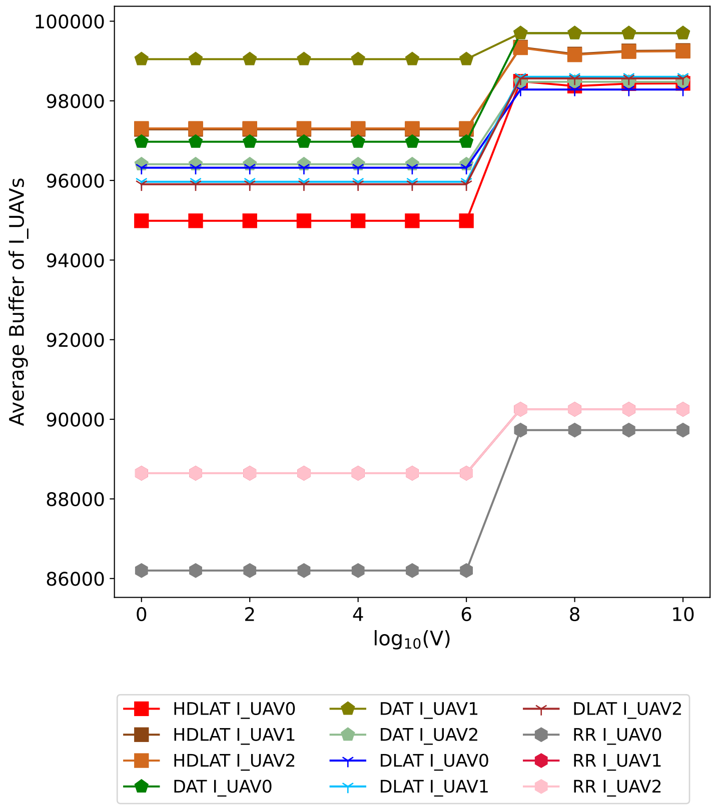

Figure 3.8: Analysis of Average Buffer of I_UAV.

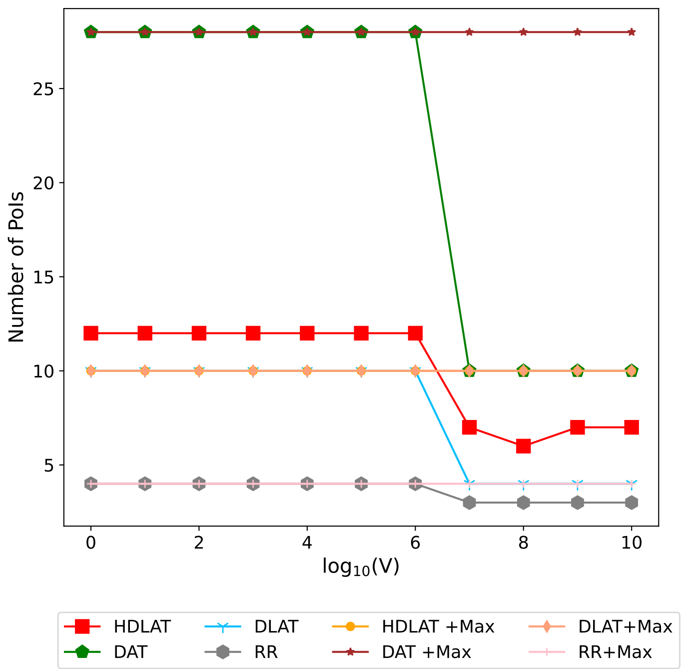

Figure 3.9: Analysis of Total Number of PoIs Covered.

is higher compared to DLAT and HDLAT. However, the A_UAV transition energy starts decreasing after the inflexion point. Similarly, DAT and HDLAT consume less battery in every time-slot as both save on transition energy by selecting the nearest I_UAV. This allows A_UAV to stay longer in the field, which is illustrated in Figure 3.4. The baseline DAT remains for a more extended time, whereas RR stays for the least number of time slots before running out of battery. The proposed approaches of DLAT and HDLAT lie between the extreme baselines for the different analyses conducted. This shows that the proposed approach has an optimized trade-off between energy saving along with the end-to-end data offloading from multiple I_UAVs.

3.6.3 I_UAVs Access Latency

Figure 3.3 shows the effect of the trade-off parameter V on the average access latency of the system. The trajectory of the A_UAV affects the access sequence and waiting times of the I_UAVs to offload their data to A_UAV. This can be observed in the proposed DLAT + Lyapunov, which has an upper bound on access latency throughout the system. Similarly, RR-based baseline approach has an access latency of 2 time-slots whereas for both DAT and DAT + MAX baseline approaches, the average access latency is higher. The average access latency for DAT baseline approach becomes worse with increasing V. The same remains stable for DLAT in both scenarios. HDLAT, as per expectation, remains between the DLAT and DAT approaches. It could be related to the fact that an increase of V causes the A_UAV to spend more time slots to collect the data from I_UAVs. An increase in flying time of A_UAV is influenced by a decrease in transition and transmission power, which increases the average access latency of DAT and HDLAT approaches. In contrast, it remains constant for DLAT and RR because of latency constraints. Our proposed approaches lie within the extreme baselines and maintain the average access latency by saving on transmission energy and transition energy in HDLAT by switching from DLAT to DAT after 75 percent of the battery is consumed.

The Average energy consumption of A_UAV includes transmission, transition, and hovering energy consumption. Similarly, the tradeoff between the average access latency and the average energy consumption can be observed as the average access latency of the system reduces, the average energy consumption increases. By the definition and from Figures 3.2, 3.5, 3.3 and 3.6, this can be observed that the RR baseline has the least average access latency as well as the highest energy consumption whereas, DAT has the highest average access latency and the least energy consumption.

From Figure 3.3, this can be observed that for HDLAT the average access latency of A_UAV is reduced by approximately 70% as compared to the greedy approach(DAT) and remains constant for DLAT. The RR baseline has the least average access latency, but the gap between HDLAT and RR is much lesser as compared to the gap between HDLAT and DAT.

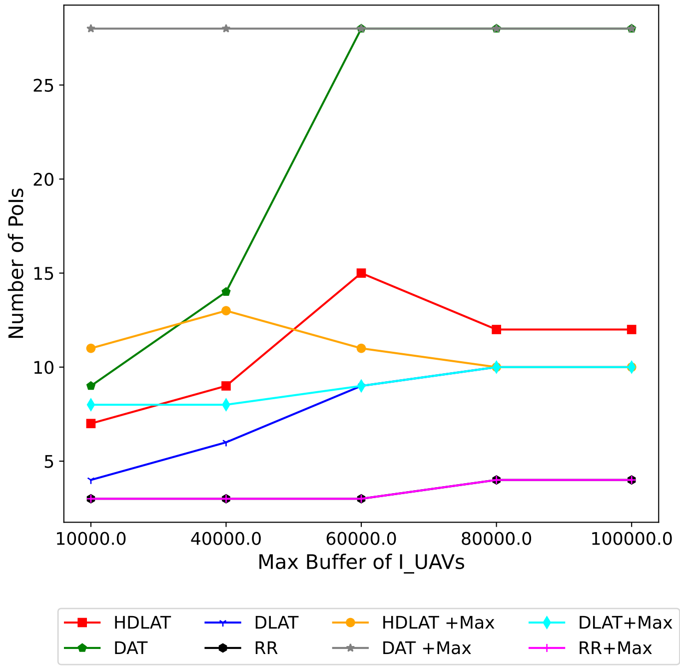

Figure 3.10: Analysis of Total Number of PoIs Covered and Max Buffer of I_UAVs.

3.6.4 Coverage of PoIs

The coverage of PoIs by the A_UAV can be defined as the number of PoI locations whose data has been offloaded to A_UAV by the I_UAVs. Figure 3.9 shows the effect of trade-off parameter V value on the number of PoIs covered in the system. It can be observed that A_UAV can serve more PoIs for both DLAT and HDLAT approaches than the RR baseline approach. The DAT-baseline approach serves relatively more PoIs than DLAT and HDLAT by saving on transition and transmission energy, but not maintaining low access latencies of I_UAVs.

In Figure 3.10, the effect of increasing the buffer size of I_UAVs on the PoIs is shown. It can be observed, the number of PoIs served increases with an increase in buffer size. In our proposed approach, the performance of DLAT and HDLAT is a tradeoff between two extreme baselines. DAT baseline approach covers more locations but at the cost of access latency, as shown in Figure 3.11. Similarly to average access latency, the tradeoff between the PoIs coverage and the average energy consumption is also evident from Figures 3.6 and 3.9. The approach with higher average energy consumption also has a reduced PoIs coverage. RR has the highest energy consumption and covers the least number of PoIs, whereas DAT has the least energy consumption but covers the maximum number of PoIs. In the optimal processing zone of log(V) between 6 and 7, HDLAT covers more than double the number of PoIs as compared to RR while consuming less energy.

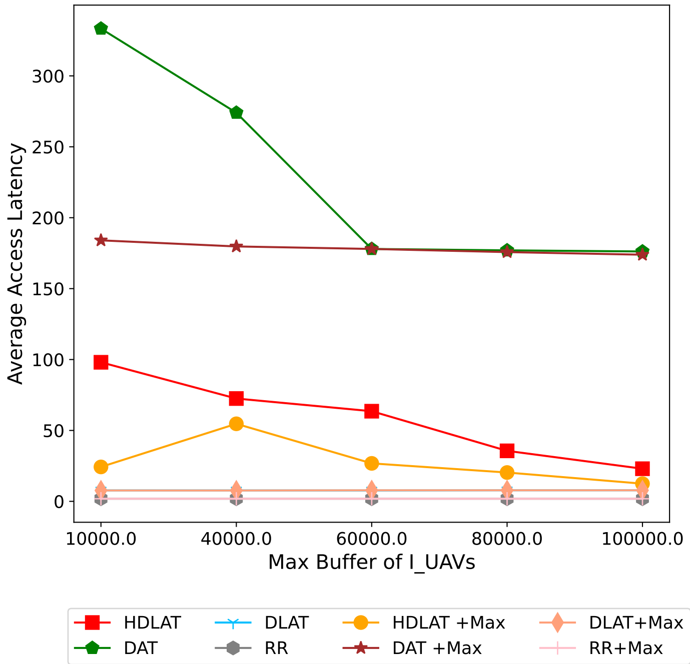

Figure 3.11: Analysis of Average Access Latency and Max Buffer of I_UAVs.

3.7 Chapter Summary

This chapter introduces the Distance and Latency Aware Trajectory (DLAT) and Hybrid Distance and Latency Aware Trajectory (HDLAT) algorithms, tailored for planning the flight paths of Access UAVs in a hierarchical multi-UAV network. These algorithms are crafted to tackle challenges such as limited infrastructure, constrained battery life, and the finite buffer capacity of each UAV. To enhance coordination in the absence of a central controller, we incorporate a unique message-based method that allows Access UAVs (A_UAVs) to estimate the locations of I_UAVs independently. Furthermore, we employ a Lyapunov-based online optimization framework to effectively manage the multilevel queue system. However, there are still areas for improvement, particularly in refining the trajectory scheduling for I_UAVs and moving towards a more autonomous system operation.

4| Age-of-Information based Multi-UAV Trajectories using Deep Reinforcement Learning

This chapter delves into the trajectory scheduling of multiple UAVs in environments with limited infrastructure, focusing on minimizing the Age of Information (AoI). It presents the Markov Decision Process (MDP) formulation for a hierarchical multi-UAV system. In this research, we explore the use of the Deep Deterministic Policy Gradient (DDPG) algorithm and the Advantage Actor-Critic (A2C) network for developing effective UAV policies. The main objective is to reduce AoI and energy consumption for access UAVs. We aim to enhance the coverage of Points of Interest (PoIs) with higher crowd density, decrease access latency to these PoIs, and optimize energy efficiency.

This chapter is divided into the following sections. In section 4.1 the proposed decentralized AoI minimal scheduling for two-level networks is discussed. The section 4.2 discusses the MDP formulation for I_UAVs and A_UAV. The section 4.5 and section 4.6 discuss the experiments and results. The chapter concludes with section 4.7, providing a concise summary of the covered content.

4.1 System Model

In this section, we introduce a multilayer network model designed for crowd surveillance utilizing multiple UAVs in an infrastructure-less environment, as illustrated in 4.1. The group of I_UAVs is tasked with monitoring an area containing randomly distributed Points of Interest (PoI) for crowd surveillance. However, due to the limited transmission range of I_UAVs, one-hop transmission becomes ineffective. Consequently, an A_UAV

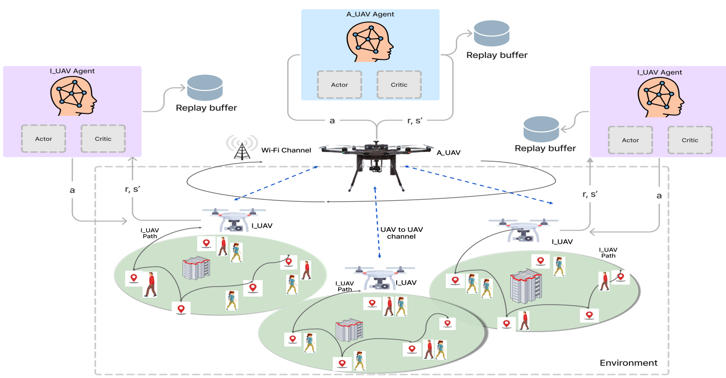

Figure 4.1: System Setup for Reinforcement Learning based Trajectory Scheduling

is deployed as an access platform to relay the data collected by I_UAVs to the base station. The PoIs are organized into distinct clusters, with each I_UAV assigned to cover one cluster of PoIs. The set of I_UAVs is denoted by $\mathcal{N}$, and a single A_UAV is deployed for assisting in data relay to the base station. Without loss of generality, the location of Point of Interest l_i is represented as $(x_i, y_i, 0)$. At time slot t, the position of I_UAV is denoted by $\mathcal{S}i(t) = (x_i(t), y_i(t), z_i(t))$, and the position of A_UAV is denoted by $\mathcal{S}_{\text{access}}(t)$. The maximum velocity of both A_UAV and I_UAVs is constrained due to mechanical limitations. Each PoI has a distinct crowd arrival rate, which is not known in advance to either set of UAVs. Each I_UAV is assumed to be equipped with a visual sensor, enabling it to detect the number of people in each frame of the PoI and store the latest information of PoI within the assigned cluster. The trajectory of I_UAVs is jointly designed while considering battery and latency to serve the PoIs. Similarly, the trajectory of A_UAV is designed to relay data from the I_UAVs to the BS while optimizing battery and AoI. The 4.1 illustrates the system design where the A_UAV interacts with I_UAVs to relay data to the BS.

4.1.1 Data Freshness

The concept of Age of Information (AoI) [38] is used to quantify the freshness of data at the j^{th} Point of Interest (PoI), which is measured by the elapsed time since the latest information was generated at that PoI.

$$AoI^{j}_{access}(l_j, t) = t - t^{'}, \tag{4.1}$$

where
$AoI^{j}_{access}(l_j, t) = t - t^{'}$ is AoI of j^{th} PoI at time slot t on A_UAV. This is the difference between current time slot and time slot $t^{'}$ when it is last accessed by I_UAV.

4.1.2 Transition Energy

The transition energy of A_UAV and I_UAV refers to the energy consumed during the movement from one point to another. It can be expressed as follows:

$$E^{trans}_{access} = \kappa \cdot ||vel(t)||^2 \tau_{trans} \tag{4.2}$$

where κ is a system constant specific to UAVs, vel(t) represents the velocity of the UAVs at time t, and τ_{trans} denotes the time taken to transit from one location to another. This equation calculates the energy consumption during the movement of UAVs based on their velocity and the time taken for transition.

4.1.3 Communication energy

I_UAV offloads the data to the A_UAV through a wireless channel. The energy consumed to transmit the data from I_UAV to the cloud is given as:

$$E_i^{comm}(t) = (2^{\frac{d_i^{off}(t)}{W\cdot\tau}} - 1) \cdot \frac{N_0 W}{\zeta} \cdot \tau_{comm} \tag{4.3}$$

where τ_{comm} is the duration for communication, $d_i^{off}(t)$ is the number of bits offloaded, W is the bandwidth, τ is the total duration of the timeslot, N_0, ζ are communication channel parameters.

4.1.4 Communication Channel

The connection between I_UAVs and A_UAV is restricted by limited communication range, which means communication is only possible when they are within each other's communication range. The path loss of a link between I_UAVs and A_UAV in the presence of both line-of-sight (LoS) and non-line-of-sight (NLoS) conditions can be expressed as:

$$L_\alpha = L_\alpha(r_0) + 10\phi \log(\frac{r^{'}}{r_0}) \tag{4.4}$$

where $\alpha \in \{\text{LoS}, \text{NLoS}\}$ and ϕ is the path loss exponent. The probability of LoS link, (P_{LoS}), depends on angle of elevation and environmental constraints (e_o and e_1) as given in Equation 4.5:

$$P_{\text{LoS}} = \frac{1}{1 + e_o.exp(-e_1[\theta - e_o])} \tag{4.5}$$

The average path-loss is calculated as:

$$L = P_{\text{LoS}}.L_{\text{LoS}} + (1 - P_{\text{LoS}}).L_{\text{NLoS}} \tag{4.6}$$

4.2 RL based Trajectory Scheduling

In this section, we propose a decentralized solution utilizing a two-level (DRL) approach to optimize the trajectory scheduling of UAVs. The solution operates in a fully distributed manner, with each UAV learning its policy for trajectory scheduling. The training and evaluation processes are conducted independently for each UAV.

The problem is formulated as a Markov Decision Process (MDP), where each I_UAV can only observe its local environment. The objective is to minimize the expected cumulative energy consumption and latency over the time horizon for each I_UAV. This objective is captured by Equation 4.7, which represents the optimization goal of the problem.

By utilizing DRL in a distributed manner, each I_UAV can learn its own trajectory scheduling policy, taking into account its local environment and optimizing for energy efficiency and latency. This approach enables effective coordination among the UAVs while achieving the joint goals of energy efficiency and minimizing AoI and latency.

$$\min_{l_i(t)} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} (\mathbb{E}[E_i(t)] + \mathbb{E}[AoI_i(t)] + \mathbb{E}[RL_i(t)]) \tag{4.7}$$

$$0 \leq \sum_{t=1}^{T} B_i(t) \leq B^{max}, \quad \forall i \in \{1, 2, .., N\}, \quad \forall t \in T \tag{4.8}$$

$$||S_i(t) - S_i(t-1)|| \leq v_{max}\tau, \quad \forall i \in \{1, 2, .., N\}, \quad \forall t \in T \tag{4.9}$$

$$0 \leq \sum_{t=1}^{T} B_{access}(t) \leq B^{max}_{access}, \quad \forall i \in \{1, 2, .., N\}, \quad \forall t \in T \tag{4.10}$$

$$||S_{access}(t) - S_{access}(t-1)|| \leq V_{max}\tau, \quad \forall t \in T \tag{4.11}$$

$$\sum_{i=1}^{N} u_i(t) = 1, \quad \forall i, \forall t \tag{4.12}$$

$$u_i(t) \in \{0, 1\}, \quad \forall i, \forall t \tag{4.13}$$

$$\sum_{i=1}^{P} x_i(t) = 1, \quad \forall i, \forall t \tag{4.14}$$

$$x_i(t) \in \{0, 1\}, \quad \forall i, \forall t \tag{4.15}$$

The problem introduces several constraints that govern the behavior of the I_UAVs. Constraint (4.8) and (4.10) sets a maximum limit on the energy or battery capacity of each I_UAV and A_UAV respectively, ensuring they does not exceed a certain threshold. Similarly, constraint (4.9) and (4.11) restricts the transition energy of the I_UAVs during each time slot to a predefined maximum value. These constraints ensure energy efficiency and prevent excessive energy consumption.

Constraints (4.14) and (4.15) enforce the selection of only one PoI by each I_UAV in each time slot. This limitation ensures that each I_UAV focuses on monitoring a single PoI at a time. Similarly, for A_UAV the constraint (4.12) and (4.13) restricts its focus to only one I_UAV in a given timeslot.

4.3 MDP Formulation

To address the problem 4.7, we reformulate it as an MDP (Markov Decision Process) for trajectory scheduling of the I_UAVs and A_UAV. An MDP is represented by the tuple $\langle O, A, R, P\rangle$, where O, A, R, and P denote the state set, action set, reward, and state transition probability, respectively. In our context, each I_UAVs and A_UAV is treated as an agent in the formulated MDP. We define the observation set, action set, and reward function for the UAV agents as follows:

- The observation space or state set includes the

$$O_i(t) = \{B_i(t), RL_{l_{1..p}}(t), S_{l_{1..p}}(t)\} \tag{4.16}$$

The current observation of the i-th I_UAV at time t is denoted as $O_i(t)$. It consists of variables such as the battery level $B_i(t)$ of the i-th I_UAV, the latency $RL_{l_{1..p}(t)}$ of the p-th Point of Interest (PoI), the location $S_{l_{1..p}}(t)$ of the p-th PoI.

$$O_{access}(t) = \{S_{access}(t), B_{access}(t), AoI_{1..j}(t), S_{l_{1..p}}(t)\} \tag{4.17}$$

The observation space of A_UAV at time t is denoted as $O_{access}(t)$, which includes information such as the current position, battery level $B_{access}(t)$, the AoI of each I_UAV ($AoI_{1..j}(t)$), and the location of the p^{th} PoI ($S_{l_{1..p}}(t)$).

- Reward: The reward function is a key component in reinforcement learning as it guides the learning process toward the optimal policy. Its quality directly affects the convergence of the network training. In case of A_UAV, the reward function is defined as a combination of two factors: the average latency of the cluster and the battery consumption of A_UAV in its assigned cluster. The reward function can be expressed as follows:

$$r_{\text{access}}(t) = -(AoI^{avg}_{\text{access}}(t)) + \frac{B^{\text{left}}_{\text{access}}(t)}{B_{\text{access}}(t)} - r^{\text{penalty}}_{\text{access}} \tag{4.18}$$

$$r_i(t) = -(RL^{avg}_{\text{i}}(t)) + \frac{B^{\text{left}}_{\text{i}}(t)}{B_{\text{i}}(t)} - r^{\text{penalty}}_{\text{i}} + CI_i \tag{4.19}$$

The equation 4.19 and 4.18 above calculates the reward function used by the reinforcement learning agent to determine the optimal policy based on the expected returns. The reward function depends on two factors: the average latency of the cluster assigned to the i^{th} I_UAV, and the ratio of the battery used in the given interval, $B^{left}_i(t)/B_i(t)$, by the I_UAV or A_UAV agent. Additionally, a penalty term introduced in equation 4.20, $r^{penalty}_i$, is introduced to capture the relationship between the average access latency value at the agent(I_UAVs and A_UAV) and the total available time slots. The last component signifies the number of people(crowd density) seen so far by i^{th} I_UAV. The constant K_i is a positive value. The first two conditions impose an additional penalty on the agent if the average latency exceeds half of the total number of time slots in any given episode. The last two conditions prevent the agent from always selecting the PoI with either the highest or the lowest AoI. These constraints encourage the agent to explore and avoid getting stuck at a single PoI during the learning process.

$$r_{\mathrm{i}}^{\mathrm{penalty}} = \begin{cases} +K_{\mathrm{i}} & \text{if } RL_{\mathrm{i}} \geq T/2 \\ +K_{\mathrm{i}} & \text{if } RL_i(t) \equiv \min(RL_i(t)) \\ 0 & \text{else} \end{cases} \tag{4.20}$$

- Action: The action of I_UAV encompasses the PoI index, while the action of A_UAV includes the index of I_UAV.

$$a_{\mathrm{access}}(t) = i \tag{4.21}$$

$$a_i(t) = l_j(t) \tag{4.22}$$

4.4 DRL based UAV Trajectory Scheduling

In the research discussed Deep Reinforcement Learning (DRL) methods like Deep Deterministic Policy Gradients (DDPG) and Advantage Actor-Critic (A2C) are employed to optimize the trajectory scheduling of I_UAVs and A_UAVs. These advanced DRL methods are particularly effective in dynamic and complex environments where traditional heuristic or rule-based approaches may not perform well.

The DDPG algorithm uses an actor-critic framework, which consists of two neural networks: the actor-network and the critic network. The actor-network at A_UAV selects actions, which in this case are the indices of I_UAVs to be served next. The critic network evaluates these actions by predicting the expected rewards.

The A2C method also uses an actor-critic approach but enhances learning stability through advantage estimation. This technique reduces the variance in the value function estimation, making the learning process more efficient and effective. The actor-network in A2C at A_UAV selects the index of the I_UAV to be served, while the critic network evaluates the chosen actions by estimating the value of the current state.

4.4.1 Complexity Analysis

Complexity based on DRL algorithm is determined by the configuration of the neural networks utilized. In our work, actor critic based DDPG and A2C are implemented so it is calculated based on the number of neurons present in each layer of both the actor and critic networks. During the training phase, this complexity is expressed as a function that considers the quantity of fully connected layers and the dimensions of the input layers. However, in the testing phase, each agent utilizes only its actor network, resulting in a simplified computational load. The algorithm's convergence is facilitated through the application of a gradient descent method.

Algorithm 3: DRL based Trajectory Scheduling

Input: LR_A_ddpg, LR_C_ddpg, LR_A_a2c, LR_C_a2c $\mathcal{Z}^{u_i}$, $\mathcal{Z}^{access}$
Initialize environment and UAV agents
Initialize DRL agent and replay buffer for each UAV $i \in N$ and A_UAV
Initialize state normalizers and exploration variables
for $UAV = u_i, u_j, ..., u_n, A\_UAV$ **do**
  for *episode=1,2,...* **do**
    Initial observation $O_i(t), O_{access}(t)$
    while $t \leq T$ *and* $B_{access}(t) > 0$ *and* $B_i(t) > 0$ **do**
      Select action $a_i(t)$ and $a_{access}(t)$
      compute battery left $B_{access}(t)$ and $B_i(t)$ for A_UAV and I_UAV
      Each I_UAV u_i makes the next observation $O_i(t)$ and receives reward $r_i(t)$ from the environment.
      A_UAV makes the next observation $O_{access}(t)$ and receives reward $r_{access}(t)$ from the environment.
      Store transition $< O_i(t), a_i(t), r_i(t), O_i(t+1) >$ in replay buffer $\mathcal{Z}^{u_i}$
      Store transition $< O_{access}(t), a_{access}(t), r_{access}(t), O_{access}(t+1) >$ in replay buffer $\mathcal{Z}^{access}$
      if $a_{access}(t) == u_i$ **then**
        update $AoI_{1...j}(t)$ at A_UAV // Considering u_i and A_UAV communicated at time t
        Update the input state as given in Equation 4.17
      end
      if $a_i(t) == i$ **then**
        update $AoI_{1...j}(t)$ at u_i // Considering u_i visited i^{th} PoI at time t
        Update the input state as given in Equation 4.16
      end
      Sample Minibatch for each UAV agent.
      Upadate the weights of network as per rules defined for A2C or DDPG in every TS steps.
    end
  end
end
Performance metrics

4.5 Experiments

In this section, we describe the simulation setup used to evaluate the effectiveness of our proposed DRL-based trajectory optimization for I_UAVs and A_UAV, followed by the presentation of results and discussions. Our experiments are conducted in an area of 400 by 400 square meters. The Points of Interest (PoIs) are distributed across the region in separate clusters.

The simulation setup includes three I_UAVs acting as data collectors and the A_UAV serving as the access platform. The optimization of AoI and energy consumption is performed jointly by the I_UAVs and A_UAV. The I_UAVs autonomously plan their data collection and data relay schedules in a decentralized manner. Each I_UAV determines the next PoI to visit for data collection based on factors such as the crowd

density observed so far, access latency of the PoI, and its current battery level. Similarly, the A_UAV considers the current AoI of the system and its own battery level to schedule the I_UAVs for the next time slot. The number of people (crowd density) present at each PoI is unknown to the I_UAVs beforehand. For simulation purposes, the crowd density at each PoI is generated from pre-defined normal Gaussian distributions. It is important to note that our proposed solution is decentralized, meaning that the I_UAVs and A_UAV make decisions independently based on their observations and objectives.
Due to limited communication and lossy channels at low altitudes, the I_UAVs are unable to directly communicate with each other. The A_UAV approaches the cluster to establish a connection and relay the data to the base station (BS).

4.5.1 Crowd Density

In our system, the crowd density at each PoI is unknown to the inspection UAVs (I_UAVs) in advance. To estimate the crowd density, each I_UAV keeps track of the average number of people observed at each PoI up to the current point in time. Using this information, the average crowd density of each PoI can be computed. Our goal is to optimize the data transmission frequency of the I_UAVs from the locations where crucial information, such as the number of people at a PoI, is available. This optimization aims to enhance the efficiency of data transmission by focusing on areas with valuable information related to crowd density. By simulating these scenarios, we aim to demonstrate the effectiveness of our proposed decentralized solution for optimizing trajectory planning and data collection for the I_UAVs and A_UAV. The simulation parameters are listed in Table 4.1

4.5.2 Baselines:

For comparison, we considered three baseline approaches as follows: greedy and random policy.

- Maximal AoI First (MaxAF): In AoI-based greedy policy, the I_UAV will always pick the PoI with maximum AoI to collect the data. Similarly, A_UAV will pick the I_UAV with maximum average access latency.

- Minimum Distance First (MinDF): In the distance-based greedy policy, both A_UAV and I_UAVs will optimize the battery consumption during the transition. As a result, I_UAVs will always schedule the nearest PoI to collect the data, whereas A_UAV will schedule the nearest I_UAVs to collect the data collected from its respective cluster.

- PSO: The Particle Swarm Optimization (PSO)[54] is a metaheuristic approach in which I_UAVs and A_UAV find the best location in each time slot. In PSO each particle represents a candidate solution. During each iteration, particles adjust their positions based on their own best solution and the best solution found by the entire

Table 4.1: List of Simulation Parameters

Parameters	**Values**
Channel Bandwidth	1 MHz
Battery capacity of I_UAV	250 kJ
Battery capacity of A_UAV	700 kJ
κ	1
Noise Power	-100 dBm
Velocity of A_UAV	50 m/s
Velocity of I_UAV	30 m/s
The path-loss constant g_0	10^{-4}
The path loss exponent θ	2 to 4
Mass of small I_UAVs	5 Kg
Mass of A_UAV	9.65 Kg
Replay Memory Buffer($\mathcal{Z}^{u_i}$, $\mathcal{Z}^{access}$)	10000
Mini-batch size	64
DDPG Actor learning rate(LR_A_ddpg)	0.00011
DDPG Critic learning rate(LR_C_ddpg)	0.00022
Optimizer method	Adam
Reward discount	0.001
Learning Rate of DQN	0.00011
A2C Actor learning rate (LR_A_a2c)	0.000035
A2C Critic learning rate (LR_C_a2c)	0.00002

swarm. By dynamically adapting their positions, particles collectively explore the search space in search of an optimal solution as defined in Algorithm **??**.

4.6 Results

In this section, we discuss the detailed analysis of our proposed DRL-based approaches comparing their performance with baseline approaches. To confirm the advantage of DRL-based trajectory scheduling across various metrics, such as cumulative reward, average crowd density, average number of timesteps per episode, average AoI of data at A_UAV, and average access latency of PoI for I_UAV, a comparison of training results for 1000 episodes is conducted.

To ensure a robust evaluation, the test results are presented for the same scenarios across 50 episodes. This rigorous examination aims to provide a thorough understanding of the performance gains achieved by our DRL-based trajectory scheduling approach in contrast to the baseline methods. The results are presented for multiple iterations. In the Particle Swarm Optimization (PSO) algorithm, we utilize a swarm consisting of 1000 particles for each UAV agent, including both the Access UAVs (A_UAVs) and the Inspection UAVs (I_UAVs), with each particle having 50 dimensions. The velocity constraints for the particles are set to ensure controlled movement during the optimization process. For the A_UAVs, the velocity components are limited to the range of [-10, 10] units. In contrast, for the I_UAVs, the velocity components are constrained within the range of

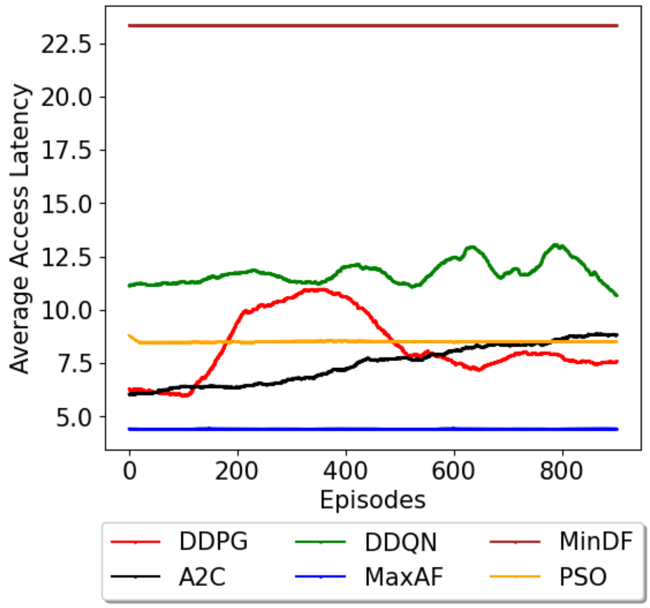

Figure 4.2: Average Access Latency of I_UAVs during Train

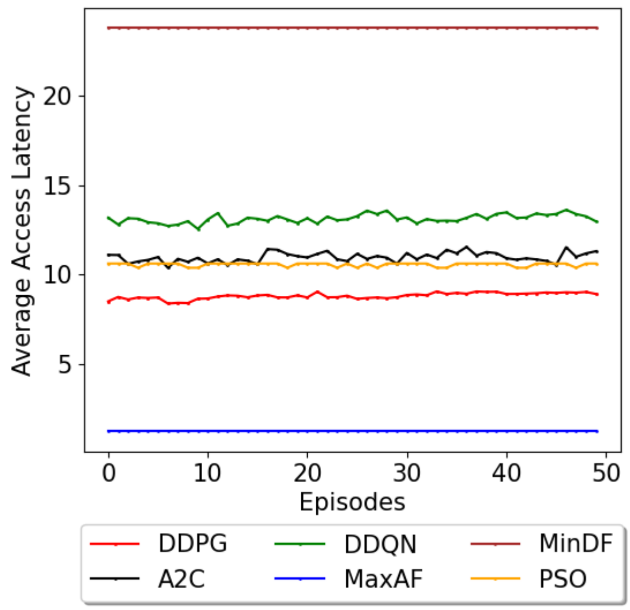

Figure 4.3: Average Access latency of I_UAV during Testing

[-3, 3] units. These parameter values are selected to strike a balance between exploration and exploitation, allowing the particles to efficiently navigate the solution space without excessive oscillations or divergence. This setup aids in optimizing the trajectory planning and data collection tasks performed by the UAVs, ensuring both effectiveness and efficiency in the operations.

4.6.1 Energy Efficiency of System

The chart depicted in 4.4 provides an overview of the total number of timeslots utilized by each approach in an episode. Initially, all approaches exhibit a relatively shorter

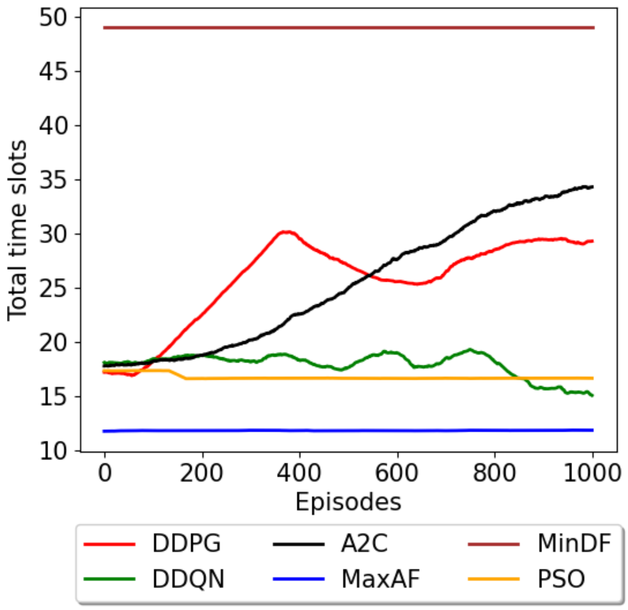

Figure 4.4: Timesteps of Different Approaches during Train

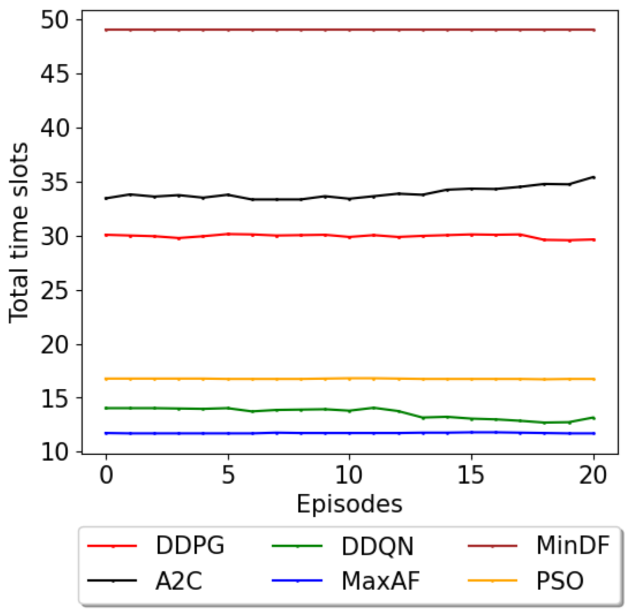

Figure 4.5: Timesteps of Different Approaches during Testing

duration of activity. However, as experience accumulates, both DDPG and A2C extend their operational time, indicating a learned optimization in battery usage. Conversely, the baseline methods PSO and MaxAF display suboptimal performance in this aspect. The MinDF approach, on the other hand, sustains a longer duration, consistently optimizing battery usage by minimizing travel in each time step.

In the testing scenario (as illustrated in the 4.5), a parallel behavior is observed. Both DDPG and A2C demonstrate prolonged field presence compared to PSO, DDQN, and MaxAF. Notably, the baseline approach MaxAF exhibits the highest battery consumption due to its consideration of the maximum AoI.

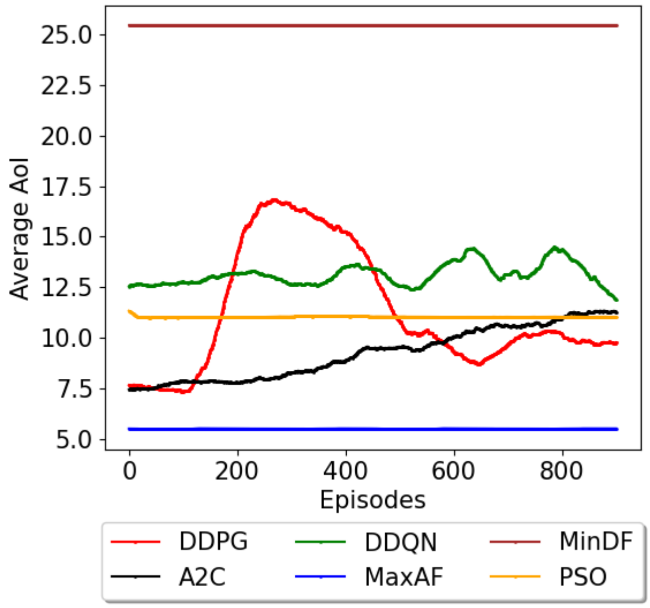

Figure 4.6: Average AoI of A_UAV during Train

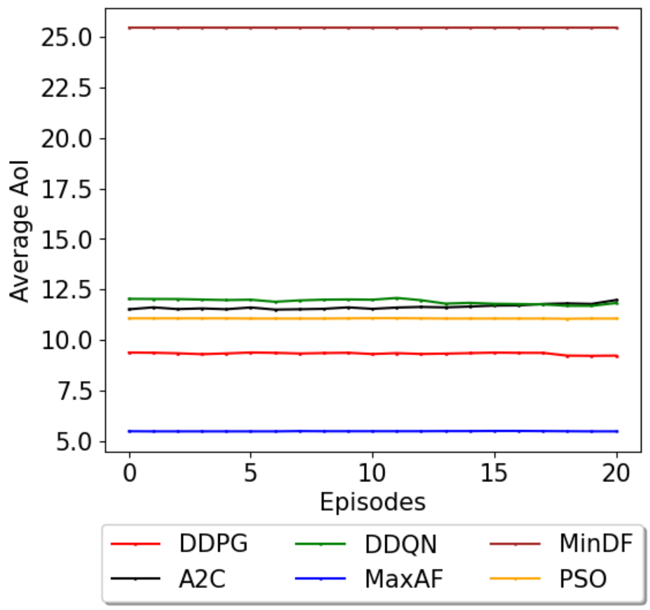

Figure 4.7: Average AoI of A_UAV during Testing

4.6.2 Analysis of Average AoI of A_UAV

The 4.6 illustrates the average AoI of data at the A_UAV for the proposed approaches and baseline methods. Notably, the MaxAF baseline approach exhibits the lowest average AoI, as it prioritizes the consideration of maximum AoI when selecting the I_UAV for data collection. However, this comes at the cost of the A_UAV traveling without considering battery constraints, leading to maximum energy consumption. However, more energy consumption leads to early termination of the episode.

Conversely, the MinDF baseline approach yields the highest average AoI since the A_UAV consistently selects the nearest I_UAV for service without considering the AoI-based

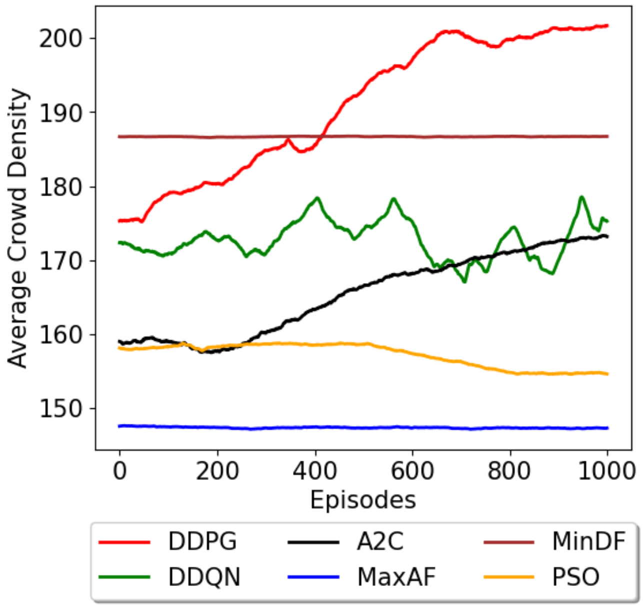

Figure 4.8: Average Crowd Density at I_UAV during Train

constraint, resulting in the maximum average AoI. The methods relying on DDQN, A2C and PSO exhibit superior performance compared to the baseline approaches; however, they do not reach the level achieved by DDPG.
In contrast, our proposed approaches, both based on DDPG and A2C, have successfully optimized the AoI, showcasing improved performance in this aspect. In the testing phase, it could be inferred from the 4.7.

4.6.3 Analysis of Average Access Latency of I_UAVs

The 4.2 presents the average access latency of the I_UAVs for both the proposed approaches and baseline methods. It can be observed that the baseline approach MaxAF maintains the minimum AoI while selecting the PoIs with the highest AoI to collect data. Similarly MinDF, the baseline approach consumes the least energy as the I_UAVs always selects the nearest PoI to serve. Conversely, our proposed approaches, employing DDPG, A2C, and PSO, outperform the baselines in terms of optimizing access latency at the I_UAVs level, all while addressing additional objectives. The DDQN approach, while slightly on the higher side, still outperforms MinDF From the testing results in 4.3, it is evident that the average access latency achieved by the DDPG-based approach is lower than that of other techniques. Notably, MinDF and MaxAF serve as the upper and lower bounds, respectively, in this evaluation.

4.6.4 Analysis of Average Crowd Density

The Figure 4.8 provides insights into the average crowd density observed in the trajectories of I_UAVs for both the proposed approaches and baseline methods. The MaxAF baseline approach maintains crowd density by covering all points based on the Age of Information

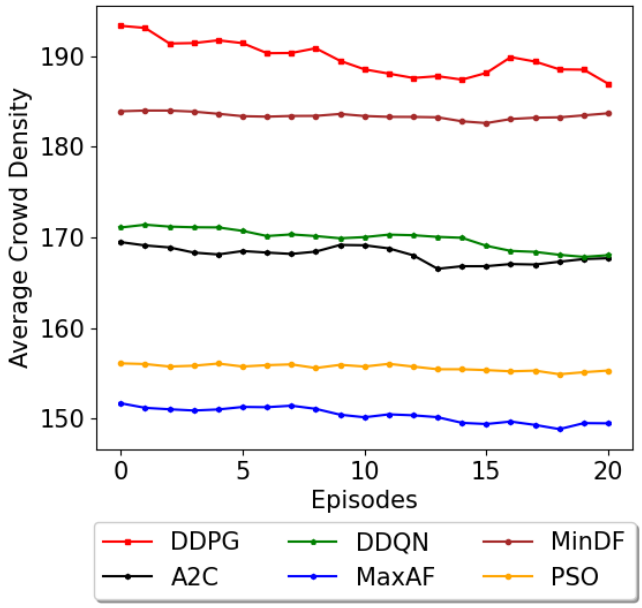

Figure 4.9: Average Crowd Density at I_UAV during Testing

(AoI) constraint. Conversely, the MinDF baseline approach, while achieving lower energy consumption, may neglect points with higher crowd density as I_UAVs consistently opt for the nearest Points of Interest (PoI) and in the Figure 4.9, it could be observed, that this approach remains constant across the episode. In contrast, our proposed approaches, both based on DDPG and A2C, demonstrate superior performance in optimizing crowd density at the I_UAVs level, in addition to addressing other objectives. For the Average crowd density metric, DDQN is not stabilizing and the PSO technique is not performing as required. The testing results, as depicted in Figure 4.9, highlight the improved performance of the DDPG-based approach compared to other algorithms.

4.6.5 Reward Analysis

The Figure 4.10 and Figure 4.11 depicts the cumulative reward obtained by A_UAVs and I_UAV respectively. Notably, the cumulative reward of over 1000 episodes is highest for DDPG, followed by A2C and PSO. This suggests that DDPG outperforms both A2C and PSO in terms of learning and optimizing energy utilization and other objectives, as evident from figures Figure 4.4 and Figure 4.5.

For detailed simulation parameters, including optimal learning rates for all DRL-based strategies, please refer to Table 4.1.

4.7 Chapter Summary

This chapter presents a learning based approach on managing a hierarchical multi-UAV network in infrastructure-deficient environments using a Markov Decision Process (MDP) framework. It highlights the deployment of advanced Reinforcement Learning algorithms viz. Deep Deterministic Policy Gradient (DDPG), Advantage Actor-Critic (A2C), and

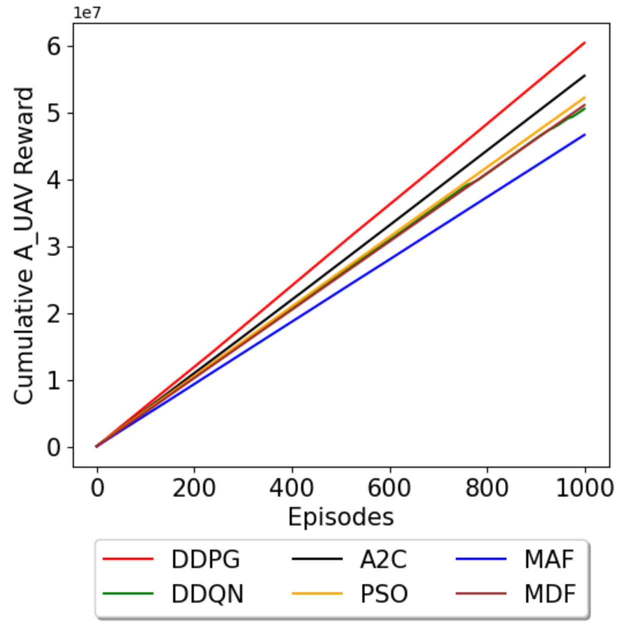

Figure 4.10: Reward of A_UAV

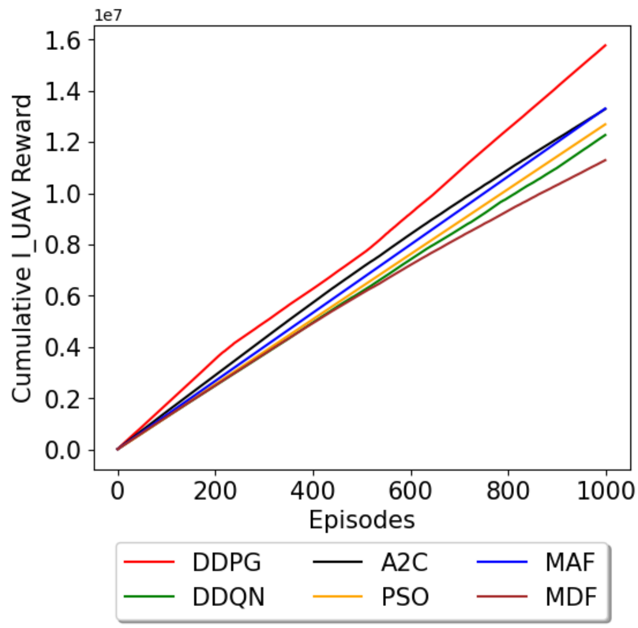

Figure 4.11: Reward of I_UAV

Double Deep Q-Network (DDQN) to optimize the objectives such as energy efficiency and the Age of Information (AoI). The I_UAVs operate in a decentralized manner, focusing on reducing access latency and energy consumption while prioritizing the coverage of high-crowd-density areas. The effectiveness of the proposed approach is demonstrated by comparing its performance against the traditional heuristic and metaheuristic approaches. The results show that DDPG and A2C algorithms perform better on various metrics as compared to the baselines.

5| Conclusions and Avenues for Future Research

5.1 Conclusion

In conclusion, the comprehensive research conducted in this thesis presents a detailed study of multiple unmanned aerial vehicle (UAV) solutions for various applications such as surveillance, target tracking, large area monitoring and others. The research highlights the applicability of deploying hierarchical multi-UAV solutions cover larger aspects of the problem particularly in enhancing the coverage and providing a comprehensive view of the targeted area However, multi-UAV-based applications bring forth the challenge of multi-UAV coordination. In other words, the complexity of coordinating them to work together effectively in achieving the system objective pose a challenge. This interaction among multiple UAVs could be based on optimizing multiple objectives in the system subject to different system constraints.

In our work, we have explored the multi-UAV coordination for achieving system objectives subject to different constraints in an infrastructure-deficient environment without a centralized controller.

In our first study, we addressed the optimization of multiple objectives such as energy, access latency, and queue backlog of the system. Our work introduced a heterogeneous multi-UAV framework that employs distance and latency-aware trajectory optimization for efficient data collection and offloading. The application of the Lyapunov optimization approach ensured system stability, adeptly managing system queue backlogs. The proposed methods have demonstrated superior performance in reducing access latency compared to other baseline strategies. Moreover, the analysis of the system trade off parameter V has highlighted a balance between queue stability and system utility. Further a detailed examination of the energy consumption has been provided for different UAV models. In this work, there is no central entity which communictaes A_UAV about the current locations of I_UAVs so we have devised a strategy to estimate the set of candidate locations of I_UAVs. Hence, we have devised a strategy to estimate the set of candidates loactions of I_UAVs. We have thus designed a baseline based on heuristics, such as greedy and Round-Robin scheduling. The total operational time of A_UAV has increased for both proposed approaches (HDLAT and DLAT) that signifies optimal use of battery. Similarly, the average access latency of A_UAV is reduced substantially for HDLAT, compared to greedy approach (DAT). The number of PoIs covered for HDLAT is 3.5 times of Round Robin and for DLAT the number of locations increased by 3 times of Round Robin baseline.

In our second work, we have established a decentralized multi-UAV framework based on Deep Reinforcement Learning principles, utilizing Deep Deterministic Policy Gradients

(DDPG) and Advantage Actor-Critic (A2C) methods. This framework excels in minimizing access latency and the AoI while optimizing system energy. In this work, we have designed the trajectory scheduling of both I_UAVs and A_UAV independently. The MDP formulation is such that the system doesn't need a centralized controller. Our simulation results validate that our proposed approach outperforms various baseline strategies, especially in terms of minimizing latency and AoI. From the results, it could be deduced the DDPG and A2C are performing better than the baselines in both training and testing environments.

5.2 Future work

Looking ahead, the potential to enhance UAV trajectory optimization in a multi-UAV network, particularly in an infrastructure-sufficient environment, is substantial. The primary areas for potential improvements include:

- Collaborative Trajectory Scheduling: Focusing on Multi Agent Deep Deterministic Policy Gradients (MADDPG) shows promise for jointly optimizing the trajectory scheduling of different UAV groups. This approach enhances the coordination and efficiency of their flight paths [55].

- Simplifying Complex Goals with Hierarchical RL: Hierarchical RL can be used to break down the complex goals of the UAV system into smaller, more manageable tasks. By focusing on these sub-goals in an ordered, step-by-step manner, the system can handle objectives more efficiently[56].

- Integrating Collision Avoidance Mechanisms: Adding another constraint to avoid obstacles, ensuring safer operations in crowded or complex environments. This involves creating algorithms that help UAVs detect and navigate around obstacles safely.

- Reward Function Refinement: In RL-based solutions, the reward function is key to achieving system objectives. Improving this function might include incorporating specific knowledge relevant to the domain or application, or using dynamic reward strategies to adapt to changing conditions[57].

- Adding more number of A_UAV: Introducing more A_UAVs can provide insights into energy optimization and the collective impact of multiple UAVs on the overall system. As the number of A_UAVs grows, coordinating their actions becomes increasingly important.

- Generating Real World Dataset: Creating and utilizing real-world data to evaluate the performance of these algorithms could confirm the effectiveness of the proposed strategies. The real-world data can offer essential insights into the actual operation of UAVs in real-life situations, helping us understand how they perform outside of simulated environments. Performance evaluation on real-world datasets not only enhances the reliability of multi-UAV system but also ensures that the UAVs are prepared for the complexities and nuances of real-world applications.

References

[1] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, Young-Han Nam, and Mérouane Debbah. A tutorial on UAVs for wireless networks: Applications, challenges, and open problems. *IEEE Communications Surveys & Tutorials*, 21(3):2334–2360, 2019.

[2] Fabio Ruggiero, Vincenzo Lippiello, and Anibal Ollero. Aerial manipulation: A literature review. *IEEE Robotics and Automation Letters*, 3(3):1957–1964, 2018.

[3] Asad Waqar Malik, Anis Ur Rahman, Tariq Qayyum, and Sri Devi Ravana. Leveraging fog computing for sustainable smart farming using distributed simulation. *IEEE Internet of Things Journal*, 7(4):3300–3309, 2020.

[4] Ebtehal Turki Alotaibi, Shahad Saleh Alqefari, and Anis Koubaa. Lsar: Multi-UAV collaboration for search and rescue missions. *IEEE Access*, 7:55817–55832, 2019.

[5] Hesam Hamledari, Shakiba Davari, Seyed Omid Sajedi, Pouya Zangeneh, Brenda McCabe, and Martin Fischer. UAV mission planning using swarm intelligence and 4d bims in support of vision-based construction progress monitoring and as-built modeling. In *Construction Research Congress 2018*, pages 43–53, 2018.

[6] Yujing Fu, Haibo Mei, Kezhi Wang, and Kun Yang. Joint optimization of 3d trajectory and scheduling for solar-powered uav systems. *IEEE Transactions on Vehicular Technology*, 70(4):3972–3977, 2021.

[7] Hazim Shakhatreh, Ahmad H Sawalmeh, Ala Al-Fuqaha, Zuochao Dou, Eyad Almaita, Issa Khalil, Noor Shamsiah Othman, Abdallah Khreishah, and Mohsen Guizani. Unmanned Aerial Vehicles (UAVs): A Survey on Civil applications and key research challenges. *IEEE Access*, 7:48572–48634, 2019.

[8] Utkarsh Singh, Jean-François Determe, François Horlin, and Philippe De Doncker. Crowd Monitoring: State-of-the-art and Future Directions. *IETE Technical Review*, 38(6):578–594, 2021.

[9] Rodrigo Saar De Moraes and Edison Pignaton De Freitas. Multi-UAV based Crowd Monitoring System. *IEEE Transactions on Aerospace and Electronic Systems*, 56(2): 1332–1345, 2019.

[10] Qamar Abbas, Syed Ali Hassan, Hassaan Khaliq Qureshi, Kapal Dev, and Haejoon Jung. Comprehensive Survey on Age of Information in Massive IoT Networks. *Computer Communications*, 197:199–213, 2023.

[11] Hao-nan Wang, Ning Liu, Yi-yun Zhang, Da-wei Feng, Feng Huang, Dong-sheng Li, and Yi-ming Zhang. Deep Reinforcement Learning: A Survey. *Frontiers of Information Technology & Electronic Engineering*, 21(12):1726–1744, 2020.

[12] Richard S. Sutton and Andrew Barto. *Reinforcement learning: an introduction*. The MIT Press, 2018.

[13] Shuo Wan, Jiaxun Lu, Pingyi Fan, and Khaled B Letaief. Towards Big Data Processing in IoT: Path Planning and Resource Management of UAV Base Stations in Mobile-Edge Computing System. *IEEE Internet of Things Journal*, 2019.

[14] Jiao Zhang, Li Zhou, Qi Tang, Edith C-H Ngai, Xiping Hu, Haitao Zhao, and Jibo Wei. Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing. *IEEE Internet of Things Journal*, 6(2):3688–3699, 2018.

[15] Saeed H Alsamhi, Ou Ma, M Samar Ansari, and Sachin Kumar Gupta. Collaboration of Drone and Internet of Public Safety Things in Smart Cities: An Overview of QoS and Network Performance Optimization. *Drones*, 3(1):13, 2019.

[16] Michael Neely. *Stochastic network optimization with application to communication and queueing systems*. Morgan & Claypool Publishers, 2010.

[17] Shuo Wan, Jiaxun Lu, Pingyi Fan, and Khaled B Letaief. Toward Big Data Processing in IoT: Path Planning and Resource Management of UAV Base Stations in Mobile-Edge Computing System. *IEEE Internet of Things Journal*, 7(7):5995–6009, 2019.

[18] Yong Zeng and Rui Zhang. Energy-efficient uav communication with trajectory optimization. *IEEE Transactions on Wireless Communications*, 16(6):3747–3760, 2017.

[19] Yinlu Wang, Ming Chen, Cunhua Pan, Kezhi Wang, and Yijin Pan. Joint optimization of uav trajectory and sensor uploading powers for uav-assisted data collection in wireless sensor networks. *IEEE Internet of Things Journal*, 2021.

[20] Muhammad Yeasir Arafat and Sangman Moh. Bio-inspired approaches for energy-efficient localization and clustering in UAV networks for monitoring wildfires in remote areas. *IEEE Access*, 9:18649–18669, 2021.

[21] Khashayar Asadi Boroujeni and Kevin Han. Perspective-based image-to-bim alignment for automated visual data collection and construction performance monitoring. In *Computing in Civil Engineering 2017*, pages 171–178. 2017.

[22] Pavel Mach and Zdenek Becvar. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials*, 19(3): 1628–1656, 2017.

[23] Bin Dai, Jianwei Niu, Tao Ren, Zheyuan Hu, and Mohammed Atiquzzaman. Towards energy-efficient scheduling of UAV and base station hybrid enabled mobile edge computing. *IEEE Transactions on Vehicular Technology*, 71(1):915–930, 2022. doi: 10.1109/TVT.2021.3129214.

[24] Seongah Jeong, Osvaldo Simeone, and Joonhyuk Kang. Mobile edge computing via a uav-mounted cloudlet: Optimization of bit allocation and path planning. *IEEE Transactions on Vehicular Technology*, 67(3):2049–2063, 2018. doi: 10.1109/TVT.2017.2706308.

[25] Yu Xu, Tiankui Zhang, Jonathan Loo, Dingcheng Yang, and Lin Xiao. Completion time minimization for UAV-assisted mobile-edge computing systems. *IEEE Transactions on Vehicular Technology*, 70(11):12253–12259, 2021. doi: 10.1109/TVT.2021.3112853.

[26] Qihui Wu, Jiaxin Chen, Yuhua Xu, Nan Qi, Tao Fang, Youming Sun, and Luliang Jia. Joint Computation Offloading, Role and Location Selection in Hierarchical Multi-coalition UAV MEC Networks: A Stackelberg Game Learning Approach. *IEEE Internet of Things Journal*, 2022.

[27] Xiaoyan Hu, Kai-Kit Wong, Kun Yang, and Zhongbin Zheng. UAV-assisted relaying and Edge computing: Scheduling and Trajectory optimization. *IEEE Transactions on Wireless Communications*, 18(10):4738–4752, 2019.

[28] Xianbang Diao, Meng Wang, Jianchao Zheng, and Yueming Cai. Fairness-Aware Offloading and Trajectory Optimization for Multi-UAV Enabled Multi-Access Edge Computing. *IEEE Access*, 8:124359–124370, 2020.

[29] Qiyu Hu, Yunlong Cai, Guanding Yu, Zhijin Qin, Minjian Zhao, and Geoffrey Ye Li. Joint offloading and Trajectory Design for UAV-enabled Mobile Edge Computing Systems. *IEEE Internet of Things Journal*, 6(2):1879–1892, 2018.

[30] Junghwa Kang, Kyeongrok Kim, Howon Lee, and Jae-Hyun Kim. Lyapunov optimization-based online positioning in uav-assisted emergency communications. *IEEE Access*, 2023.

[31] Chao Sun, Wei Ni, and Xin Wang. Joint Computation Offloading and Trajectory Planning for UAV-Assisted Edge Computing. *IEEE Transactions on Wireless Communications*, 20(8):5343–5358, 2021.

[32] Cheng Zhan, Han Hu, Xiufeng Sui, Zhi Liu, and Dusit Niyato. Completion time and energy optimization in the UAV-enabled mobile-edge computing system. *IEEE Internet of Things Journal*, 7(8):7808–7822, 2020.

[33] Shuhang Zhang, Hongliang Zhang, Boya Di, and Lingyang Song. Cellular cooperative Unmanned Aerial Vehicle networks with sense-and-send protocol. *IEEE Internet of Things Journal*, 6(2):1754–1767, 2018.

[34] Jingzhi Hu, Hongliang Zhang, and Lingyang Song. Reinforcement Learning for Decentralized Trajectory design in cellular UAV networks with sense-and-send protocol. *IEEE Internet of Things Journal*, 6(4):6177–6189, 2018.

[35] Mengjie Yi, Xijun Wang, Juan Liu, Yan Zhang, and Bo Bai. Deep Reinforcement Learning for fresh Data Collection in UAV-assisted IoT Networks. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 716–721. IEEE, 2020.

[36] Wen Fan, Ke Luo, Shuai Yu, Zhi Zhou, and Xu Chen. AoI-driven Fresh Situation Awareness by UAV Swarm: Collaborative DRL-based Energy-Efficient Trajectory Control and Data Processing. In *2020 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 841–846. IEEE, 2020.

[37] Zhaoyang Han, Yaoqi Yang, Weizheng Wang, Lu Zhou, Tu N Nguyen, and Chunhua Su. Age Efficient Optimization in UAV-Aided VEC Network: A Game Theory Viewpoint. *IEEE Transactions on Intelligent Transportation Systems*, 23(12): 25287–25296, 2022.

[38] Roy D. Yates, Yin Sun, D. Richard Brown, Sanjit K. Kaul, Eytan Modiano, and Sennur Ulukus. Age of Information: An Introduction and Survey. *IEEE Journal on Selected Areas in Communications*, 39(5):1183–1210, 2021. doi: 10.1109/JSAC.2021.3065072.

[39] Wuhui Chen, Xiaoyu Qiu, Ting Cai, Hong-Ning Dai, Zibin Zheng, and Yan Zhang. Deep Reinforcement Learning for Internet of Things: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, 23(3):1659–1692, 2021.

[40] Biplav Choudhury, Prasenjit Karmakar, Vijay K Shah, and Jeffrey H Reed. Learning-based Age of Information Minimization in UAV-relayed IoT Networks. *arXiv preprint arXiv:2203.04227*, 2022.

[41] Angelo Trotta, Ufuk Muncuk, Marco Di Felice, and Kaushik R. Chowdhury. Persistent Crowd Tracking Using Unmanned Aerial Vehicle Swarms: A Novel Framework for Energy and Mobility Management. *IEEE Vehicular Technology Magazine*, 15(2): 96–103, 2020. doi: 10.1109/MVT.2020.2982244.

[42] Biplav Choudhury, Vijay K. Shah, Aidin Ferdowsi, Jeffrey H. Reed, and Y. Thomas Hou. AoI-Minimizing Scheduling in UAV-relayed IoT Networks. *CoRR*, abs/2107.05181, 2021. URL https://arxiv.org/abs/2107.05181.

[43] Omar Sami Oubbati, Mohammed Atiquzzaman, Abderrahmane Lakas, Abdullah Baz, Hosam Alhakami, and Wajdi Alhakami. Multi-UAV-enabled AoI-aware WPCN: A Multi-agent Reinforcement Learning Strategy. In *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6, 2021. doi: 10.1109/INFOCOMWKSHPS51825.2021.9484496.

[44] Shuo Wan, Jiaxun Lu, Pingyi Fan, and Khaled B. Letaief. Toward Big Data Processing in IoT: Path Planning and Resource Management of UAV Base Stations in

Mobile-Edge Computing System. *IEEE Internet of Things Journal*, 7(7):5995–6009, 2020. doi: 10.1109/JIOT.2019.2954825.

[45] Wenjing Xiao, Miao Li, Bander Alzahrani, Reem Alotaibi, Ahmed Barnawi, and Qingsong Ai. A Blockchain-Based Secure Crowd Monitoring System Using UAV Swarm. *IEEE Network*, 35(1):108–115, 2021. doi: 10.1109/MNET.011.2000210.

[46] Rodrigo Saar de Moraes and Edison Pignaton de Freitas. Multi-UAV Based Crowd Monitoring System. *IEEE Transactions on Aerospace and Electronic Systems*, 56(2): 1332–1345, 2020. doi: 10.1109/TAES.2019.2952420.

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

[48] Akram Al-Hourani, Sithamparanathan Kandeepan, and Simon Lardner. Optimal lap altitude for maximum coverage. *IEEE Wireless Communications Letters*, 3(6): 569–572, 2014.

[49] Naser Hossein Motlagh, Tarik Taleb, and Osama Arouk. Low-altitude Unmanned Aerial Vehicles-based Internet of Things services: Comprehensive Survey and Future perspectives. *IEEE Internet of Things Journal*, 3(6):899–922, 2016.

[50] Carmelo Di Franco and Giorgio Buttazzo. Energy-aware coverage path planning of UAVs. In *2015 IEEE International Conference on Autonomous Robot Systems and Competitions*, pages 111–117. IEEE, 2015.

[51] Seongah Jeong, Osvaldo Simeone, and Joonhyuk Kang. Mobile Edge Computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning. *IEEE Transactions on Vehicular Technology*, 67(3):2049–2063, 2017.

[52] Jiaxun Lu, Shuo Wan, Xuhong Chen, Zhengchuan Chen, Pingyi Fan, and Khaled Ben Letaief. Beyond empirical models: Pattern formation driven placement of uav base stations. *IEEE Transactions on Wireless Communications*, 17(6):3641–3655, 2018.

[53] Yuyi Mao, Jun Zhang, and Khaled B Letaief. A Lyapunov optimization approach for green cellular networks with hybrid energy supplies. *IEEE Journal on Selected Areas in Communications*, 33(12):2463–2477, 2015.

[54] Tareq M Shami, Ayman A El-Saleh, Mohammed Alswaitti, Qasem Al-Tashi, Mhd Amen Summakieh, and Seyedali Mirjalili. Particle Swarm Optimization: A Comprehensive Survey. *IEEE Access*, 10:10031–10061, 2022.

[55] Yukai Hou, Jin Zhao, Rongqing Zhang, Xiang Cheng, and Liuqing Yang. Uav swarm cooperative target search: A multi-agent reinforcement learning approach. *IEEE Transactions on Intelligent Vehicles*, 9(1):568–578, 2024. doi: 10.1109/TIV.2023.3316196.

[56] Hongyue Kang, Xiaolin Chang, Jelena Mišić, Vojislav B Mišić, Junchao Fan, and Yating Liu. Cooperative uav resource allocation and task offloading in hierarchical aerial computing systems: A mappo based approach. *IEEE Internet of Things Journal*, 2023.

[57] Armaan Garg and Shashi Shekhar Jha. Directed explorations during flood disasters using multi-uav system. In *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, pages 2154–2161, 2022.