POSITIONING FLYING WIRELESS BASE STATIONS FOR OPTIMAL COMMUNICATION COVERAGE

LIM YEN KHAI

A project report submitted in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

October 2023

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature	:	
Name	:	LIM YEN KHAI
ID No.	:	2005211
Date	:	3/10/2023

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"POSITIONING FLYING WIRELESS BASE STATIONS FOR OPTIMAL COMMUNICATION COVERAGE"** was prepared by **LIM YEN KHAI** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature	:	
Supervisor	:	Dr. Lee Ying Loong
Date	:	4 October 2023

Signature	:	kckhor
Co-Supervisor	:	Dr. Khor Kok Chin
Date	:	4/10/2023

ACKNOWLEDGEMENT

I would like to express sincere gratitude to my research supervisor Dr. Lee Ying Loong and co-supervisor Dr. Khor Kok Chin for their constant support throughout the project. I am grateful for their knowledge, companionship, enthusiasm and openness to communication, which made this research project a very insightful and enjoyable experience.

In addition, I would like to extend my thanks to Dr. Lee Ying Loong and research senior Jiang Sheng Qi for their guidance, advice and patience in helping me with my research, as well as for assisting me with writing and submitting a research paper on this project to the Indonesian National Research and Innovation Agency's 10th International Conference on Computer, Control, Informatics and its Applications (IC3INA 2023), which has been accepted as of 21st September 2023. They have helped me greatly in optimising my research methodology and presenting the results in a clear and concise manner.

ABSTRACT

Unmanned aerial vehicle-mounted base stations (UAV-BS) have the potential to revolutionize fifth-generation (5G) networks and beyond. Their ability to traverse virtually any terrain allows them to be positioned in the air to provide coverage and connectivity to nearby users. However, determining the UAV-BS's optimal positions is challenging, as several factors must be considered, including quality of service (QoS), collision avoidance, and fair QoS provisioning for efficient usage of UAV-BS's limited transmit power. Thus, we propose a fairness-aware three-dimensional multi-UAV-BS placement scheme based on the artificial hummingbird algorithm (AHA). First, we formulate a joint UAV-BS and user association problem to maximize a proportional fairness utility function, subject to collision avoidance and user QoS requirements. Next, we develop a joint UAV-BS placement and user association scheme using the AHA and greedy algorithm. Results show that our proposed scheme significantly outperforms existing baseline schemes in blocking probability, Jain's fairness index and data rate.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

5G	Fifth-Generation
AHA	Artificial Hummingbird Algorithm
ATG	Air-to-Ground
BS	Base Station
CN	Core Network
DCA	Department of Civil Aviation
DE	Differential Evolution
DP	Dynamic Programming
DQN	Deep-Q Network
DRL	Deep Reinforcement Learning
eMBB	Enhanced mobile broadband connectivity
GA	Genetic Algorithm
GT	Ground Terminal
GSA	Gravitational Search Algorithm
ICT	Information and Communication Technology
IMT-2020	International Mobile Telecommuncations-2020
IRCD	Iterative Redundant Circle Deletion
IoT	Internet of Things
ITU-R	International Telecommunications Union-Radio Communication sector
LoS	Line of Sight
MWA	Maximum Weighted Area
mMTC	Massive Machine-Type Communications
NHPSO	Network-Based Heterogenous Particle Swarm Optimisation
PBSR	Placement-Based Sparse Recovery
PRDDQN	Double Deep-Q Network with Prioritised Replay
PSO	Particle Swarm Optimisation
QoS	Quality of Service
RAN	Radio Access Network
RL	Reinforcement Learning
SA	Simulated Annealing
SNR	Signal-to-Noise Ratio

SPA	Spiral Placement Algorithm
TLBO	Teaching Based Learning Optimisation
UAV	Unmanned Aerial Vehicle
UAV-BS	Unmanned Aerial Vehicle-Mounted Base Station
URLCC	Ultra-Reliable Critical Communication Services
RA	Random UAV-BS Placement Algorithm
PP	Partition-based UAV-BS Placement Algorithm

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

In recent years, unmanned aerial vehicles (UAV) have been gaining traction in military, agricultural and geological technology use, thanks to its ease of use, maneuverability and increasing accesibility. In the area of communications, specifically fifth generation (5G) communications, UAVs can be used to carry mobile base stations (BSs) to provide and improve network coverage in a given area, such as those with impacted connection due to natural disasters, high user density, and locations far from fixed network terminals. Relative to traditional terrestrial base station networks, UAV-BS networks possess greater flexibility in coverage area, being able to traverse virtually any terrain, cost effectiveness in providing temporary coverage, and the potential to provide better quality coverage in general, as UAV-BSs are able to fly at high altitudes, providing broader coverage with few deadzones.

To fully realise the potential of UAV-BS 5G networks, the three-dimensional (3D) positioning of UAV-BSs plays a vital role, as their positions determine the optimal coverage area and can be vastly affected by the wireless environment. The wireless environment can significantly impact the performance of 5G UAV-BS networks, and notable factors such as path loss, shadowing, and fading can result in reduced coverage area and network performance. Additionally, environmental noise and interference from other wireless devices in the same airspace can further deteriorate the signal output.

In the current literature, a variety of optimization algorithms have been applied to UAV positioning for 5G networks such as particle swarm optimisation (PSO), dynamic programming (DP), genetic algorithm (GA), and other novel ad hoc heuristics to factor in specific needs for the respective problem (Cicek et al., 2019). It is worth noting that the complex nature of UAV-BS positioning for 5G networks has led to the fact that optimal approaches such as brute forcing not being considered as their execution times are significantly longer than heuristic approaches. Nonetheless, current UAV-BS positioning techniques for 5G remain far from optimum. The objective of this project is to investigate and develop a more optimal positioning method for 5G UAV-BS networks.

In this project, a UAV positioning scheme based on a new bio-inspired meta-heuristic algorithm called the Artificial Hummingbird Algorithm (AHA) will be developed and investigated. The AHA-based UAV-BS positioning scheme possesses the potential to overcome challenges faced by existing solutions, as well as provide a more efficient way to dynamically determine suitable UAV-BS positions while addressing fair QoS provisioning and collision avoidance - constraints widely forsaken in the current literature.

1.2 Problem Statement

1.2.1 Limitations of Single UAV-BS Systems

The main study goal of the UAV positioning problem is to minimise transmit power while maximising network coverage (Shakhatreh et al., 2017). Despite multiple studies being carried out to achieve this, they mostly consider only single UAV-BS networks (Lim et al., 2021). Due to limited transmit power of a single UAV-BS, these networks face notable issues when dealt with vast coverage area, or uneven terrain that results in path loss, requiring the UAV-BS to expend excess transmit power to achieve its quality of service (QoS) requirement.

Utilizing a multi-UAV-BS system can create coverage that greatly outperforms a single UAV-BS network in many metrics including data rate, reliability and of course, coverage area. However, position optimization for multiple UAV-BSs is a complex problem, with much investigation required to develop a multi-UAV-BS positioning scheme.

1.2.2 Challenges of Multi-UAV-BS Systems

When deploying multiple UAVs into a shared airspace, the risk of collision among UAVs is unavoidable. This highlights the importance of collision avoidance in a multi-UAV-BS positioning scheme as UAV-BS collision will likely result in affected devices being damaged, disrupting the QoS provided by the 5G network coverage, potentially resulting in network deadzones or service interruptions. Another challenge of multi-UAV-BS systems is fair QoS provisioning among users. This is to ensure that each user achieves QoS satisfaction without jeopardising other users' QoS satisfaction.

Though current literature have begun to shift focus towards developing multi-UAV-BS positioning schemes over single UAV-BS schemes, collision avoidance and fair QoS provisioning are seldom accounted for, leaving a research gap that should be addressed to ensure development of a practical solution.

1.2.3 High Complexity Solutions

Reducing complexity is a critical step to develop efficient algorithms for solving the UAV base station positioning problem. Highly complex algorithms require substantial amounts of processing power and computation time, which are not practical for UAV-BSs with limited battery life and processing capability. Due to this widely known fact, the existing literature on this problem have highlighted the development of efficient low complexity algorithms, including simulated annealing (Lim et al., 2021), PSO (Li et al., 2018)and genetic algorithm (Chen et al., 2018). These algorithms have proved effective in obtaining satisfactory solutions with minimal algorithmic complexity.

However, the dynamic nature of users and their environment presents a never ending need to develop algorithms with further reduced complexities, to further optimise computating resources, processing time and UAV-BS battery life duration. Furthermore, it is also beneficial to develop an algorithm which can be easily implemented in 5G systems, as it will greatly reduce the difficulty of future optimisation efforts and increases application flexibility.

1.3 Aim and Objectives

To address the research gaps mentioned above and ensure fair QoS provisioning among users, a fairness-aware multi-UAV-BS placement scheme with downlink priority is proposed.

The project objectives are:

1. To formulate a joint multi-UAV-BS placement and user association problem to maximize a proportional-fairness utility function that encourages fair QoS provisioning, subject to collision avoidance and limited user capacity of UAV-BSs.

2. To propose a modified artificial hummingbird algorithm for UAV-BS placement that implements a user association approach to solve the joint UAV-BS placement and fair QoS provisioning problem.
3. To demonstrate that the proposed AHA-based UAV-BS placement algorithm outperforms baseline schemes in terms of Jain's fairness index, blocking probability, and data rate.

1.4 Proposed Solution

As mentioned previously, to solve the problem of UAV BS positioning for 5G network coverage, an algorithm based on the AHA will be developed to design a dynamic multi-UAV positioning scheme with collision avoidance. Figure 1.1 illustrates the application of the multi-UAV-BS positioning scheme.

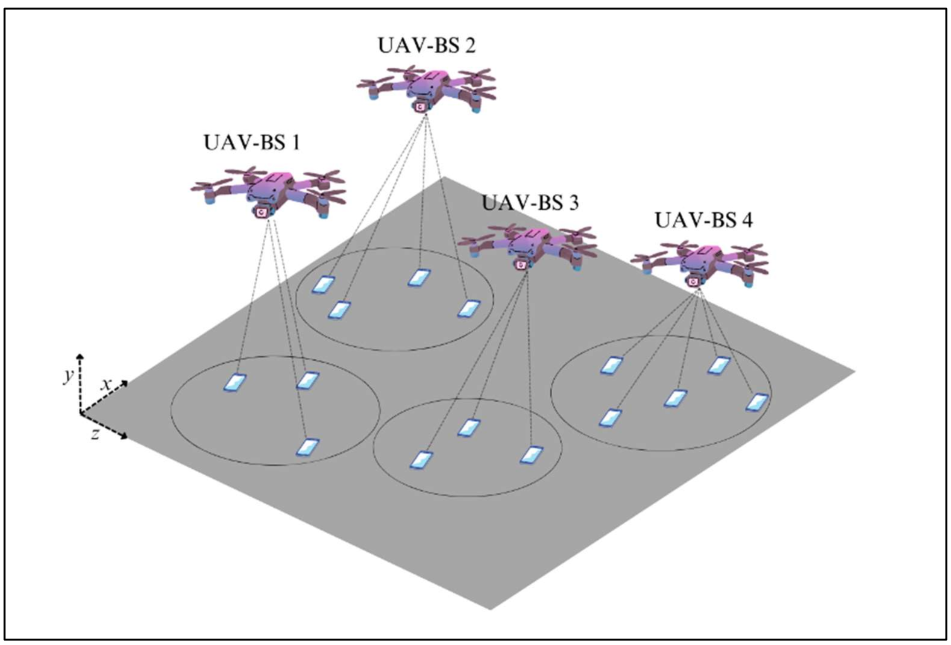


Figure 1.4.1: 5G network coverage using multiple UAV-BSs

The AHA is a bio-inspired swarm intelligence-based optimisation algorithm that has been shown to successfully solve various complex multidimensional optimization problems. This algorithm is developed based on the flight patterns and foraging behaviors of hummingbirds, which dictate the movement of the hummingbirds in finding and locating food sources. Tests have shown that the AHA has a high convergence rate and precise results across a spectrum of benchmark functions, while also outperforming other optimisation algorithms in the Friedman test, showing the best overall performance against algorithms such as the PSO, Differential Evolution (DE), Gravitational Search Algorithm (GSA) over a large set of benchmark functions including separable, nonseparable, unimodal and multimodal functions (Zhao, Wang and Mirjalili, 2022). The limited power source and processing capabilities of UAVs will also be addressed by the AHA's low algorithmic complexity and fast processing speed. Considering the AHA's flexibility and ease of implementation, it is chosen to solve the UAV-BS placement problem.

However, it is important to note that the original AHA does not address constraints. As the project considers collision avoidance and fair QoS provisioning, the AHA cannot be directly applied. Therefore, a penalty function approach will be leveraged in the design of the proposed scheme to address collision avoidance, while a greedy-based user association approach will be implemented to ensure fair QoS provisioning.

To test the effectiveness of the proposed solution, we will conduct simulations within a realistic system model to evaluate the algorithm's performance. The simulation will be performed using a simulation tool that can model the UAVs' movements and communication links. The solution will be evaluated based on the Jain's fairness index, loss rate, and data rate.

1.5 Scope and Limitations of Study

1.5.1 Scope

This project investigates and develops an algorithm for 3D positioning of UAV-BS to achieve optimal 5G network coverage. To outline the project scope, several points have been highlighted.

Firstly, the project investigates and develops an algorithm that dynamically optimises the positions of each UAV-BS in a multi-UAV-BS model to meet user QoS requirements without factoring in channel allocation for BSs. Secondly, the positions of UAV-BSs have to be coordinated in such a way that provides fair QoS provisioning while ensuring that the number of users that can be served by the UAV-BSs is maximized(Mozaffari et al., 2018). The scheme also only optimises downlink connection from UAV-BSs to ground users as the project focuses on using UAV-BSs to provide 5G network coverage to users, data transmitted in the uplink direction is typically relatively insignificant. Lastly, the algorithm will be designed and developed assuming that all ground users exist on the same altitude, while UAV-BSs are positioned three-dimensionally in the above airspace.

1.5.2 Limitation of Study

The implementation of the fairness-aware multi-UAV-BS placement scheme will require multiple simultaneously flying UAV-BSs and a large number of user devices. To obtain the UAVs and BSs will incur substantial costs, and another challenge is that each UAV is required to have a permit issued by the Department of Civil Aviation Malaysia (DCA) before flying can commence. A large airspace is also required to thoroughly test the algorithm, and depending on the location of the airspace, may require further permits before commencing. Furthermore, obtaining several hundred user devices is also unfeasible due to its high cost. Therefore, testing and implementation of the scheme will be performed in a simulation environment.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

As mentioned in the previous chapter, the application of UAV-BSs for 5G network coverage is the most researched topic in its field. Researchers have tested and proposed many solutions over the years, with varying development methods and effectiveness. Hence, the following literature review provides a brief introduction to 5G networks, an overview of some popular UAV-BS positioning techniques, and a high level breakdown of the AHA's mechanism and characteristics.

2.2 5G Networks and Beyond

Fifth-generation (5G) mobile communication networks are a comprehensive system that facilitates a fully mobile and interconnected society. The conception of 5G networks is a result of technological and architectural enhancements to 4G networks to meet new requirements, resulting in a multi-component network architecture which consists of a radio access network (RAN), core network (CN), and cloud infrastructure. 5G networks provide a comprehensive range of improvements over previous generation communication networks, such as lower latency, higher throughput and internet speed, higher mobility range, increased reliability, and greater connectivity density. There are three main usage scenarios that 5G networks serve: Enhanced mobile broadband connectivity (eMBB), massive machine-type communications (mMTC), and ultra-reliable critical communication services (URLCC), which can be applied across many industries, such as healthcare, transportation, information and communication technology (ICT) and agriculture (Zhang, Wang and Zhou, 2019).

Historically, the growing demand for mobile broadband services has been the driving force behind the evolution of mobile communication technology. To meet the needs of newly emerging applications and performance demands of mobile device users, eMBB is one of the most important usage scenarios of 5G technology due to its ubiquitous applications in improving performance of mobile networks in both hotspots and wide-area coverage to surpass 4G technology (Dahlman, Parkvall and Sköld, 2018). eMBB's enable allocation of higher data rates and network capacity to increase internet speed in hotspots, while prioritising service mobility and seamlessness for wide-area coverage, with lower speed requirements. To cover use cases with strict latency, reliability, and availability requirements, 5G technology introduces URLCC (Dahlman, Parkvall and Sköld, 2018). Examples of URLCC use cases include vehicle-to-vehicle communication, wireless control of industrial equipment, remote medical surgery and tactile internet. These applications share a common trait of having important functions that can be detrimentally affected by latency and network downtime. Driving Internet of Things (IoT) technology, 5G mMTC aims to connect a large number of devices with low power and data rates such as remote sensors and actuators to achieve high connection density, with a focus on maximising the number of interconnected devices (Bockelmann et al., 2018).

According to the International Mobile Telecommuncations-2020 (IMT-2020) standards's foreseen use cases for next-generation radio technology, these three usage scenarios – eMBB, URLCC, mMTC provide a key grouping of widely applicable use cases that can be used to point out relevant performance requirements for 5G technology (Dahlman, Parkvall and Sköld, 2018). However, it is important to note that these scenarios do not cover all possibilities, and that 5G technology must be able to adapt to evolving use cases.

5G technology is poised to bring significant impacts on smart technology, enhancing speed, network reliability, latency and power efficiency (Ni et al., 2019). The latest standard for 5G networks is the IMT-2020 presented by the International Telecommunication Union-Radiocommunication Sector (ITU-R), which highlights eight key capabilities that technologies must possess to support 5G usage scenarios.

Table 2.2.1 presents the details on the relationship between the target performance values of the 8 key technological capabilities for 5G support and its corresponding usage scenarios.

Table 2.2.1: Key Technological Capabilities and Target Performances for 5G applications in IMT-2020

Parameter	**Priority Levels for Different Usage Scenarios**			**Target Performance**
	eMBB	**URLCC**	**mMTC**	
Peak data rate	High	Low	Low	Downlink: 20Gb/s Uplink: 10Gb/s
User-experienced data rate	High	Low	Low	Downlink: 100Mb/s Uplink: 50Mb/s
Spectrum efficiency	High	Low	Low	3 x IMT-Advanced
Mobility	High	High	Low	500km/h
Latency	Medium	High	Low	URLLC: 1ms eMBB: 4ms
Connection density	Medium	Low	High	1 million devices per km^2
Network energy efficiency	High	Low	Medium	100 x IMT-Advanced
Area traffic capacity	High	Low	Low	10Mb/s/m^2

In Table 2.2.1, IMT-Advanced is the performance standard for 4G technology.

Though the standardisation of 5G has been established, the IMT-2020 highlights the requirements to further guide the development of 5G network technology. However, as more advanced applications begin to demand even higher data rates and coverage areas than before, exploration beyond-5G networks is inevitable. Among the most researched technologies in this field are UAV-BSs, which present a significant challenge in terms of positioning.

Therefore, the next section will provide an introduction to the problem and a review of current solutions that have been used to address this issue.

2.3 UAV-BS Positioning

Due to the mobility and ease of use of UAV-BSs, they hold large potential for providing mobile communication services in areas without infrastructural coverage, beyond the capabilities of current renditions of 5G technology. Some examples of this are disaster sites and other similar areas with hazardous terrain, battlefields, intermittent hotspots and rural areas (Lyu et al., 2017). The potential that UAV-BS hold far surpasses any other model of mobile BS, as flying UAV-BS are virtually deployable from anywhere, over any terrain and along any physically feasible trajectory. The flexibility of UAV-BSs highlights the key challenge in this technology, which is the optimal positioning of UAV-BSs to provide adequate levels of communication services (Cicek et al., 2019).

This section will provide a comprehensive analysis of the similar work related to UAV-BS positioning, highlighting the different approaches that have been proposed and identifying the similarities, differences and limitations of these approaches. A critical review of existing research will provide a clear understanding of the state-of-the-art in UAV-BS positioning and facilitate identification of research gaps that remain to be addressed in future research.

2.3.1 Single UAV-BS Systems

In a paper by Li et al., (2018), a UAV-BS positioning strategy that focuses on energy efficiency is proposed. The scheme focuses on the UAV-BS's antenna beam angle and flight altitude as parameters to maximise its coverage capability, as it is stated that there is a fine balance between the antenna beam angle and flight altitudes to ensure maximum area coverage with minimal path loss. Energy efficiency is a key consideration in this due to UAV-BS's limited transmit power and power supply, driving this paper's research objective of maximising the UAV-BS's effective coverage area, within a given transmit power. The research methodology involved formulating the problem as a 2D optimisation problem, followed by application of PSO to find the optimal flight altitude and antenna beam angle combinations. However, this paper only considers maximising the coverage of a single UAV-BS, and considers the horizontal position of the UAV-BS to be fixed.

While the research done by Li et al., (2018) focuses on maximising the coverage area of a single UAV-BS, Lai, Chen and Wang, (2019) proposed a unique approach that addresses the varying demands and density of users in a given area. Therefore, a demand-driven density-aware 3D UAV-BS positioning algorithm is developed to serve arbitrarily distributed users in a given area. In this paper, the UAV-BSs altitude is predefined to facilitate the paper's focus on its horizontal deployment position, while coverage radius is adjusted based on user density as larger coverage radius tends to amount cause a decrease in QoS. The UAV-BS positioning problem is then formulated as a knapsack-like problem, and solved with a novel solution based on the genetic algorithm (GA). The algorithm shows strong performance in meeting varying QoS requirements, however it does not truly make use of 3D positioning as the UAV-BS is at a fixed altitude. Additionally, although the usage of the variable coverage radius and horizontal positions help the UAV-BS to provide stronger QoS provisioning to areas of higher demand, it causes poor QoS experiences for other areas with sparse user distribution.

Another similar research publication worth highlighting is by Shakhatreh et al., (2017). Though some time has passed since the paper's publication, its unique problem statement and research objectives remain quite relevant. Unlike many other UAV-BS positioning schemes that focus on downward coverage for ground users, the paper proposes a solution that uses PSO to optimally position a UAV-BS to provide horizontally projected network coverage for uniformly distributed users in a multiple-floor high rise building. The problem is formulated to find the optimal 3D coordinates for a single UAV-BS that balances the relationship between UAV-BS's distance and incident angle to the building to meet user QoS and coverage requirements, while addressing the constraints of limited UAV-BS transmit power and 3D coordinate restrictions, though it does not explicitly address building collision avoidance.

The aforementioned research publications highlight some very important aspects of the UAV-BS positioning problem, primarily on the relationship between the mobility, limited coverage radius and limited transmit power of UAV-BSs. However, these papers address single UAV-BS systems which pose significant limitations due to its limited coverage radius and low transmit power, making it difficult to provide services that can fulfil QoS and area coverage requirements. As coverage areas and user densities increase, there is a need for more advanced techniques. A solution to that is to utilize multiple UAV-BSs to create network-based coverage over a wider area, which has the potential to offer significant improvements over single UAV-BS positioning techniques. Therefore, the next section will provide a review on research that has been conducted on multi-UAV-BS systems.

2.3.2 Multi-UAV-BS Systems

In the realm of multi-UAV-BS positioning schemes, one of the most highly cited research publications is by Lyu et al., (2017), where the goal is to develop a multi-UAV-BS positioning algorithm to minimise the number of UAV-BSs required to service a group of ground terminals (GT). Though the paper is old, it remains significant as many later publications used the solution developed here as a benchmark. The algorithm used by the researchers is called the spiral placement algorithm (SPA), which works by searching for the optimal UAV-BS positions in the pattern of an inward spiral, assigning positions to UAV-BSs successively. In this paper, the UAV-BSs altitudes and transmit power are assumed to be fixed, and collisions between UAV-BSs are not considered. It is also assumed that all GTs have the same QoS requirements.

Similar to Lyu et al., (2017) , Huang et al., (2020) proprosed a sparsed recovery-based algorithm called placement-based sparse recovery (PBSR) to solve the multi-UAV-BS placement problem. This algorithm is designed with the objective to create a placement scheme that provides consistent performance even when the number of UAV-BSs increase, while ensuring the minimum number of UAV-BSs for serving a number of GTs. In this paper, the positions of the GTs are known, and UAV-BSs are assumed to have the same transmit power and flying altitudes. The problem is formulated in a way that it ensures not more than one UAV-BS is appointed to a GT at any given time, to reduce the likelihood of signal interference and inefficient use of coverage bandwidth. To find the optimal UAV-BS positions, the problem is formulated as a sparse optimization problem which is solved with the reweighted 1-norm algorithm. From this, the resulting solutions are then adjusted using an iterative redundant circle deletion (IRCD) algorithm to ensure fair QoS provisioning, by assigning each GT to only one UAV-BS. However, the researchers do not account for true 3D UAV-BS positioning because UAV-BSs have fixed altitudes. Additionally, although collision avoidance is not explicitly mentioned, it could be argued that it is a biproduct of the IRCD algorithm.

A similarity between these papers are that the UAV-BS's network coverage is meant to account for GTs which are assumed to have transmit capabilities of their own. This implies that the necessary UAV-BS coverage range only needs to cover a small number of GTs. If it is applied to a scenario of providing coverage for large numbers of users distributed in the coverage area with UAV-BS coverage radii, it is likely to cause network deadzones, and if the coverage radii are decreased, more UAV-BSs will be required.

Contrary to the above two papers, a 3D multi-UAV-BS positioning algorithm aimed towards providing coverage for ground users is proposed by Alzenad, El-Keyi and Yanikomeroglu, (2018). In this paper, the researchers aim to solve the problem of meeting various user QoS requirements, a problem not addressed in the paper by Lyu et al., (2017). To develop a solution, the problem is formulated to maximise coverage area for as many users as possible with the UAV-BS's limited transmit power, while meeting unique QoS requirements, evaluated in terms of the signal-to-noise ratio (SNR) threshold. To simplify the problem, it is decoupled into two parts with one for vertical placement and the other for horizontal placement. A novel algorithm called the Maximum Weighted Area (MWA) algorithm is developed and applied to the first part of the problem to obtain the optimal altitude, followed by a traditional branch and cut method to obtain the optimal horizontal positions. The developed solution is tested in a simulation environment containing uniformly distributed stationary users that are partitioned into sets with differing QoS requirements. However, the research conducted does not consider the possibility of UAV-BS collisions and signal overlap from multiple UAV-BSs serving the same area, thus effectively neglecting fair QoS provisioning.

Du et al., (2020) proposes a unique network-based heterogenous PSO algorithm (NHPSO) for a 3D multi-UAV-BS positioning scheme. This algorithm is able to overcome the PSO's most significant caveat: Trapped solutions in local optima. This paper has a strong emphasis on the NHPSO, benchmarking it against many PSO variations to prove its superiority. Its improvements come from implementing structural heterogeneity through adaptive learning patterns between particles during the search process, allowing it to leave the local optima in search of potentially better solutions, while keeping the computational complexity to a minimum. In the system model, users are considered to be uniformly distributed in known positions on a horizontal plane. Similar to what is researched by Alzenad, El-Keyi and Yanikomeroglu, (2018), users in different area portions have different QoS requirements. The scheme involves partitioning users based on their QoS requirements, followed by application of NHPSO to position UAV-BSs to meet their respective requirements. The algorithm is found to be relatively easy to implement and shows potential for a wide spectrum of algorithms. However, this paper shows insufficient constraint handling, as it doesn't feature fair QoS provisioning, and also does not consider collision avoidance between UAV-BSs.

Lim et al., (2021) proposes a 3D multi-UAV-BS positioning scheme using the simulated annealing (SA) algorithm. The paper addresses the challenge for UAV-BSs and users to maintain ubiquitous connection as signal strength varies based on a users distance to the UAV-BS. It is also one of the few papers in this literature review that explicitly covers collision avoidance between UAV-BSs. In the system model, the researchers consider user groups with differing QoS requirements, on a horizontal plane, and fixed and equal transmission power among all UAV-BSs. In its problem formulation, collision avoidance is added into the formula as a penalty function, to ensure a minimum distance between the UAV-BSs. The problem is then modified from a minimisation problem into a throughput maximisation problem, before finally applying the SA algorithm. The SA algorithm is tested against a random placement-based static UAV-BS positioning scheme, where it is found to greatly outperform the random placement scheme. It is also found that the penalty function has minimum effects on the algorithm performance. However, fair QoS provisioning remains unaddressed, and there is a caveat in the testing phase of this research, as it was only tested against a single baseline, the random UAV-BS placement algorithm.

From the research discussed above, it can be observed that a large number of UAV-BS positioning schemes are based on heuristic algorithms. This is likely due to the application of heuristic algorithms being logical and rule based, making them relatively easy to use, even by personnel with minimal training. However, these papers are largely conceptual, though constraints such as collision avoidance and differing user QoS requirements have been considered, it is difficult to take into account environmental factors into the algorithm's development. To overcome this issue, there have been several research publications on taking a more modern machine learning approach to this issue.

Gopi and Magarini, (2021) proposes a reinforcement learning (RL) based multi-UAV-BS positioning system. In addition to satisfying user QoS requirements, the researchers emphasized UAV-BS collision avoidance and prevention of UAV-BSs from flying out of the search area. This solution is developed based on the Q-learning algorithm, a RL technique which functions by allowing RL agents to determine the UAV-BS's direction of flight and objective between exploration or exploitation, based on a reward function value. To reduce computation costs, the UAV-BSs are set to move in steps, across points in a square grid. The agents perform an episode of a fixed number of steps, in which the more episodes an agent performs, the higher its accuracy becomes. During the testing phase, the Q-learning based algorithm is trained for 2000 90 step episodes before being applied to get the optimum UAV-BS positions. The accuracy of the obtained positions are benchmarked against positions obtained through brute forcing techniques, where the Q-learning based algorithm is able to achieve 60.2% of the brute force technique's accuracy. The algorithm is also especially effective in scenarios where user location data is imperfect.

Qiu, Lyu and Fu, (2020) also propose a UAV-BS placement optimization scheme that uses deep reinforcement learning (DRL) to assign UAV-BSs to their optimum positions. As stated earlier, one of the problem statements that this research addresses is the lack of realistically applicable UAV-BS placement schemes. Another issue that it aims to solve is that many other algorithms face significant performance issues when dealt with large numbers of UAV-BSs and large coverage area. This solution follows a two step design approach, where a double deep-Q network (DQN) is enhanced with Prioritized Replay (PRDDQN) in each step. In this scheme, UAV-BSs have fixed flight altitudes and a maximum of one UAV-BS is assigned to each user. In the preliminary design step, PRDDQN is applied to find a set of optimum UAV-BS locations and their achieved coverage rates. In the advanced design phase, a 3D terrain map is used to obtain a coverage bitmap, which the PRDDQN can be applied to, to obtain optimal UAV-BS positions that are more realistic to the environment. The DRL agent was trained for a total of 2500 episodes before being used for testing. Similarly to several papers discussed earlier, collision avoidance is not explicitly addressed but it indirectly handled by the constraint of no more than 1 UAV-BS to a user.

These machine learning approaches show promising results towards realistically applicable UAV-BS positioning schemes, both considering multi-UAV-BS usage and putting great emphasis on basing the UAV-BS positions not just based on maximising data rate, but on the external environment as well. Due to the large number of training episodes applied to these machine learning techniques, they are more adaptable to changing data and environments, as opposed to highly rule-based heuristic algorithms. However, this highlights multiple issues with machine learning techniques. Most significantly, machine learning algorithms require large amounts of data, time and computational resources for sufficient training before they can be applied to a problem. This becomes an issue when in scenarios requiring immediate deployment of UAV-BSs, such as disaster sites and warzones, where limited training data, time and computing resources are available.

2.3.3 Overview of Related Work

The above sections have provided a high level overview of some notable previous works in the UAV-BS positioning field. Each paper has a set of constraints, assumptions, solutions and test results, which can be summarised to identify the most common research gaps and limitations in this field. Table 2.3.1 illustrates a summary of characteristics of the research papers discussed above.

Table 2.3.1: Overview of Related Works

Paper	Single/Multi UAV-BS (S/M)	Algorithm	QoS metric	True 3D UAV-BS positioning	Highlights	Limitations
The Energy-efficient UAV-based BS Coverage in Air-to-Ground Communications (Li et al., (2018)	S	PSO	SNR	No	Energy efficient system modelling	Does not address horizontal positioning of UAV-BS
On-Demand Density-Aware UAV Base Station 3D	S	GA-based (novel)	Data rate	No	User density Aware UAV-BS	Fixed UAV-BS altitude, single

Placement for Arbitrarily Distributed Users With Guaranteed Data Rates (Lai, Chen and Wang, 2019)					placement	UAV-BS system
Efficient 3D placement of a UAV using particle swarm optimization (Shakhatreh et al., 2017)	S	PSO	Path loss	Yes	Provide coverage for users in a high-rise building (vertical plane)	UAV-BS and building collision not addressed, single UAV-BS system
Placement Optimization of UAV-Mounted Mobile Base Stations (Lyu et al., 2017)	M	SPA (novel)	SNR	No	Algorithm to minimise number of UAV-BS required	Only considers providing coverage to GTs, no UAV-BS collision avoidance, fixed UAV-BS altitude

UAV-Mounted Mobile Base Station Placement via Sparse Recovery (Huang et al., 2020)	M	PBSR (novel)	-	No	Algorithm minimises number of UAV-BS neede to cover large number of GTs, while ensuring fair QoS provisioning	Only considers providing coverage to GTs, fixed UAV-BS altitude, no UAV-BS collision avoidance, focuses on coverage over QoS requirement
3-D Placement of an Unmanned Aerial Vehicle Base Station for Maximum Coverage of Users with Different QoS	M	MWA (novel)	SNR	Yes	Allows true 3D UAV-BS positioning, addresses varying user QoS requirements	No UAV-BS collision avoidance, does not address fair QoS provisioning

Requirements (Alzenad, El-Keyi and Yanikomeroglu, 2018)						
Network-Based Heterogeneous Particle Swarm Optimization and Its Application in UAV Communication Coverage (Du et al., 2020)	M	NHPSO (novel)	Data rate	Yes	Novel PSO variant where solution does not get trapped in local optima, addresses varying user QoS requirements	No UAV-BS collision avoidance, no fair QoS provisioning
Coverage Optimization for UAV Base Stations using Simulated Annealing (Lim et al., 2021)	M	SA (novel)	Data rate	Yes	Algorithm addresses UAV-BS collision avoidance	Insufficient testing, no fair QoS provisioning
Reinforcement Learning Aided UAV	M	Q-learnin	Data rate	Yes	Algorithm addresses	Large amounts of

Base Station Location Optimization for Rate Maximization (Gopi and Magarini, 2021)		g-based (novel)			UAV-BS collision avoidance and UAV-BS exceeding search area	training data and time needed
Placement Optimization of Aerial Base Stations with Deep Reinforcement Learning (Qiu, Lyu and Fu, 2020)	Multi	PRDDQN (novel)	SNR	No	Algorithm can handle large number of UAV-BSs and users, addresses LoS loss caused by environment	Fixed UAV-BS altitude, Large amounts of training data and time needed

As seen in Table 2.3.3.1, most of the papers propose novel algorithms to find the optimal UAV-BS positions. Half of the discussed papers address true 3D positioning of UAV-BS while the other half either assume fixed UAV-BS altitudes or horizontal positions. The most commonly used QoS metrics are data rate followed by SNR and path loss, the exception being Huang et al., (2020), which does not state a specific QoS metric. All except for one paper models the system with users on a horizontal plane, with Shakhatreh et al., (2017) modelling a scenario of providing coverage to users in a multi-level high-rise building. From this overview, it is found that the most commonly encountered limitation is collision avoidance and fair QoS provisioning. It is also worth noting that while the Artificial Hummingbird Algorithm (AHA) is proposed in this project as the solution to the UAV-BS positioning problem, it has yet to be researched for this specific application.

2.4 Summary

The UAV-BS positioning problem is highly researched and has strong potential to revolutionise network communications beyond 5G networks. However, the implications of utilising this technology are still uncertain, as research is often highly conceptual and are mostly conducted in simulation environments. Therefore, from the information gathered through this literature review, the current standards for 5G networks can be used as a reference for the development of a new AHA-based 3D multi-UAV-BS positioning scheme that aims to address some of the common limitations highlighted in similar works, while displaying comparable or improved performance.

CHAPTER 3
RESEARCH PLAN AND METHODOLOGY

3.1 Introduction

The methodology and work plan for this project will be presented in this chapter. As it is not possible to perform physical testing with UAV-BSs, the problem will be formulated mathematically for testing in a simulation environment.

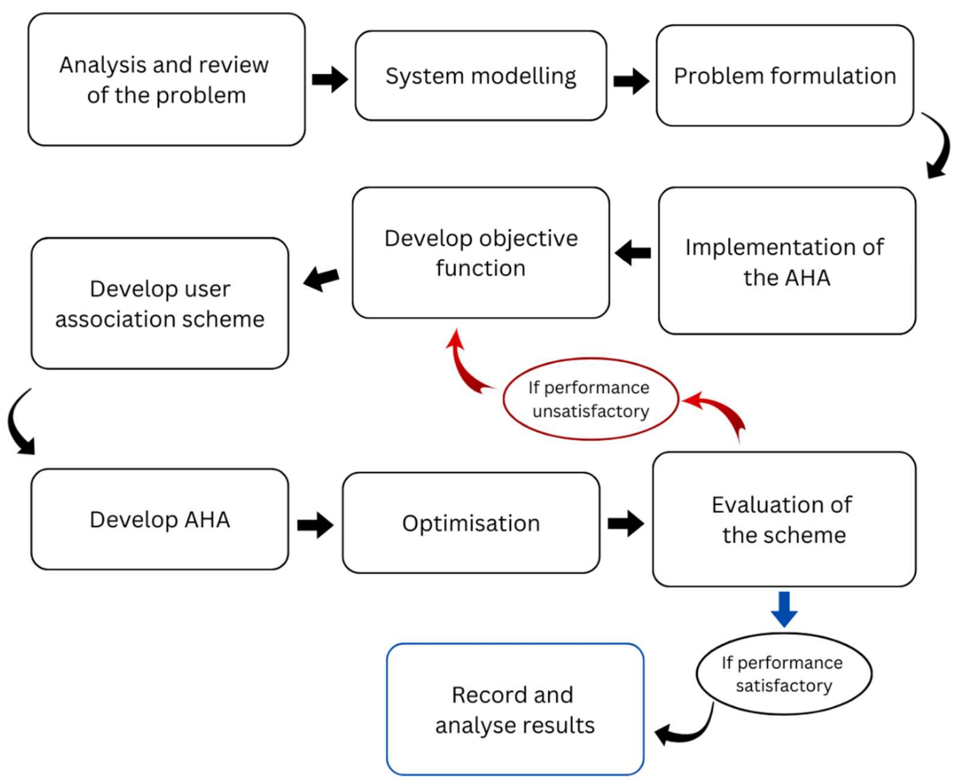


Figure 3.1.1: Fairness-aware UAV-BS placement scheme research summary

3.2 Analysis and Review of the Problem

The goal of this project is to develop a multi-UAV-BS positioning scheme based on the AHA. This algorithm is expected to be able to support the simultaneous use of multiple UAV-BSs, to ensure that they are able to satisfy user QoS requirements. The QoS metric evaluated in this project will be the downlink data rate received by the end user's devices, as well as Jain's fairness index and blocking probability. In this project, multiple UAV-BSs will be occupying the same air space, therefore collision avoidance among UAV-BSs should be addressed. A limitation of not more than one UAV-BS to each user will be implemented to ensure fair QoS provisioning, which can prevent QoS oversatisfaction, allowing the UAV-BS's transmit power to be utilised more efficiently.

3.3 System Modelling

The UAV-BS positioning algorithm is developed by modelling the system, followed by mathematical formulation of problems into equations. The considers a multi-UAV-BS coverage scenario, and the system is first modelled with an air-to-ground (ATG) communication model as shown in figure 1.4.1, where the network performance is highly dependent on the line-of-sight (LoS) path between the UAV-BSs and the users. This system model is commonly used for this problem as seen in research by Li et al., (2018), Lai, Chen and Wang, (2019) and Du et al., (2020). The initial modelling will be done in reference to the research by Du et al., (2020) due to its similarities to the proposed project.

The set of UAV-BSs is denoted as *J* and the set of users is denoted as *K*, therefore the total numbers of UAV-BSs and users can also be denoted respectively as *J* and *K*. For UAV-BS placement, the 3D Cartesian coordinates of UAV-BS $j \in J$ and user $i \in K$ are denoted as (x_j, y_j, z_j) and (x_i, y_i, z_i), respectively. As it is assumed that the users will always be positioned on the ground, the value of z_i is always zero. Therefore, the distance $d_{j,i}$ between each UAV-BS j and user i can be calculated as

$$d_{j,i} = \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2 + {z_j}^2} \tag{3-1}$$

Meanwhile, the probability of having a LoS path $P_{j,i}$ between UAV-BS j and user i is given by Al-Hourani, Kandeepan and Lardner, (2014) as

$$P_{j,i} = \frac{1}{1+ae^{-b\left(\theta_{j,i}-a\right)}} \tag{3-2}$$

where a and b are environmental constants, and $\theta_{j,i}$ is the elevation angle between UAV-BS j and user i. Next, the path loss of the LoS link $L_{j,i}^{LoS}$ is given as

$$L_{j,i}^{LoS} = 20\log\left(\frac{4\pi f d_{j,i}}{c}\right) + \eta_{LoS} \tag{3-3}$$

While the path loss of the non-LoS (NLoS) link $L_{j,i}^{NLoS}$ is given as

$$L_{j,i}^{NLoS} = 20\log\left(\frac{4\pi f d_{j,i}}{c}\right) + \eta_{NLoS} \tag{3-4}$$

In the equations (number, number) the η_{LoS} and η_{NLoS} are respectively the additional mean LoS and $NLoS$ path losses, while c represents the speed of light, and f represents the carrier frequency. Using these values, the final path loss $L_{j,i}$ between UAV-BS j and user i can be computed using the equation

$$L_{j,i} = P_{j,i}L_{j,i}^{LoS} + \left(1 - P_{j,i}\right)L_{j,i}^{NLoS} \tag{3-5}$$

The downlink SNR $\gamma_{j,i}$ received by user i from UAV-BS j can then be calculated using the equation

$$\gamma_{j,i} = \frac{P_t L_{j,i}}{P_n} \tag{3-6}$$

where P_t is the downlink transmission power of each UAV-BS, and P_n denotes additive Gaussian noise power. From this, the Shannon's capacity equation is implemented to calculate the received data rate $R_{j,i}$ (in b/s/Hz) of user i from UAV-BS j, given as

$$R_{j,i} = \log_2\left(1 + \gamma_{j,i}\right) \tag{3-7}$$

3.4 Problem Formulation

As this project involves multiple UAV-BSs providing coverage over a service area, the positions of the UAV-BSs are limited within the 3D space boundary over the service area as follows:

$$x_{min} \leq x_j \leq x_{max} \quad \forall j \in J, \tag{3-8}$$

$$y_{min} \leq y_j \leq y_{max} \quad \forall j \in J, \tag{3-9}$$

$$z_{min} \leq z_j \leq z_{max} \; \forall j \in J, \tag{3-10}$$

where x_{min}, y_{min} and z_{min} are the minimum boundaries, and x_{max}, y_{max} and z_{max} are the maximum boundaries of the 3D Cartesian space. To address collision avoidance between UAV-BSs, a minimum horizontal distance d^{safe} is maintained between the x-y dimensions of any two UAV-BSs (i.e.,$\forall j, j \in J, k \neq j$). It is given as

$$d_{j,k}{}^{uav} = \sqrt{\left(x_j - x_k\right)^2 + \left(y_j - y_k\right)^2} \tag{3-11}$$

$$d_{j,k}{}^{uav} > d^{\,safe} \tag{3-12}$$

The computation of d^{safe} omits vertical z-axis separation, as ensuring a safe horizontal distance between UAV-BSs is sufficient to ensure collision avoidance, as UAV-BSs will not collide regardless of their altitude. This allows the applied scheme to explore a greater variety of solutions while addressing the necessary constraint. The next constraint to be addressed is fair QoS provisioning. This is achieved through a user association system to assign each user with not more than one UAV-BS. The user association variable is defined as $c_{j,i}$

$$c_{j,i} \in \{0,1\} \quad \forall j \in J \,, \forall i \in K \tag{3-13}$$

where $c_{j,i} = 1$ if user i associates with UAV-BS j, otherwise $c_{j,i} = 0$. As each user is assigned only one UAV-BS at a time, an additional constraint is given as

$$\sum_{j \in J} c_{j,i} \leq 1 \quad \forall i \in K \tag{3-14}$$

It is considered that each user is required to achieve a target data rate of q_i for QoS provisioning for user i. Therefore, the user needs to associate with a UAV-BS that can satisfy the data rate requirement, and if there is no UAV-BS that meets the requirement, the user will be blocked from the network.

$$\sum_{j\in J} c_{j,i}R_{j,i} \geq \sum_{j\in J} c_{j,i}q_i \quad \forall i \in K \tag{3-15}$$

To represent fair QoS provisioning among all users, a QoS fulfilment ratio is defined as the ratio of achievable data rate of user i to its data rate requirement q_i.

$$w_i = \frac{\sum_{j\in J} c_{j,i}R_{j,i}}{q_i} \tag{3-16}$$

The above equation represents that w_i<1 implies failure to meet user QoS requirements, $w_i = 1$ implies that the QoS requirement of user i is met, and w_i >1 indicates that QoS requirements of user i is over-satisfied.

This study aims to position the UAV-BSs in such a way that the number of users being served with QoS fulfilments is maximised, while ensuring that the QoS fulfilment ratios among users are being satisfied in a fair manner. Therefore, an α-fairness approach is proposed, where setting the α value controls the fairness level of the scheme by reducing the QoS discrepancy among users as the α value increases (Lin et al., 2022). As such, the following α-fairness sum utility function is to be maximised

$$f_\alpha(w_i) = \begin{cases} \ln(w_i), & \alpha = 1 \\ w_i^{1-\alpha} \big/ (1-\alpha), & \alpha \neq 1, \alpha \geq 0 \end{cases}$$

$$\boldsymbol{P}: \max_{x,y,z,c} \sum_{i\in K} f_\alpha(w_i) \tag{3-17}$$

where it is subject to (3-8) – (3-10) and (3-12) – (3-15). Generally, maximising the α-function results in proportional fairness when α=1, which distributes more QoS to users who’s positions allow them to better utilise the resources, while at

$\alpha=\infty$, it results in max-min fairness, which aims to equally meet the minimum QoS requirements of every user (Lee et al., 2020). By maximising the sum logarithmic function of w_i, proportional fairness can be achieved in QoS provisioning among users, that is, a tradeoff between fairness and maximisation of w_i of each user can be achieved (Lee et al., 2018).

3.5 Implementation of AHA

Problem **P** is a mixed-integer programming (MIP) problem as $c_{j,i}$ is a discrete binary-valued variable, while x_j, y_j and z_j are continuous-valued variables. The non-convex nature of this problem makes it challenging to solve. Thus, any exhaustive approaches are excluded, as they can be computationally prohibitive.

3.5.1 Background

Recently, a new metaheuristic known as AHA has been proposed by Zhao, Wang and Mirjalili, (2022), which is a bio-inspired swarm intelligence-based optimisation algorithm that has been shown to successfully solve various complex multidimensional optimisation problems. The main reasons for choosing the AHA over other metaheuristic optimisation algorithms are summarised as follows:

1. The AHA is able to handle high-dimensional continuous search spaces, which is a requirement for this project.
2. The AHA has relatively few hyperparameters as it is highly probability based, and random number coefficient ranges are uniformly distributed between zero and one.
3. The AHA has yet to be applied to the UAV-BS coverage optimisation placement problem.

The AHA's performance in solving high-dimensional continuous search spaces is demonstrated through its performance in solving a Wilcoxon-signed rank test with a 5% significance level, significantly outperforming the PSO, Teaching-Learning-Based Optimisation (TLBO), DE, GSA and many more in terms of multimodal and composite functions (Zhao, Wang and Mirjalili, 2022). This algorithm is developed based on the flight patterns and foraging behaviors of hummingbirds, which dictate the movement of the hummingbirds in finding and locating food sources. Considering the AHA's flexibility and ease of implementation, it is adopted for solving the continuous variables of problem **P,** while a greedy user association approach will be adopted to solve the discrete variable. Though the AHA's source code does not contain constraint handling, research by Zhao, Wang and Mirjalili (2022) show that the algorithm is able to perform well under penalty functions. Therefore, the AHA will be used as a basis for developing a new solution that is able to implement the necessary UAV-BS positioning constraints of this project. The multi-UAV-BS positioning algorithm will be developed and tested using MATLAB. The main factor that makes MATLAB the programming language of choice is the wide range of built in tools and functions for mathematical and scientific computations, making it ideal for algorithm development and testing.

3.5.2 Developing the Scheme

In AHA, each hummingbird *n* represents a search agent finding a solution for a given problem. More precisely, the position of each hummingbird represents a candidate food source (i.e., a solution for the given problem). To solve problem P, the first food source of hummingbird n is designated as

$$X_n = \{x_1^n, y_1^n, z_1^n, x_2^n, y_2^n, z_2^n, \dots, x_{|J|}^n, y_{|J|}^n, z_{|J|}^n\} \tag{3-18}$$

where $\{x_j^n, y_j^n, z_j^n\}$ are the 3D coordinates of UAV-BS j found by hummingbird n. Then, the hummingbirds will be randomly placed throughout the search space (i.e., the solution space) defined by (3-8), (3-9) and (3-10) for the 3D coordinates. Next, a visit table of previously visited food sources is initialised as

$$VT_{n,m} = \begin{cases} 0 & if\ n \neq m \\ null & if\ n = m \end{cases} \quad n, m = 1,2,\cdots,N \tag{3-19}$$

where N is the number of hummingbirds. $VT_{n,m} = null$ indicates that a hummingbird is taking food at its own specific food source, while $VT_{n,m} = 0$ indicates that the food source found by hummingbird m has just been visited by hummingbird n.

The AHA functions similarly to other popular swarm-intelligence meta-heuristics as it is executed iteratively. In each iteration, the hummingbirds have equal probabilities of being assigned one of the following three flight patterns: Axial, diagonal, and omnidirectional flight, which are modelled mathematically as (3-20), (3-21) and (3-22), respectively:

$$D_n^{(v)} = \begin{cases} 1 \ \ if \ \ v = randi([1,d]) \\ 0 \qquad otherwise \end{cases} \quad v = 1, \dots, d \tag{3-20}$$

$$D_n^{(v)} = \begin{cases} 1 \ \ if \ \ v = P(j), \forall\, j \in [1,k] \\ 0 \qquad otherwise \end{cases} \quad v = 1, \dots, d \tag{3-21}$$

$$D_n^{(v)} = 1 \quad v = 1, \dots, d \tag{3-22}$$

where randi([1,d]) is a random number ranging from 1 to d with d being the maximum number of dimensions (i.e., the maximum number of variables and thus d = 3|J| is the number of elements in vector X_n) and $D_n^{(v)}$ represents the flight pattern chosen by hummingbird n in dimension v. Also, $P = randperm(k)$is a 1-by-k vector consisting of random permutation of integers from 1 to k where $k \in [2, \lceil r_1 \cdot (d-2) \rceil + 1]$, and r_1 is a normally distributed random value between 0 and 1.

After performing the selected flight pattern, the hummingbird is assigned either guided foraging or territorial foraging with a set probability.

Guided foraging involves the hummingbirds searching for food sources in the area of a previously visited food source. This is characterised by the hummingbird's ability to remember the locations of the food sources and prioritise those with higher nectar concentration. However, it should be noted that the nectar concentration is modelled as nectar refill rate to simplify the problem. Guided foraging is simulated by the algorithm's ability to maintain a visit table of candidate food sources and use that information to guide the search for new food sources with higher objective function values. The mathematical model for guided foraging is given as

$$V_n(t+1) = X_n^{tar}(t) + \alpha \cdot D_n \cdot \left(X_n(t) - X_n^{tar}(t)\right) \tag{3-23}$$

Algorithm 1: Guided foraging

If $f\left(V_n(t+1)\right) < f(X_n t)$
  $X_n(t+1) = V_n(t+1)$
  For *j*th food source from 1 to *N* ($j \neq tar, n$)
    $\text{VisitTable}(n, \text{j}) = \text{VisitTable}(n, j) + 1$
  End
  For *j*th food source from 1 to *N*
    $\text{VisitTable}(n, j) = \max_{l \in n \text{ and } l \neq j}\left(\text{VisitTable}(j, l)\right) + 1$
  End
Else
  For *j*th food source from 1 to *N* ($j \neq tar, n$)
    $\text{VisitTable}(n, \text{j}) = \text{VisitTable}(n, j) + 1$
  End
  $\text{VisitTable}(n, tar) = 0$
End

Figure 3.5.1: Pseudocode of guided foraging

where α denotes a random guided factor between 0 and 1, $D_n = [D_n^{(1)}, D_n^{(2)}, \ldots, D_n^{(d)}]$, $X_n^{(t)}$ denotes the current food source, and $X_n^{tar}(t)$ denotes the target food source.

On the other hand, territorial foraging involves each hummingbird exploring the search space around its current position. This facilitates thorough exploration and evaluation of the hummingbird's immediate surroundings, preventing premature convergence and ensures that multiple promising solutions are explored throughout the search space, promoting diversity and improving the likelihood of finding a global optimum. The mathematical model for territorial foraging is given as

$$V_n(t+1) = X_n(t) + \beta \cdot D_n \cdot X_n(t) \tag{3-24}$$

Algorithm 2: Territorial foraging

If $f\left(V_n(t+1)\right) < f(X_n t)$
    $X_n(t+1) = V_n(t+1)$
    For *j*th food source from 1 to N ($j \neq n$)
        $\text{VisitTable}(n, \text{j}) = \text{VisitTable}(n, j) + 1$
    End
    For *j*th food source from 1 to N
        $\text{VisitTable}(n, j) = \max\limits_{l \in n \text{ and } l \neq j}\left(\text{VisitTable}(j, l)\right) + 1$
    End
Else
    For *j*th food source from 1 to N ($j \neq n$)
        $\text{VisitTable}(n, \text{j}) = \text{VisitTable}(n, j) + 1$
    End
End

Figure 3.5.2: Pseudocode of territorial foraging

where β denotes a random territorial factor between 0 and 1, and. To prevent the solution from being trapped in local optima, the algorithm introduces migration foraging when it reaches a certain number of iterations, where the hummingbird at the worst food source is randomly repositioned within the solution space defined by (3-8), (3-9) and (3-10).

Migrational foraging is characterised by hummingbird's ability to explore new areas and adapt to environmental changes. It is simulated by introducing a condition where after a certain number of iterations, the hummingbirds at the current worst food sources will abandon their current locations and move to new locations randomly. This random repositioning allows the algorithm to discover new areas of the solution space that may contain better solutions and prevents the solution from being stuck in local optima. The mathematical model for migration foraging is given as

$$X_{wor}(t+1) = Low + r \cdot (Up - Low) \tag{3-25}$$

Algorithm 3: Migration foraging
1. **For** *j*th food source from 1 to *N* $(j \neq wor)$
2. $\text{VisitTable}(wor, \text{j}) = \text{VisitTable}(wor, j) + 1)$
3. **End**
4. **For** *j*th food source from 1 to *N*
5. $\text{VisitTable}(j, wor) = \max_{l \in n \text{ and } l \neq j} \big(\text{VisitTable}(j, l)\big) + 1$
6. **End**

Figure 3.5.3: Pseudocode of migration foraging

where *r* denotes a random migration factor between 0 and 1, $X_{wor}(t+1)$ denotes the current worst food source, *Up* and *Low* denote the upper and lower boundaries of the solution space as defined by (3-8), (3-9) and (3-10). After each hummingbird *n* completes its foraging process, the fitness function of the hummingbird is evaluated. Therefore, the fitness function is designed as

$$F = \sum_{i \in J} f_\alpha(w_i) - f_p, \tag{3-26}$$

where $f_p = \partial \sum_{j \in J} \sum_{k \in J \setminus \{j\}} \big(d_{j,k}^{safe} - d^{uav}\big)$ is the penalty function derived from constraint (3-12), with ∂ being the penalty coefficient. It is important to note that computation of the (fitness function) requires the solution of $c_{j,i}$, which is not present in the hummingbird food source. To obtain the solution of $c_{j,i}$ for each hummingbird, the 3D coordinates of each UAV-BS are extracted from the food source of the hummingbird to perform Algorithm 4.

Algorithm 4: User association and fitness function calculation
1. Initialize $c_{j,i} = 0$ for all $j \in J$ and $i \in K$
2. **For** $i \in K$
3. Obtain $c_{j,i}$ using (3-14).
4. **If** $\sum_{j \in J} c_{j,i} R_{j,i} < q_i$
5. Set $c_{j,i} = 0$.
6. **End if**
7. **End for**
8. Evaluate F_n using (3-26).

Figure 3.5.4: Pseudocode of greedy user association algorithm

Algorithm 4 determines $c_{j,i}$ based on a greedy user association strategy as follows:

$$c_{m,i} = \begin{cases} 1 & m = arg\,max_{j \in J}\, R_{j,i} \\ 0 & otherwise. \end{cases} \tag{3-27}$$

Equation (3-14) selects the UAV-BS m that provides the highest data rate for association with user i. After the selection, Algorithm 4 checks whether constraint (3-15) is satisfied. If the constraint is not satisfied, user i is blocked, as enforced in lines 4-6 of Algorithm 4. After the solution of $c_{j,i}$ is obtained, the fitness function of the hummingbird n, F_n is evaluated using (3-26).

Following the fitness function evaluation, the food source of the hummingbird n is updated by

$$X_n(t+1) = \begin{cases} X_n(t) & if\ \ F_n\big(X_n(t)\big) \geq F_n\big(V_n(t+1)\big) \\ V_n(t+1) & otherwise. \end{cases} \tag{3-28}$$

After that, the visit table of the hummingbird n is updated, depending on the chosen foraging behavior (Zhao, Wang and Mirjalili, 2022).

Algorithm 5: Proposed scheme

For $n = 1{:}N$
    Initialize X_n and $VT_{n,m}$ for all $m = 1, \ldots, N$.
End for
For: $t = 1{:}T_{\max}$
    For $n = 1{:}N$
        Create a random value *rand* between 0 and 1.
        If $rand<1/3$
            Perform *diagonal flight* in (3-21).
        Else if $rand>2/3$
            Perform *omnidirectional flight* in (3-22).
        Else
            Perform *axial flight* in (3-20).
        End if
        Create a random value *rand* between 0 and 1.
        If $rand < var$
            Perform *Guided Foraging* in (3-23).
            Perform Algorithm 1.
            Update $VT_{n,m}$ for all $m = 1, \ldots, N$ using Algorithm 1.
        Else
            Perform *Territorial Foraging* in (3-24).
            Perform Algorithm 1.
            Update $VT_{n,m}$ for all $m = 1, \ldots, N$ using Algorithm 2.
        End if
        If $\mathrm{mod}(t,2N) == 0$
            Perform *Migration Foraging* in (3-25).
            Update $VT_{n,m}$ for all $m = 1, \ldots, N$ using Algorithm 3.
        End if
    End for
    Set $m = \arg\max_{n=1,\ldots,N} F_n(X_n)$.
    Set $F_{\text{best}} = F_m(X_m)$ and $X_{\text{best}} = X_m$.
    Set $t = t + 1$.
End for

Figure 3.5.5: Pseudocode of the AHA-based UAV placement scheme

Algorithm 5 summarises the proposed AHA-based joint UAV-BS placement and user association scheme, where N is the number of hummingbirds and T_{max} is the maximum number of iterations. The computational complexity of Algorithm 5 can be approximated as of $O(NT_{max}|K||J|)$, which indicates that the time complexity will grow no faster than a polynomial function of $NT_{\max}|K||J|$.

Comparisons between the complexity of similar schemes are as follows:

1. PBSR (Huang et al., 2020) : $O(\frac{5}{4}K^6)$.
2. MWA (Alzenad, El-Keyi and Yanikomeroglu, 2018): $O(2^K \mathrm{K}^{3.5}\log(\varepsilon^{-1}))$, where ε is an algorithm parameter.
3. SPA (Lyu et al., 2017): $O(K^3)$
4. SA (Lim et al., 2021): $O(T_{max}|K|)$

Generally, exponential time complexities are less desirable due to their extreme increase in run time as the variable size increases. The most similar time complexity is achieved by the SA-based approach proposed by Lim et al., (2021), which is less complex than the proposed scheme. However, the AHA is a swarm intelligence based algorithm and the proposed scheme contains a user association algorithm that is nested within the AHA, causing more variables to be involved in the computation. The higher complexity of the proposed scheme is acceptable in this case, as it addresses an additional constraint - fair QoS provisioning.

3.5.3 Optimisation

Once the system model, objective function and all relevant algorithms have been set up in MATLAB, the scheme is tested against the objective function to obtain results for further optimisation. First, the UAV-BS settings and environmental parameters used in the scheme will be derived from existing literature by obtaining results from running the scheme and selecting the values that produced the most feasible results.

The scheme will then be evaluated on its ability to handle the collision avoidance constraint. It will be tested against the objective function as seen in equation (3-26), at penalty coefficients of zero and non-zero to evaluate the scheme in the presence and absence of a penalty function. The results will be compared to determine if further optimisation is necessary.

After verifying the constraint handling capabilities of the AHA have, testing will be conducted on adjusting the *var* parameter in figure 3.5.5 to modify probabilities of selecting between flight methods (3-23) and (3-24). The probabilities of selecting each flight method will be adjusted and tested between 0.1 to 0.9. The probabilities producing the best fitness value is selected to be used in the final AHA.

Further testing will be conducted to determine the best degree of fairness for the scheme, by testing the scheme with varying α fairness values. This will identify the fairness degree that constitutes the best trade off between proportional fairness and max-min fairness.

3.6 Evaluation Metrics

The developed scheme will be evaluated and compared against baseline schemes using the following performance metrics:

1. Blocking probability: The ratio of the number of users blocked from the network to the total number of users. In this scheme, users who's QoS requirements are unable to be satisfied are blocked from the network. This metric represents the percentage of users for whom the scheme did not meet their QoS requirements. It is defined as

$$\frac{|K|-\sum c_{j,i}}{|K|}. \tag{3-29}$$

2. Jain's fairness index: An indicator of fairness in QoS provisioning among users (Seid et al., 2021). A high Jain's fairness index percentage represents fair QoS provisioning. It is defined as

$$\frac{(\sum w_i)^2}{\sum w_i^2|K|}. \tag{3-30}$$

3. Sum of data rate: Sum of user received data rate, computed in bits/seconds/Hertz. It is defined as

$$\sum_{i \in K} \sum_{j \in J} c_{j,i} R_{j,i}. \tag{3-31}$$

Fitness value is not used as a main metric in the result analysis, as the ideal result of this scheme is a high Jain's fairness index, high sum of data rate and low blocking probability. A high fitness value generally represents a good result, however it is possible for a high fitness value to be achieved as a result of a heavy skew towards one of the three metrics.

3.7 Gantt Chart

This project's duration spans over two trimesters. The general work plan for the first and second trimesters are illustrated in Figure 3.7.1 and Figure 3.7.2 respectively, in work breakdown structures (WBS) and Gantt charts. Note that when referring to "predecessors" in the work breakdown structure, it includes tasks that need to be started before the current task, not only tasks that have already been completed.

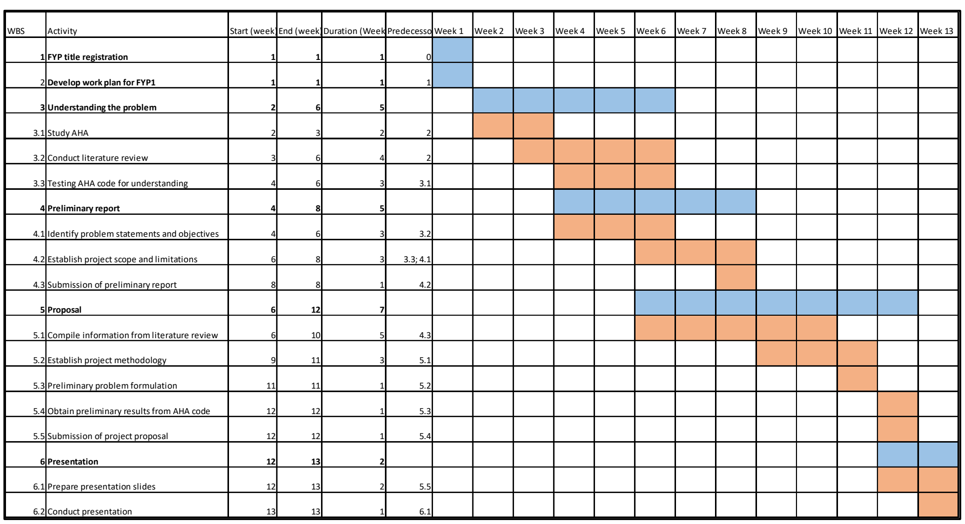


Figure 3.7.1: WBS and Gantt chart for FYP 1

In the first trimester, the goal of the project is to develop a clear, relevant and detailed proposal of the project goals. This time is allocated to developing a preliminary report, studying the relevant literature, structuring the project methodology and obtaining preliminary results to confirm the feasibility of the project.

WBS	Descriptions	Start (week)	End (week	Duration (week)	Predecess	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14
1	**Develop UAV-BS placement scheme**	**1**	**7**	**7**															
1.1	Formulate problem	1	5	5	0														
1.2	Coding	2	6	5	1.1														
1.3	Present code to supervisor	4	6	3	1.2														
1.4	Testing and optimisation	4	7	4	1.3														
1.5	Result collection and analysis	5	7	3	1.4														
1.6	Conference paper writing	5	7	3	1.4														
2	**Final Report**	**5**	**12**	8															
2.1	Report writing	5	12	8	1.4														
2.2	Report submission	12	13	1	2.1														
3	**FYP Poster**	**9**	**12**	**4**															
3.1	Make FYP poster	9	11	3	2.1														
3.2	Submit FYP poster	12	12	1	2.2; 3.1														
4	**Presentation**	2	13	12															
4.1	Preparation of presentation	2	13	12	1.1														
4.2	Conduct presentation	14	14	1	4.1														

Figure 3.7.2: WBS and Gantt chart for FYP 2

In the second trimester, the project's goal moves fully toward developing the project components that were proposed in the first trimester. This includes development of the multi-UAV-BS positioning scheme from the problem formulation to coding and testing. This time is allocated towards coding, completion of necessary deliverables and finalisation of work. During the second trimester, the contents of this project will be submitted as a research paper to the 2023 IC3INA online conference, organised by Indonesia's National Research and Innovation Agency (BRIN).

3.8 Summary

Overall, a number of similar works on the UAV-BS positioning problem provide a reference point for developing a research methodology, system modelling, problem formulation and benchmarking. Though there has been no research conducted on utilising the AHA for the UAV-BS positioning problem thus far, the existing literature on the AHA documents its strength in solving optimisation problems, demonstrating a number of its applications and the relative simplicity of its implementation, strongly suggesting that this project is feasible with achievable goals.

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Introduction

The section covers the results of testing the convergence of the general AHA, followed by the proposed fairness-aware AHA-based multi-UAV placement scheme. The results obtained provide proof of the proposed scheme's feasibility in solving the UAV placement problem with fair QoS provisioning. The scheme is fine-tuned through testing with varying stochastic parameters and alpha fairness values, while environmental constants were set based on existing literature. The proposed scheme and baseline schemes are evaluated and compared at varying numbers of users, in terms of the performance metrics stated in Section 3.6: blocking probability (3-29), Jain's fairness index (3-30), and sum of data rate (3-31).

4.2 General Convergence Performance of the AHA

To demonstrate the general performance of the AHA, it will be tested with a multimodal benchmark function: the Rastrigin function. As real-world scenarios often involve problems with multiple solutions, multimodal functsions test the AHA's feasibility more effectively than unimodal functions. The global minimum is given as

$$f(x) = 0; x(i) = 0, i = 1{:}n. \tag{4-1}$$

The Rastrigin function is a highly multimodal test function that produces a large number of uniformly distributed local minima (Pohlheim, 2005). The function definition is given as

$$f_6(x) = 10 \cdot n + \sum_{i=1}^{n} (x^2 - 10 \cdot \cos(2 \cdot \pi \cdot x_i)), \tag{4-2}$$

subject to solution space

$$-5.12 \le x_i \le 5.12. \tag{4-3}$$

The AHA is used to solve the Rastrigin function with the hummingbird population size set to 10. The results are averaged over 50 realisations, and each realisation consisted of 500 iterations and a dimensionality of 300 to evaluate the algorithm's performance in dealing with problems of high dimensionality. Dimensionality refers to the number of variables that the algorithm must optimise to find an optimum solution. The figures below illustrate the results obtained from testing the AHA with a population size of 10.

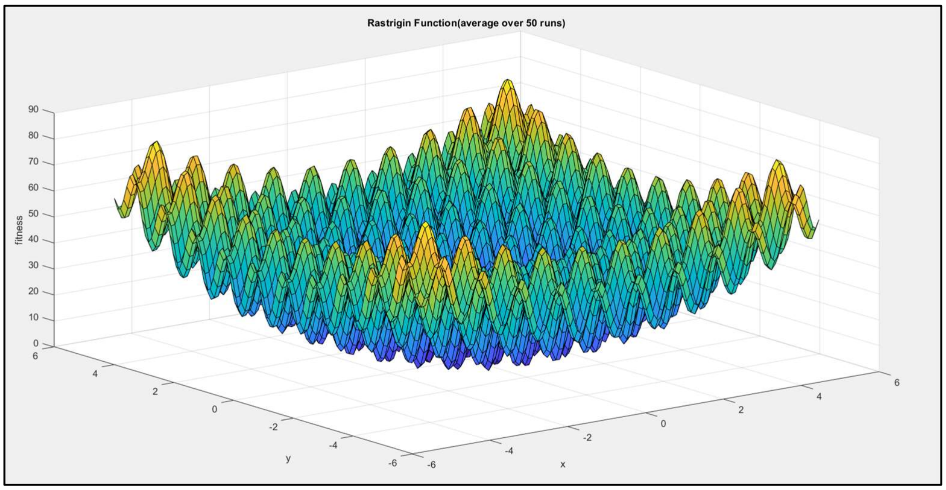


Figure 4.2.1: Surface plot of Rastrigin function

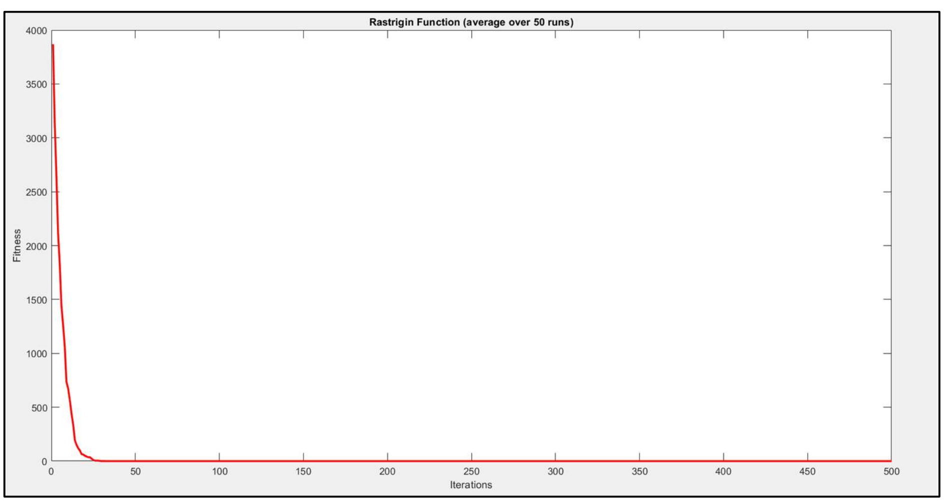


Figure 4.2.2: Convergence of Rastrigin function

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
main	1	7.042 s	0.052 s	
AHA	50	6.831 s	3.342 s	
SpaceBound	250000	1.948 s	1.948 s	
BenFunctions	261750	1.558 s	1.558 s	
newplot	2	0.054 s	0.038 s	
surf	1	0.047 s	0.008 s	

Figure 4.2.3: Time chart of Rastrigin function

The results in Figures 4.2.1 – 4.2.3 show that the AHA is able to find the global minima before completing 50 iterations in virtually every realisation. This proves that the AHA is able to converge to an accurate solution in a high dimensionality problem within a low number of iterations. Overall run time is fast as well, with the entire test being completed in around seven seconds.

4.3 Initialising Parameters and Coefficients

The proposed scheme is denoted as U-AHA. To evaluate the performance of the U-AHA, a service area of 800 × 800 m^2 is considered. The boundaries of the search space were set to x_{min} = 0 m, x_{max} = 800 m, y_{min} = 0 m, y_{max} = 800 m, z_{min} = 100 m, z_{max} = 500 m, while the UAV-BS parameters were set as follows: carrier frequency f_c = 2 GHz, transmit power P_t = 20 dBm (Simunek, Pechac and Fontan, 2011), noise power P_n = -110 dBm (Wu, Zeng and Zhang, 2018. The environmental parameters were set as follows: a = 0.6, b = 0.11, ηLoS = 1, $\eta NLoS$ = 20 (Niu, Zhao and Li, 2021). The user requirement is set to q_i = 75.8 b/s/Hz for all $i \in K$, and the penalty coefficient is set to ∂ = 0.3. In this section, the U-AHA will be tested at 200 iterations in an environment where the number of users, |K| = 200. Results are averaged over 30 realisations.

4.3.1 Collision Avoidance

The U-AHA is first evaluated on handling the collision avoidance penalty function. The scheme will be tested with the absence and presence of a penalty function, through the values of the penalty coefficient. The penalty coefficient

∂ is set to 0.3 as it is found to be able to guarantee collision avoidance without limiting the scheme's exploration of the search space.

Table 4.3.1: Performance of U-AHA in the absence and presence of a penalty function

Penalty coefficient	Blocking probability (%)	Jain's fairness index (%)	Sum of data rate (b/s/Hz)
0	2.5	97.499	14875.6835
0.3	3.5	96.499	14733.4174

Table 4.3.1.1 demonstrates that there is a small drop in performance of the scheme when the penalty function is introduced. However the difference in performance is extremely small, and therefore deemed insignificant. This test concludes that the U-AHA is able to ensure collision avoidance with little effect on its performance.

To determine the best probabilities between guided foraging (3-23) and territorial foraging (3-24), they are tested at probabilities between 0.1 to 0.9, in increments of 0.1.

Table 4.3.2: Foraging technique probabilities and corresponding fitness values

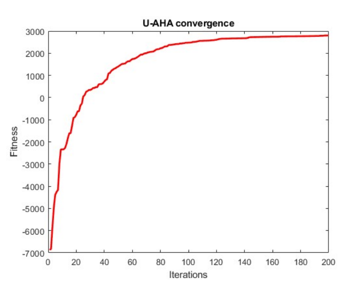
Probability of guided foraging	**Probability of territorial foraging**	**Fitness value**
0.1	0.9	1576.2255
0.2	0.8	1886.2048
0.3	0.7	2022.5864
0.4	0.6	2173.0033
0.5	0.5	2121.6362
0.6	0.4	2298.0739
0.7	0.3	2798.8181
0.8	0.2	2353.3515
0.9	0.1	2240.5277

Table 4.3.1 shows that guided foraging is generally a better at finding solutions as seen from the generally low fitness values achieved when a higher probability is given to territorial foraging. This is due to guided foraging's focus on exploration of the search space as opposed to territorial foraging's focus on exploitation within a small local area. The fitness value peaks when the probabilities of guided and territorial foraging are set to 0.7 and 0.3 respectively. When the probability of guided foraging is set to 0.8 and above, the fitness value drops significantly from its peak, suggesting that although territorial foraging is generally weaker than guided foraging at exploring the search space, it still acts as a method to account for the shortcomings of guided foraging. This is the essence of the mechanics of metaheuristic algorithms, defined by the fine balance between exploration and exploitation of the search space (Abdel-Basset, Abdel-Fatah and Sangaiah, 2018). Therefore, for the following tests, the probabilities of guided foraging and territorial foraging will be set to 0.7 and 0.3 respectively.

4.3.2 Degree of α-Fairness

To determine the ideal α-fairness measure, the U-AHA is tested at varying α values, i.e., α = 1, 2, 3.

Table 4.3.3: Convergence of U-AHA at various α values.

Convergence at different α values	**Fitness value**	**Jain's fairness index (%)**	**Sum of data rate (b/s/Hz)**	**Blocking probability (%)**
α = 1 	2785.566	97.126	14998.1131	2.6
α = 2	2273.426	97.499	14895.8209	1.8

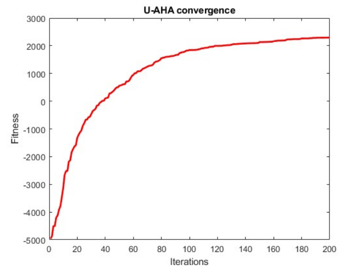
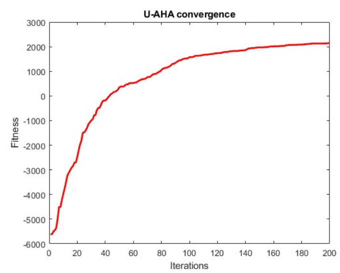
				
$\alpha = 3$ 	2157.194	97.999	14695.9573	2.2

Table 4.3.2 illustrate the performance of the proposed scheme at different α values. In all cases, the proposed scheme starts to converge around 100 iterations, which is reasonably efficient for a real world-based problem. It is observed that the U-AHA at α=1 converges to the highest fitness value, followed by α=2 and α=3. The Jain's fairness index values indicate that the larger α value results in increased fairness in QoS provisioning, with α=3 achieving the highest Jain's fairness index, followed by α=2 and α=1. However, there is a tradeoff between fairness and QoS provisioning as seen in the high blocking probability at α=1 despite the high sum of data rate. The tradeoff is better at α=2, where the sum of data rate drops slightly but the blocking probability decreases significantly. At α=3, the tradeoff starts to skew towards fairness over maximising QoS provisioning as the Jain's fairness index increases, but the sum of data rate is low enough to cause an increase in blocking probability. Therefore, despite α=1 achieving the highest fitness value, α=2 will be used for the following tests as it provides the best tradeoff between maximisation and fairness of QoS provisioning.

4.4 Scheme Evaluation

The U-AHA's performance is evaluated against the following two baseline schemes for comparison:

1. Random UAV-BS placement algorithm with user association, denoted as RA (Lim, Yu and Lee, 2022). This scheme operates by randomly placing the UAV-BSs throughout the service area over many iterations, the random UAV-BS positions that result in the highest fitness value is chosen as the best solution.
2. Service area partitioning-based UAV-BS placement with user association, denoted as PP (Huang and Savkin, 2022). This scheme operates by first partitioning the service area into the same number of partitions as the number of UAV-BSs. Each UAV-BS is then assigned a partition and randomly placed within its respective partition. This is repeated over many iterations, and the UAV-BS positions that result in the highest fitness value is chosen as the best solution.

For a fair comparison, the user association scheme developed for the U-AHA will also be used in the baseline schemes.

The three schemes will be tested in terms of the performance metrics stated in section 4.2, at varying numbers of users, i.e., $K = 50, 100, 150, 200, 250$ and 300. For each value of K, the results are obtained and averaged over 30 realisations. This is then tested over varying numbers of UAV-BSs, i.e., $J = 4, 8, 12$. The proposed U-AHA scheme's population size is set to $N = 20$, and the number of iterations is set to $I_{max} = 100$. The low number of iterations is meant to simulate a real-world scenario where the system operators may have a limited amount of time to deploy the scheme.

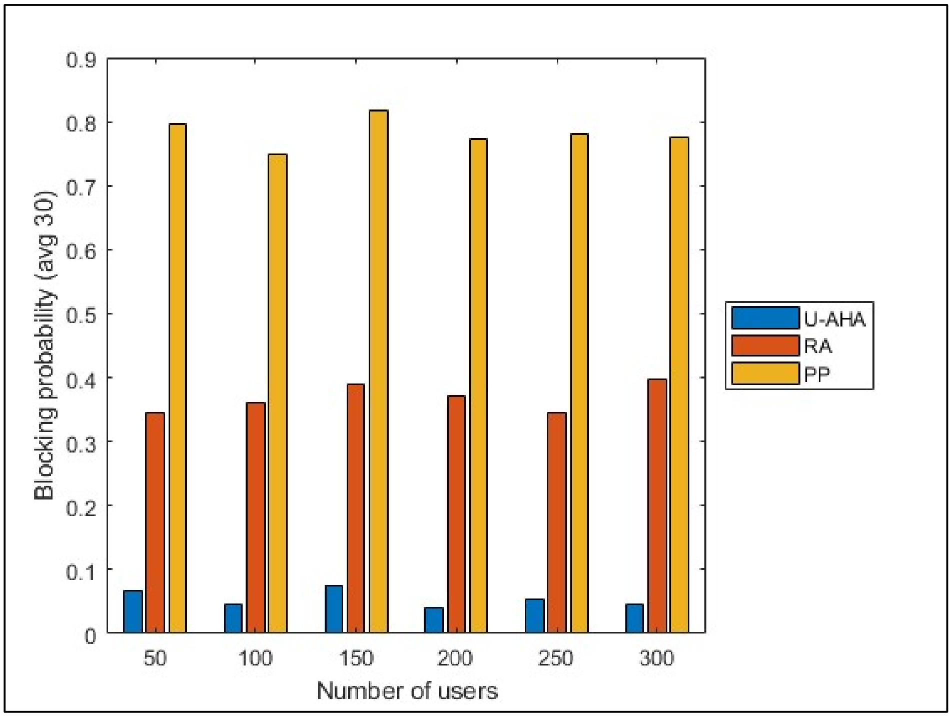


Figure 4.4.1: Blocking probability when $J = 4$

Table 4.4.1: Summary of blocking probability at $J = 4$

Number of Users	Blocking probability (%)		
	U-AHA	**RA**	**PP**
50	6.0	32.5	79.5
100	5.0	34.5	74.2
150	6.5	38.2	82.4
200	4.0	36.8	76.2
250	5.5	33.2	77.6
300	5.0	40.0	76.1

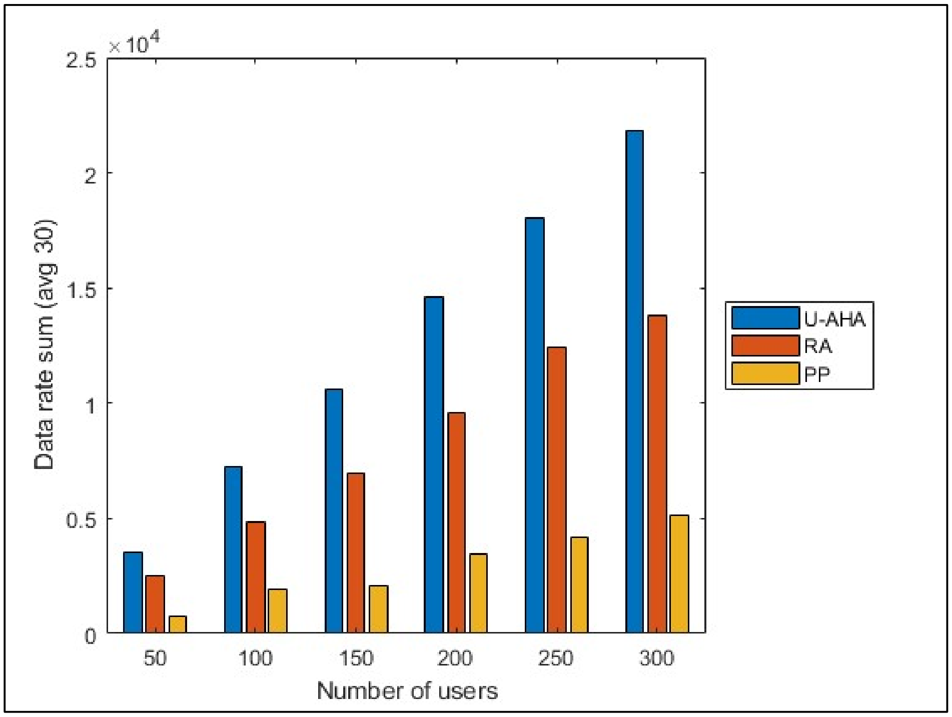


Figure 4.4.2: Sum of data rate when $J = 4$

Table 4.4.2: Summary of sum of data rate at $J = 4$

Number of Users	**Sum of data rates (b/s)**		
	U-AHA	**RA**	**PP**
50	4332.3246	2872.4847	862.9185
100	7412.7584	4682.5561	1982.1276
150	11296.7642	6385.8779	1997.8534
200	14824.9684	8671.3753	3426.4452
250	17863.6587	12863.8164	4192.1268
300	22475.3485	13782.1736	5006.7328

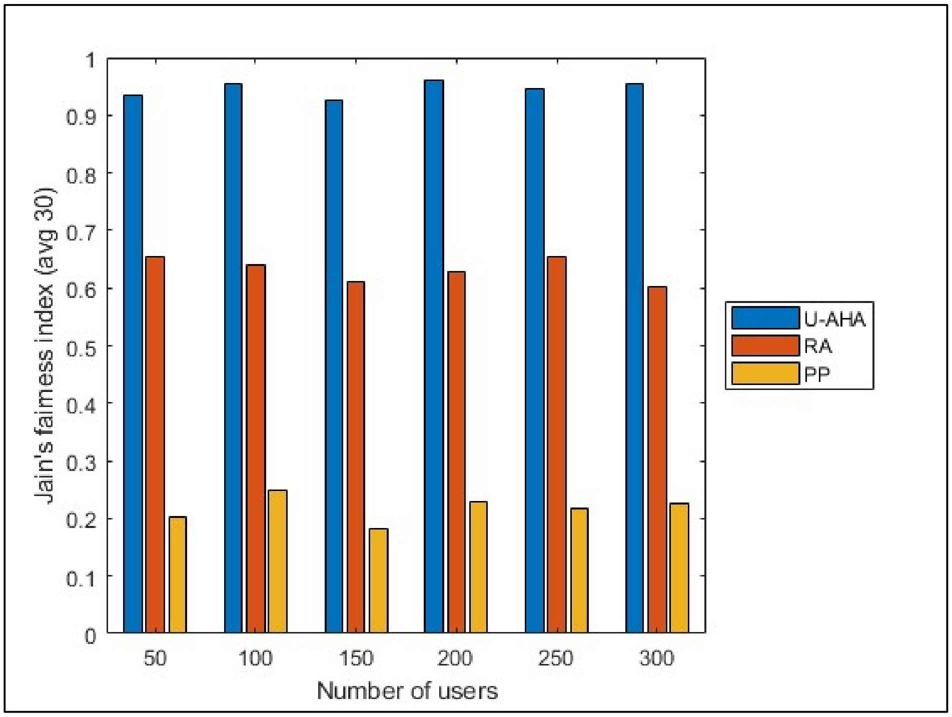


Figure 4.4.3: Jain's fairness index when $J = 4$

Table 4.4.3: Summary of Jain's fairness index at $J = 4$

Number of Users	**Jain's fairness index (%)**		
	U-AHA	**RA**	**PP**
50	93.223	64.938	19.981
100	95.812	63.273	24.922
150	92.829	60.162	18.172
200	96.274	62.728	22.562
250	95.121	65.182	21.849
300	95.992	60.133	22.119

Figure 4.4.1 – 4.4.3 and Table 4.4.1 – 4.4.3 illustrates the results obtained with four UAV-BSs. Figure 4.4.1 and Table 4.4.1 shows the blocking probability achieved by the three schemes. The proposed U-AHA scheme consistently obtains the lowest user blocking probability across all values. Figure 4.4.2 and Table 4.4.2 shows that the sum of user received data rates achieved by all schemes generally increased over different numbers of users. It is observed that the proposed U-AHA scheme outperforms the baseline schemes. Figure 4.4.3

and Table 4.4.3 shows that the proposed U-AHA scheme provides a fairer QoS fulfilment among users, with the achieved Jain's fairness indices of above 0.9 across the different numbers of users. This is attributed to the fact that the proposed U-AHA scheme is able to maximise the proportional fairness function for QoS provisioning far better than the baseline schemes. Meanwhile, Jain's fairness index values achieved by the baseline schemes are lower, and even show a slight decrease as the number of users increases.

To further verify that the findings remain consistent across varying scheme conditions, the 3 schemes are tested with the same metrics and varying user numbers, but with a different number of UAV-BSs $J = 8$.

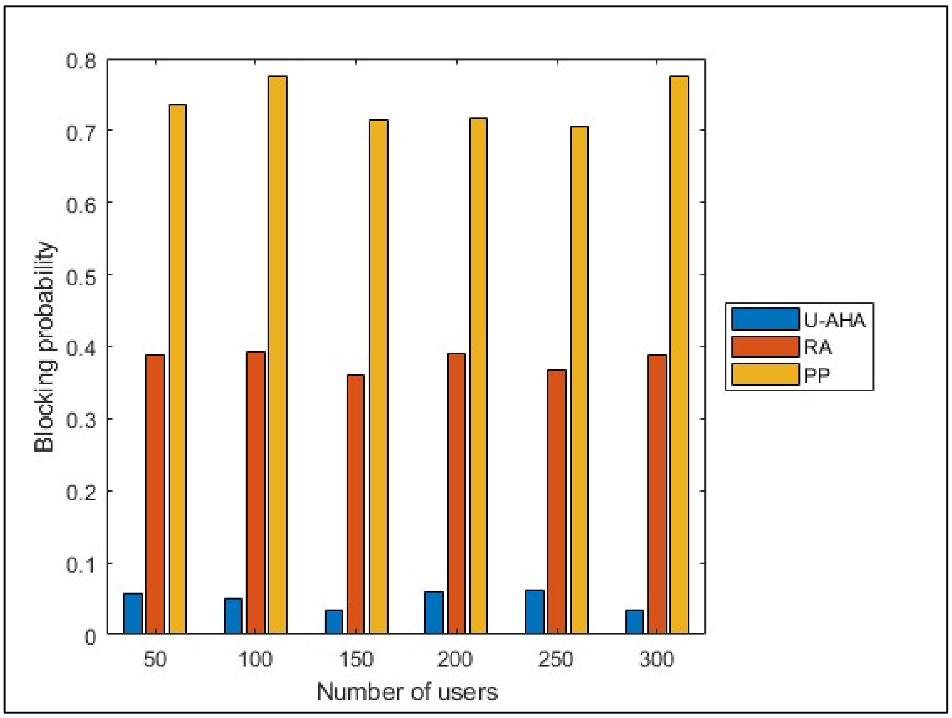


Figure 4.4.4: Blocking probability when $J = 8$

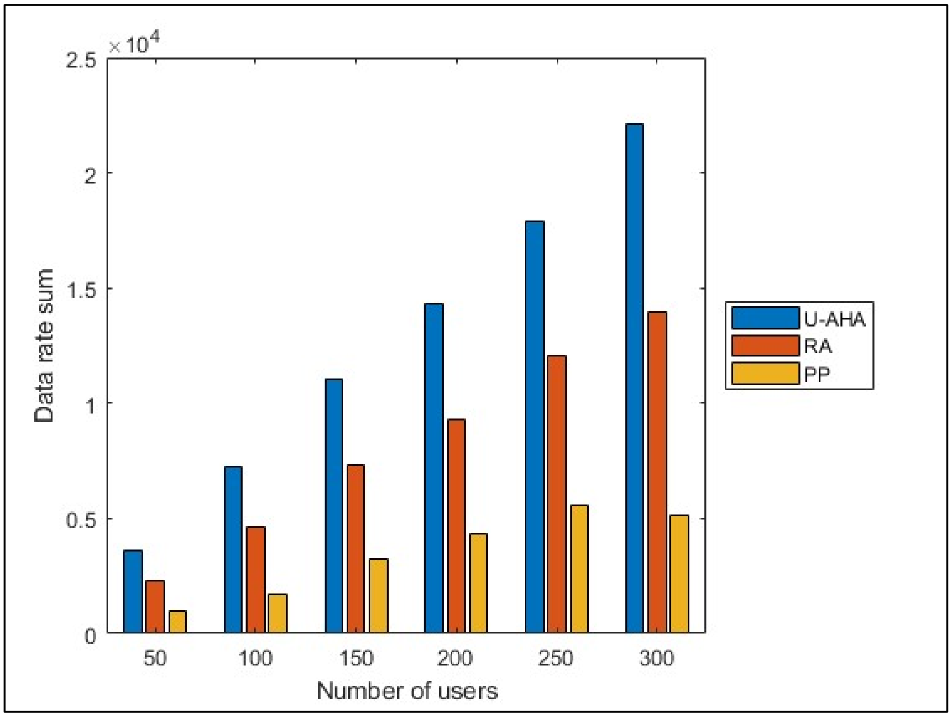


Figure 4.4.5: Sum of data rate when $J = 8$

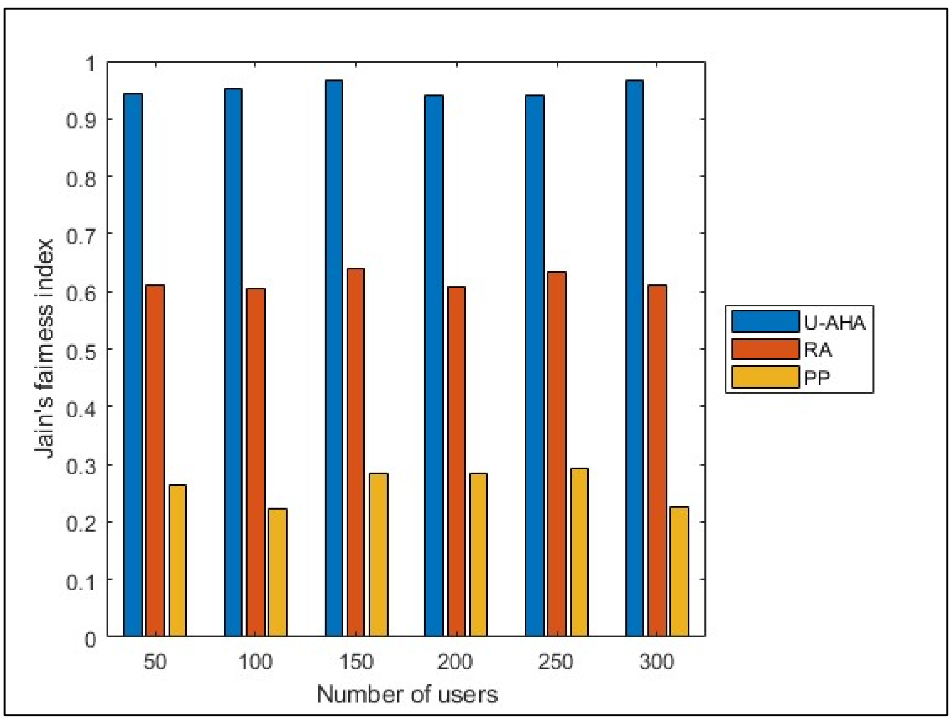


Figure 4.4.6: Jain's fairness index when $J = 8$

Figure 4.4.4 to 4.4.6 illustrates the results obtained when using 8 UAV-BSs. The performance trend in all metrics remain unchanged, with the U-AHA significantly outperforming RA and PP in every metric. A third test is conducted using 12 UAV-BSs.

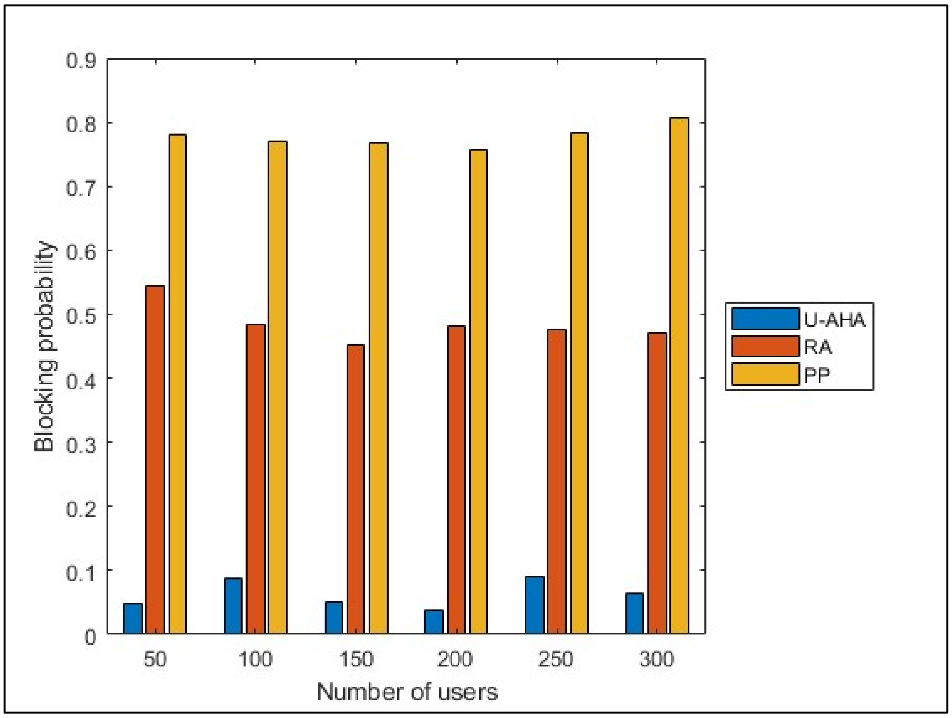


Figure 4.4.7: Blocking probability when $J = 12$

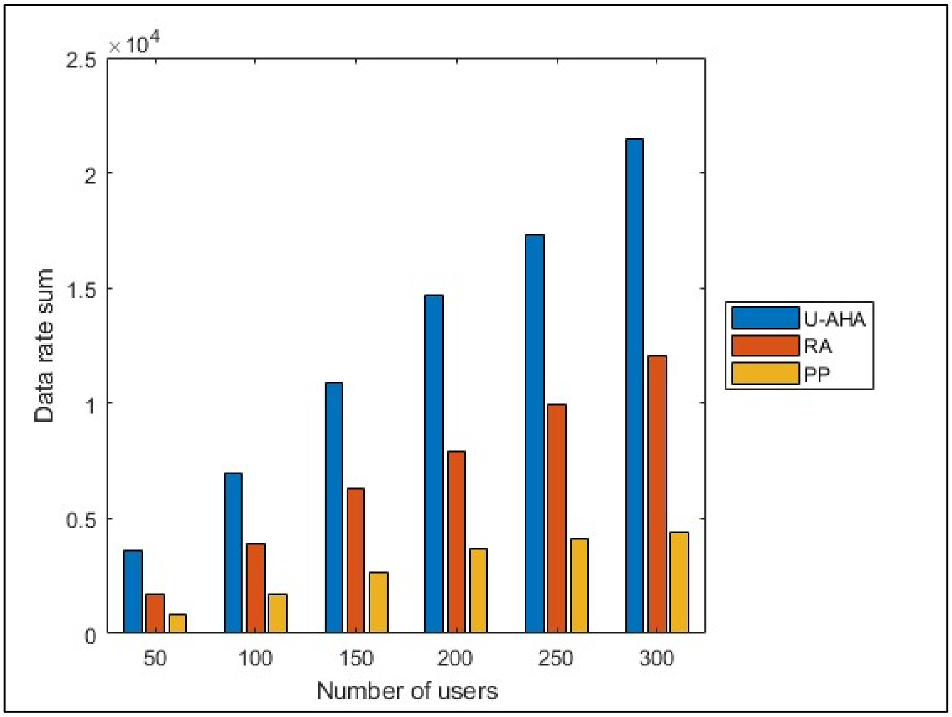


Figure 4.4.8: Sum of data rate when $J = 12$

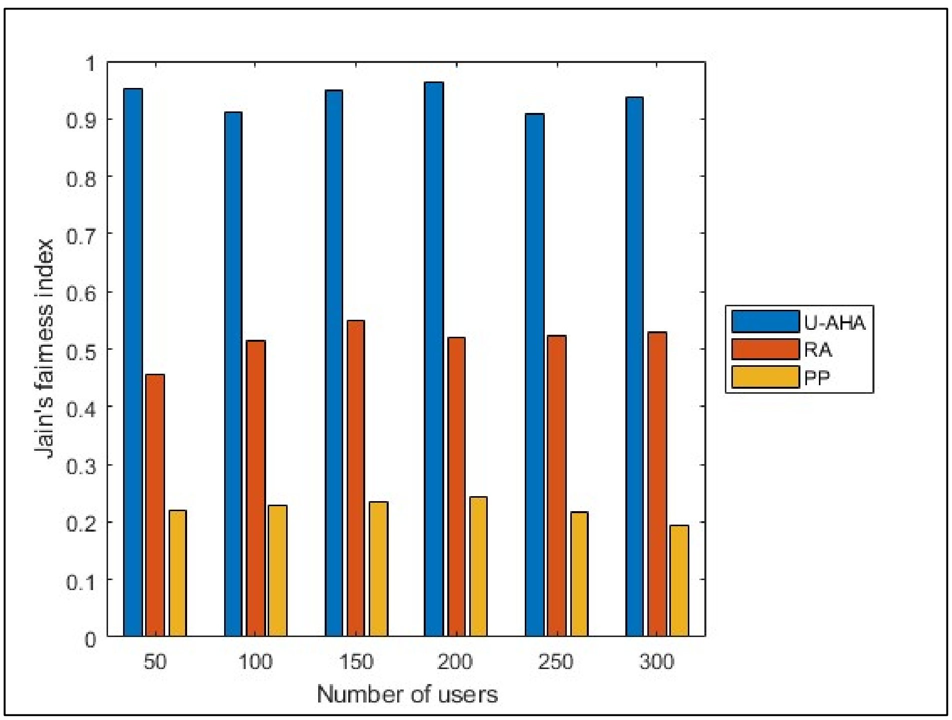


Figure 4.4.9: Jain’s fairness index when $J = 12$

Table 4.4.4: Performance ranking of schemes in each metric at all numbers of UAV-BSs

Metric	**Ranking (best to worst)**		
	1st	**2nd**	**3rd**
Blocking probability	U-AHA	RA	PP
Sum of data rate	U-AHA	RA	PP
Jains fairness index	U-AHA	RA	PP

Once again, Figures 4.4.7 – 4.4.9 show that the U-AHA continues to outperform the RA and PP in all performance metrics even as the number of UAV-BSs increases to 12. From these results, it can be inferred that the U-AHA will outperform the baseline schemes at any number of users and any number of UAV-BSs. The overall performance rankings of each scheme is tabulated in Table 4.4.4, clearly reflecting the U-AHA's superior performance.

RA expectedly demonstrates poorer performance than U-AHA due to its random nature, as the U-AHA is a metaheuristic algorithm that makes educated guesses according to a framework based on the natural behavior on hummingbirds in the wild. The PP's poor overall performance is due to the limitations set by the search space partitions. On paper, it seems as if the PP should outperform the RA as it fairly partitions the search space to place each UAV-BS, which should maximise QoS fairness among users. However, testing results show that this is not the case, as it greatly limits the potential of each UAV-BS by confining it to a small space. The RA's flexibility in exploring the search space allows for it to achieve better solutions if given enough iterations, which is ultimately how it consistently outperforms PP.

4.5 Simulated Visualisation of Real-World Results

To present the physical outcome of the scheme, the results of running the U-AHA with varying numbers of UAV-BSs J=4, 8, 12 and number of users K=200 is recorded. The resulting coordinates of users i and UAV-BSs j are then plotted and labelled on a 3D graph to visualise the simulation environment.

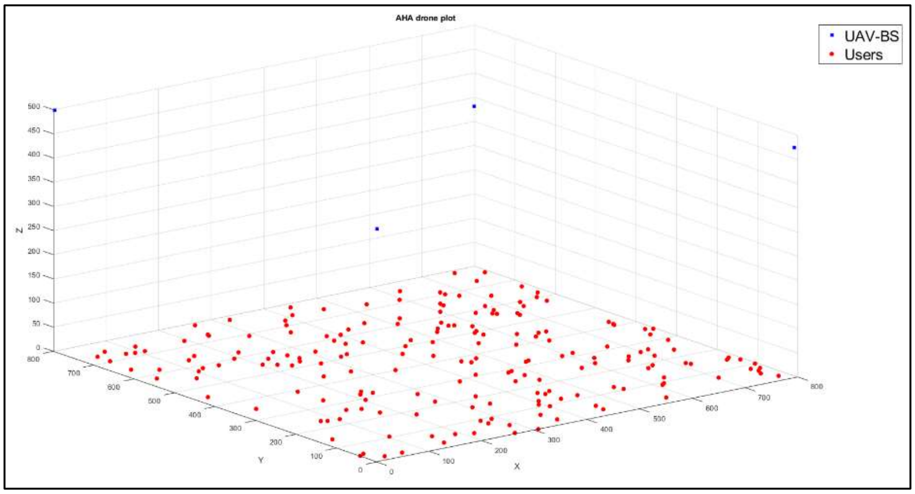


Figure 4.5.1: Simulation environment results at $J = 4$

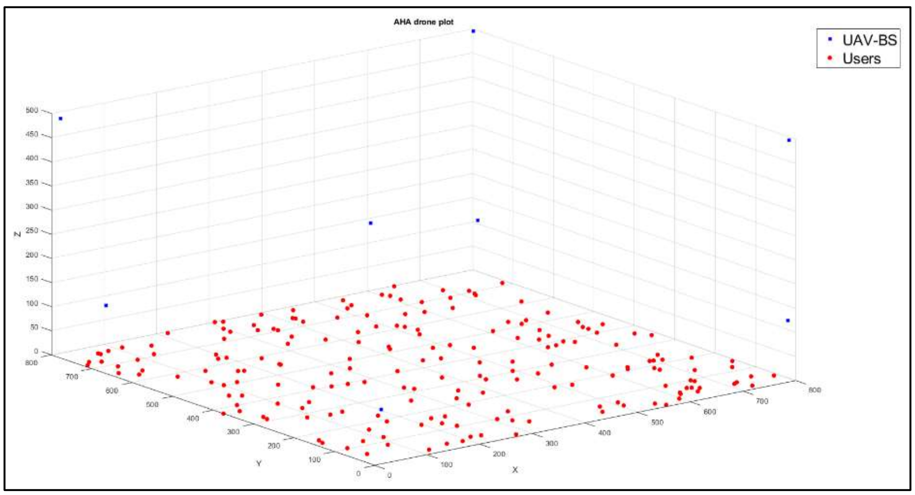


Figure 4.5.2: Simulation environment results at $J = 8$

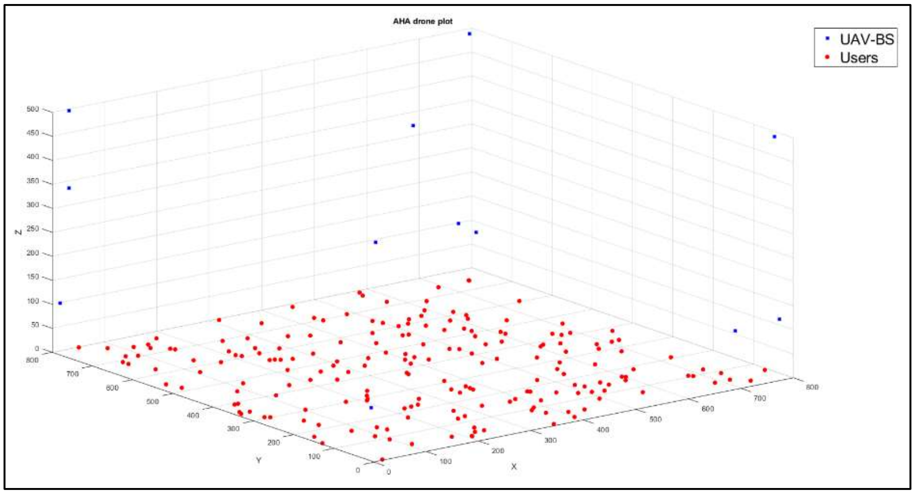


Figure 4.5.3: Simulation environment results at J = 12

Figures 4.5.1 – 4.5.3 illustrate the results of the test environment, which reflects the system model in Figure 1.4.1. The uniformly distributed random positions of the 200 ground users are indicated by red dots, while the aerially-positioned UAV-BSs are represented by blue dots. It is also observed that there is ample space between the UAV-BSs at all three J values, which clearly demonstrates the scheme's collision avoidance feature.

4.6 Summary

The obtained results show that the U-AHA is able to efficiently solve optimisation problems. It solves fairness-aware multi-UAV-BS placement problem by maximising the objective function (3-26), which addresses fairness aware provisioning of user's received data rates and collision avoidance. Results show that the U-AHA is able to outperform the RA and PP in terms of blocking probability, sum of user's received data rate and Jain's fairness index, while the visualisation of the simulation environment demonstrates the collision avoidance feature. At varying numbers of UAV-BSs, the U-AHA is able to consistently satisfy the QoS requirements of 90% of users while ensuring maximum fairness, achieving a Jain's fairness index of over 90% in all conducted tests. Taking into account the cost of a real-world UAV-BS network system, the high Jain's fairness index rating dratiscally reduces operating costs as it minimises the number of UAV-BSs needed to meet user QoS requirements.

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

Fairness-aware QoS provisioning is crucial to efficiently utilising UAV-BSs to provide good quality network coverage for users, while collision avoidance is a crucial safety factor in multi-UAV-BS networks. The proposed AHA-based UAV-BS placement scheme successfully maximises the objective function, leading to more fair service to users as well as a larger sum of user's received data rates than random UAV-BS placement and partition-based UAV-BS placement. Overall, the results demonstrated by the proposed scheme act as a proof of concept of the scheme's feasibility, however further research and testing is required before real-world implementation can be tested.

5.2 Limitations and Recommendations

5.2.1 Limited Channel Bandwidth

As stated in section 1.5, this project does not address limited channel bandwidth of UAV-BSs. In future works, this should be considered as the proposed scheme assumes that the UAV-BSs have unlimited channel bandwidth and can serve any number of users. Fairness provision is achieved in this project through a one-way association between a user and a specific UAV-BS. To limit the number of users that can be serviced by a single UAV-BS, a two-way user-UAV-BS association problem can be formulated into the objective function, while other relevant environmental parameters can be derived from existing literature. This should result in a scheme that better reflects the real-world capabilities of UAV-BSs.

5.2.2 Mobile Users

The proposed scheme does not address the movement of users throughout the service area, but assumes that users are static. As it is virtually impossible for users to be fully static in a real-world scenario, this should be considered in future works. A suggestion would be to recalibrate the positions of UAV-BSs at fixed time intervals based on the continuously changing positions of users. Machine learning approaches may also be implemented to predict how users may move around the service area instead of collecting real-time data on user positions.

5.2.3 Machine Learning Approaches

Furthermore, this scheme uses a metaheuristic approach, which negates the long computation time needed for exhaustive approaches, by using an algorithm guided by a certain framework to make an educated guess of a "good enough" solution for complex optimisation problems (Abdel-Basset, Abdel-Fatah and Sangaiah, 2018). However, real-world applications involve further challenges including the path-planning, flying speed and limited battery life of UAV-BSs. Machine learning schemes may be able to develop solutions that address more of these constraints when provided with sufficient training data, however their biggest limitations lie in difficulty of implementation and processing time. Therefore, it could be beneficial to mix machine learning and meta-heuristic approaches to develop a solution that can overcome this challenge.

REFERENCES

Abdel-Basset, M., Abdel-Fatah, L. and Sangaiah, A.K., 2018. Metaheuristic Algorithms: A Comprehensive Review. In: *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*. [online] Elsevier. pp.185–231. https://doi.org/10.1016/B978-0-12-813314-9.00010-4.

Al-Hourani, A., Kandeepan, S. and Lardner, S., 2014. Optimal LAP Altitude for Maximum Coverage. *IEEE Wireless Communications Letters*, [online] 3(6), pp.569–572. https://doi.org/10.1109/LWC.2014.2342736.

Alzenad, M., El-Keyi, A. and Yanikomeroglu, H., 2018. 3-D Placement of an Unmanned Aerial Vehicle Base Station for Maximum Coverage of Users with Different QoS Requirements. *IEEE Wireless Communications Letters*, 7(1), pp.38–41. https://doi.org/10.1109/LWC.2017.2752161.

Bockelmann, C., Pratas, N.K., Wunder, G., Saur, S., Navarro, M., Gregoratti, D., Vivier, G., De Carvalho, E., Ji, Y., Stefanovic, C., Popovski, P., Wang, Q., Schellmann, M., Kosmatos, E., Demestichas, P., Raceala-Motoc, M., Jung, P., Stanczak, S. and Dekorsy, A., 2018. Towards Massive Connectivity Support for Scalable mMTC Communications in 5G Networks. *IEEE Access*, [online] 6, pp.28969–28992. https://doi.org/10.1109/ACCESS.2018.2837382.

Chen, Y., Li, N., Wang, C., Xie, W. and Xv, J., 2018. A 3D Placement of Unmanned Aerial Vehicle Base Station Based on Multi- Population Genetic Algorithm for Maximizing Users with Different QoS Requirements. In: *2018 18th IEEE International Conference on Communication Technology*. Chongqing, China: IEEE. https://doi.org/10.1109/ICCT.2018.8600206.

Cicek, C.T., Gultekin, H., Tavli, B. and Yanikomelogru, H., 2019. UAV Base Station Location Optimization for Next Generation Wireless Networks: Overview and Future Research Directions. In: *Proceedings of the 1st International Conference on Unmanned Vehicle Systems (UVS), Muscat, Oman,* *5-7 February, 2019*. Muscat, Oman: IEEE. https://doi.org/10.1109/UVS.2019.8658363.

Dahlman, E., Parkvall, S. and Sköld, J., 2018. *5G NR: the Next Generation Wireless Access Technology*. [online] Elsevier. https://doi.org/10.1016/C2017-0-01347-2.

Du, W., Ying, W., Yang, P., Cao, X., Yan, G., Tang, K. and Wu, D., 2020. Network-Based Heterogeneous Particle Swarm Optimization and Its Application in UAV Communication Coverage. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(3), pp.312–323. https://doi.org/10.1109/TETCI.2019.2899604.

Gopi, S.P. and Magarini, M., 2021. Reinforcement Learning Aided UAV Base Station Location Optimization for Rate Maximization. *Electronics*, [online] 10(23), p.2953. https://doi.org/10.3390/electronics10232953.

Huang, H. and Savkin, A. V., 2022. Deployment of Heterogeneous UAV Base Stations for Optimal Quality of Coverage. *IEEE Internet of Things Journal*, [online] 9(17), pp.16429–16437. https://doi.org/10.1109/JIOT.2022.3150292.

Huang, M., Huang, L., Zhong, S. and Zhang, P., 2020. UAV-Mounted Mobile Base Station Placement via Sparse Recovery. *IEEE Access*, [online] 8, pp.71775–71781. https://doi.org/10.1109/ACCESS.2020.2987851.

Lai, C.-C., Chen, C.-T. and Wang, L.-C., 2019. On-Demand Density-Aware UAV Base Station 3D Placement for Arbitrarily Distributed Users With Guaranteed Data Rates. *IEEE Wireless Communications Letters*, [online] 8(3), pp.913–916. https://doi.org/10.1109/LWC.2019.2899599.

Lee, Y.L., Chuah, T.C., El-Saleh, A.A. and Loo, J., 2018. User Association for Backhaul Load Balancing With Quality of Service Provisioning for Heterogeneous Networks. *IEEE Communications Letters*, [online] 22(11), pp.2338–2341. https://doi.org/10.1109/LCOMM.2018.2867181.

Lee, Y.L., Tan, W.L., Lau, S.B.Y., Chuah, T.C., El-Saleh, A.A. and Qin, D., 2020. Joint Cell Activation and User Association for Backhaul Load Balancing in Green HetNets. *IEEE Wireless Communications Letters*, [online] 9(9), pp.1486–1490. https://doi.org/10.1109/LWC.2020.2994643.

Li, B., Chen, C., Zhang, R., Jiang, H. and Guo, X., 2018. The Energy-efficient UAV-based BS Coverage in Air-to-Ground Communications. In: *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*. https://doi.org/10.1109/SAM.2018.8448770.

Lim, N.H.Z., Lee, Y.L., Tham, M.L., Chang, Y.C., Sim, A.G.H. and Qin, D., 2021. Coverage Optimization for UAV Base Stations using Simulated Annealing. In: *2021 IEEE 15th Malaysia International Conference on Communication (MICC)*. [online] IEEE. pp.43–48. https://doi.org/10.1109/MICC53484.2021.9642136.

Lim, S., Yu, H. and Lee, H., 2022. Optimal Tethered-UAV Deployment in A2G Communication Networks: Multi-Agent *Q* -Learning Approach. *IEEE Internet of Things Journal*, [online] 9(19), pp.18539–18549. https://doi.org/10.1109/JIOT.2022.3161260.

Lin, X.-H., Bi, S.-Z., Cheng, N., Dai, M.-J. and Wang, H., 2022. An α-Fairness Approach to Balancing the Energy Consumption Among Sensors for UAV–IoT Systems. *IEEE Internet of Things Journal*, [online] 9(18), pp.17965–17978. https://doi.org/10.1109/JIOT.2022.3164403.

Lyu, J., Zeng, Y., Zhang, R. and Lim, T.J., 2017. Placement Optimization of UAV-Mounted Mobile Base Stations. *IEEE Communications Letters*, 21(3), pp.604–607. https://doi.org/10.1109/LCOMM.2016.2633248.

Mozaffari, M., Saad, W., Bennis, M., Nam, Y.-H. and Debbah, M., 2018. A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems. [online] Available at: <http://arxiv.org/abs/1803.00680>.

Ni, Y., Liang, J., Shi, X. and Ban, D., 2019. Research on Key Technology in 5G Mobile Communication Network. In: *2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS)*. [online] IEEE. pp.199–201. https://doi.org/10.1109/ICITBS.2019.00054.

Niu, H., Zhao, X. and Li, J., 2021. 3D Location and Resource Allocation Optimization for UAV-Enabled Emergency Networks under Statistical QoS Constraint. *IEEE Access*, 9, pp.41566–41576. https://doi.org/10.1109/ACCESS.2021.3065055.

Qiu, J., Lyu, J. and Fu, L., 2020. Placement Optimization of Aerial Base Stations with Deep Reinforcement Learning. In: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. [online] IEEE. pp.1–6. https://doi.org/10.1109/ICC40277.2020.9149258.

Shakhatreh, H., Khreishah, A., Alsarhan, A., Khalil, I., Sawalmeh, A. and Othman, N.S., 2017a. Efficient 3D placement of a UAV using particle swarm optimization. In: *2017 8th International Conference on Information and Communication Systems (ICICS)*. [online] IEEE. pp.258–263. https://doi.org/10.1109/IACS.2017.7921981.

Shakhatreh, H., Khreishah, A., Alsarhan, A., Khalil, I., Sawalmeh, A. and Othman, N.S., 2017b. Efficient 3D placement of a UAV using particle swarm optimization. In: *2017 8th International Conference on Information and Communication Systems (ICICS)*. [online] IEEE. pp.258–263. https://doi.org/10.1109/IACS.2017.7921981.

Simunek, M., Pechac, P. and Fontan, F.P., 2011. *Excess Loss Model for Low Elevation Links in Urban Areas for UAVs*. *RADIOENGINEERING*, .

Wu, Q., Zeng, Y. and Zhang, R., 2018. Joint trajectory and communication design for multi-UAV enabled wireless networks. *IEEE Transactions on Wireless Communications*, 17(3), pp.2109–2121. https://doi.org/10.1109/TWC.2017.2789293.

Zhang, S., Wang, Y. and Zhou, W., 2019. Towards secure 5G networks: A Survey. *Computer Networks*, [online] 162, p.106871. https://doi.org/10.1016/j.comnet.2019.106871.

Zhao, W., Wang, L. and Mirjalili, S., 2022. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. *Computer Methods in Applied Mechanics and Engineering*, [online] 388, p.114194. https://doi.org/10.1016/j.cma.2021.114194.

APPENDICES

APPENDIX A: MATLAB code - AHA

```
function
[AHA_coordinates,AHA_fitness,HisBestFit,AHA_data_rate,AHA_Jain_
fairness,AHA_loss_rate]=AHA2(MaxIt,nPop,Low,Up,Dim,UAV_num,UE_n
um,PopPos,Pos_U,A_k)
    %% initial agent positions PopPos
    PopFit=zeros(1,nPop);

        for i=1:nPop

[fitness,~,~,~]=ObjFunction(UAV_num,UE_num,A_k,Pos_U,PopPos(i,:
));
            PopFit(i) = fitness;
        end

    AHA_fitness= -inf; % maximisation problem
    AHA_coordinates=[];

    for i=1:nPop
        if PopFit(i)>=AHA_fitness
            AHA_fitness=PopFit(i);
            AHA_coordinates=PopPos(i,:);
        end
    end

    % Initialize visit table
    HisBestFit=zeros(MaxIt,1);
    VisitTable=zeros(nPop) ;
    VisitTable(logical(eye(nPop)))=NaN;

    for It=1:MaxIt
        DirectVector=zeros(nPop,Dim);% Direction vector/matrix

        for i=1:nPop
            r=rand;
            if r<1/3     % Diagonal flight
                RandDim=randperm(Dim);
                if Dim>=3
                    RandNum=ceil(rand*(Dim-2)+1);
                else
                    RandNum=ceil(rand*(Dim-1)+1);
                end
                DirectVector(i,RandDim(1:RandNum))=1;
            else
                if r>2/3  % Omnidirectional flight
                    DirectVector(i,:)=1;
                else  % Axial flight
                    RandNum=ceil(rand*Dim);
                    DirectVector(i,RandNum)=1;
                end
            end

            if rand<0.7   % Guided foraging

[MaxUnvisitedTime,TargetFoodIndex]=max(VisitTable(i,:));
```

```
MUT_Index=find(VisitTable(i,:)==MaxUnvisitedTime);
                if length(MUT_Index)>1
                    [~,Ind]= min(PopFit(MUT_Index));
                    TargetFoodIndex=MUT_Index(Ind);
                end


newPopPos=PopPos(TargetFoodIndex,:)+randn*DirectVector(i,:).*(P
opPos(i,:)-PopPos(TargetFoodIndex,:));

newPopPos=SpaceBound(newPopPos,Up,Low); %spacebound function

[newPopFit,~,~,~]=ObjFunction(UAV_num,UE_num,A_k,Pos_U,newPopPo
s);

                if newPopFit>PopFit(i)
                    PopFit(i)=newPopFit;
                    PopPos(i,:)=newPopPos;
                    VisitTable(i,:)=VisitTable(i,:)+1;
                    VisitTable(i,TargetFoodIndex)=0;
                    VisitTable(:,i)=max(VisitTable,[],2)+1;
                    VisitTable(i,i)=NaN;
                else
                    VisitTable(i,:)=VisitTable(i,:)+1;
                    VisitTable(i,TargetFoodIndex)=0;
                end
            else    % Territorial foraging
                newPopPos=
PopPos(i,:)+randn*DirectVector(i,:).*PopPos(i,:);
                newPopPos=SpaceBound(newPopPos,Up,Low);

[newPopFit,~,~,~]=ObjFunction(UAV_num,UE_num,A_k,Pos_U,newPopPo
s);
                if newPopFit>PopFit(i)
                    PopFit(i)=newPopFit;
                    PopPos(i,:)=newPopPos;
                    VisitTable(i,:)=VisitTable(i,:)+1;
                    VisitTable(:,i)=max(VisitTable,[],2)+1;
                    VisitTable(i,i)=NaN;
                else
                    VisitTable(i,:)=VisitTable(i,:)+1;
                end
            end
        end

        if mod(It,2*nPop)==0 % Migration foraging
            [~, MigrationIndex]=min(PopFit);
            PopPos(MigrationIndex,:) =rand(1,Dim).*(Up-
Low)+Low;

[PopFit(MigrationIndex),~,~,~]=ObjFunction(UAV_num,UE_num,A_k,P
os_U,PopPos(MigrationIndex,:));

VisitTable(MigrationIndex,:)=VisitTable(MigrationIndex,:)+1;

VisitTable(:,MigrationIndex)=max(VisitTable,[],2)+1;
            VisitTable(MigrationIndex,MigrationIndex)=NaN;
        end
```

```
        for i=1:nPop
            if PopFit(i)>AHA_fitness
                AHA_fitness=PopFit(i);
                AHA_coordinates=PopPos(i,:);
            end
        end


[~,AHA_Jain_fairness,AHA_loss_rate,AHA_data_rate]=ObjFunction(U
AV_num,UE_num,A_k,Pos_U,AHA_coordinates);
        HisBestFit(It)=AHA_fitness;
    end
end % end of function
```

APPENDIX B: MATLAB code – Objective Function

```
function
[fitness,jain_fairness,loss_rate,sum_data_rate]=ObjFunction(UAV
_num,UE_num,A_k,Pos_U,PopPos)
%% coefficient settings
a = 0.6; b = 0.11; % urban
f_c = 2*(10^9); % 2 GHz, Carrier frequency, Cation: Chen, Liao,
and Chen, ¡®End-to-End Delay Analysis in Aerial-Terrestrial
Heterogeneous Networks¡¯.
c = 3 * (10^8); % speed of light
w_1 = 0.3; % penalty cofficient of A_k
% eta_LoS = 1;
% eta_NLoS = 20;
alpha = 3;
F = 20*log10((4*pi*f_c)/c)+23; % path loss in 0DB
P_t = 0.1; %  30 dBm, 1 watt
P_N = 10^(-14); %  -110 dBm, 10^(-14) watt
D_sec = 25; % security distance between UAV
%% PROBLEM start by removing this nPop
        % ---- UAV get the global information ---- (is this
necessary? cuz the values of the 2d matrix are all the same
anyway)
        for j = 1:UAV_num
            n = 3*j;
            % ---- distance of UAV with UE ----
            for k = 1:UE_num
                UAV_UE_distance(j,k) = sqrt( (PopPos(n-2)-
Pos_U(k,1))^2 + (PopPos(n-1)-Pos_U(k,2))^2 + (PopPos(n))^2 ); %
the distance of UAV/drone and UE
                UAV_UE_angle(j,k) =
(180/pi)*acos(PopPos(n)/UAV_UE_distance(j,k)); % angle of
UAV/drone and UE
            end % end for k

            for i=1:UAV_num
                p = 3*i;
                UAV_UAV_distance(j,i) = sqrt( (PopPos(n-2)-
PopPos(p-2))^2 + (PopPos(n-1)-PopPos(p-1))^2 + (PopPos(n)-
PopPos(p))^2 );
            end
        end
        link_Pro_LoS = 1./( 1+ a.*exp( -b.*(UAV_UE_angle -
a) ) ); % LoS probability function, coefficient b=0.11,
cofficient a=12.08.
```

```
        link_Path_loss = 10.^((10*log10(UAV_UE_distance.^2)+((-
21.4)*link_Pro_LoS)+F)/10); % mean path loss function, L =
P_LoS*L_LoS + P_NLoS*L_NLoS.
        link_SNR = (P_t.*link_Path_loss)/(P_N);
        link_data_rate = log2(1+link_SNR);

        % ---- Update the connection state ----
        % greedy algorithm
        C = zeros(UAV_num,UE_num);

        for k = 1:UE_num
            C(:,k) = 0;
            [ff idx] = max(link_data_rate(:,k)); % select the
max value to serve user
            C(idx,k) = 1;
            % ---- less than the requirment ----
            if link_data_rate(idx,k) < A_k(k)
                C(idx,k) = 0;
            end
        end

        link_user_data_rate = sum(C.*link_data_rate, 1); % here
calaculate the data rate provided by each UAV
        %% received data/user requiremrent
        w = link_user_data_rate./A_k;

        for  i=1:UE_num
            if w(i) == 0
                link_fair(i) = -100;
                 % the fairness of j_th UAV,alpha = others
            else
                if alpha == 1
                    link_fair(i) = log(w(i));

                else
                    if alpha~=1 && alpha>=0
                    link_fair(i)=(w(i)^(1-alpha))/(1-alpha);
                    end
                end
            end
        end

        loss_rate =  (UE_num - sum(sum(C)))/(UE_num) ;  % loss
of user connectivity

        jain_fairness =
( (sum(w)).^2 )/( UE_num*( sum((w).^2) ) ) ;% Jain's fairness
        sum_data_rate = sum(link_user_data_rate);

        fitness = sum(link_fair) - w_1*sum( sum(D_sec-
UAV_UAV_distance) );
```

APPENDIX C: MATLAB code - RA

```
function
[Random_pos,Random_fitness,Random_Jain_fairness,Random_loss_rat
e,Random_data_rate]=RA(MaxIt,XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,Dim,
UAV_num,UE_num,Pos_U,A_k)

Random_pos_temp = zeros(1,Dim);
Random_pos = zeros(1,Dim);
Random_fitness = -inf; %change this to -inf for maximization
problems
Random_Jain_fairness = zeros(1,1);
Random_loss_rate = zeros(1,1);
Random_data_rate = zeros(1,1);

%% initial values
%[fitness,jain_fairness,loss_rate,sum_data_rate]=ObjFunction(UA
V_num,UE_num,A_k,Pos_U,PopPos(i,:));
    for i=1:MaxIt

                for j = 1:Dim
                    % X-coordinate
                    Random_pos_temp(1, (j - 1) * 3 + 1) = XMIN
+ (XMAX / 3 - XMIN) * rand;

                    % Y-coordinate
                    Random_pos_temp(1, (j - 1) * 3 + 2) = YMIN
+ (YMAX / 4 - YMIN) * rand;

                    % Z-coordinate
                    Random_pos_temp(1, (j - 1) * 3 + 3) = (ZMAX
- ZMIN) * rand + ZMIN;
                end


[fitness,jain,loss_rate,data_rate]=ObjFunction(UAV_num,UE_num,A
_k,Pos_U,Random_pos_temp(1,:));
            if fitness>Random_fitness
                Random_pos = Random_pos_temp(1,:);
                Random_fitness = fitness;
                Random_Jain_fairness = jain;
                Random_loss_rate = loss_rate;
                Random_data_rate = data_rate;
            end
        %end % nPop
    end %max iter
    if all(Random_pos(:) == 0)

                for j = 1:Dim
                    % X-coordinate
                    Random_pos((j - 1) * 3 + 1) = XMIN + (XMAX
/ 3 - XMIN) * rand;

                    % Y-coordinate
                    Random_pos((j - 1) * 3 + 2) = YMIN + (YMAX
/ 4 - YMIN) * rand;

                    % Z-coordinate
```

```
                      Random_pos((j - 1) * 3 + 3) = (ZMAX - ZMIN)
* rand + ZMIN;
                end
    end
end
```

APPENDIX D: MATLAB code - PP

```
function
[Static_pos,Static_fitness,Static_Jain_fairness,Static_loss_rat
e,Static_data_rate]=SP(iter,XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX,Dim,U
AV_num,UE_num,Pos_U,A_k)

    %% ------ UAV BSs Settings ------
%Fixed altitude (midpoint between ZMAX and ZMIN)
altitude = (ZMAX + ZMIN) / 2;
Static_fitness = -inf;
pos = zeros(iter,Dim);
    for i=1:iter
        if Dim == 12
            %% 4 UAVs
            % ---- UAV 1 ----
             pos(i,1) = XMIN + (XMAX/2-XMIN)*rand;
             pos(i,2) = YMIN + (YMAX/2-YMIN)*rand;
             pos(i,3) = altitude;
             % ---- UAV 2 ----
             pos(i,4) = XMAX/2 + (XMAX-XMAX/2)*rand;
             pos(i,5) = YMIN + (YMAX/2-YMIN)*rand;
             pos(i,6) = altitude;
             % ---- UAV 3 ----
             pos(i,7) = XMIN + (XMAX/2-XMIN)*rand;
             pos(i,8) = YMAX/2 + (YMAX-YMAX/2)*rand;
             pos(i,9) = altitude;
             % ---- UAV 4 ----
             pos(i,10) = XMAX/2 + (XMAX-XMAX/2)*rand;
             pos(i,11) = YMAX/2 + (YMAX-YMAX/2)*rand;
             pos(i,12) = altitude;
        elseif(Dim == 24)
             % ---- UAV 1 ----
             pos(i,1) = XMIN + (XMAX/2-XMIN)*rand;
             pos(i,2) = YMIN + (YMAX/2-YMIN)*rand;
             pos(i,3) = altitude;
             % ---- UAV 2 ----
             pos(i,4) = XMAX/2 + (XMAX-XMAX/2)*rand;
             pos(i,5) = YMIN + (YMAX/2-YMIN)*rand;
             pos(i,6) = altitude;
             % ---- UAV 3 ----
             pos(i,7) = XMIN + (XMAX/2-XMIN)*rand;
             pos(i,8) = YMAX/2 + (YMAX-YMAX/2)*rand;
             pos(i,9) = altitude;
             % ---- UAV 4 ----
             pos(i,10) = XMAX/2 + (XMAX-XMAX/2)*rand;
             pos(i,11) = YMAX/2 + (YMAX-YMAX/2)*rand;
             pos(i,12) = altitude;

            % ---- UAV 5 ----
            pos(i,13) = XMIN + (XMAX/4-XMIN)*rand;
            pos(i,14) = YMIN + (YMAX/2-YMIN)*rand;
            pos(i,15) = altitude;
```

```
        % ---- UAV 6 ----
        pos(i,16) = XMAX/4 + (XMAX/2-XMAX/4)*rand;
        pos(i,17) = YMIN + (YMAX/2-YMIN)*rand;
        pos(i,18) = altitude;

        % ---- UAV 7 ----
        pos(i,19) = XMIN + (XMAX/4-XMIN)*rand;
        pos(i,20) = YMAX/2 + (YMAX-YMAX/2)*rand;
        pos(i,21) = altitude;

        % ---- UAV 8 ----
        pos(i,22) = XMAX/4 + (XMAX/2-XMAX/4)*rand;
        pos(i,23) = YMAX/2 + (YMAX-YMAX/2)*rand;
        pos(i,24) = altitude;
    elseif Dim == 36
      %% 12 uav
      % Initialize a matrix to store the random positions
        % ---- UAV 1 ----
        pos(i, 1) = XMIN + (XMAX/3 - XMIN) * rand;
        pos(i, 2) = YMIN + (YMAX/4 - YMIN) * rand;
        pos(i, 3) = altitude;

        % ---- UAV 2 ----
        pos(i, 4) = XMAX/3 + (2*XMAX/3 - XMAX/3) * rand;
        pos(i, 5) = YMIN + (YMAX/4 - YMIN) * rand;
        pos(i, 6) = altitude;

        % ---- UAV 3 ----
        pos(i, 7) = 2*XMAX/3 + (XMAX - 2*XMAX/3) * rand;
        pos(i, 8) = YMIN + (YMAX/4 - YMIN) * rand;
        pos(i, 9) = altitude;

        % ---- UAV 4 ----
        pos(i, 10) = XMIN + (XMAX/3 - XMIN) * rand;
        pos(i, 11) = YMAX/4 + (YMAX/2 - YMAX/4) * rand;
        pos(i, 12) = altitude;

        % ---- UAV 5 ----
        pos(i, 13) = XMAX/3 + (2*XMAX/3 - XMAX/3) * rand;
        pos(i, 14) = YMAX/4 + (YMAX/2 - YMAX/4) * rand;
        pos(i, 15) = altitude;

        % ---- UAV 6 ----
        pos(i, 16) = 2*XMAX/3 + (XMAX - 2*XMAX/3) * rand;
        pos(i, 17) = YMAX/4 + (YMAX/2 - YMAX/4) * rand;
        pos(i, 18) = altitude;

        % ---- UAV 7 ----
        pos(i, 19) = XMIN + (XMAX/3 - XMIN) * rand;
        pos(i, 20) = YMAX/2 + (3*YMAX/4 - YMAX/2) * rand;
        pos(i, 21) = altitude;

        % ---- UAV 8 ----
        pos(i, 22) = XMAX/3 + (2*XMAX/3 - XMAX/3) * rand;
        pos(i, 23) = YMAX/2 + (3*YMAX/4 - YMAX/2) * rand;
        pos(i, 24) = altitude;

        % ---- UAV 9 ----
        pos(i, 25) = 2*XMAX/3 + (XMAX - 2*XMAX/3) * rand;
```

```
            pos(i, 26) = YMAX/2 + (3*YMAX/4 - YMAX/2) * rand;
            pos(i, 27) = altitude;

            % ---- UAV 10 ----
            pos(i, 28) = XMIN + (XMAX/3 - XMIN) * rand;
            pos(i, 29) = 3*YMAX/4 + (YMAX - 3*YMAX/4) * rand;
            pos(i, 30) = altitude;

            % ---- UAV 11 ----
            pos(i, 31) = XMAX/3 + (2*XMAX/3 - XMAX/3) * rand;
            pos(i, 32) = 3*YMAX/4 + (YMAX - 3*YMAX/4) * rand;
            pos(i, 33) = altitude;

            % ---- UAV 12 ----
            pos(i, 34) = 2*XMAX/3 + (XMAX - 2*XMAX/3) * rand;
            pos(i, 35) = 3*YMAX/4 + (YMAX - 3*YMAX/4) * rand;
            pos(i, 36) = altitude;
        end


[fitness,jain,loss_rate,data_rate]=ObjFunction(UAV_num,UE_num,A
_k,Pos_U,pos);
            if fitness>Static_fitness
                Static_pos = pos(i,:);
                Static_fitness = fitness;
                Static_Jain_fairness = jain;
                Static_loss_rate = loss_rate;
                Static_data_rate = data_rate;
            end
    end
end
```

APPENDIX E: Conference Acceptance Notice - IC3INA 2023

[IC3INA 2023] Your paper #1570952604 ('Fairness-Aware Unmanned Aerial Vehicle-Mounted Base Station Placement with Quality of Service Provisioning') Inbox ×

IC3INA 2023 <ic3ina2023-chairs@edas.info>
to me, Shengqi, Kok, Ying, Feng, Zaenal, Arafat, Hayuning, Ira, Ekasari

21 Sept 2023, 10:52 (12 days ago)

Dear Mr. Yen Khai Lim:

Congratulations, Your paper #1570952604 ('Fairness-Aware Unmanned Aerial Vehicle-Mounted Base Station Placement with Quality of Service Provisioning') for IC3INA 2023 has been **accepted** and will be presented in the The 10th International Conference on Computer, Control, Informatics and its Applications (IC3INA) 2023. Some revisions are required for it to be presented in the IC3INA 2023.

Please read the rest of this email carefully to find important information related to revising and presenting your paper.

1. Paper revision:

   a. Please read the reviews for revision. Please reflect the reviewers' comments into your final manuscript. Committee will check whether the revision has been performed or not. Fail to do so, we have a right to exclude your paper from the proceedings.
   Note: Your paper has a similarity score of 21%, it must be at maximum of 20%. Please, revise accordingly.

   b. Strictly follow IEEE style and format. Please find the complete guideline in our website https://conference.brin.go.id/ic3ina2023/ , under the section "Submission"

   c. Send your revision no later than September 22nd, 2023 through your EDAS account.

   d. Please make sure the authors' name appear on EDAS the same with the names on the final manuscripts.

2. Acceptance letter: Formal acceptance letter will be issued after we receive your final manuscript.

3. Presentation: Please note that every paper has to be presented at least by one of the authors. We will notify by email for further details about the online conference program.

4. Registration payment. The payment invoice and further registration details can be seen at the following link: https://bit.ly/registration_ic3ina2023.

We would greatly appreciate it if you would inform us and withdraw your paper as soon as possible in the event that you or a co-author cannot attend the conference to present the paper.

The reviews can be found at link reviews 1570952604.

Blind review 1

Novelty/Originality: Rate the novelty and originality of the ideas or results presented in the paper

The ideas are interesting and the authors show good knowledge on the subjects (4)