

**LEBANESE AMERICAN UNIVERSITY**

3D Deployment of UAVs in Wireless Networks for Traffic  
Offloading and Edge Computing

By

Rania Islambouli

A thesis

Submitted in partial fulfillment of the requirements  
for the degree of Master of Science in Computer Science

School of Arts and Sciences

May 2019

© 2019

Rania Islambouli

All Rights Reserved

## THESIS APPROVAL FORM

Student Name: Rania Islambouli I.D. #: 201403924

Thesis Title: 3D Deployment of UAVs in Wireless Networks for Traffic Offloading and Edge Computing

Program: MS in Computer Science

Department: Computer Science and Mathematics

School: Arts and Sciences

The undersigned certify that they have examined the final electronic copy of this thesis and approved it in Partial Fulfillment of the requirements for the degree of:

Master of Science in the major of Computer Science

Thesis Advisor's Name: Dr. Sanaa Sharafeddine

Signature:  Date: 13 / 05 / 2019  
Day Month Year

Committee Member's Name: Dr. Haidar Harmanani

Signature:  Date: 13 / 05 / 2019  
Day Month Year

Committee Member's Name: Dr. Samer Habre

Signature:  Date: 13 / 05 / 2019  
Day Month Year

## THESIS COPYRIGHT RELEASE FORM

### LEBANESE AMERICAN UNIVERSITY NON-EXCLUSIVE DISTRIBUTION LICENSE

By signing and submitting this license, you (the author(s) or copyright owner) grants the Lebanese American University (LAU) the non-exclusive right to reproduce, translate (as defined below), and/or distribute your submission (including the abstract) worldwide in print and electronic formats and in any medium, including but not limited to audio or video. You agree that LAU may, without changing the content, translate the submission to any medium or format for the purpose of preservation. You also agree that LAU may keep more than one copy of this submission for purposes of security, backup and preservation. You represent that the submission is your original work, and that you have the right to grant the rights contained in this license. You also represent that your submission does not, to the best of your knowledge, infringe upon anyone's copyright. If the submission contains material for which you do not hold copyright, you represent that you have obtained the unrestricted permission of the copyright owner to grant LAU the rights required by this license, and that such third-party owned material is clearly identified and acknowledged within the text or content of the submission. IF THE SUBMISSION IS BASED UPON WORK THAT HAS BEEN SPONSORED OR SUPPORTED BY AN AGENCY OR ORGANIZATION OTHER THAN LAU, YOU REPRESENT THAT YOU HAVE FULFILLED ANY RIGHT OF REVIEW OR OTHER OBLIGATIONS REQUIRED BY SUCH CONTRACT OR AGREEMENT. LAU will clearly identify your name(s) as the author(s) or owner(s) of the submission, and will not make any alteration, other than as allowed by this license, to your submission.

Name: Rania Islambekli

Signature: 

Date: 13 / 05 / 2019

Day Month Year

## PLAGIARISM POLICY COMPLIANCE STATEMENT

I certify that:

1. I have read and understood LAU's Plagiarism Policy.
2. I understand that failure to comply with this Policy can lead to academic and disciplinary actions against me.
3. This work is substantially my own, and to the extent that any part of this work is not my own I have indicated that by acknowledging its sources.

Name: R [REDACTED] al:

Signature: [REDACTED] Date: 13 / 05 / 2019  
Day Month Year

## ACKNOWLEDGMENT

This work would not have been possible without the support of many people.

I would like to express my sincere gratitude to my advisor Dr. Sanaa Sharafeddine for her continuous support, motivation, and immense knowledge. This thesis could not have been completed without her guidance and supervision.

Also, many thanks to my committee members Dr.Haidar Harmanani and Dr. Samer Habre for their insightful comments and encouragement.

Finally, I would like to thank my family and friends for supporting and encouraging me throughout my studies.

# 3D Deployment of UAVs in Wireless Networks for Traffic Offloading and Edge Computing

Rania Islambouli

## ABSTRACT

Unmanned aerial vehicles (UAVs) have recently emerged as enablers for multitude use cases in 5G networks leading to interesting industrial and business applications. 5G networks envision a multi-service network promoting various applications with a distinct set of performance and service demands. In this thesis, we leverage the high flexibility, low-cost, and mobility of UAVs to scale up and improve the efficiency of IoT and mobile networks. We study the utilization of UAVs to increase the capacity and coverage in wireless networks on one side and to extend low computational capabilities and mitigate battery limitations in constrained devices on another side. However, to unlock these promising use cases of UAVs, we address the challenges coupled with UAV utilization mainly 3D deployment and device association.

First, we address the problem of deploying multiple UAVs to act as aerial base stations (ABS) in 3D space while autonomously adapting their positions as users move around within the network. We formulate the problem as a mixed integer program and then propose a novel autonomous positioning approach that can efficiently gear the UAV positions in a way to maintain target quality requirements.

Next, we leverage the mobility and agility of UAVs and use them as mobile edge servers or cloudlets to offer computation offloading opportunities to IoT devices. This being said, computation tasks generated by IoT devices can be processed in less latency and with much lower energy consumption at the devices. To optimally deploy UAVs as mounted cloudlets, we formulate our problem as mixed

integer program and then use an efficient meta-heuristic algorithm to generate optimized results for large scale IoT networks. The simulation results presented in this thesis demonstrate the effectiveness of the proposed solutions and algorithms compared to the optimal solutions and related work in the literature for various network scenario

Keywords: Aerial base station deployment and planning, Drone cells, Traffic offloading, 5G networks, UAV cloudlets, IoT networks, Latency sensitive applications, Edge computing.



# TABLE OF CONTENTS

<b>1. Introduction</b>	<b>1</b>
1.1. Motivation and Background Information . . . . .	1
1.2. Thesis Contribution . . . . .	2
1.3. Thesis Organization . . . . .	3
<b>2. UAVs in 5G Networks and Beyond</b>	<b>4</b>
2.1. The Growing Importance of UAV Networks . . . . .	4
2.2. Features and Characteristics of UAV Networks . . . . .	6
2.2.1. Air-Ground Channel Characteristics . . . . .	8
2.3. Technical and Research Challenges in UAV Networks . . . . .	11
<b>3. 3D Deployment of UAVs as ABS in Wireless Networks for Traffic Offloading</b>	<b>14</b>
3.1. Motivation and Background Information . . . . .	14
3.2. Related Literature . . . . .	18
3.3. System Model . . . . .	21
3.4. Problem Formulation . . . . .	23
3.5. Autonomous Deployment of ABSs in Wireless Networks with User Mobility . . . . .	27
3.5.1. The Need for An Autonomous Algorithm . . . . .	27
3.5.2. Network Changes and User Mobility . . . . .	28
3.5.3. Autonomous Force Algorithm . . . . .	29
3.6. Simulation Results and Performance Analysis . . . . .	33
3.7. Implementation and Testbed . . . . .	37

3.8. Conclusion . . . . .	40
<b>4. Optimized 3D Deployment of UAV-Mounted Cloudlets to Support Latency-Sensitive Services in IoT Networks</b>	<b>42</b>
4.1. IoT Networks . . . . .	42
4.1.1. Overview . . . . .	42
4.1.2. Applications and Use Cases . . . . .	44
4.1.3. Technical and Research Challenges . . . . .	47
4.2. Optimized 3D Deployment of UAV-Mounted Cloudlets . . . . .	50
4.2.1. Motivation and Background Information . . . . .	50
4.2.2. Related literature . . . . .	52
4.2.3. System Model . . . . .	54
4.2.4. Problem Formulation . . . . .	58
4.2.5. Ion Motion Algorithm . . . . .	63
4.2.6. Simulation Results and Performance Analysis . . . . .	67
4.2.7. Conclusion . . . . .	71
<b>5. Conclusion and Future Work</b>	<b>72</b>
<b>Bibliography</b>	<b>73</b>

## LIST OF FIGURES

Figure	Page
1. UAV signal propagation in urban environment. . . . .	9
2. Average bit rate for different ABS heights. . . . .	10
3. ABS deployment in public safety and disaster situations. . . . .	15
4. ABS deployment for enhancing coverage and capacity of terrestrial infrastructure. . . . .	16
5. ABS aided relaying. . . . .	17
6. System model with aerial base stations serving mobile ground users and adapting their positions as the users move to maintain the target performance. . . . .	22
7. Proposed 3D autonomous force deployment algorithm flowchart. .	31
8. Tradeoff between the selected number of ABSs and the network average rate as a function of $\lambda$ . . . . .	34
9. Average number of ABSs required to cover the users for the optimal solution compared to the force approach, as a function of total number of users. . . . .	34
10. Execution time of the optimal solution versus the force approach, as a function of the total number of users. . . . .	35
11. Average bitrate in Mbps for the force approach compared to the optimal solution, as a function of total number of users. . . . .	35
12. Average bitrate in Mbps for the force approach compared to the spiral solution, as a function of time. . . . .	36

13.	Network snapshot showing the initial ABS deployments based on the execution of the force algorithm. Users associated with each ABS are marked using the same color. . . . .	37
14.	Network snapshot after 5 min showing the adaptation of the ABSs positions based on the execution of the force algorithm. Initial ABS locations are marked by small filled squares and final locations are marked by larger filled squares. . . . .	37
15.	Testbed setup. . . . .	38
16.	Messages exchanged between the ground users and the drone. . .	39
17.	Trajectory of a single ABS serving a group of ground users. . . .	40
18.	Trajectory of a single ABS serving a group of ground users after moving around in the area. . . . .	40
19.	General definition of IoT. . . . .	43
20.	UAV-mounted cloudlets in agriculture scenario. . . . .	52
21.	Operation steps of offloading computation tasks to UAV-mounted cloudlets. . . . .	56
22.	Behaviour of ions in liquid state . . . . .	66
23.	Behaviour of ions in solid state . . . . .	66
24.	Ion motion algorithm flowchart. . . . .	68
25.	Average number of UAVs required to serve the IoT devices for the optimal solution compared to the meta-heuristic approach, as function of total number of devices. . . . .	69
26.	Average number of UAVs required to serve the IoT devices for the optimal solution compared to the meta-heuristic approach, as function of total number of devices and in different industry vertices.	70
27.	Average number of UAVs required to serve the IoT devices in different industry verticals, as function of requests rate. . . . .	71
28.	Average number of UAVs and achieved time delay, as function of time deadline in Tactile Internet applications. . . . .	71

## LIST OF TABLES

Table	Page
3.1. System parameters for simulation results . . . . .	33
4.1. Industry verticals time deadlines [1] . . . . .	69

# Chapter One

## Introduction

### 1.1 Motivation and Background Information

Current research efforts are enabling the various and wide applications of unmanned aerial vehicles (UAVs) in different areas and industries including search and rescue, health-care, agriculture, and telecommunications [2]. According to recent reports issued by the red cross and United Nations, UAVs are recognized as highly effective and practical solutions in emergency relief and rescue operations [3]. This is mainly because of the inherent attributes offered by UAVs including wireless connectivity, sensing and surveillance abilities, flexibility , mobility, and computational power. In addition, recent studies conducted by the Federal Aviation Administration (FAA) estimates that the size of drone fleet will be doubled by 2022 and thus reaching 2.4 million vehicles from an approximated 1.1 million in 2017 [4].

Truly the advances in UAV technologies including hardware and software unlocked many future and current applications and enabled low-cost solutions for an extensive number of challenges. UAVs are typically aircrafts that are operated without any human pilot on board and they are also commonly known as drones. The technological advances allowed the development for a wide range of UAVs with different properties and features. Currently, unmanned aerial vehicles are categorized based on their wing type namely fixed or rotary wing and based

on their altitude whether they are considered high or low altitude platforms [5]. Owing to their flexibility, mobility, adjustable altitude, and ability to achieve a high line of sight probability, UAVs nowadays are considered a promising solution in a broad spectrum of telecommunications applications [6].

However, to enable the promising opportunities and applications of UAVs, we should first tackle the technical challenges coupled with deploying UAVs in different environments and scenarios like 3D positioning, path planning, and managing energy consumption [5]. In specific, if correctly deployed, planned and utilized, UAVs can act as key enabling technologies for solving numerous challenges in the telecommunications industry. In this thesis, we leverage the numerous characteristics of UAVs and adopt them as enabling technologies to solve major challenges that emerge in wireless networks and IoT applications.

## 1.2 Thesis Contribution

This thesis addresses the problem of 3D UAV deployment in wireless networks and IoT applications. In specific, we leverage the flexibility, mobility and low-cost characteristics of UAVs to deploy them first as aerial base stations (ABSs) in wireless networks and second as mobile edge computing cloudlets in IoT applications.

ABSs are used in wireless networks to offload traffic, enhance capacity and coverage of existing wireless infrastructure, and to replace damaged or missing terrestrial base stations. To effectively and efficiently deploy ABSs in wireless networks. We initially formulate the problem as a mixed integer program (MIP) and then we propose an efficient and lightweight solution based on electrostatic forces that are capable to adapt to network changes and user mobility. We also tested our proposed algorithm on a commercial drone and using a testbed setup.

On the other hand, UAVs provide a promising solution that enables a plethora of IoT services. In this thesis, we tackle the problem of energy consumption and limited power in IoT devices by deploying UAVs as mounted cloudlets to serve

IoT devices in an effective and orderly fashion. We initially formulate the problem as MIP and then we propose an effective meta-heuristic algorithm that is based on the behavior of ions in different states and conditions.

This thesis work resulted in three manuscripts, one journal article [7], one conference proceeding [8], and another journal article to be submitted for possible publication.

### **1.3 Thesis Organization**

The rest of this thesis is organized as follows. In Chapter 2 we discuss the main applications, use cases, characteristics, and challenges of deploying UAVs in wireless networks. Then, in Chapter 3 we study the 3D deployment of ABSs in wireless networks for traffic offloading and we propose robust, reliable, and efficient algorithms. Chapter 4 deals with deploying UAV-mounted cloudlets in IoT environments where we utilize UAVs to support latency-sensitive services in IoT networks. Finally, in Chapter 5 we conclude and consider possible future work and extensions.



# Chapter Two

## UAVs in 5G Networks and Beyond

### 2.1 The Growing Importance of UAV Networks

The rapid technological advancements in mechanical systems, sensors, electronics, and embedded systems have paved the way for the fast, low cost and reliable production of unmanned aerial vehicles (UAVs). These UAVs can fly in an autonomous way or can even be operated from a distance without the need of any human staff to be on board [9]. Due to their low cost, high mobility, flexibility, and simple installation, UAVs have been utilized for a plethora of applications throughout the past few years [10]. First, UAVs were used in military applications and for surveillance purposes. However, with the continuous advancement in technology and the huge reduction in costs now, UAVs are accessible to the public and are used in a wide range of civil, commercial, and research applications. Advanced and concerted research in this area enabled a huge number of applications including telecommunications, rescue missions, traffic monitoring, weather tracking and relay for ad hoc networks [11] [12].

Amid the numerous applications powered by UAVs, the role of UAVs in achieving wireless communications is becoming more significant and vital in future communication systems [2]. Specifically, deploying UAVs in a correct and op-

timal way can present robust and flexible wireless communication solutions for a huge number of applications [13]. Indeed, the value of solutions that utilizes UAVs in telecommunications applications is measured to be approximately 6.3 billion USD [14].

On an industrial level, there exists numerous projects that have been studied and implemented to deploy and utilize UAVs for wireless connectivity. Examples include:

1. Google Loon project [15]:

This project aims to provide connectivity to rural areas where no existing infrastructure exists. This is done by deploying air balloons that are in charge of relaying the radio communications to the terrestrial infrastructure.

2. Facebook Aquila project [16] :

The principal aim of this project is to provide internet coverage in urban areas. Unmanned aircrafts are deployed in the sky and can rise up to a 20 km altitude. The aircrafts are called Aquila and are self-powered by the use of solar panels installed on their wings.

3. Nokia F-Cell project [17]:

This project focuses on reducing the cost incurred upon the installation of many small cellular cells. F-cells are small drones that are powered by light energy.

4. Eurecome Perfume [6]:

This project studies the use of UAVs as relays to support existing infrastructure. This prototype implemented and tested a machine learning algorithm that is intelligent enough to deploy UAVs in wireless networks for the aim of relaying connections from the ground users to the existing base stations.

5. Huawei Digital Sky [18]:

The Huawei Wireless X Lab started in 2017 to work on a project that strives to test different trials of various use cases enabled by UAVs. After that, the Shanghai city created a huge ecosystem that integrates multiple participants including mobile operators, software firms, hardware manufacturers, and public institutions to work on possible UAV applications.

## 2.2 Features and Characteristics of UAV Networks

Currently, UAVs are available in multiple forms that can be essentially categorized into two main types, large and small UAVs [19]. Large UAVs can be used alone in a critical mission or for a specific use case. Small UAVs can be deployed as swarms where multiple UAVs are utilized to provide a service or execute a mission. Currently, small UAVs are being heavily used in civilian and military applications like search operations, managing wildfire, and communication relaying [20]. In addition, the boost and improvement in sensor, surveillance, and electronics technology paved the way for a various set of UAV applications including localization and remote sensing [21].

Because of the nature of UAVs, certain characteristics distinctive to aerial networks emerge that differ from regular terrestrial networks. Indeed, the use of UAVs enables us to exploit the many features not regularly available in ground base stations.

Thus by using UAVs in telecommunication applications, we benefit from lower costs and better quality of services due to unique features granted by UAVs. The diverse features are detailed below [21] [22] [5]:

- Fast to Deploy:

Compared to terrestrial and ground base station UAVs are faster and easier to deploy. They can provide on-demand services and coverage in a short period of time without the need for extensive planning and preparation.

- Mobile and Flexible:

Due to their mobility UAVs can alter their locations according to the changing demands and sudden network changes. In this way, UAVs can cope with user mobility, weather shifts, and possible network failures.

- Unrestricted Deployment Locations:

Due to their nature, UAVs can be installed in any environment. UAVs can be used in places where there is no infrastructure like rural and remote areas. They can be utilized in disaster and emergency scenarios.

- Enhanced Line of Sight(LoS):

UAVs can adjust their heights to avoid obstacles and thus enhancing LoS. Since the pathloss depend on all distance, height, and elevation angle better communication channels can be achieved by adjusting the 3D position and coping with environmental characteristics.

- Scalable:

UAVs can be deployed based on demand and according to service requirements and number of users. Thus, it is possible to deploy extra UAVs in case of increased demand. UAV networks can easily adjust according to the size of the network, amount of demand, and distribution of users.

- Varying associations:

In UAV networks user associations with UAVs are not static nor restricted. Users can connect to the UAV providing a better quality of service. Users can move and change connections easily. The network load can be better distributed among available UAVs to enable a satisfying and enhanced experience for everybody.

### 2.2.1 Air-Ground Channel Characteristics

As discussed earlier, the deployment of UAVs in wireless networks has been gaining a lot of attention. These vehicles are being used for extending communications and increasing coverage. Furthermore, they are employed to conduct critical missions and collect sensitive data. These applications deploy UAVs at different altitudes and in 3D spaces where UAVs are remotely operated or fully autonomous. Thus, in order to ensure high reliability and good quality of service, it is greatly essential to accurately characterize the air-ground channels. Indeed, many research and work have been done to stabilize a unique and correct UAV channel model. For example, a specialized committee has been established by the Radio Technical Commission for Aeronautics (RTCA) in 2013 to model and specify the different performance characteristics and standards of UAV channels [23].

Channel models and radio transmissions in terrestrial wireless systems have been extensively studied and are considered well defined and stabilized. However, these models are not consistent with UAV networks due to the unique characteristics of UAV channels that are based on altitude changes, airframe shadowing, and temporal and spatial variations. As depicted in Fig. 1, radio signals emitted by a UAV propagate in free space until reaching the urban environment where they incur shadowing and scattering caused by the man-made structures. Thus, the pathloss model for UAV base stations should reflect the environmental factors that effects the LoS probabilities and should consider the UAV position with respect to the ground users.

Indeed, accurate channel characterization is essential for the performance optimization and design of efficient UAV communication systems. A lot of work has been done to model the air-ground channel between UAVs and ground users [24]. In this thesis, we consider the channel model represented in [25] since it models the air-ground channel by considering multiple factors that might affect the channel status and not just the distance also, it reflects the unique characteristics of non-stationary UAV channels that is usually not reflected in other models.

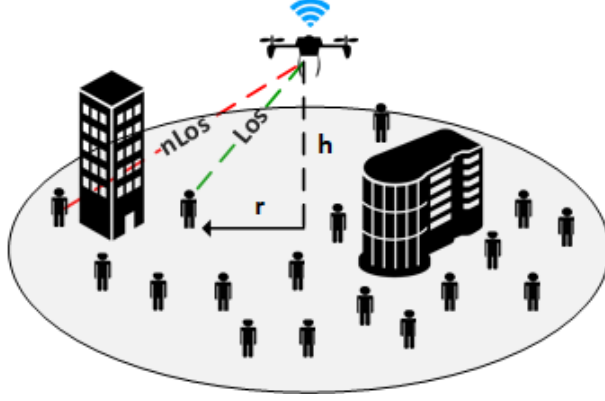


Figure 1: UAV signal propagation in urban environment.

This pathloss model depends on environmental parameters, while considering the UAV altitude, elevation angle from the ground, distance between the UAV and the ground user and angle formed with respect to the served user. By considering multiple factors and focusing on UAV specific characteristics this model can provide coverage analysis for optimal UAV positions. It is an analytical model that depends on the height and angle formed with respect to the served user, resulting in the following line of sight (LoS) probability:

$$p_{LoS} = \frac{1}{1 + \mu * \exp(-\beta (\arctan(\frac{h}{r}) - \mu))}, \quad (2.1)$$

where  $h$  and  $r$  are the height and the horizontal distance between the UAV and the ground user, respectively and are represented in Fig. 1. In addition,  $\mu$  and  $\beta$  are constants that depend on the environment.

The channel model between a UAV and a ground user can then be modeled as follows:

$$PL_{i,j} = \frac{P_T}{P_R} = p_{LoS} \eta_{LoS} \left( \frac{4\pi f_c d_{i,j}}{c} \right)^\alpha + p_{NLoS} \eta_{NLoS} \left( \frac{4\pi f_c d_{i,j}}{c} \right)^\alpha, \quad (2.2)$$

where  $f_c$  is the carrier frequency in Hz,  $c$  is the speed of light in m/s,  $\alpha$  is the pathloss exponent,  $\eta_{LoS}$  and  $\eta_{NLoS}$  are, respectively, the losses corresponding to LoS and non-LoS connections depending on the environment and  $d_{i,j}$  is the

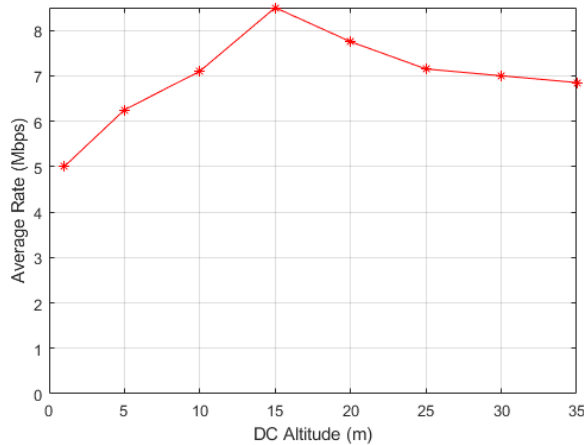


Figure 2: Average bit rate for different ABS heights.

distance between UAV  $i$  and ground user  $j$ .

This model captures the different unique characteristics of the channel formed between the UAV and ground users. The pathloss model depends not only on distance but on all height, horizontal distance, elevation angle, and LoS probabilities. This model was utilized in all of our simulations and algorithms represented in this thesis. To further explain the tradeoffs offered by this model we plot in Fig. 2 the average bit rate of ground users when deploying a single UAV on multiple heights while considering the system parameters represented in Table 3.1. We can clearly see that the altitude at which a UAV is installed at greatly influences the received user bit rate. At low heights the achieved user bit rate was increasing until reaching a maximum of approximately 8.25 Mbps at a height of 15 meter. This is due to the LoS achieved at higher heights where at low height LoS is hard to achieve due to obstacles and high density of objects. After that, the rate started to decrease. This is because of the high distance between the ground users and the UAV. Thus, a clear trade-off exists between distance and height where height increases LoS probability but also increases the transmission distance. Hence, an optimal height is where a good line of sight is achieved but with an acceptable transmission distance.

## 2.3 Technical and Research Challenges in UAV Networks

The use of UAVs in communication networks can enable a wide range of applications and solve many problems. In comparison with the regular terrestrial station, UAVs are more flexible and mobile, faster to deploy, and might result in enhanced communication quality due to better LoS probabilities. However, to enable this spectrum of applications we should consider all the challenges incurred upon the use of UAVs in communication networks. Truly, the use of UAVs in communication networks introduces many challenges that should be considered for effective use of UAVs in different applications and scenarios. Some of the key challenges are discussed below [22] [5] [26] [6].

- Optimizing UAV deployment position : One of the main difficulties in enabling UAV applications in telecommunication is their optimal deployment in 3D spaces. Truly, the mobility of UAVs and adjustable altitude enables them to move freely and achieve better communication links. However, this introduces the challenge of optimally deploying UAVs in communication networks compared to deploying terrestrial networks that are usually static and immobile. The deployment task is particularly challenging since it depends on environmental and geographical factors, interference, user locations, and channel modeling. Furthermore, deploying more than one UAV in the same network makes this task much more challenging since we should also study resource allocation, interference between multiple UAVs and distances separating the deployed UAVs.
- Optimizing the trajectory of UAVs: In UAV-aided communications it is necessary to choose a path and route that shortens the distance crossed while serving all required users and maintaining a good quality of service. However, path planning is a challenging task in UAV network due to capacity, energy, QoS, and environmental constraints.



- Deciding on number of UAVs: Deciding on the number of UAVs to install or utilize in a certain scenario is another challenge. The number of UAVs should be sufficient enough to serve all users within the acceptable QoS requirements. This also depends on the available resources and distribution of ground users.
- Enabling energy-efficient deployment of UAVs: Despite, the huge progress in technology and specifically energy storage, the performance and use of UAVs are still constrained by the limited power and energy available on-board. Thus, it is necessary to consider energy efficient deployment and operation of UAVs.
- Recharging of UAVs: In addition to reducing the energy consumption another way to extend the short lifetime of UAVs is achieved by deploying recharging stations. However, it is crucial to deploy these stations in optimal positions to make the recharging process efficient and fast. Hence, another major challenge is the optimal deployment of recharging stations and the design of an efficient process for replacing exhausted UAVs with new charged UAVs.
- Ensuring safety and a collision free Environment: Since UAVs are mobile they can move freely and change their positions to adjust with network changes. However, this calls for the need for safety-critical functions since UAVs may collide with each other. Trajectories taken by UAVs should never meet and paths should be planned accordingly. On the other hand, some UAVs may encounter hardware or software errors that might cause emergency landing or even crashing. Thus, it is extremely vital to ensure the safety of ground users throughout the whole service period.
- Mitigating interference: Due to their mobility, managing interference in UAV networks is much harder than in regular terrestrial networks. Interference with neighboring UAVs should be well studied and reduced. In

addition, it is necessary to ensure that neighboring UAVs and peer connections do not interfere with backhaul links. Thus, it is necessary to establish a robust interference management system.

- Ensuring security and privacy: Since 2007, the number of recorded cyber-attacks on drones has been tragically increasing [27]. An attack does not only cause loss in connection but, might also lead to sudden crashes and threatening of lives. Since UAVs now are publicly used and easily accessible it is necessary to ensure a private and completely secure environment when deploying UAVs for telecommunication applications.

# Chapter Three

## 3D Deployment of UAVs as ABS in Wireless Networks for Traffic Offloading

### 3.1 Motivation and Background Information

Unmanned aerial vehicles (UAVs) have recently attracted various industry verticals to enable and create new services and markets. Integrating UAVs into 5G networks brings forward many use cases in search and rescue, disaster management, V2X infotainment services, data gathering in Internet of Things (IoT), and many others. Due to their mobility and ability to cover unreachable sites, UAVs are gaining increased interest in deploying them as aerial base stations to provide wireless connection in high demand and rural areas [28]. Employing UAVs as aerial base stations is strongly effective in situations where large obstacles degrade the quality of wireless links between users and ground base stations and is considered an effective solution to improve the network quality and capacity by offloading traffic from the ground base station in dense locations [22] [29]. Unlike terrestrial base stations, aerial base stations (ABSs) can be dynamically deployed and can also adjust their positions to support mobile users with different quality of service (QoS) requirements [30].

Indeed, UAV-mounted aerial base stations acting as ABSs have numerous applications in 5G networks. Below, we discuss some of the major applications for the use of ABSs [31] [29].

- Disasters and Public Safety:

Natural disasters are often hard to avoid and stop and thus it is important to quickly act to reduce the number of losses in possessions and lives and to obviate any extra damages. Disasters like earthquakes, floods, fires, and extreme weather conditions often yield devastating consequences in various environments. When these unanticipated events occur, the available base stations could be destroyed or disabled. Base stations might also become overloaded and not able to handle the increased number of requests as indicated in a study of cellular networks during disasters in [32]. Hence, it is crucial to find an effective and fast alternative to secure robust communication between victims like injured or trapped people and first responders. Thus, the use of on-demand ABSs is crucial in these situations where ensuring a reliable connection does not only extend the available communication network but also saves lives. As shown in Fig. 3 a disaster might result in huge fires and flames that can hinder the communication with terrestrial base stations. In this scenario, a fleet of 3 ABSs was launched to replace the existing infrastructure and secure a reliable connection.



Figure 3: ABS deployment in public safety and disaster situations.

- Coverage and Capacity Enhancement in 5G Networks:

The demand for reliable and low latency wireless connection is increasing drastically due to the fast propagation of greatly intelligent devices and IoT applications. When high demand suddenly arises the existing wireless infrastructure might not be able to withstand the high demand causing uncovered areas. As such, introducing new technologies and solutions to overcome such challenges is necessary. Actually, UAVs are being envisioned as an essential solution for complementing existing networks in heterogeneous environments and thus overcoming the many challenges caused by high connection demands. For example, in concerts, large crowds might gather in a single location while requesting access to wireless connection to stream live content, download data, or use online applications. In these cases, ABSs can be deployed in overloaded areas to extend the coverage of existing terrestrial base stations. As shown on the left in Fig. 4 an overloaded base station is being complemented by an ABS to extend the capacity and coverage capabilities. In addition, ABSs can be used whenever a base station is inactive due to hardware or software errors and thus providing a fast and reliable solution while working on fixing and repairing the damage in the terrestrial infrastructure. Also, ABSs can be used in rural and remote places where there is no existing infrastructure and thus providing a low cost and on-demand connection.

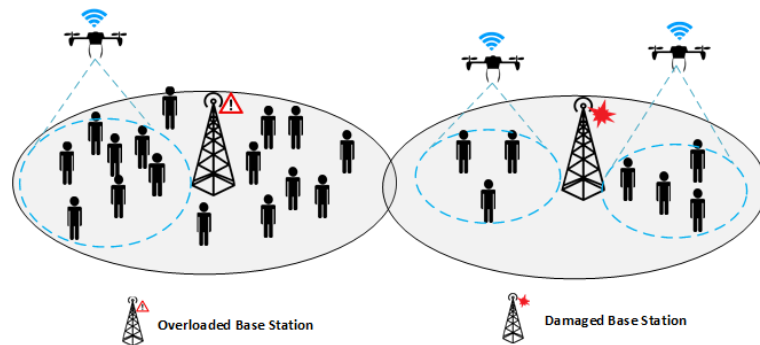


Figure 4: ABS deployment for enhancing coverage and capacity of terrestrial infrastructure.

- Relaying in Wireless Networks using ABSs:

ABSs can be also used as relays in wireless networks when large obstacles hinder the communication between a ground user and a base station or another ground user. In case of large objects affecting the wave propagation and diminishing the power of received signal strength, ABSs are deployed to act as relays between users and existing terrestrial base stations. The use of ABSs in wireless networks and environments can actually mitigate the effects of shadowing and large obstacles affecting coverage. For example, in Fig. 5 a large natural mountain is affecting the communication between some ground users and the base station. This is solved by deploying an ABS that acts as a relay between the users on the left and the terrestrial base station on the right. In this way users that are far from the coverage area of the base station are covered and served. This helps in alleviating the effects of natural obstacles like mountains and rocks and human made obstacles like buildings and towers

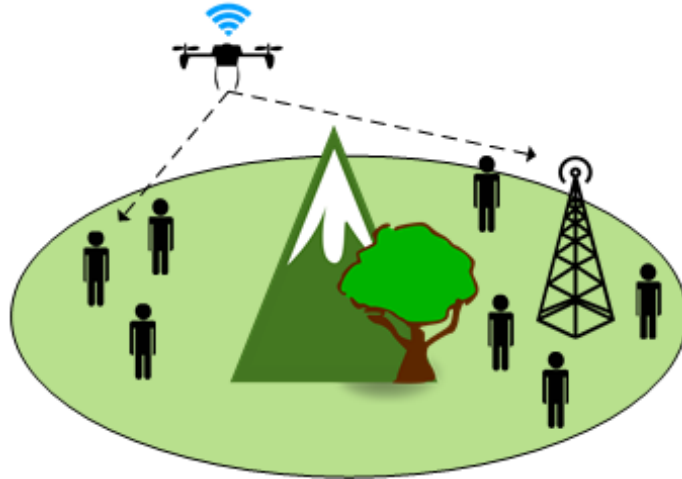


Figure 5: ABS aided relaying.

In this chapter, we study the deployment of UAVs to act as aerial base stations in different scenarios and environments like disasters or increase in connection outage. By using UAVs as ABSs we can benefit from their low cost, flexibility,

mobility, and fast deployment. However, utilizing UAVs as ABS comes with major technical challenges that include determining the number of needed UAVs and their locations in a 3D search space while taking into account practical performance aspects such as cost, complexity, and QoS. Thus, in this chapter we study the 3D deployment of ABSs in wireless networks and more specifically we leverage the mobility and flexibility of UAVs to deploy them in dynamic networks with user mobility. We first formulate the problem as mixed integer problem (MIP) and then we propose an efficient and lightweight solution based on electrostatic forces that is capable to adapt to network changes and user mobility.

The remainder of this chapter is organized as follows. In Section 3.2, we cover the related literature and highlight the main contributions of this work. Section 3.3 presents the system model and its key components. Section 3.4 formulate the problem of 3D deployment of multiple UAVs in wireless systems as a MIP. In Section 3.5.1 and 3.5.2 we motivate the need for an autonomous algorithm that considers user mobility. Section 3.5.3 describes the proposed low complexity autonomous algorithm for deploying multiple UAVs in wireless systems with user mobility. Section 3.6 presents performance results for various scenarios and highlights the effectiveness of the proposed algorithm. In Section 3.7 we implement the force algorithm on a single drone using a testbed setup. Finally, conclusions are drawn in section 3.8.

## 3.2 Related Literature

Previous work available in the literature tackled either the problem of optimizing the position of a number of ABSs [33] [34] [35] [36] [37] [38] [39] or planning the trajectory of a single ABS where the start point, trajectory path, and end point are optimized [40] [41] [42] [43]. A large number of work exists that study the positioning of static ABSs within wireless networks. In these static wireless environments, the horizontal location and altitude of an ABS are jointly or individually optimized while accommodating for multiple QoS requirements. Some work

focused on only optimizing the altitude at which UAVs are deployed. The authors in [33] worked on optimizing the altitude of one UAV to accomplish a maximum coverage radius. They studied the height in terms of the maximum pathloss and different environmental parameters. Similarly, in [34] the optimal altitude of a UAV was derived while maximizing the coverage area and minimizing the needed transmit power. The authors extended their work by considering a scenario where two UAVs are being deployed. They studied two cases an interference-free and a full interference scenario.

Other work focused on solely optimizing the 2D position. The work in [35] optimized the horizontal position of multiple ABSs with the goal of minimizing the total number of deployed UAVs. The authors suggested a polynomial time spiral algorithm to position the UAVs where ABSs are first placed on the perimeter to cover the maximum number of users and then distributed along a spiral route towards the center till successfully covering all users in the area. The work in [36] studied the problems associated with the use of UAVs as wireless base stations in emergency situations. Initially, the authors considered the scenario where all UAVs are launched from the same position and proposed a polynomial time algorithm. Then, this was extended to include the more general case where UAVs are launched from various unique locations. They first modeled a dynamic program to formulate the problem and then solved it by using a pseudo-polynomial time solution.

Few works tackled the UAV positioning problem in 3D space. In [37], the deployment of one UAV as an ABS was studied as a 3-D circle placement problem. The horizontal and vertical placement of the UAV was decoupled to simplify the problem and then formulated as integer problem while maximizing the ground user coverage probability. Similarly, the authors in [38] investigated the 3D placement of a single ABS where the goal is to cover the highest number of users. First, they represented the problem as a multiple circles placement problem. Then, they propose a solution based on exhaustive search (ES) where the optimal height is



searched for within a bounded interval. To further reduce the complexity, they suggested a weighted area algorithm that produced close to the ES simulation results. In [39] the authors utilized geometrical relaxation and clustering methods to deploy multiple ABSs. The problem of deploying multiple UAVs was modeled as a mixed integer problem with non convex constraints and then solved using a k-means clustering method that takes a certain area and separates into k divisions. Finally, a robust procedure was applied to compensate for losses due to inaccurate user location information.

On the other hand, some work focused on optimizing the ABS trajectory while considering multiple QoS requirements. In [40] the authors considered an ABS that is dispatched to cover and serve the maximum number of users before exhausting all its energy resources. The problem of jointly optimizing the trajectory, scheduling, and user associations were studied. First, the trajectory optimization problem was modeled as a mixed integer linear problem. Then an efficient iterative solution that divides the problem into multiple sub-problems was suggested. The solution was further enhanced to account for inaccurate user location information where two techniques were introduced to tackle this problem. Also, in [41] a Q-learning technique was applied for planning the path of one ABS while maximizing the received rate by the group of ground users. The ABS was treated as an autonomous agent that takes movement decision without any knowledge of the environment.

In [42] the trajectory of multiple UAVs assisting in wireless networks was studied. The authors suggested a three-step approach to design the trajectory of the UAVs while considering user mobility. In the first step a Q-learning technique is utilized to deploy the UAVs according to the available users' positions. After that, they implemented an algorithm based on echo state networks in order to predict the positions of the ground users in the future. Finally, a Q-learning algorithm is applied to figure the deployment coordinates of each UAV within each time slot. The authors in [43] considered a dynamic environment where users

are moving and changing the network environment. Reinforcement learning was utilized to adjust the position of a single ABS that supports multiple terrestrial base stations. A Q-learning technique was applied to position the ABS that was shown to be successful in compensating for QoS lost due to user mobility.

The aforementioned research contributions considered the deployment of UAVs mostly in a 2D plane, with more recent work studying the deployment of a number of UAVs in a 3D space. In addition, most work considered scenarios where users are fixed or analyzed the deployment of UAVs depending on a network snapshot within a specific time slot. In this work, we leverage the mobility and flexibility of UAVs to deploy them in dynamic networks with user mobility. The main contribution of this work is to study the 3D autonomous deployment of UAVs in wireless networks where user mobility is taken into account. We propose an efficient and lightweight solution based on electrostatic forces that is capable to adapt to network changes.

### 3.3 System Model

We consider a wireless communication system where a fleet of UAV-mounted base stations, denoted as aerial base stations (ABSs), is deployed to serve a number of ground users in a specific area, where the terrestrial infrastructure was destroyed or had not been installed. These UAVs can dynamically move, when needed, to effectively serve the mobile users. Our objective is to identify the minimum number of ABSs required and their optimized 3D locations to provide efficient connectivity to the set of ground users.

With a 3D Cartesian coordinate system, we denote by  $c_i = (x_i, y_i, z_i)$  the three-dimensional coordinate of each ABS  $i$  where  $x_i$  and  $y_i$  represent the horizontal position and  $z_i$  represents the altitude. We consider a downlink scenario in which each ABS is equipped with a directional antenna.  $K_T$  represents the total number of users in the area and  $N$  is the number of needed ABSs out of a maximum of  $N_D$  available ABSs.

We adopt the channel model suggested in [25] which depends on the height and angle formed with respect to the served user. This channel model was discussed in details in Chapter 2. Our system model discussed in this section is depicted in Fig. 6. In this figure two ABSs are deployed to serve the ground users. Users are associated to a single UAV for example the blue colored users are associated with the ABS with blue colored signal and the orange colored users are associate with the ABS with the orange colored signal. When users move, ABSs adapt to the network and environment changes. For example, the first UAV with the orange colored signal had to update its position and move to the right due to the network changes and user movement.

Thus, our goal in this work is to decide first on the number of required ABSs

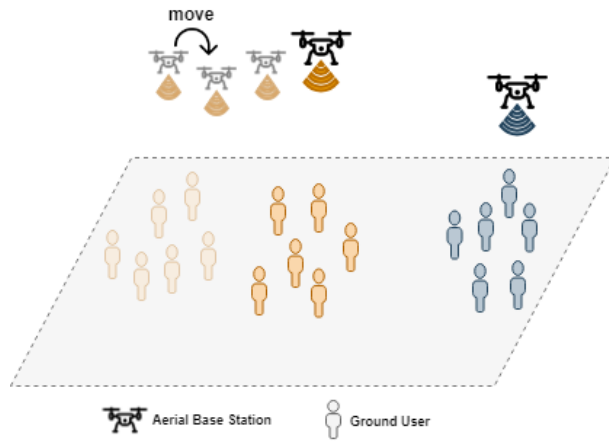


Figure 6: System model with aerial base stations serving mobile ground users and adapting their positions as the users move to maintain the target performance.

to serve the ground users and on their initial 3D positions. Then, we aim to adopt an efficient model that enables the dynamic and adaptive deployment of ABSs where an ABS accommodate network and user changes. Thus, each ABS should be able to adjust its position according to the status of the users and their positions. In the next section we formulate our problem as a MIP and then we explain our autonomous and mobile algorithm.

### 3.4 Problem Formulation

This work mainly aims to determine the location and number of ABSs to be deployed to serve ground users. In order to minimize the cost we aim to deploy the minimum number of ABSs to serve the users with a specific outage probability. In this section we formulate the problem as a mixed integer programming optimization problem.

Our main goals are 1) to minimize the deployment cost by minimizing the required number of UAVs and 2) to intelligently position the ABSs in such a way that maximizes the overall quality of service measured by the received power for all users. First, we introduce a decision variable  $u_i$  to indicate whether an ABS  $i$  will be deployed or not.

$$u_i = \begin{cases} 1, & \text{if ABS } i \text{ is deployed} \\ 0, & \text{otherwise} \end{cases} \quad (3.1)$$

We also define a  $K_T(N_D + 1)$  association matrix  $\mathbf{A}$  where  $a_{ki}$  is a binary value indicating whether or not user  $k$  is associated with ABS  $i$ .

$$a_{ki} = \begin{cases} 1, & \text{if user } k \text{ is associated to ABS } i \\ 0, & \text{otherwise} \end{cases} \quad (3.2)$$

We assume a user is in the coverage region of ABS if the power received satisfies the QoS requirement. For a given transmission power of the ABS, let  $\Gamma$  represent the signal to noise ration (SNR) threshold corresponding to the QoS requirement. That is user  $k$  is covered by ABS  $i$  if

$$SNR_k = \frac{P_{k,i}}{\sigma^2} \geq \Gamma, \quad (3.3)$$

where  $P_{k,i}$  is the received power and  $\sigma^2$  is the thermal noise. By utilizing the channel model in (2.2), we can calculate the received power for each user covered

by an ABS. The power budget set to each user by an ABS is expressed as  $\frac{P_D}{K_D}$  where  $P_D$  is the maximum transmit power of the ABS and  $k_D$  is the maximum capacity of the ABS.

Based on the above, the problem is formulated as follows:

$$\min_{d,A} \sum_{i=1}^{N_D} \sum_{k=1}^{K_T} \lambda u_i - (1 - \lambda) \frac{P_{k,i}}{P_T} a_{k,i} \quad (3.4)$$

$$\text{subject to } a_{k,i} \leq u_i \quad \forall i, \forall k \quad (3.5)$$

$$\sum_{i=1}^{N_D} a_{k,i} \leq 1 \quad \forall k \quad (3.6)$$

$$\sum_{k=1}^{K_T} a_{k,i} \leq K_D \quad \forall i \in [1, N_D] \quad (3.7)$$

$$\sum_{k=1}^{K_T} \sum_{i=1}^{N_D} a_{k,i} \geq (1 - \beta) K_T \quad (3.8)$$

$$P_{k,i} \geq \sigma^2 \Gamma a_{k,i} \quad \forall i, \forall k \quad (3.9)$$

$$\begin{aligned} & \sum_{j=1}^{N_D} \sum_{k=1}^{N_D} \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2 + (z_j - z_k)^2} \\ & \geq \theta \quad \forall i, j \in [1, N_D] \end{aligned} \quad (3.10)$$

The objective function represented in equation (3.4) minimizes the deployment cost by minimizing the number of ABSs deployed and placing them over high demand areas. It also improves the network quality by maximizing the received power for each user. The first part in equation (3.4) represented by  $u_i$  is responsible for minimizing the number of deployed ABSs. The second part accounts for the total received power normalized by the transmit power  $P_T$ . Since our objective function consists of two components, we introduce a new parameter  $\lambda$  to balance the need between maximizing the network performance quality and minimizing the number of deployed ABSs. This parameter is configurable by the network operator to either favor minimizing the number of ABSs or maximizing the total received power.

The first constraint represented by equation (3.5) ensures that a wireless node will be only linked to an available ABS. Thus, a user is not associated with an ABS if it is not picked to be deployed in the network. Users are only associated with deployed ABSs and a user associated with an ABSs  $i$  means that this ground user is connected and served by this ABS  $i$ . So, if ABS  $i$  is not deployed then, no users could be associated to it. In this case,  $u_i$  is set to be zero and users should not be associated to it. When  $u_i$  is set to zero the variable  $a_{k,i}$  should be also set to zero. Indeed, this is achieved by the inequality in equation (3.5) that forces the value of  $a_{k,i}$  to be always less than or equal to  $u_i$ . On the other hand, if ABS  $i$  is deployed then a user  $j$  can be associated with ABS  $i$ . In this case,  $u_i$  is set to one and  $a_{k,i}$  could be set to one or zero indicating whether a user is associated with ABS  $i$  or not. Also, this is achieved by the inequality in (3.5) where  $a_{k,i}$  could be set to zero or one.

Constraint two defined in equation (3.6) states that each wireless node should be served by a single ABS. A covered user must only be connected and served by one ABS and not multiple ABSs at the same time. To guarantee this we introduce equation (3.6). In this equation we are ensuring that each ground user is covered by a single unique UAV by forcing the association matrix of a single ground user to sum to a number less than or equal to one. Setting this summation to one implies that user  $j$  is associated and covered by exactly one ABS. Setting it to zero implies that a user  $j$  is not covered by any ABS whether deployed or not.

Constraint three defined in equation (3.7) guarantees that the number of users linked to each ABS does not exceed the defined capacities where  $K_D$  represents the maximum number of users that can be served by a single ABS. Each ABS can withstand a specific number of connections at the same time thus, it is critical to ensure that number of connection per ABSs is less than or equal to the maximum number of allowed connections. This is achieved by the inequality represented by equation (3.7). In this equation the association matrix of each single ABS is summed up and forced to be less than or equal to the capacity of each ABS. Here,

we consider that all ABSs being deployed and utilized have the same capacity  $K_D$ .

The fourth constraint in (3.8) ensures that the number of uncovered users is not more than the allowed outage ratio denoted by  $\beta$ . So, the deployed ABSs should be able to cover the ground users while leaving a small percentage uncovered. This is particularly important in some cases where outliers exist in the environment.  $\beta$  represents the allowed outage ratio out of the whole number of users. For example, if the number of available ground users is 100 and  $\beta$  is equal to 0.1 then, this indicates that ten percent of the users could be not covered in our solution. In this case, equation (3.8) forces the summation of each ABS association matrix to be set to a value equal to or greater than  $(1 - 0.1)K_D = 0.9K_D$

The fifth constraint in (3.9) ) ensures a minimum SNR threshold for a ground user to be served by an ABS  $i$ . This constraint is particularly responsible of ensuring a minimum QoS. Users should be served by an ABS where a good quality connection could be established to ensure a pleasant experience and good service. The quality of connection depends on the SNR that depend in its turn on the distance between an ABS and ground user, established LoS and received power.

Finally, the last constraint in (3.10) enforces a safety distance between any pair of deployed ABSs. The distance between any two ABSs is calculated according to the Euclidean distance where both horizontal and vertical positions are considered and reflected. In fact, it is extremely necessary to ensure a minimum distance between any two ABSs to avoid any sudden crashes or damage that might not only cause connection disruption but also threatens the safety and lives of ground users. A minimum distance also decreases the interference between UAVs and ensures a well maintained and planned network.

Some of the constraints represented above are non linear in their nature. Thus, we use the Taylor series linearization to handle the non-linearity in (3.9) and (3.10). First, we start by an initial approximation for all variables. Second,

we solve for the first order derivatives. Third we calculate the revised values of all variables by referring to the first order derivatives ( $x = x_0 + dx$ ). Using these newly revised values, we repeat the second and third step. This procedure continues until the calculated first order derivatives are small enough to bring all variables to an acceptable accuracy; hence, the solution converges.

The problem proposed above is a mixed integer programming problem that is very hard to solve. Specifically, our problem can be easily coupled with the continuous capacitated facility location (CCPL) problem that is a well known NP-hard problem [44] [45]. The main objective of this problem is to find the positions of  $F$  facilities that can handle each a capacity equal to  $X$  while optimizing the distances from the facilities to the connected nodes. The CCPL problem can be mapped to our problem if we set  $K_D$  to  $X$ , and  $\beta \Gamma$  to 0. Thus, the problem discussed in this section is also NP-hard and it is really very difficult to solve the formulated optimization problem in linear time for large test cases.

## **3.5 Autonomous Deployment of ABSs in Wireless Networks with User Mobility**

This section lays down the importance of an autonomous and mobile algorithm. We also shed light on the main differences between regular and autonomous algorithms. Then, we discuss the value of considering user mobility in our algorithm. Finally, we suggest an autonomous algorithm for deploying ABSs in wireless networks with user mobility.

### **3.5.1 The Need for An Autonomous Algorithm**

Most of the work available in the literature and discussed in Section 3.2 assumes the knowledge of exact user locations and coordinates. This is particularly challenging when it comes to the actual implementation of the solution since user locations are hard to record and track. The previous knowledge about user lo-



cations is not logical when implementing the solution in different environments due to the expensive and hard task that is needed to be done before running the algorithm in order to get user locations. In addition, recording exact and accurate user locations is not always achievable.

Indeed, collecting user locations requires a dedicated algorithm and protocol [46]. For example, users might have to send their own locations to the ABSs or to a centralized server. In both cases, this incurs extra messaging and might cause overhead in the network and thus would affect the quality of the available connection. It might also require extra centralized servers and thus dedicated hardware and increased costs.

Another way to track user locations is by actively locating them using available base stations or deployed ABSs. This would add further load on UAVs given that localization algorithms are heavy and still under research [47] [48] [49]. On the other hand, actively localizing users is not very accurate and might be challenging in certain environments and conditions [50]. In addition, running the localization algorithms on the ABSs might lead to high energy consumption and quick battery depletion.

Hence, collecting user locations and actively locating users is not practical and would incur extra costs. In addition, users might move and change locations and thus this would require extra updates. Hence, it is essential to develop an autonomous algorithm that is capable of running without the need for previous knowledge about user locations. To overcome this challenge we present in this section a self-deploying algorithm that is capable of running without the need for any prior knowledge. This algorithm works once deployed and does not require any previous tasks or extra investigation.

### **3.5.2 Network Changes and User Mobility**

In most previous work available in the literature and discussed in Section 3.2 authors consider a static environment. However, this does not reflect real life

scenarios where users in wireless networks might move and update their positions [51]. In fact, the changes in user positions play a vital role in performance analysis in wireless networks. Hence, it is crucial to consider user movements when deploying ABSs in wireless environments.

The movement of users actually affect the quality of the received signal and might even cause sudden disconnection especially when a user moves outside the coverage region of a base station [52] [53]. In this case associations between users and base stations might need to change where some users are disconnected from a base station and connected to another base station. Also, in our case, ABSs might need to move in order to adapt to network changes. When users move, coverage areas change and ABSs could handle this problem by moving to better positions to achieve the required coverage percentage and to meet and enhance the expected QoS and minimum signal strength.

Thus, it is extremely important to consider an algorithm that adapts with user movements when deploying UAVs. Hence, in this section, we propose an autonomous algorithm that can quickly cope with user mobility and adapt according to the network change and updates in user locations.

### **3.5.3 Autonomous Force Algorithm**

Due to the high complexity of the formulated optimization problem, we study and propose in this section an efficient and practical solution. The proposed solution autonomously deploys ABSs in wireless environments while taking user mobility into account. Our aim is to position the ABSs in the 3D space while minimizing their number and attaining the minimum allowed quality of service. The algorithm utilizes the laws of electrostatic forces to place the ABSs in the best possible position and is adopted from [54].

In the first step, a number of ABSs based on the capacity constraints are released. Then, each of the ABSs and users is assigned a charge forming a non balanced electrical field. ABSs are assigned dynamic positive charges and users

are assigned fixed negative charges. Hence, the force formed between the different ABSs is repulsive, while the force formed between the ABSs and the users is attractive. According to Coulomb's Law, the force between two electrically charged points is determined as follows:

$$\vec{F}_{1,2} = \frac{Q_1 Q_2}{d_{12}^2} \vec{c}_{12}, \quad (3.11)$$

where  $Q_1, Q_2$  are the charges of the respective charged points,  $\vec{c}$  is the direction vector, and  $d_{12}$  is the distance separating both points. Since the received signal strength and the measured distance are inversely proportional as per (2.2), we model the force among ABSs and users as follows:

$$\vec{F}_{1,2} = Q_1 Q_2 R_{12} \vec{c}_{12}, \quad (3.12)$$

where  $R$  is the received signal strength from one user/ABS to another. This step allows the design of a dynamic and autonomous ABS positioning algorithm since the signal strength can be readily measured at the ABS in real time without any knowledge of the users' specific locations or mobility patterns. In addition, the direction of the read signal can be recorded by for example using directional antennas [55].

Hence, users are allocated static negative charges that are set to  $-1$ . However, the charges of the ABSs are always updated and are inversely proportional to the number of users associated to them. This being said, ABSs with high number of users get less ability to attract additional users. Hence, the charge of an ABS  $i$  is calculated as follows:

$$Q_{a_i} = \frac{\alpha}{k_i + 1}, \quad (3.13)$$

where  $\alpha$  is between 0 and 1 and  $k_i$  is the number of users associated with ABS  $i$ .

Based on the above, an electric field is formed between ABSs and users allowing ABSs to be attracted to users due to their opposite charges while they repel from each other due to similar charges. Hence, the formed forces will make

ABSs move until electrostatic equilibrium is achieved where the sum of forces is balanced and ABSs reside at fixed positions. The ABS movement is calculated according to the sum of forces exerted on each ABS with a specific step size. Fixing the step size we calculate the ABS new position as follows:

$$P_{a_i}^{\tau+1} = P_{a_i}^{\tau} + \eta \frac{\vec{F}_{a_i}}{\|\vec{F}_{a_i}\|}, \quad (3.14)$$

where  $P_{a_i}^{\tau}$  is the position at iteration  $\tau$ ,  $P_{a_i}^{\tau+1}$  is the new position at iteration  $\tau + 1$  and  $\eta$  is the step size.

To accommodate the necessary outage probability, we first deploy a fixed number of ABSs and perform the force algorithm. After that, if the number of users covered does not meet the outage probability then, we increase the number of ABSs and rerun the force algorithm. This is depicted in Fig. 7. By utilizing

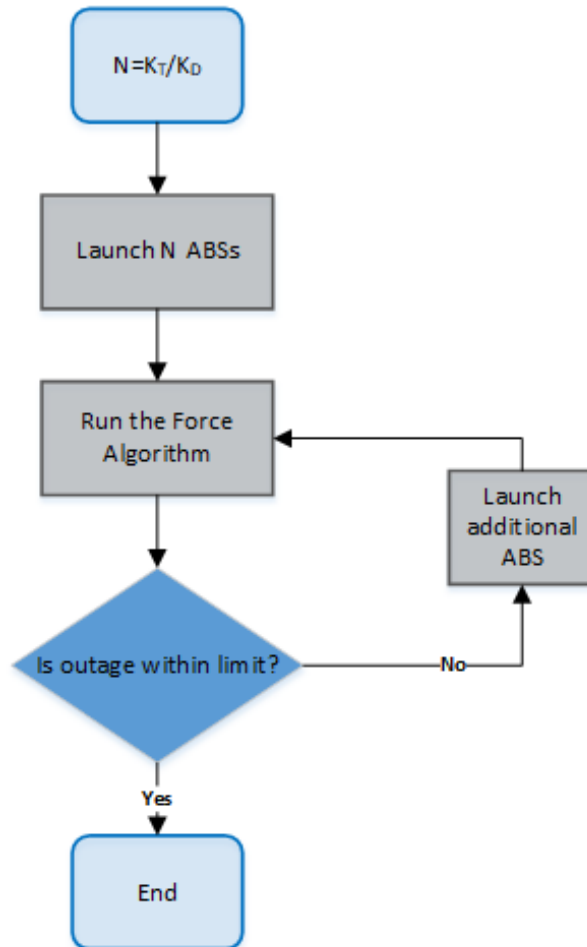


Figure 7: Proposed 3D autonomous force deployment algorithm flowchart.

the rules of electrostatic forces we effectively and autonomously deploy ABSs in wireless networks. After the initial deployments, ABSs can change their positions according to network changes and user mobility. When users move, this affects the equilibrium established and will require ABSs to move to restore the electrostatic equilibrium. In this way, ABSs move to meet new requirements and to maintain a given QoS. The algorithm described in this section is summarized in Algorithm 1.

---

**Algorithm 1** Force

---

**Input:**

$D$  : Set of  $N_D$  ABSs

$U$  : Set of ground users

**Output:**

$P$  : Set of coordinate vectors of all ABSs

```

1: procedure FORCE
2:    $N \leftarrow \frac{K_T}{K_D}$ 
3:   while ( Outage ratio not achieved) do
4:      $Q \leftarrow 0$ 
5:      $F \leftarrow 0$ 
6:     while ( Equilibrium not achieved ) do
7:       for each  $d_i \in D$  do
8:          $K \leftarrow 0$ 
9:         while ( $K < N_D$ ) do
10:            associate nearest visible user  $u_i$ 
11:             $U \leftarrow U - u_i$ 
12:             $K \leftarrow K + 1$ 
13:         end while
14:          $q_i \leftarrow \frac{\alpha}{K+1}$ 
15:       end for
16:       for each  $d_i \in D$  do
17:         for each  $v_j \in D \cup U$  do
18:            $F_i \leftarrow F_i + Force(d_i, v_j)$ 
19:         end for
20:       end for
21:       for each  $d_i \in D$  do
22:          $P_i \leftarrow P_i + \eta \frac{F_i}{\|F_i\|}$ 
23:       end for
24:     end while
25:     if ( $OutageRatio > \beta$ ) then
26:        $N \leftarrow N + 1$ 
27:     end if
28:   end while
29:   return  $P$ 
30: end procedure

```

---

## 3.6 Simulation Results and Performance Analysis

This section presents numerical and implementation results to evaluate the performance of our suggested solution. We consider a 100 m x 100 m area where users are randomly distributed. We apply a random walk model [56], with pedestrian ground users moving at a speed randomly picked from [1.25 m/s, 1.5 m/s] [57]. We run various number of simulations and then average the results according to the number of iteration which is guaranteed to be large enough. In our simulations we utilized the system parameters shown in Table 3.1. Fig. 8 plots the tradeoff between the number of needed ABSs and average user rate as the value of  $\lambda$  varies, where it is shown that values between 0.6 and 0.7 lead to a balanced point between both components of the objective function. Therefore, in the sequel, we set the value of  $\lambda$  to 0.6.

Table 3.1: System parameters for simulation results

Parameter	Value
ABS transmit power $P_D$	5 Watts
ABS maximum capacity $K_D$	30
path loss exponent $\alpha$	2
thermal noise power $\sigma$	$10^{-6}$ Watts
SNR threshold $\Gamma$	2 db
carrier frequency $f_c$	2.5 GHz
$\mu$ in (2.1)	9.61
$\beta$ in (2.1)	0.16
$\eta_{LoS}$	1
$\eta_{NLoS}$	20
step size $\eta$	0.5

First, we start by comparing the optimal solution to the autonomous force algorithm. To do so, we resolve the optimization problem at each time interval. Fig. 9 shows the average number of deployed ABSs with respect to the number of users in the network. We can see that the force algorithm provides close-to-optimal results. In fact, our solutions are trying to optimize the number of ABSs by minimizing the deployed ABSs. Minimizing the number of ABSs deployed in

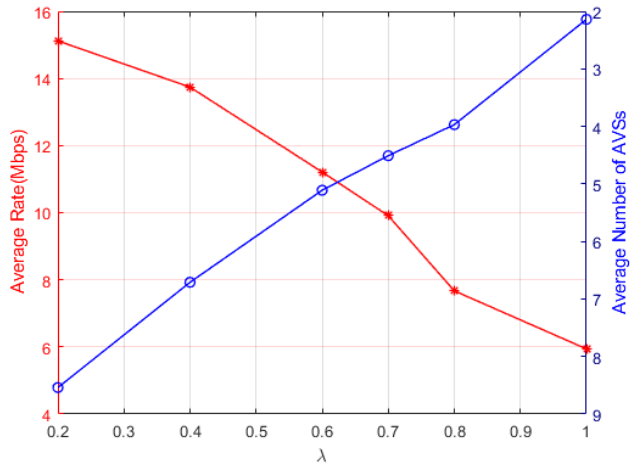


Figure 8: Tradeoff between the selected number of ABSs and the network average rate as a function of  $\lambda$ .

wireless networks leads to lower costs and better utilization of available resources. Fig. 10 shows the execution time of both optimal solution and force algorithm

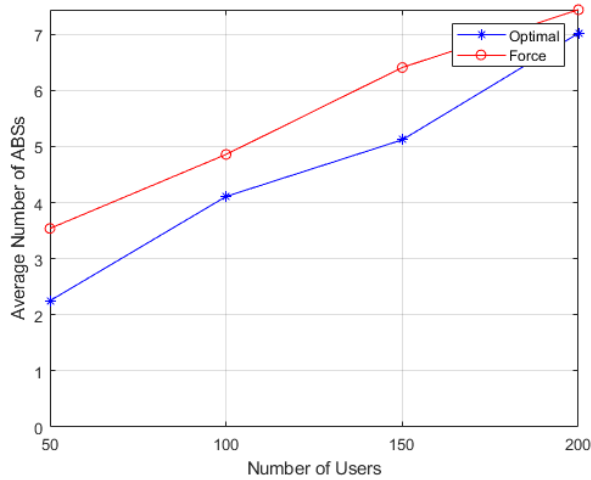


Figure 9: Average number of ABSs required to cover the users for the optimal solution compared to the force approach, as a function of total number of users.

as a function of the number of users. The execution time of the optimal solution increases exponentially with the number of users and is significantly higher than the proposed force algorithm. In Fig. 11, the force algorithm is shown to produce an average rate close to the optimal solution for different number of users. Thus, the force algorithm is capable of achieving an acceptable QoS.

To further evaluate the effectiveness of the proposed force algorithm, we com-

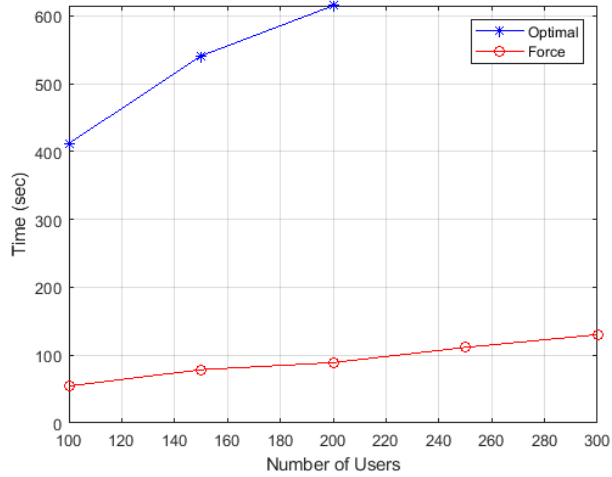


Figure 10: Execution time of the optimal solution versus the force approach, as a function of the total number of users.

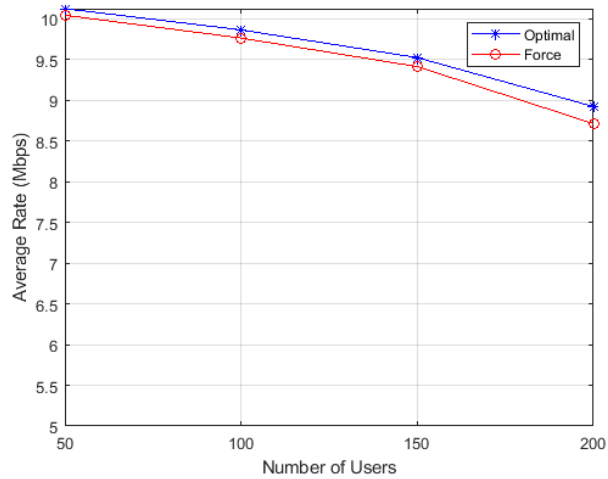


Figure 11: Average bitrate in Mbps for the force approach compared to the optimal solution, as a function of total number of users.

pare its performance to the work in [58], which is based on a spiral-based approach to deploy ABSs in a wireless network. The algorithm starts by placing the ABSs from the outer boundary and then iteratively moves towards the center to cover all users. To adapt the approach in [58] to network scenarios with user mobility, we rerun the solution from the beginning at each time interval. The key advantage of the force algorithm is its autonomous and dynamic behaviour. Without the need to run the force algorithm from the beginning in each time slot, the ABSs automatically adapt their positions incrementally as users move in the network.

Fig. 12 shows the change in average rate over time as users move, and clearly



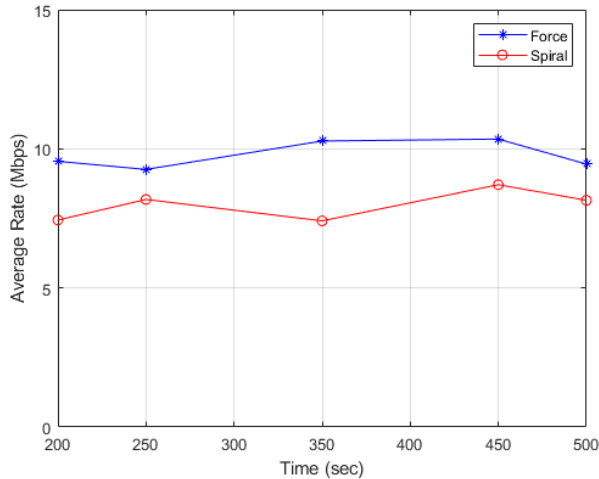


Figure 12: Average bitrate in Mbps for the force approach compared to the spiral solution, as a function of time.

demonstrates the superior performance of the force algorithm. The force algorithm is capable of achieving higher user rates and thus better quality of service and connection.

To better illustrate the behaviour of our proposed solution, we present network snapshot results in Fig. 13 and 14. In these figures the ABSs are represented as filled squares and the ground users are represented as empty squares. Users are connected and associated with ABSs of the same color. For example, in Fig. 13 the green users represented by empty squares are connected to the green ABS represented by green filled square. In Fig. 13, we can see the initial positions of the users and their two serving ABSs after the execution of the force algorithm. The ABSs were deployed over high demand areas and associated with users according to the QoS requirements. After 5 min, the users have moved and the new positions are depicted in Fig. 14. The old ABS positions are represented by small filled squares and the new updated ABS positions are represented by larger filled square. We can see the trajectory that both ABSs took to reach the final destinations. This is due to the electrostatic environment and charged fields created in the execution of the force algorithm. This demonstrates the ability of the algorithm to adapt to user mobility in an efficient and autonomous manner. When users moved the equilibrium in the environments is lost and the ABSs are

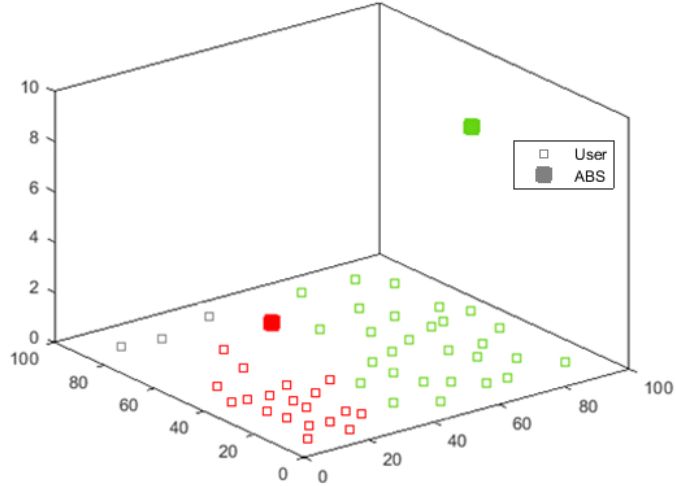


Figure 13: Network snapshot showing the initial ABS deployments based on the execution of the force algorithm. Users associated with each ABS are marked using the same color.

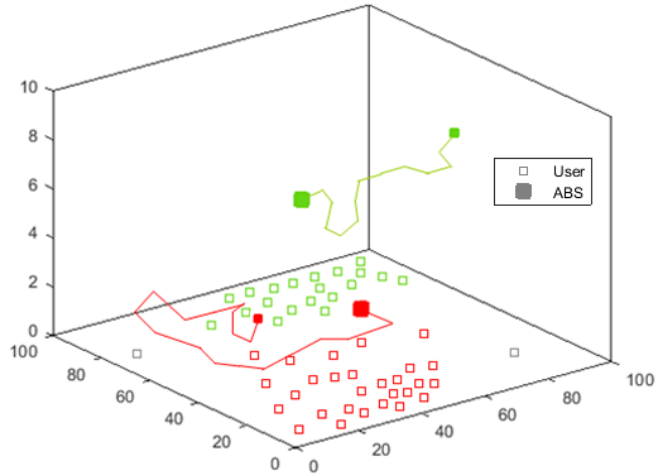


Figure 14: Network snapshot after 5 min showing the adaptation of the ABSs positions based on the execution of the force algorithm. Initial ABS locations are marked by small filled squares and final locations are marked by larger filled squares.

forced to move to reestablish the electrostatic equilibrium and thus reestablish the quality of service.

### 3.7 Implementation and Testbed

In this section we describe the utilized testbed and implementation. We implemented the force algorithm by using one drone acting as an ABS and a set of mobile devices acting as the ground users. We run the force algorithm on the

drone and we record and plot the drone trajectory.

The testbed consisted of one drone and a set of mobile devices as shown in Fig. 15. The drone utilized is a Parrot Bebop 2 drone with an ARM Architecture and a Linux kernel. We also used a set of Samsung S2 devices to act as the ground users. The devices are grouped into mobile and static devices. The mobile devices are devices that move around in our area and they are depicted in the left part of Fig. 15. The drone reads the RSSI value of each user and move

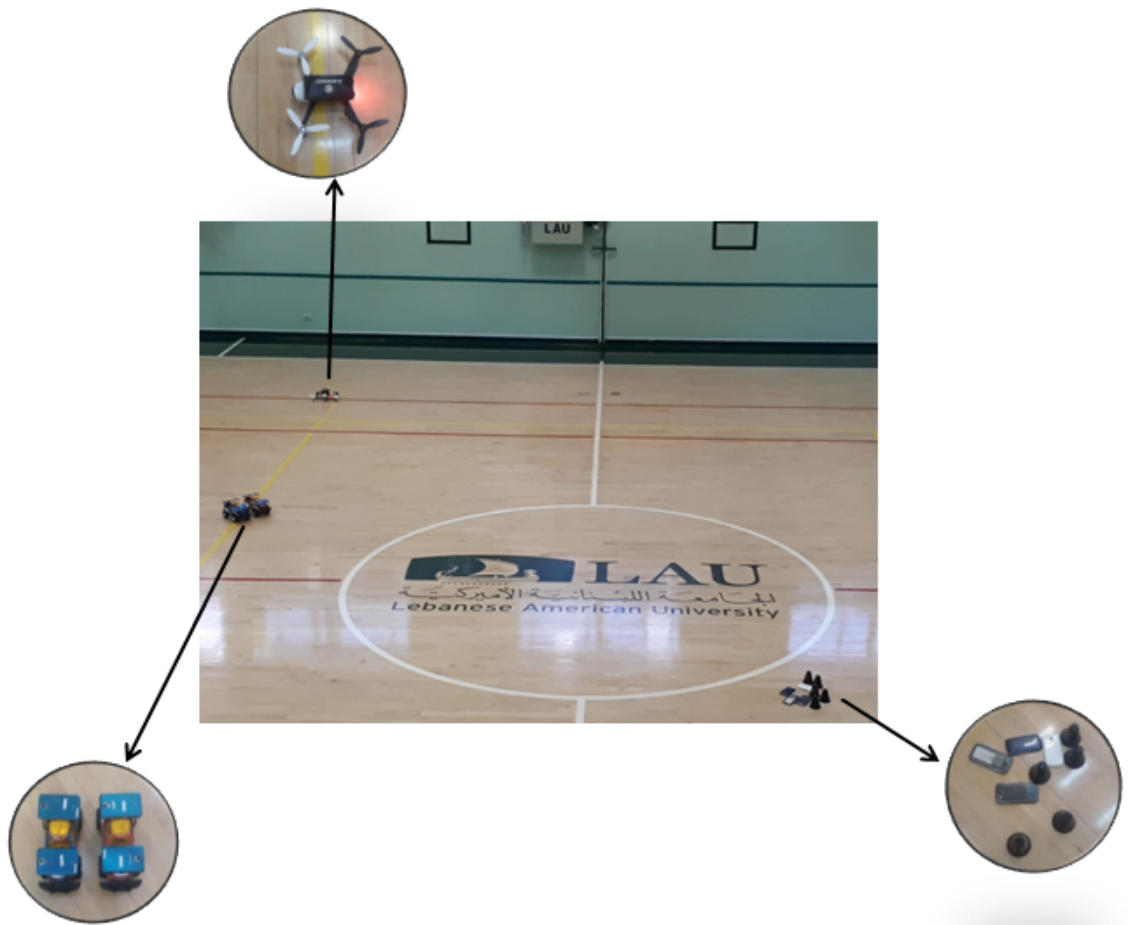


Figure 15: Testbed setup.

according to the force algorithm to finally reach the equilibrium state where all forces are stabilized. The final position is the equilibrium position where the drone will settle. The drone is reading new RSSI values each 20 seconds. Whenever the drone senses a change in RSSI values and change in forces where equilibrium is lost, it changes its position according to the new forces. The force algorithm also requires us to know the direction of the signal coming from each

user. In our implementation we assume that the devices are capable of sharing their direction with the drone. Thus, the mobile devices send to the drone their direction. Having both signal strength values and users' direction the drone is able to execute the force algorithm and move according to the different calculated forces. The communication between the ground users and the drone is depicted in Fig. 16.

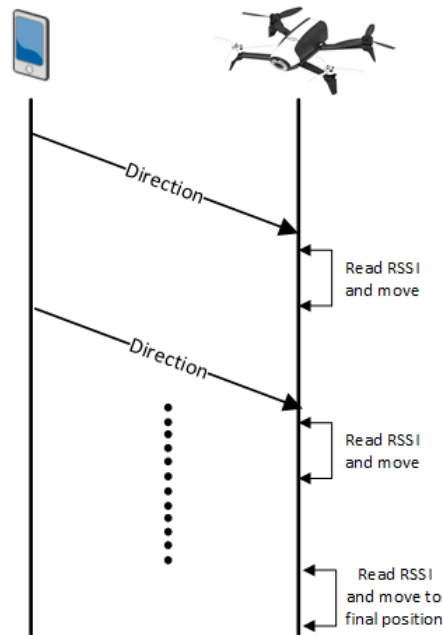


Figure 16: Messages exchanged between the ground users and the drone.

To program the drone we use the java language and Android development. An Android application was installed on the ground devices to enable the communication with the drone. This application connects to the drone server and sends the users' directions. Another application was deployed on the drone. This application represents the drone's server and it is responsible for receiving users' direction, read signal strength values, and execute the force algorithm.

We tested our implementation on different scenarios and test cases. We used one drone that was always released on a fixed and stable height. In figures 17 and 18 we consider a 6 m x 6 m area where ground users are deployed. The drone representing the ABS is launched from the (0, 0) position. The drone will collect the directions and read the different signal strength values to execute and move

according to the force algorithm. The users colored red are mobile users who can move and the blue users are static users. In Fig. 17 the drone moved until it reached the equilibrium position to serve all ground users in the area. When the users moved in Fig. 18 the drone moved again from the old equilibrium position to reach the new equilibrium position.

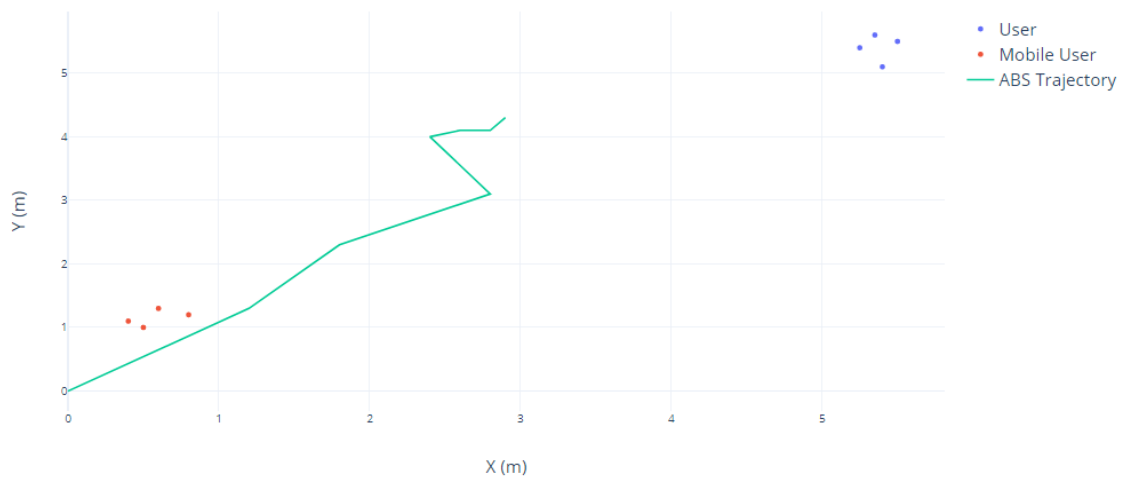


Figure 17: Trajectory of a single ABS serving a group of ground users.

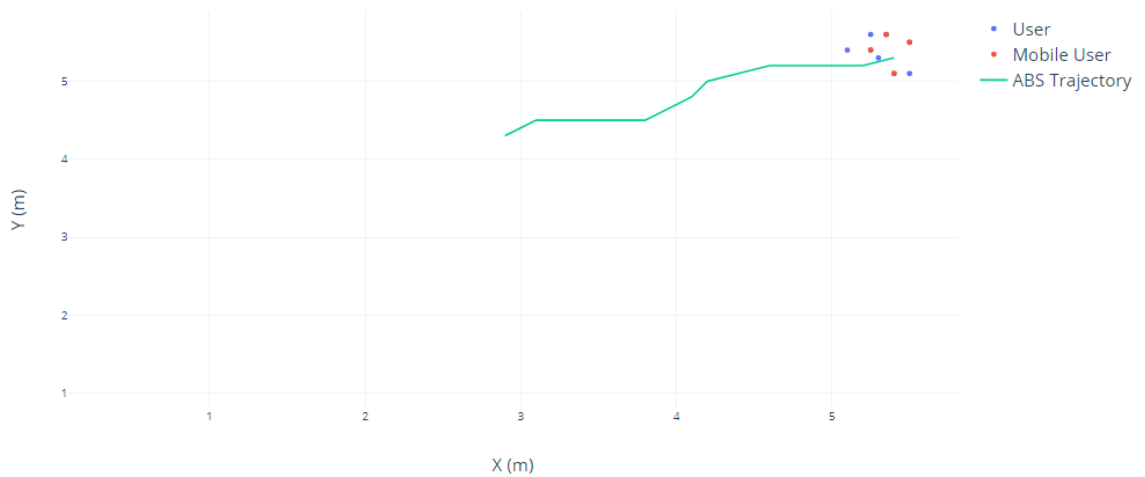


Figure 18: Trajectory of a single ABS serving a group of ground users after moving around in the area.

### 3.8 Conclusion

In this chapter, we studied the problem of autonomously deploying UAVs to act as ABSs in 3D space with user mobility. We first formulated the 3D positioning

problem as a mixed integer program to obtain optimized results for evaluation purposes. Then, we proposed an efficient and autonomous 3D positioning algorithm that depend on the notion of electrostatic forces. The proposed algorithm works without any knowledge about the network topology or the users' distribution, and easily adapts to network changes and user mobility. Results are presented as a function of various system parameter, and demonstrate close performance compared to the optimal solution and superior performance compared to recent related work from the literature.

# Chapter Four

## Optimized 3D Deployment of UAV-Mounted Cloudlets to Support Latency-Sensitive Services in IoT Networks

### 4.1 IoT Networks

#### 4.1.1 Overview

Since the first successful connection of two computers, the Internet did not stop evolving. It first started in the late 1960s where two computers were successfully connected. After that, in the early 1980s, the IP and TCP stacks were introduced and then the Internet began to be used commercially in the late 1980s. Next, the world wide web developed in 1991. Actually, the introduction of the world wide web led to an increase in Internet popularity and thus a huge rise in the number of users. Following that, mobile devices started to connect to the Internet which induced the creation of the mobile Internet. In this era, mobile devices, computers, and people started to connect to the Internet and use it on a daily basis for different services and applications like e-commerce, education, social networking, and gaming [59].

Alongside, the advances in the Internet and its technologies, sensors networks and embedded actuators have been also growing and improving. Sensors became more accurate and reliable and thus enabling the quick detection of information that even humans cannot detect. Robots evolved and were shipped with huge capabilities that defeat any physical restriction. These huge advancements in the Internet and in game-changing technologies lead to the vision of a new paradigm where machines are connected to the Internet and are capable of communicating with each other. Truly, from here emerged the notion of the Internet of Things (IoT) [60] [61].

The term IoT was first suggested in 1999 by the British Scientist Kavin Ashton. Indeed, the word IoT can be divided into two main components. The first is the Internet reflecting the network and communication part. The second part is the things that not only refers to computers but also to sensors, actuators, electronic devices, cameras, vehicles, living things, and non-living things. The IoT paradigm envisions an environment where things are connected anywhere, anytime and with anything and thus providing any service [61]. While there is no single and standardized definition, IoT mainly states that anything can be armed with the necessary capabilities to connect to the Internet and provide any service or access any application anytime, and anywhere (Fig.19) .

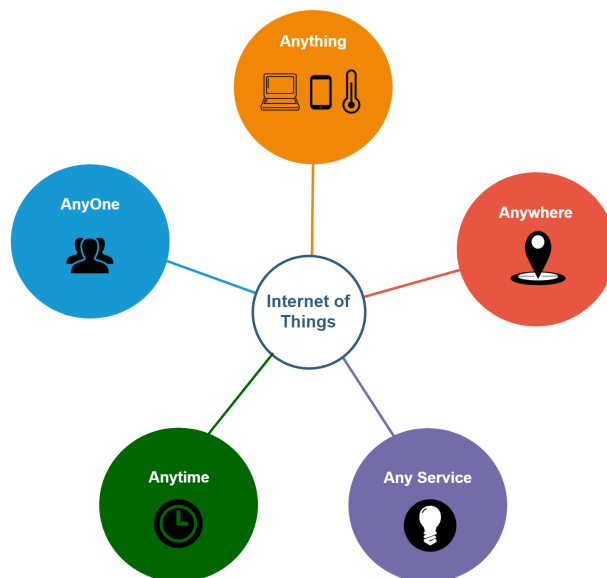


Figure 19: General definition of IoT.



IoT networks nowadays are enabling a huge variety of applications and services like smart farming, smart health services, smart transportation, and environment monitoring. Indeed IoT networks are unique in their characteristics and nature. Below we detail some of the main IoT characteristics [62] [63].

- **Interconnectivity:** IoT envisions an environment where anything could be connected to the Internet or with other devices. Anything and anyone can easily and directly connect to the network infrastructure to provide or access a service.
- **Heterogeneity:** Devices deployed in IoT networks are diverse in their nature. Anything actually could be deployed and act as an enabler for IoT services including but not limited to cameras, sensors, cars, refrigerators, watches, mobile devices, actuators, and computers.
- **Huge scale:** The number of connected devices in IoT networks is very large and is expected to become 5 times more than the number of connected computers to the Internet by 2025.
- **Progressive variations:** In IoT networks, the state of the connected devices continuously varies. For example, a device might change its location or speed, disconnect, sleep, or turn off.

#### **4.1.2 Applications and Use Cases**

The advances in IoT devices, networks, and systems enabled a plethora of applications in various areas including health, industry, and environment. Below we discuss some of the main applications and services provided by IoT systems [64] [65].

- **Health-care:** IoT networks have been widely studied and investigated hoping that they can improve the quality of human lives. Indeed, IoT networks have been widely used in medical and health care to combat difficulties faced by doctors and patients. For example, sensors could be deployed to

monitor a patient throughout his hospitalization. These sensors are normally connected to the main control system that displays all information to doctors in real time. This can help doctors in monitoring the patient's health in a fast and efficient way. It can be specifically important to report sudden emergencies or health complications. On the other hand, IoT can be also used to track the medication effects on a patient's body and health. It can detect any side effects, allergies, or misbehaviour.

- Smart industry: IoT networks could be also utilized in industrial plants and factories. For example, radio-frequency identification (RFID) tags could be deployed on products to track the manufacturing line and monitor the status of each product. Professionals could use the sensed data to improve production lines in terms of efficiency and quality and to detect any malfunctions. Now, it is also possible for the manufacturer to track the product even after production and until it is received by the customer. This helps in improving the delivering process and ensuring a good customer experience. On the other hand, IoT systems could be deployed in factories and mines to monitor gas and chemical levels in the air and ground. This guarantees a safe and healthy environment in industrial buildings.
- Smart agriculture: Nowadays IoT systems have been widely used in agriculture and farming. Indeed, IoT systems could be utilized to monitor animals in farms and track their location, health, weight, and eating habits. In addition, IoT systems are being deployed in greenhouses to monitor the air, temperature, and light. This can help in automatically adjusting the greenhouse environment according to growing patterns and plants' status. Furthermore, sensors are deployed to measure the number of minerals and vitamins in the sand. They could be also used to plan and execute watering schedules for different crops and plants.
- Intelligent transportation: Installing IoT devices on roads, vehicles, and

highways can guarantee a safer environment for drivers and pedestrians. Sensors could be deployed to monitor traffic and thus direct vehicles towards better and less congested paths. IoT devices fitted in vehicles could also detect any emergencies or accidents and thus quickly warning the driver.

- Smart environment: Ensuring a healthy environment and perceiving nature is greatly important. IoT systems can make this mission easier on us by monitoring the air composition and checking for any polluting or dangerous gases. IoT devices could be also useful in checking the cleanliness and purity of water in rivers and seas. They can efficiently and correctly evaluate whether different water sources are safe to drink or not. In addition, deployed sensors can quickly detect fires in forests or woods and thus help in reducing extra damages or further expansion of fire flames.
- Smart cities: IoT can help in planning and designing smart cities by monitoring lightning patterns in the city and available parking spots. Moreover, IoT can assist in monitoring the status of different buildings, structures, and bridges. Also, IoT devices could be installed to monitor and limit energy consumption in buildings and on roads. Sensors can monitor heating, cooling, and lighting systems and adjust them based on environmental factors.
- Smart law enforcement: By placing IoT devices and sensors in public places the government can force better surveillance and thus can detect any violations or crimes actively and respond in a fast manner. Furthermore, IoT systems can help in predicting crimes and thus can stop them even before occurring. IoT systems can improve the effectiveness and intelligence of regular surveillance devices.

Indeed, many industries and companies started to use IoT in numerous services and applications [66]. For example, Amazon has created the Amazon Go concept which is actually a store that is not operated by cashiers but by sensors

and embedded machines that can give you better shopping experience. After picking a product the sensors will directly identify it and add its cost to your account. If you leave any product behind it will again automatically deduct the cost from your account. When leaving the shop, the total amount according to your purchases will be deducted.

Another main example that shows the power of IoT systems is the current DHL tracking and monitoring system [67]. DHL launched in 2018 a smart tracking system that utilized sensors and embedded chips in trucks to track and monitor products. Also, this is currently being used to monitor the roads and guide drivers to better paths and thus avoiding any delay. Customers are being updated about the status of their products in real time.

### **4.1.3 Technical and Research Challenges**

Indeed, the wide range of IoT devices and the advances in communication and sensing technologies are allowing the emergence of new and novel applications. However, to correctly and efficiently enable these applications and services, one should first tackle and solve the many challenges arising in IoT systems. Next, we discuss some of the common challenges faced in IoT systems and networks [68] [64] [69].

- Privacy and security: Since IoT devices are plugged into the Internet, security and privacy threats will naturally flow. Thus, it is very important to secure the communication between any pair of communicating IoT peripherals. Data integrity should be forced to ensure that no one can change the content of the data on its way without the consent of the sender nor the receiver. Also, IoT devices should be able to authenticate themselves and other devices to ensure correct dispensing of information between different nodes. So, once an IoT node is contacted by another node it should first authenticate this node and then share the confidential information. On the other hand, the confidentiality of the collected and shared data is increas-

ingly important. The sender should be confident that his shared data is not breached on its way to the receiver and is not used by other nodes for any other purposes.

- Network foundation: The number of connected IoT devices is extremely large and requires fast, reliable, and flexible connectivity. The limitation in the current architecture of the Internet, and its availability act as primary obstacles in front of IoT applications. In addition, a reliable and standardized identity management scheme should be implemented to manage unique and different devices.
- Management of heterogeneity: IoT devices connected together are of different types and forms. These devices are expected to communicate and operate together in the same environment. Thus, it is very important to manage the relationship between the different heterogeneous devices especially that each device might have its own communication standards and operation rules. It is also crucial to define common standards to enable better interoperability of devices.
- Data storage and memory limitations: IoT devices and sensors produce a large amount of data the needs to be stored for future analysis and study. On the other hand, IoT devices have limited memory storage. Thus, a major challenge is to store these large amounts and sizes of data in an efficient and secure way to ensure reliable and fast transactions and queries.’
- Designing sensor mechanisms: Correct monitoring of different environments calls for the accurate and well-studied design and manufacture of various sensors. In hard and exceptional conditions sensing information from the surrounding might become a challenging and hard task. Thus, it is crucial to design robust and reliable sensors that are capable of operating in different conditions and circumstances. It is also necessary to focus on the automated configuration of sensors to guarantee correct and accurate

behavior in different situations and circumstances.

- Real-time processing: IoT networks and systems require the fast and reliable processing of data, transactions, and queries. Thus, it is crucial to enable fast processing and real-time response in IoT networks where time is an essential factor for taking critical and sensitive decisions and actions. The response delay should be minimized to enable efficient and flexible control of the environment and surrounding.
- Computational limitations: Processors installed on IoT devices are very limited in terms of computational capabilities and capacities. IoT devices can only act as simple sensors and actuators. Thus, IoT devices cannot run heavy applications or process information in a fast and efficient manner. Hence, to enable real-time and reliable applications one might need to look into other solutions.
- Energy limitations: Usually IoT devices are equipped with limited batteries and thus have restricted energy and power capabilities. These devices often go to sleep to extend their lifetime and save energy. Thus, it crucial to study different ways to decrease energy consumption and extend the effective operation time of IoT devices.
- Mobility: A lot of the utilized IoT devices are mobile and portable like health care tools and IoT devices used in transportation and transit applications. The mobility of IoT devices incurs extra challenges in terms of connection, communication, and stability.

In this work, we aim to mitigate the challenges induced by the low processing power in IoT devices. We suggest new solutions that can enable the effective utilization of IoT devices in smart environments and mitigate different environmental, hardware, and software difficulties.

## 4.2 Optimized 3D Deployment of UAV-Mounted Cloudlets

### 4.2.1 Motivation and Background Information

As stated earlier, the proliferation of Internet-of-Things caused a major shift in computing and communication. IoT paradigm envisions an environment where things are connected anywhere, anytime and with anything and thus providing any service. These devices are transforming our physical environment into smart and interactive platforms. However, IoT devices are usually energy limited and possess weak processing capabilities. Indeed, it becomes very challenging for IoT devices with small computational power and limited energy to run latency-sensitive services that need high computation capabilities.

Realizing the large number of services that IoT devices can provide demands a significant rethinking of the way these constrained devices are utilized, installed, and integrated within a smart, reliable, and efficient environment. It is especially important to tackle the problem of limited energy and low computational power in IoT devices to effectively enable an extensive range of latency sensitive and on-demand applications. These applications like industrial automation, smart grids, and video streaming demand very low latency and high reliability. For example, IoT systems can be used in factories to track and manage the production line. Sensors and RFID tags can be deployed on manufactured products and machines to track the manufacturing process, detect any problems, and automatically act upon sudden accidents or possible errors in the production line. However, it is very hard to envision these applications with the current resources available within IoT devices. Thus, an efficient and reliable solution should be proposed to bring these applications to life.

Mobile edge computing (MEC) has been introduced as a promising solution to offload latency-sensitive traffic to edge cloudlets that are capable of processing

requests in a fast and efficient manner [70]. This solution provides powerful servers that IoT devices can use to offload their computations to and benefit from fast processing and quick delivery. However, because of their power constraints, IoT devices are not capable of transmitting over a long distance. In addition, IoT environments are always prone to dynamic changes and continuous shifts in device positions, number, and types of services requested. Hence, in this work, we propose an efficient solution that utilizes UAV cloudlets to support latency-sensitive applications in IoT networks. By utilizing moving UAVs we provide an efficient and low cost solution that can be flexibly deployed in different environments and scenarios. UAVs will bring edge cloudlets and powerful servers much closer to IoT devices and thus can decrease transmission latency and provide better on-demand services.

By the means of uplink and downlink communications mobile users can offload their computational tasks to the UAVs. For example, moving cloudlets can play an essential role in farming applications [64] where sensors determine soil moisture levels and are used to analyze the surrounding environment and for detecting plant disorders [71]. In the scenario represented in Fig.20, sensors can collect data and offload to the UAV cloudlets; the UAV cloudlets process the data and then send the appropriate order to the water sprinklers. In this way soil moisture is monitored and perfect water levels are always achieved leading to healthier plants and less water wastage. UAV-Mounted cloudlets can be used in multitude of applications to assist existing IoT networks and perform the heavy and power consuming computations.

The rest of this chapter is organized as follows. In Section 4.2.2, we cover the related literature and highlight the main contributions of this work. Section 4.2.3 details the system model and its key components. In section 4.2.4, we formulate the problem of 3D deployment of multiple UAVs in IoT networks as a MIP. Section 4.2.5 describes the proposed meta-heuristic algorithm for deploying multiple UAVs in IoT systems. Section 4.2.6 presents performance results



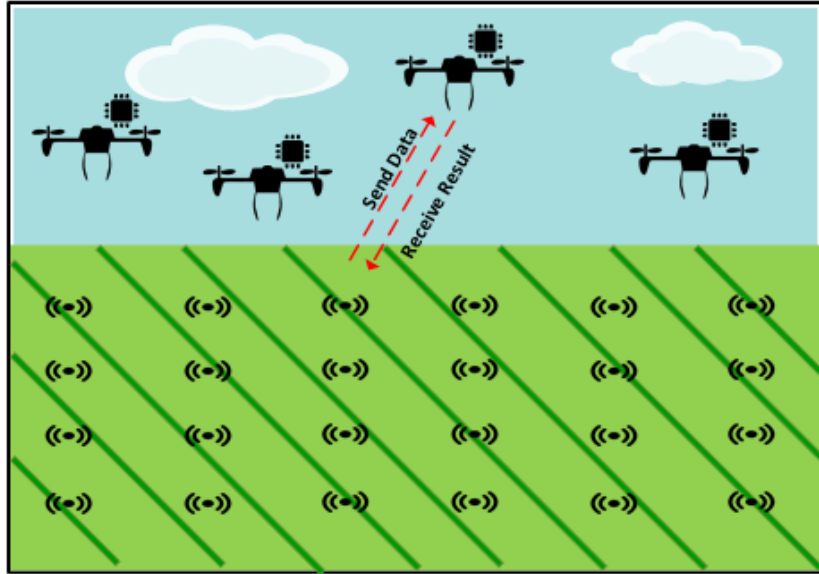


Figure 20: UAV-mounted cloudlets in agriculture scenario.

for various scenarios and highlights the effectiveness of the proposed algorithm. Finally, conclusions are drawn in section 4.2.7.

## 4.2.2 Related literature

Mobile edge computing has been widely used as an effective solution to combat the downsides of constrained power and computational capabilities in IoT devices. However, to be able to effectively use edge computing in IoT environments one should be able to clearly answer some of the basic questions on MEC implementation like where to place edge servers? How to offload traffic? And How to assign user requests to edge servers? Indeed, there has been a lot of research to answer the many questions related to MEC utilization and implementation in IoT networks.

For example, in [72] the authors studied the problem of Latency-Aware Workload Offloading in edge computing. First, they suggested an architecture for edge computing based on SDN networks. Then, they formulated the problem with the goal of minimizing the response time of the end users. They proved that this problem is NP-hard by a reduction from the famous partition problem. At the end, they suggest a heuristic algorithm to solve the formulated problem based on

a greedy strategy. Another work represented in [73] jointly optimized the resource allocation and user requests offloading in edge computing. In addition, they considered the transmission power of IoT devices and formulated their problem to minimize the utilized power. At the end, they suggested an efficient heuristic algorithm to solve the optimization problem based on an iterative algorithm.

Other work focuses on virtualizing the services and network functions available within IoT environments. For example, [74] focused on distributing IoT based applications on edge servers. Specifically, they assigned a unique group of IoT based applications to each edge server and defined specific stringent latency requirements for each application. Also, they associated each IoT device with multiple computing edges while aiming to minimize the total response time. Another work in [75] performed an extensive study to evaluate the performance gains while using MEC in different scenarios and environments. They specifically considered a game application and studied the effect of device association and server positions on the effectiveness of the deployed solutions. Their conclusions state that the response time greatly depends on the position of the edge servers and the task assignment between end users and edge servers.

On another hand, currently, UAVs are playing an important role in achieving effective and low energy IoT environments. UAVs have been utilized in multiple IoT scenarios to extend IoT capabilities. In [76] the authors deployed UAVs in 3D space to harvest data from IoT devices. They jointly optimized the 3D deployment positions of IoT devices and the applied transmit power. They formulated their problem while minimizing the total power and associating different IoT devices to various UAVs. Similarly, in [77] the path of multiple UAVs was optimized to effectively collect data generated by IoT devices. Initially, they deployed the UAVs within a static network consisting of immobile IoT devices. This step consisted of grouping devices together and associating each group to a single static UAV. Then, they consider the scenario of mobile and moving UAVs where they studied the optimal path of multiple UAVs to collect IoT data while

optimizing the energy consumed.

Research related to deploying UAV-mounted cloudlets is actually still in its infancy. Some work started to tackle the problems associated with using and deploying UAV-mounted cloudlets. For example, in [78] the authors suggest using UAVs as computing cloudlets where they focused on optimizing the allocated bitrates. They carefully designed the path of a single UAV equipped with high computing capabilities while optimizing both uplink and downlink bit allocation. While in [79] UAVs were used to support the edge network and act as caching devices. The authors suggested and described a new architecture that can enable effective use of UAVs for both caching and cloud computing purposes.

Deploying and utilizing UAVs as mounted cloudlets is not extensively studied yet and is still in its infancy. Most previous work considers UAVs only for collecting data or for acting as caching devices. Also, they consider only one single UAV that is used to serve all users. However, in this work, we study the optimal deployment of UAV-mounted cloudlets in IoT networks to support latency-sensitive applications while considering user demands, QoS, and resources available. Specifically, we model our system to enable efficient use of resources while benefiting from UAVs unique properties like flexibility and enhanced LoS. We work on deploying an optimal number of UAV-mounted cloudlets to support IoT stringent applications while meeting all demands and quality constraints. Our work aims to optimize all number of deployed UAVs, their 3D positions, and devices associations.

### **4.2.3 System Model**

We consider an area where a large number of resource-constrained IoT devices are deployed to regularly monitor and collect information that needs to be processed in a timely manner for proper decision making. Requests to process the collected information are issued to UAV-mounted cloudlets, which offer computation offloading services. Every IoT device is expected to generate requests to

process computation tasks; whereby requests of multiple devices are aggregated and delivered to a designated UAV-mounted cloudlet that can fully process the requests within their delay limit. Due to the low transmission power of IoT devices to reach their respective UAV-mounted cloudlets, more UAVs are deployed that may only act as relay nodes and thus are not equipped with cloudlet capabilities. Consequently, based on the expected demand of IoT devices, a swarm of UAVs are deployed in 3D space as a fully meshed network with selected UAVs equipped with cloudlet resources while others serve as relay nodes to connect IoT devices with their cloudlet. It may also occur that one UAV takes both roles, as a relay to a number of IoT devices and as a cloudlet to other devices. In this work, we aim at determining and placing the minimum number of UAVs in 3D space to handle all computational workload generated by the IoT devices in the network. Each IoT device is associated to one of the nearby UAVs denoted as serving UAV that may either relay the tasks to another UAV for computation or compute the tasks locally. The UAV responsible for computing the offloaded task is denoted as the processing UAV.

Figure 21 presents a sample network with six IoT devices offloading tasks to three UAV-mounted cloudlets. The device, first, associates with a nearby UAV referred to as serving UAV and uploads its computation tasks. Afterwards, the serving UAV may compute the tasks locally and deliver the results to the respective IoT device upon completion or offload to another UAV that can handle the requests. If offloaded to another UAV, the latter computes the task and sends the result back to the serving UAV to deliver it to the requesting IoT device. For example in Fig. 21 the IoT devices colored yellow is being connected to one UAV but the offloaded task are computed by another UAV. So, in this case one UAV acts as the relay and thus the serving UAV and another UAV act as the processing cloudlet and thus the processing UAV. While the IoT device colored blue for example is being served by a close UAV and the tasks offloaded are being processed by the same UAV.

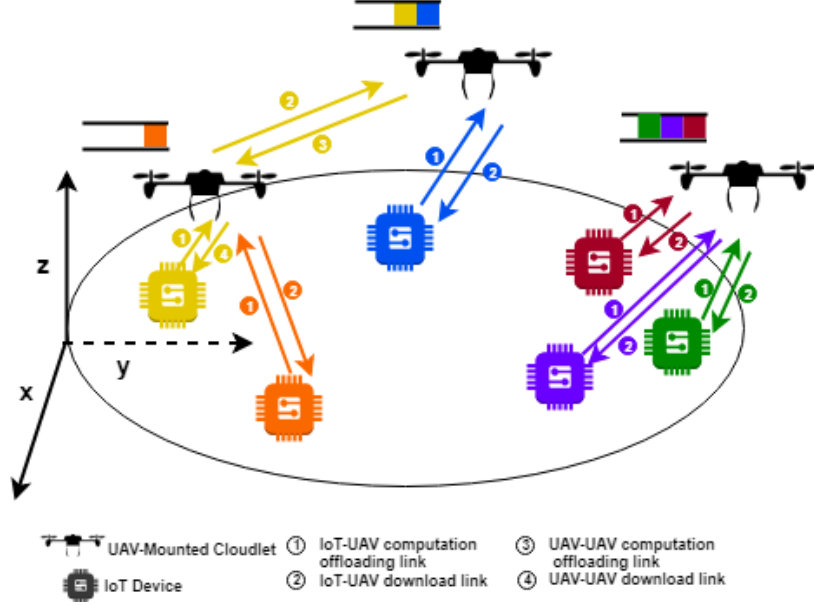


Figure 21: Operation steps of offloading computation tasks to UAV-mounted cloudlets.

We denote the set of IoT devices in the environment as  $\mathcal{U} = 1, 2, \dots, U$  and the group of UAVs as  $\mathcal{D} = 1, 2, \dots, D$ . Every UAV can serve a maximum of  $K_D$  IoT devices. For the path loss model we use the one presented in [25] that depends on the angle and height established between the UAV and the IoT device resulting in the equations discussed in Section 2.2.

In what follows, we model the IoT device computation tasks, UAV computation resources, and task uploading transmissions.

### IoT Computation Tasks

We consider that each IoT device  $i$  generates computation requests based on a Poisson distribution where the average rate is  $\lambda_i$ . Requests of multiple devices are aggregated and delivered to one UAV for processing. It follows that the arrival process of requests to one UAV  $j$  also follows a Poisson distribution with an average rate equal to the summation of the individual rates at which each IoT device generates tasks and is represented as  $\sum_i \lambda_i$ . Moreover, we consider computation tasks to be completely processed within a given delay limit depending on the type of data collected by the device and the offered service based on this data. Each

IoT device  $i$  then requires its tasks to be fulfilled before a time limit  $T_i$ .

### UAV Computation Resources

Each UAV  $j$  is assumed to execute IoT requests in an exponentially distributed manner with an average service time equal to  $\frac{1}{\mu_j}$ , where  $\mu_j$  is the average service rate of UAV  $j$ . If the processing capacity of UAV  $j$  is  $f_j$  cycles/sec, then the service rate in requests/sec is  $f_j/L$ , where  $L$  is the average size of computation tasks in cycles. Therefore we model the processing of IoT requests by a UAV-mounted cloudlet as an M/M/1 queueing system with arrivals following a Poisson distribution with rate  $\sum_i \lambda_i$  that represents the summation of all rates at which IoT devices that are assigned to cloudlet UAV  $j$  generate computation tasks.

### Total Task Delay

In case IoT device  $i$  offloads its tasks to one of the UAVs, then the incurred delay comprises the time  $t_{up,ij}$  to upload the data to the serving UAV  $j$ , the time  $t_{U2U,jk}$  to deliver the task from the serving UAV  $j$  to the processing UAV  $k$ , the time  $t_{process,ik}$  to fully process the task at the processing UAV  $k$ , and the time to deliver the result back to the IoT device through the serving UAV. All UAVs are assumed to be fully meshed and the transfer delay  $t_{U2U,jk}$  of data from one UAV to another is assumed to be fixed. As assumed in other related work including [77], the size of the task output is in general much smaller than the input task data uploaded by the IoT device and thus the delay to transfer the result from the serving UAV to the IoT device in the downlink is ignored. Hence, the total delay experienced by tasks of IoT device  $i$  can be calculated as follows:

$$t_{total,i} = t_{up,ij} + 2 * t_{U2U,jk} + t_{process,ik}, \quad (4.1)$$

The upload delay  $t_{up,ij}$  depends on the resulting bit rate of the uplink from the IoT device  $i$  to the serving UAV  $j$ . For simplicity, we assume that the uplink bandwidth is equally distributed among all active devices  $U$  associated with UAV

$j$  resulting in a bit rate

$$R_{ij} = \frac{B_j}{U} \log(1 + SNR_{ij}), \quad (4.2)$$

where  $B_j$  is the total uplink bandwidth of UAV  $j$  and  $SNR$  is the received signal to noise ratio of IoT device  $i$  at UAV  $j$ . As for the delay  $t_{process,ik}$ , which represents the total delay spent at the processing UAV including the waiting time and the execution time, it is computed according to Little's law knowing that the cloudlet UAV is modeled as M/M/1 system as described earlier. This being said,  $t_{process,ik}$  is derived as follows:

$$t_{process,ij} = \frac{1}{\frac{f_j}{L} - \sum_n \lambda_n}, \quad (4.3)$$

where  $\sum_n \lambda_n$  is the total task generation rates of all IoT devices  $n$  assigned to cloudlet UAV  $j$  for processing their tasks.

#### 4.2.4 Problem Formulation

We mathematically formulate our problem in this section to determine the minimum number of UAVs in addition to the 3D optimal positions to serve all IoT requests before their delay limits. Our optimization problem should be able to decide on the number of UAVs to be deployed, the 3D optimal position of each UAV, the UAV-IoT association and whether each UAV is considered a serving or processing UAV for each device associated.

To formulate the problem, we introduce a decision variable  $d_j$  to indicate whether UAV  $j$  is deployed or not.

$$d_j = \begin{cases} 1, & \text{if UAV } j \text{ is deployed} \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

Since each IoT device may use up to two UAVs, a serving UAV that connects

the device to the UAV network and a processing UAV that acts as a cloudlet and processes the computation tasks, we make use of two binary decision variables,  $a_{i,j}$  and  $b_{i,j}$ .  $a_{i,j}$  specifies whether UAV  $j$  is the serving UAV of IoT device  $i$  and is defined as follows:

$$a_{i,j} = \begin{cases} 1, & \text{if IoT device } i \text{ is associated with UAV } j \\ 0, & \text{otherwise} \end{cases} \quad (4.5)$$

The other decision variable  $b_{i,j}$  indicates whether UAV  $j$  is the processing UAV of IoT device  $i$  and is defined as:

$$b_{i,j} = \begin{cases} 1, & \text{if tasks of IoT device } i \text{ are processed by UAV } j \\ 0, & \text{otherwise} \end{cases} \quad (4.6)$$

In that case, an IoT device connects to a nearby UAV and delivers its task data with little transmission power. For an IoT device  $i$  to be considered connected to UAV  $j$ , the received signal to noise ratio (SNR) at the UAV should be above a target threshold  $\Gamma$ . Hence,

$$P_{ij} \geq a_{ij}\sigma^2\Gamma, \quad (4.7)$$

constitutes the received power constraint for device  $i$  to associate with UAV  $j$ , where  $P_{ij}$  is the received power level that is calculated according to Eq.2.2 and  $\sigma^2$  is the thermal noise power.

According to the above, the problem is formulated as a mixed integer programming as follows:



$$\text{minimize } \sum_{j=1}^D d_j \quad (4.8)$$

$$\text{subject to } a_{ij} \leq d_j \quad \forall i \in [1, U] \quad \forall j \in [1, D] \quad (4.9)$$

$$b_{ij} \leq d_j \quad \forall i \in [1, U] \quad \forall j \in [1, D] \quad (4.10)$$

$$\sum_{j=1}^D a_{ij} = 1 \quad \forall i \in [1, U] \quad (4.11)$$

$$\sum_{j=1}^D b_{ij} = 1 \quad \forall i \in [1, U] \quad (4.12)$$

$$P_{ij} \geq a_{ij} \sigma^2 \Gamma \quad \forall i \in [1, U] \quad \forall j \in [1, D] \quad (4.13)$$

$$\sum_{j=1}^D \left( a_{ij} t_{up,ij} + \frac{b_{ij}}{\frac{f_j}{L} - \sum_{n=1}^U b_{nj} \lambda_n} \right) + \bar{a}_{ij} b_{ij} \sum_{k=1}^D a_{ik} t_{U2U,jk} \leq T_i \quad \forall i \in [1, U] \quad (4.14)$$

$$\frac{f_j}{L} - \sum_{i=1}^U b_{ij} \lambda_i \geq 0 \quad \forall j \in [1, D] \quad (4.15)$$

$$\sum_{i=1}^U a_{ij} \leq k_D \quad \forall j \in [1, D] \quad (4.16)$$

$$\sum_{j=1}^D \sum_{k=1}^D \sqrt{(x_j - x_k)^2 + (y_j - y_k)^2 + (z_j - z_k)^2} \geq \theta \quad \forall i, j \in [1, D] \quad (4.17)$$

This problem attempts to optimize the deployment of UAVs while meeting different constraints related to the computational tasks, time limits, load balancing, computational resources available and power constraints.

The objective function in (4.8) minimizes the number of deployed UAVs. This is done by simply minimizing the summation of the whole matrix representing whether or not a UAV  $j$  is deployed or not.

The first constraint represented in (4.9) ensures that an IoT device  $i$  is associated and serviced by only a deployed UAV  $j$ . Thus, a device is not associated

with a UAV if it is not picked to be deployed in the network. Devices are only associated with deployed UAVs. So, if UAV  $j$  is not deployed then, no devices could be associated to it. In this case,  $d_j$  is set to be zero and devices should not be associated to it. When  $d_j$  is set to zero then variable  $a_{i,j}$  should be also set to zero. Indeed, this is achieved by the inequality in equation (4.9) that forces the value of  $a_{i,j}$  to be always less than or equal to  $d_j$ . On the other hand, if UAV  $j$  is deployed then a device  $i$  can be associated with UAV  $j$ . In this case,  $d_j$  is set to one and  $a_{i,j}$  could be set to one or zero indicating whether a device is associated with UAV  $j$  or not.

The constraint in (4.10) ensures that the processing UAV of a device  $i$  is also a deployed UAV  $j$ . The logic applied here is similar to the one explained in (4.9). Constraints in (4.11) and (4.12) enforce each device  $i$  to be assigned only to one serving UAV and one processing UAV respectively. It is possible that one UAV acts as both a serving UAV and a processing UAV to the same device. Constraint (4.13) ensures that the received power of IoT device  $i$  at its serving UAV  $j$  exceeds a target value. This constraint is particularly responsible of ensuring a minimum QoS. Devices should be served by a UAV where a good quality connection could be established to ensure a pleasant experience and good service.

The next constraint in (4.14) limits the total time experienced by tasks of each IoT device  $i$  to a maximum value  $T_i$  that resembles the desired deadline of the respective tasks. As demonstrated in (4.1), this delay constitutes different components including the upload time to the serving UAV  $j$ , exchange delay between serving UAV  $i$  and processing UAV  $k$ , processing time at the cloudlet UAV  $k$ , and finally the download time. The upload time is calculated based on (4.2) where  $N$  in this case is  $\sum_{i=1}^U a_{ij}$ . The second component of (4.14) determines the total time the task spends at the processing UAV as in (4.3) while the last component resembles  $d_{U2U,jk}$  that is nonzero whenever the serving UAV is different than the cloudlet UAV. So, we actually consider the UAV to UAV delay whenever there is a relay UAV called the serving UAV and another processing UAV called the

cloudlet UAV. Thus, to correctly calculate the U2U delay we should consider the case when a device task is being processed by a cloudlet UAV but not served by it at the same time. Hence in this case we should take the inverse of  $a_{i,j}$  to make sure that the device  $i$  is not served by UAV  $j$  and we consider  $b_{i,j}$  to ensure that device  $i$  traffic is being processed by UAV  $j$ . We introduce the summation  $\sum_{k=1}^D a_{ik} t_{U2U,jk}$  to get the delay between both the serving UAV and the cloudlet UAV.

To explain this more we are going to take a specific example. Suppose device 1 is served by UAV 1 and its task are also processed by UAV 1. In this case the serving and the processing UAVs are the same UAVs. So,  $a_{1,1}$  will evaluate to 1 and thus we will consider the transmission time represented by  $t_{up,11}$ . Also,  $b_{1,1}$  will evaluate to 1 since the cloudlet and serving UAVs are the same UAVs and thus we will consider the processing delay represented by  $\frac{b_{11}}{\frac{f_j}{L} - \sum_{n=1}^U b_{n1} \lambda_n}$ . On the other hand the third delay component representing the UAV to UAV delay is always set to zero this is because  $b_{1,j}$  is always set to zero for all  $j$  from 2 till  $U$ .

In another example, let us consider a device 2 that is being served by UAV 1 but its task are being processed and computed by UAV cloudlet 3. In this case  $a_{1,2}$  will evaluate to 1 and  $b_{1,3}$  will also evaluate to 1. We will first consider the transmission delay incurred by sending the data to UAV 2. This is actually achieved by the first term  $a_{1,2} t_{up,12}$ . Then we also consider the processing delay of UAV 2 as reflected by the second term  $\frac{b_{13}}{\frac{f_j}{L} - \sum_{n=1}^U b_{n1} \lambda_n}$ . At the end, we consider the UAV to UAV delay between UAV 2 and UAV 3. This is reflected in the last term where  $a_{1,3}$  is set to zero and thus  $\bar{a}_{1,3}$  will evaluate to 1.

The constraint represented in (4.15) ensures a stable queuing system at the processing UAV where the arrival rate of requests does not exceed the service rate of the UAV. Constraint (4.16) makes sure that each serving UAV does not exceed its user capacity  $k_D$ . Thus the number of devices served by a UAV  $j$  is always restricted to a threshold of value  $k_D$ . This is achieved by summing the association matrix of UAV  $j$  and making sure that the whole some of the

associated and connected devices does not exceed  $k_D$ . The final constraint (4.17) is crucial to separate the positions of the deployed UAVs  $j$  and  $k$  by a minimum distance  $\theta$ .

The formulated problem has a very high complexity due to the high number of variables and constraints. We can easily show that our problem is an Np-hard problem by dividing it into two problems. The first sub-problem is the UAV deployment problem and the second sub-problem is the task offloading and assignment mission. These first problem can be proven NP-hard by reducing it form the capacitated facility location problem that is well known to be Np-hard [80]. This problems considers a number of facilities that have to be deployed in order to serve a number of customers. We can reduce our problem form the facility location problem by considering the UAVs as the facilities and the IoT devices as the customers. The second problem can be proven Np-hard by reducing it form the assignment problem that is also well known to be Np-hard [81]. This problem requires the distribution of specific objects to different bins. We can consider the objects as the IoT tasks and the cloudlet UAVs as the bins. Hence, it is very hard to obtain an optimal solution to our problem and thus we have top resort to efficient approximation algorithms.

#### 4.2.5 Ion Motion Algorithm

Due to the high complexity of the formulated problem, in this section we propose a meta-heuristic approach that iteratively searches for the best number and positions of UAVs in addition to the device and task associations . The suggested Meta-heuristic provides an efficient, effective,and reliable search algorithm that is based on the rules of chemistry and physics.

The ion motion algorithm was first suggested in [82] and is derived from the natural behavior of ions within different environments. It is a population based meta-heuristic where candidate solutions are represented as anions and cations. Anions are ions assigned negative charges and cations are ions assigned

positive charges. Half of the population is considered as anions and the other half is considered as cations. When all these ions exist in the same environment, they form an electrically charged field where unlike ions attract and likewise ions repel. So, in an electrically charged environment anions move towards cations and cations move towards anions. However, the behaviour of these ions and particles depend on the states of matters in which these particles are formed in. Mainly in this algorithm we consider two states the liquid and solid state. Thus, the proposed algorithm is based on the behavior of ions in liquid and solid states. Below we represent the different phases of the suggested meta-heuristic.

### Initialization

Considering the three variables  $d_j$ ,  $a_{ij}$ , and  $b_{ij}$  we initialize our population and divide them equally into anions represented by  $AI$  and cations represented by  $CI$ . So each  $AI$  represents a single solution and this solution is represented by a single matrix containing  $P$ , the positions of each UAV,  $A$ , the association matrix representing which device is connected by which UAV, and  $B$  the association matrix representing which device is served by which UAV cloudlet. Each cation  $CI$  represents also a single solution and is composed of the same matrix described above. The population is initialized as follows:

$$AI_{ij} = Min + (Max_j - Min_j)r_1 \quad (4.18)$$

$$CI_{ij} = Min + (Max_j - Min_j)r_2 \quad (4.19)$$

Where the integer  $i$  represents the index of the ion and the integer  $j$  represents the dimension. The dimension is used to index the different variables in a single solution, the variables are the UAV positions and the two association matrices. The upper bounds of the different variables and association matrices are expressed by the Max values and the lower bounds are expressed by Min values according to the designated variables and matrices. The variables  $r_1$  and  $r_2$  are random

numbers that are chosen from the 0 till 1. The fitness value of each candidate solution is calculated according to (4.20)

$$\sum_{j=1}^D d_j + \sum_{j=1}^D f_j + \sum_{j=1}^U k_j \quad (4.20)$$

Where  $f_j$  and  $k_j$  is the number of cloudlet UAVs and IoT devices not satisfying the constraints represented in equations 4.9 till 4.17. The association between the devices and serving UAVs are calculated according to (4.21) and (4.22) where  $a_{kf}$  represents whether or not user  $k$  is associated and connected to UAV  $f$  and  $AI_{ij}$  is the anion representing solution number  $i$  and  $j$  is the index of the variable in the anion matrix. The same is for  $CI_{ij}$ . To infer the association between the devices and cloudlet UAVs the same calculations are done for  $b_{kf}$  representing whether or not a user  $k$  is served by cloudlet UAV  $f$ .

$$a_{kf} = \lfloor |AI_{ij}| \bmod 2 \rfloor \quad (4.21)$$

$$a_{kf} = \lfloor |CI_{ij}| \bmod 2 \rfloor \quad (4.22)$$

### Liquid Phase

In the liquid phase ions attract each other. So, ions are meant to be attracted to better ions of opposite charges. Cations will move towards the best anion and the anions move towards the best cations. Repulsion forces are neglected. This behavior is represented in Figure 22 We consider the distance the only factor affecting the force between opposite ions and thus the force  $FA_{ij}$  applied on anion  $i$  is calculated according to (4.23)

$$FA_{ij} = \frac{1}{1 + \exp^{\frac{-0.1}{DA_{ij}}}} \quad (4.23)$$

Where  $DA_{ij}$  is the distance between the anion and the best cation  $j$  calculated as  $|A_{ij} - BestC_j|$ .  $BestC_j$  represents the best cation that is the cation with the

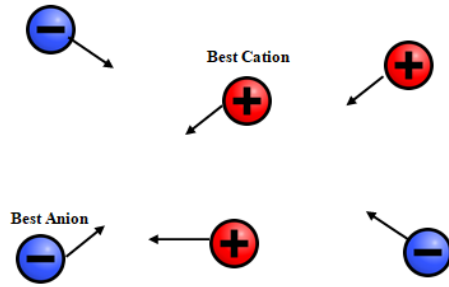


Figure 22: Behaviour of ions in liquid state

least fitness value. The forces on cations are calculated in a similar manner.

After calculating the forces applied on each ion the position of each anion is modified according to (4.24).

$$A_{ij} = A_{ij} + FA_{ij}(BestC_j - A_{ij}) \quad (4.24)$$

Similarly the forces and positions of cations are calculated according to the same behaviour.

### Solid Phase

After the liquid phase, the ions converge to an optimal solution but this convergence may get trapped in a local optima. To avoid getting stuck in a local optima we simulate the behavior of ions in the solid state. When the ions are in the solid state and if an external force is applied to them, the resultant force will crack this solid and the ions will move away from each other in random directions. This behavior is represented in Figure 23

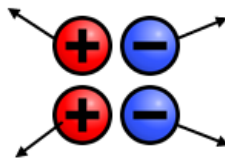


Figure 23: Behaviour of ions in solid state

The behavior of ions in solid phase is represented below in Algorithm 2 where

$\mu_1$  and  $\mu_2$  are random numbers between  $-1$  and  $1$  and  $\text{random}()$  returns a random number between  $0$  and  $1$ .  $BestCfit$  is the cation with the best fitness value and  $BestAfit$  is the anion with the best fitness value. On the other hand,  $WorstCfit$  represents the cation with the worst fitness value and  $WorstAfit$  is the anion with the worst fitness value.

---

**Algorithm 2** Solid phase

---

```

1: procedure SOLID-PHASE
2:   if ( $BestCfit \geq \frac{WorstCfit}{2}$  and  $BestAfit \geq \frac{WorstAfit}{2}$ ) then
3:     if ( $\text{random}() \geq 0.5$ ) then
4:        $AI_i = AI_i + \mu_1(BestC - 1)$ 
5:     else
6:        $AI_i = AI_i + \mu_1(BestC)$ 
7:     end if
8:     if ( $\text{random}() \geq 0.5$ ) then
9:        $CI_i = CI_i + \mu_2(BestA - 1)$ 
10:    else
11:       $CI_i = C_i + \mu_2(BestA)$ 
12:    end if
13:    if ( $\text{random}() \leq 0.05$ ) then
14:      Randomly re-initialize all  $A_i$  and  $C_i$ 
15:    end if
16:  end if
17: end procedure

```

---

### Termination

The algorithm stops whenever the number of iterations performed reaches the predefined maximum. So if the maximum number of iteration is not reached we re-enter the liquid state and then the solid state. If the maximum number of iterations is achieved the best feasible candidate solution is returned. The entire ion motion meta-heuristic is represented in Fig.24.

### 4.2.6 Simulation Results and Performance Analysis

In this section we present simulation results to evaluate the efficiency and reliability of the proposed solutions and algorithms. In addition, we study how different parameters effect our solution. We also study the effect of the requested task



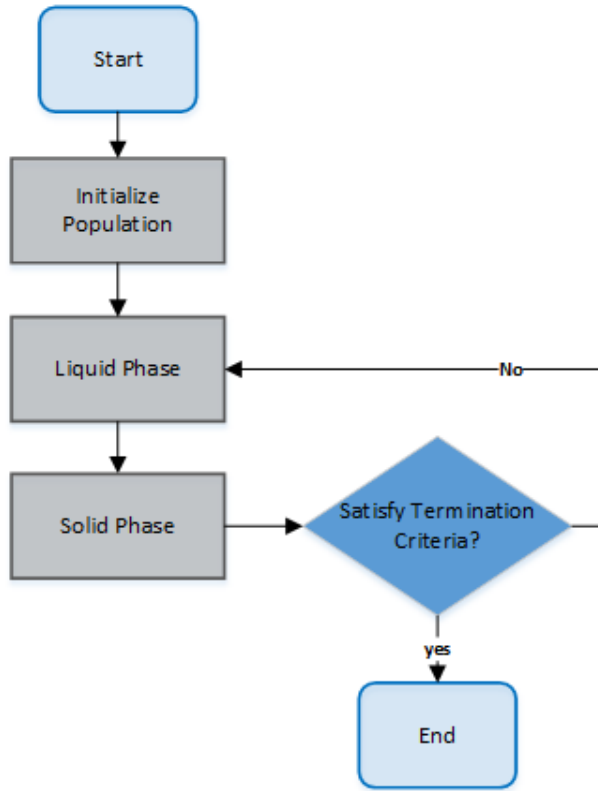


Figure 24: Ion motion algorithm flowchart.

types on the results.

We consider a 200 m x 200 m area where IoT devices are randomly deployed in the environment. We consider the same network parameters depicted in Table 3.1. First, we start by comparing the optimal solution to the meta-heuristic proposed algorithm. Fig. 25 shows the average number of deployed UAVs with respect to the number of IoT devices in the network. We can see that the meta-heuristic algorithm provides close-to-optimal results. In fact, our solutions are trying to optimize the number of UAVs by minimizing the deployed number of UAVs acting as both relays and UAV cloudlets. Minimizing the number of UAVs deployed in IoT environments leads to lower costs and better utilization of available resources.

Next, we deploy different applications on the IoT devices. We consider various industry verticals that have different time limits and unique characteristics. The utilized industry verticals in this section are presented in Table 4.1. We run our simulations using the meta-heuristic algorithm due to the high complexity of the optimal solution.

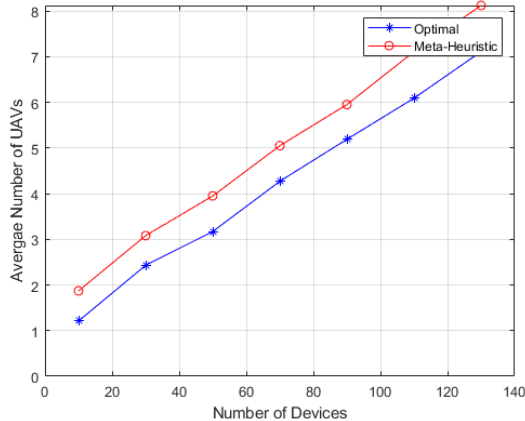


Figure 25: Average number of UAVs required to serve the IoT devices for the optimal solution compared to the meta-heuristic approach, as function of total number of devices.

Table 4.1: Industry verticals time deadlines [1]

Industry Vertical	Required Time Deadline	Applied Time Deadline
Smart Grid	3 – 20 ms	20 ms
Factory Automation	0.25 – 10 ms	10 ms
Tactile Internet	1 – 10 ms	5 ms

Smart grids are electricity grids that are usually used to measure energy consumption and to execute different services. Factory automation is related to deploying IoT systems in factories and plants. Different applications exist like automating the production line or tracking products throughout their manufacturing process until received by the customer. Tactile internet is a new vision that support ultra reliable low latency applications. These applications include hepatic communications where humans and machines are allowed to communicate together in real-time. A famous and important example and use case is remote surgery operations where the patient usually resides in the hospital and the physician is in another place performing the surgery on the patient by the use of automated technology and machines.

In Fig. 26 we plot the average number of deployed UAVs as a function of the number of IoT devices available in the network. We compare the three different verticals that have different time deadlines and characteristics. We clearly infer that as the number of devices are increasing and thus the number of requests

is increasing the number of deployed UAVs is also increasing. The number of utilized UAVs is highest for tactile internet as it requires the highest resources due to its stringent time limit and support.

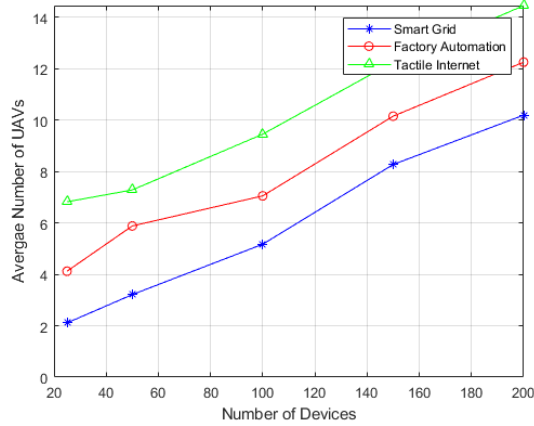


Figure 26: Average number of UAVs required to serve the IoT devices for the optimal solution compared to the meta-heuristic approach, as function of total number of devices and in different industry verticals.

In Fig.27 we study the average number of UAVs deployed for different verticals and according to different request rates.  $\lambda$  was varied between 60 and 100 requests/sec. The figure shows that as the the request rate increases the number of deployed UAVs also increases. In addition as the time deadline between the different vertical decreases the number of deployed UAVs increases. For example for a  $\lambda$  of value 60 requests/sec the smart grid vertical required the deployment of around 2.25 UAVs whereas the factory automation applications with lower time deadline equals to 10 ms the number of deployed UAVs was higher and approximately equals to 3.25.

To further study the effects of time deadlines and response times we plot in Fig.28 the average number of UAVs deployed in addition to the actual achieved time delay as a function of the time deadline. We vary the required time deadline between 1 and 10 ms. We can see that as the deadline increases the required number of UAVs decreases. Also the achieved response time is always greater than or approximately equal to the required time deadline.

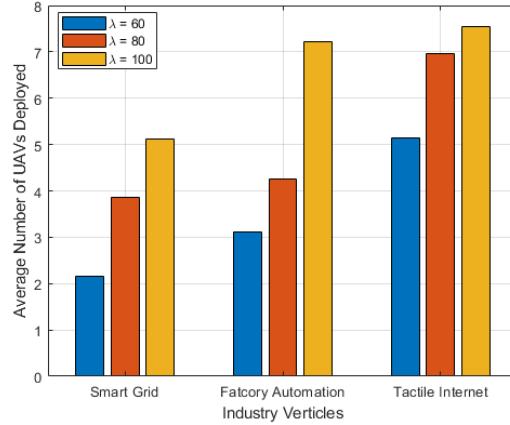


Figure 27: Average number of UAVs required to serve the IoT devices in different industry verticals, as function of requests rate.

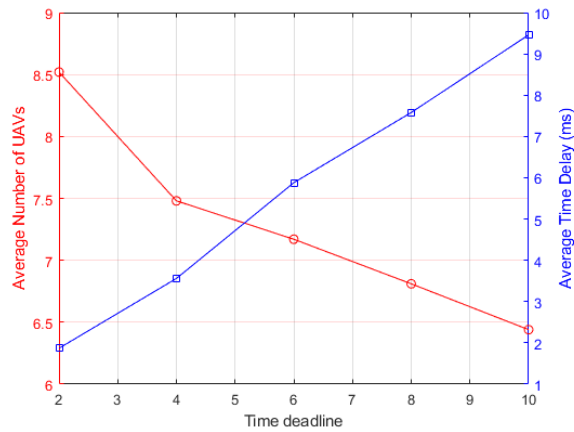


Figure 28: Average number of UAVs and achieved time delay, as function of time deadline in Tactile Internet applications.

#### 4.2.7 Conclusion

In this chapter, we have studied the 3D deployment of UAV-Mounted cloudlets in IoT networks to support latency-sensitive services and applications. We first formulate the 3D positioning problem as a mixed integer program. Then, we propose a meta-heuristic algorithm to approximate the solution in an efficient way. Results are presented as a function of various system parameter, and demonstrate close performance results compared to the optimal solution.

# Chapter Five

## Conclusion and Future Work

In this thesis, we leveraged the mobility, low cost, and flexibility of UAVs to deploy them as moving ABSs and cloudlets in wireless networks. We motivated the problems with interesting and real-life scenarios and use cases. Then, we discussed the main challenges of deploying these applications and implementing them in different environments.

We studied the problem of deploying UAVs in 3D space to offload traffic in wireless networks. We formulated the problem as a mixed integer program and then suggested an autonomous force based algorithm that adapts with user mobility. We showed the effectiveness of our solution by simulating different environments and while considering mobile users and dynamic environments. We also tested and implemented the force algorithm on a commercial drone and using a testbed setup. Our suggested solution proved to be reliable and efficient in comparison with the optimal solution and with related literature and other proposed solutions.

Motivated by the need of an efficient and fast solution to mitigate the effects of low power and computational constraints in IoT devices we then studied the 3D deployment of UAV-mounted cloudlets in IoT networks to support latency-sensitive applications. We formulated our problem as a mixed integer program and then we effectively solved it using an ion based meta-heuristic. We studied the performance of our solutions by simulating our algorithms on different IoT

environments and while changing the various system parameters. We tested our solution on different industry verticals and applications and then analyzed the effect of time deadlines and request rates.

Indeed, UAVs are promising solutions to be used in wireless networks to extend coverage, capacity, and constrained devices. Next, we aim to consider scheduling and resource allocation problems in UAV cloudlets to better adapt our solution according to incoming requests. We aim to study live scheduling of incoming devices to be served by UAV cloudlets. It is also interesting to study the movement of UAV cloudlets to adapt with user requests and changing time constraints. Another possible research direction is to consider energy consumption in UAVs where we should optimally aim to minimize energy consumed during the UAV service time.

# Bibliography

- [1] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel, *et al.*, “Latency critical iot applications in 5g: Perspective on the design of radio interface and network architecture,” *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, 2017.
- [2] S. Hayat, E. Yanmaz, and R. Muzaffar, “Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2624–2661, 2016.
- [3] A. R. Cross, “Drones for disaster response and relief operations,” 2015. [www.issuelab.org/resources/21683/21683.pdf](http://www.issuelab.org/resources/21683/21683.pdf).
- [4] HankPrice, “Federal aviation administration (faa) forecast fiscal years 2017-2038,” 2018. [https://www.faa.gov/news/fact\\_sheets/news\\_story.cfm?newsId=22594/](https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=22594/).
- [5] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, “A tutorial on uavs for wireless networks: Applications, challenges, and open problems,” *IEEE Communications Surveys & Tutorials*, 2019.
- [6] A. Fotouhi, H. Qiang, M. Ding, M. Hassan, L. G. Giordano, A. Garcia-Rodriguez, and J. Yuan, “Survey on uav cellular communications: Practical aspects, standardization advancements, regulation, and security challenges,” *IEEE Communications Surveys & Tutorials*, 2019.

- [7] S. Sharafeddine and R. Islambouli, “On-demand deployment of multiple aerial base stations for traffic offloading and network recovery,” *Computer Networks*, 2019.
- [8] R. Islambouli and S. Sharafeddine, “Autonomous 3d deployment of aerial base stations in wireless networks with user mobility,” in *2019 IEEE Symposium on Computers and Communications (ISCC)*, 2019.
- [9] G. Goh, S. Agarwala, G. Goh, V. Dikshit, S. L. Sing, and W. Y. Yeong, “Additive manufacturing in unmanned aerial vehicles (uavs): Challenges and potential,” *Aerospace Science and Technology*, vol. 63, pp. 140–151, 2017.
- [10] K. P. Valavanis and G. J. Vachtsevanos, *Handbook of unmanned aerial vehicles*. Springer, 2015.
- [11] N. H. Motlagh, T. Taleb, and O. Arouk, “Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives,” *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899–922, 2016.
- [12] J. George, P. Sujit, and J. B. Sousa, “Search strategies for multiple uav search and destroy missions,” *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1-4, pp. 355–367, 2011.
- [13] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, *et al.*, “Scenarios for 5g mobile and wireless communications: the vision of the metis project,” *IEEE Communications Magazine*, vol. 52, no. 5, pp. 26–35, 2014.
- [14] M. Mazur, A. Wisniewski, and J. McMillan, “Clarity from above: Pwc global report on the commercial applications of drone technology,” *Warsaw: Drone Powered Solutions, PriceWater house Coopers*, 2016.



- [15] S. Kaur and S. Randhawa, "Google loon: Balloon-powered internet for everyone," in *AIP Conference Proceedings*, vol. 2034, p. 020006, AIP Publishing, 2018.
- [16] T. Perry, "Facebook aims to remake telecom with millimeter waves and tether-tennas," *IEEE Spectrum*, Apr, 2017.
- [17] NOKIA, "F-cell technology from nokia bell labs revolutionizes small cell deployment by cutting wires, costs and time," Oct 2016. <https://www.nokia.com/news/releases/2016/10/03/f-cell-technologyfrom-nokia-bell-labs-revolutionizes-small-cell-deployment-by-cuttingwires-costs-and-time>.
- [18] Huawei, "Connected aerial vehicle live," May 2018. <https://www.huawei.com/en/industry-insights/outlook/mobile-broadband/xlabs/use-cases/mbbf2017-connected-aerial-vehicle-live>.
- [19] S. Norouzi Ghazbi, Y. Aghli, M. Alimohammadi, and A. Akbari, "Quadrotors unmanned aerial vehicles: A review.," *International Journal on Smart Sensing & Intelligent Systems*, vol. 9, no. 1, 2016.
- [20] K. Daniel and C. Wietfeld, "Using public network infrastructures for uav remote sensing in civilian security operations," tech. rep., DORTMUND UNIV (GERMANY FR), 2011.
- [21] I. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (fanets): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.
- [22] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.
- [23] R. T. C. for Aeronautics (RTCA), "Minimum operational performance standards for unmanned aircraft systems," May 2013. <https://www.rtca.org/content/sc-228>.

- [24] A. A. Khuwaja, Y. Chen, N. Zhao, M.-S. Alouini, and P. Dobbins, “A survey of channel modeling for uav communications,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2804–2821, 2018.
- [25] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal lap altitude for maximum coverage,” *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 569–572, 2014.
- [26] S. K. Singh *et al.*, “A comprehensive survey on fanet: challenges and advancements,” *Int. J. Comput. Sci. Inf. Technol*, vol. 6, no. 3, pp. 2010–2013, 2015.
- [27] A. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, “Cyber security threat analysis and modeling of an unmanned aerial vehicle system,” in *2012 IEEE Conference on Technologies for Homeland Security (HST)*, pp. 585–590, IEEE, 2012.
- [28] M. Erdelj, E. Natalizio, K. R. Chowdhury, and I. F. Akyildiz, “Help from the sky: Leveraging uavs for disaster management,” *IEEE Pervasive Computing*, vol. 16, no. 1, pp. 24–32, 2017.
- [29] I. Bor-Yaliniz and H. Yanikomeroğlu, “The new frontier in ran heterogeneity: Multi-tier drone-cells,” *IEEE Communications Magazine*, vol. 54, no. 11, pp. 48–55, 2016.
- [30] L. Gupta, R. Jain, and G. Vaszkun, “Survey of important issues in uav communication networks,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1123–1152, 2016.
- [31] S. Chandrasekharan, K. Gomez, A. Al-Hourani, S. Kandeepan, T. Rasheed, L. Goratti, L. Reynaud, D. Grace, I. Bucaille, T. Wirth, *et al.*, “Designing and implementing future aerial communication networks,” *IEEE Communications Magazine*, vol. 54, no. 5, pp. 26–34, 2016.

- [32] K. Gomez, A. Hourani, L. Goratti, R. Riggio, S. Kandeepan, and I. Bucaille, “Capacity evaluation of aerial lte base-stations for public safety communications,” in *2015 European Conference on Networks and Communications (EuCNC)*, pp. 133–138, IEEE, 2015.
- [33] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, “Efficient 3-d placement of an aerial base station in next generation cellular networks,” in *2016 IEEE international conference on communications (ICC)*, pp. 1–5, IEEE, 2016.
- [34] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, “Drone small cells in the clouds: Design, deployment and performance analysis,” in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, IEEE, 2015.
- [35] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, “Placement optimization of uav-mounted mobile base stations,” *IEEE Communications Letters*, vol. 21, no. 3, pp. 604–607, 2017.
- [36] X. Zhang and L. Duan, “Optimal deployment of uav networks for delivering emergency wireless coverage,” *arXiv preprint arXiv:1710.05616*, 2017.
- [37] M. Alzenad, A. El-Keyi, F. Lagum, and H. Yanikomeroglu, “3-d placement of an unmanned aerial vehicle base station (uav-bs) for energy-efficient maximal coverage,” *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 434–437, 2017.
- [38] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, “3-d placement of an unmanned aerial vehicle base station for maximum coverage of users with different qos requirements,” *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 38–41, 2018.
- [39] J. Sun and C. Masouros, “Deployment strategies of multiple aerial bss for user coverage and power efficiency maximization,” *IEEE Transactions on Communications*, 2018.

- [40] J. Sun and C. Masouros, “Energy aware trajectory optimization for aerial base stations,” *arXiv preprint arXiv:1901.07938*, 2019.
- [41] H. Bayerlein, P. De Kerret, and D. Gesbert, “Trajectory optimization for autonomous flying base station via reinforcement learning,” in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, IEEE, 2018.
- [42] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, “Trajectory design and power control for multi-uav assisted wireless networks: a machine learning approach,” *arXiv preprint arXiv:1812.07665*, 2018.
- [43] R. Ghanavi, E. Kalantari, M. Sabbaghian, H. Yanikomeroglu, and A. Yongacoglu, “Efficient 3d aerial base station placement considering users mobility by reinforcement learning,” in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, IEEE, 2018.
- [44] H. J. Carlo, F. Aldarondo, P. M. Saavedra, and S. N. Torres, “Capacitated continuous facility location problem with unknown number of facilities,” *Engineering Management Journal*, vol. 24, no. 3, pp. 24–31, 2012.
- [45] L. A. Meira and F. K. Miyazawa, “A continuous facility location problem and its application to a clustering problem,” in *Proceedings of the 2008 ACM symposium on Applied computing*, pp. 1826–1831, ACM, 2008.
- [46] B. Sidhu and H. Singh, “Location management in cellular networks,” *World Academy of Science, Engineering and Technology*, vol. 25, pp. 314–319, 2007.
- [47] B. Peng and L. Li, “An improved localization algorithm based on genetic algorithm in wireless sensor networks,” *Cognitive Neurodynamics*, vol. 9, no. 2, pp. 249–256, 2015.
- [48] X. Liu, S. Zhang, and K. Bu, “A locality-based range-free localization algorithm for anisotropic wireless sensor networks,” *Telecommunication Systems*, vol. 62, no. 1, pp. 3–13, 2016.

- [49] Y. Qu and Y. Zhang, “Cooperative localization against gps signal loss in multiple uavs flight,” *Journal of Systems Engineering and Electronics*, vol. 22, no. 1, pp. 103–112, 2011.
- [50] L. Cheng, C. Wu, Y. Zhang, H. Wu, M. Li, and C. Maple, “A survey of localization in wireless sensor network,” *International Journal of Distributed Sensor Networks*, vol. 8, no. 12, p. 962523, 2012.
- [51] C. Bettstetter, “Mobility modeling in wireless networks: categorization, smooth movement, and border effects,” *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 3, pp. 55–66, 2001.
- [52] V. C. Zandy and B. P. Miller, “Reliable network connections,” in *Proceedings of the 8th annual international conference on Mobile computing and networking*, pp. 95–106, ACM, 2002.
- [53] I. M. Moraes, M. E. M. Campista, L. H. M. Costa, O. C. M. Duarte, J. L. Duarte, D. G. Passos, C. V. N. de Albuquerque, and M. G. Rubinstein, “On the impact of user mobility on peer-to-peer video streaming,” *IEEE Wireless Communications*, vol. 15, no. 6, pp. 54–62, 2008.
- [54] M. K. Khandani, P. Saeedi, Y. P. Fallah, and M. K. Khandani, “A novel data clustering algorithm based on electrostatic field concepts,” in *Computational Intelligence and Data Mining, 2009. CIDM’09. IEEE Symposium on*, pp. 232–237, IEEE, 2009.
- [55] R. R. Choudhury, X. Yang, R. Ramanathan, and N. H. Vaidya, “Using directional antennas for medium access control in ad hoc networks,” in *Proceedings of the 8th annual international conference on Mobile computing and networking*, pp. 59–70, ACM, 2002.
- [56] K.-H. Chiang and N. Shenoy, “A 2-d random-walk mobility model for location-management studies in wireless networks,” *IEEE Transactions on Vehicular Technology*, vol. 53, no. 2, pp. 413–424, 2004.

- [57] M. M. Ishaque and R. B. Noland, “Behavioural issues in pedestrian speed choice and street crossing behaviour: a review,” *Transport Reviews*, vol. 28, no. 1, pp. 61–85, 2008.
- [58] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, “Placement optimization of uav-mounted mobile base stations,” *IEEE Communications Letters*, vol. PP, no. 99, pp. 1–1, 2016.
- [59] R. Cohen-Almagor, “Internet history,” in *Moral, Ethical, and Social Dilemmas in the Age of Technology: Theories and Practice*, pp. 19–39, IGI Global, 2013.
- [60] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, “Context aware computing for the internet of things: A survey,” *IEEE communications surveys & tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
- [61] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [62] S. Li, L. Da Xu, and S. Zhao, “The internet of things: a survey,” *Information Systems Frontiers*, vol. 17, no. 2, pp. 243–259, 2015.
- [63] A. Whitmore, A. Agarwal, and L. Da Xu, “The internet of things a survey of topics and trends,” *Information Systems Frontiers*, vol. 17, no. 2, pp. 261–274, 2015.
- [64] H. U. Rehman, M. Asif, and M. Ahmad, “Future applications and research challenges of iot,” in *2017 International Conference on Information and Communication Technologies (ICICT)*, pp. 68–74, IEEE, 2017.
- [65] K. K. Patel, S. M. Patel, *et al.*, “Internet of things-iot: definition, characteristics, architecture, enabling technologies, application & future challenges,” *International journal of engineering science and computing*, vol. 6, no. 5, 2016.

- [66] M. Meeker and L. Wu, “Internet trends 2018,” 2018.
- [67] S.-H. Lee and D.-W. Lee, “Changes in distribution industry using information and communication technology,” *JOURNAL OF ADVANCED INFORMATION TECHNOLOGY AND CONVERGENCE*, vol. 8, no. 1, pp. 23–30, 2018.
- [68] Y.-K. Chen, “Challenges and opportunities of internet of things,” in *17th Asia and South Pacific design automation conference*, pp. 383–388, IEEE, 2012.
- [69] D. Bandyopadhyay and J. Sen, “Internet of things: Applications and challenges in technology and standardization,” *Wireless personal communications*, vol. 58, no. 1, pp. 49–69, 2011.
- [70] L. Tong, Y. Li, and W. Gao, “A hierarchical edge cloud architecture for mobile computing,” in *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE*, pp. 1–9, IEEE, 2016.
- [71] X. Sun and N. Ansari, “Edgeiot: Mobile edge computing for the internet of things,” *IEEE Communications Magazine*, vol. 54, no. 12, pp. 22–29, 2016.
- [72] X. Sun and N. Ansari, “Latency aware workload offloading in the cloudlet network,” *IEEE Communications Letters*, vol. 21, no. 7, pp. 1481–1484, 2017.
- [73] T. X. Tran and D. Pompili, “Joint task offloading and resource allocation for multi-server mobile-edge computing networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 1, pp. 856–868, 2019.
- [74] M. Jia, W. Liang, and Z. Xu, “Qos-aware task offloading in distributed cloudlets with virtual network function services,” in *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, pp. 109–116, ACM, 2017.

- [75] P. J. Braun, S. Pandi, R.-S. Schmoll, and F. H. Fitzek, “On the study and deployment of mobile edge cloud for tactile internet using a 5g gaming application,” in *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 154–159, IEEE, 2017.
- [76] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, “Mobile unmanned aerial vehicles (uavs) for energy-efficient internet of things communications,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 11, pp. 7574–7589, 2017.
- [77] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, “Mobile internet of things: Can uavs provide an energy-efficient mobile architecture?,” in *Global Communications Conference (GLOBECOM), 2016 IEEE*, pp. 1–6, IEEE, 2016.
- [78] S. Jeong, O. Simeone, and J. Kang, “Mobile edge computing via a uav-mounted cloudlet: Optimization of bit allocation and path planning,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2049–2063, 2018.
- [79] N. Cheng, W. Xu, W. Shi, Y. Zhou, N. Lu, H. Zhou, *et al.*, “Air-ground integrated mobile edge networks: Architecture, challenges and opportunities,” *arXiv preprint arXiv:1804.04763*, 2018.
- [80] L.-Y. Wu, X.-S. Zhang, and J.-L. Zhang, “Capacitated facility location problem with general setup cost,” *Computers & Operations Research*, vol. 33, no. 5, pp. 1226–1241, 2006.
- [81] S. O. Krumke and C. Thielen, “The generalized assignment problem with minimum quantities,” *European Journal of Operational Research*, vol. 228, no. 1, pp. 46–55, 2013.
- [82] B. Javidy, A. Hatamlou, and S. Mirjalili, “Ions motion algorithm for solving optimization problems,” *Applied Soft Computing*, vol. 32, pp. 72–79, 2015.