

# DOCTORAL THESIS

---

## Obstacle Detection and Infrastructure Deployment for Efficient Millimeter-wave Communications

---

*A thesis submitted to the Indian Statistical Institute  
in partial fulfilment of the requirements for the degree of  
Doctor of Philosophy (in Computer Science)*

by

**Subhojit Sarkar**



**Supervisor :**

**Prof. Sasthi C. Ghosh**

Advanced Computing & Microelectronics Unit

Indian Statistical Institute, Kolkata

December 2023



*To Dip, Bua, Mukut, Ryan, Subho, and Kutush...*

*and to Dida, who left me midway through the thesis*



---

# *Acknowledgements*

Majority of this thesis is a result of relentless efforts by my supervisor Prof. Sasthi C. Ghosh, who refused to give up on me even when I had reached the lowest point of my career. I can never thank him enough for persistently calling me, even when I stubbornly refused to pick his calls. Any other supervisor in his place would have thrown me out, he did not. It is because of his endless efforts that I could start afresh, and can submit this thesis. I am ever grateful to him for bearing with me, for giving me invaluable input, and for his constant encouragement. I hope more and more supervisors are like him.

I am thankful to my co-authors for all their input and help. I would like to extend my heartfelt gratitude to all the faculty members of the Advanced Computing and Microelectronics Unit at Indian Statistical Institute for all their support. In particular, I would like to thank Prof. Nabanita Das (retd.) for her wise words of encouragement. She would come to Lab 507 just to talk to me, discuss my worries, and provide me ever encouraging advice, which I tried to follow along, but failed most of the time. My sincere gratitude to the ACMU administrative staff, particularly Sachi da, Suchitra di, Rakesh da, and Rajen da for all the help they provided, always with a smile. I thank the maintenance staff at ISI for all their support; a special thanks to the security staff whom I badgered to open and close the unit long outside elevator service hours.

A PhD journey is not complete without fellow research scholars. I am grateful to have had the best seniors that a junior researcher could ask for, particularly Avirup da (now Dr. Avirup Das). From discussing photography to short film making, from politics to tourism, we shared a lot of non-research based discussions, along with the more serious stuff. My sincere thanks to my fellow scholars Satya da, Pratham da, Priyadarshi da, Soumen, Abhinav, Arun, Laxmikanta, Bibhuti, Sankar, and Rathin for all the good times we had, for all the discussions over tea at Sahadev da's shop, that threw up new problems, and subsequently, solutions. Bibhuti, Laxmikanta, Rathin, and myself formed a special quad in the later years of the journey. A special thanks to Rathin (incidentally a co-author to a few of my works in this thesis) for all his input, and especially his coding expertise. He has remained my fail-safe backend support for a lot of my simulations, and  $\text{\LaTeX}$ formatting. I can never thank him enough.

I have been lucky to have had a few very close friends outside ISI, who have been friends unconditionally. To Labonno, Sayana, Anirban, Deboleen, Supritam, Shibashish, Srinjoy, Soham... from the bottom of my heart, thanks.

I have been plagued by severe mental health issues for a majority of my tenure as a PhD scholar. The pandemic induced lockdown only worsened my condition. I heartily thank my psychiatrist Dr. Rajarshi Neogi (Associate Professor, R. G. Kar Medical College, Kolkata), and my therapist Dr. Nilanjana Sanyal (retd. Professor, Department of Psychology, Calcutta University) for their help and guidance during the

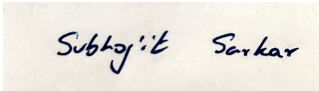
treatment. Nilanjana ma'am has been an absolute asset; her words have been kind and warm, managing to fill the darkness of my mind with rekindled hope. Dr. Rezaul Karim (former Professor, Department of Radiology, West Bengal University of Health Sciences) has been an absolute gem, always providing a calm, and reassuring voice when I was stressed. Mrs. Suparna Bhattacharya has been another source of encouragement, and rock solid support; I cannot express enough, in words, my gratitude and love for her. I take this opportunity to mention that mental health is a serious disease, though sadly often neglected due to societal pressure. I have been lucky to have had the best help possible, that helped me on the road to recovery; I am trying to recover even now.

There have been several outstanding resources available online, which made my life easier. Communities such as [StackOverflow](#) have literally been lifesavers. Apart from that, websites that promote open source publication of research, provided invaluable support. I am ever grateful.

A few of my family members have been really supportive throughout the journey. Special mention should be made of my uncle Mr. Anjan Sarkar, my cousins Debabrata and Dhruvajyoti Sarkars, and my sisters in law Smita Sarkar, and Rumki Sarkar. Along with them, Professors Ajoy Dutta and Subir Debnath (both faculty members, Department of Production Engineering, Jadavpur University) have provided invaluable support. I am ever grateful to Mrs. Dipali Das and Mrs. Tapati Dutta Choudhury for the love they have for me.

My father Prof. Bijan Sarkar (Head, Department of Production Engineering, and former Dean, Faculty of Engineering and Technology, Jadavpur University) has been a guiding light in his own way, cheering for me all along. He has been my biggest philosopher, and guide for all my life. I hope to make him proud of me again, one day. My mother Nila Sarkar has been my strictest teacher, and my biggest well-wisher, with my best interests at heart. Her advice, although not always correct, has made me the person I am today. Her sacrifices have been many, just to bring me up. Like with baba, one day I hope, she will again be proud of me, and smile like she used to.

And lastly, I am grateful to my best friend Rituparna for not giving up on me. I am a difficult person to deal with. She has been doing so for a long time. She remains my strongest cheerleader, my rock solid support, my Patronus. I hope one day I prove to be worthy of her.



Subhojit Sarkar

Subhojit Sarkar

Kolkata

July, 2023

# Abstract

There has been growing interest in millimeter-wave (mmWave) communication due to the promising high speeds and immense amounts of unused bandwidth available. However, mmWaves suffer from unusually high attenuation, through free space, and especially through obstacles, which necessitates an obstacle free line-of-sight (LOS) transmission path. This thesis deals with establishment of such LOS paths, through obstacle detection and deployment of network infrastructure.

The usual approach to avoid static obstacles on transmission paths is to use satellite imagery to detect the presence of static obstacles, an approach which apart from raising proprietary concerns, is not able to capture smaller obstacles. We propose a simple learning based approach to detect the presence of static as well as dynamic obstacles, without having apriori access to any data regarding their location from satellite imagery. We then use this knowledge to efficiently select an appropriate transmission path for a user equipment (UE), lowering the chance of allocating an obstacle prone link.

Dynamic obstacles are usually tracked by dedicated tracking hardware like RGB-D cameras, which usually have small ranges, and hence lead to prohibitively increased deployment costs to achieve complete camera coverage of the deployment area. We propose an altogether different approach to track dynamic obstacles in an mmWave network, solely based on short-term historical link failure information, without resorting to any dedicated tracking hardware. Using the obtained trajectories, we perform proactive handoffs for at-risk links. We compare our approach with an RGB-D camera-based approach and show that our approach provides better handoff performances when the camera coverage is low to moderate, which is often the case in real deployment scenarios.

Stability of allocated transmission paths is an important problem in the domain of mmWave communication. The quality of an allocated transmission path depends not only upon the present time, but also upon the maintenance of the said path in the near future; the fragile nature of mmWaves necessitates this. Thus, allocating the base station (BS) which provides the highest received signal strength (RSS) at the current time instant is not always the best idea, considering UE mobility, and presence of obstacles. We propose a simple geometric approach

to allocate stable transmission paths which are less likely to be broken in the near future.

One way to deal with obstacle free strict LOS requirements of mmWaves is to densely deploy small range mmWave BSs, to overcome outage due to obstacles. Low cost reflectors have also been proposed to augment the transmission environment, and reflect mmWaves in the desired direction, thereby bypassing the obstacles. We argue that considering the placement of mmWave BSs and reflectors independently may lead to suboptimal coverage. We consider an urban deployment scenario, and attempt to maximally cover it by jointly placing the mmWave BSs and reflectors. Given the hardness of the joint problem, we first develop a set cover based greedy solution, and also provide a linear programming (LP) relaxation based solution. With extensive simulations, we show that with a fixed number of available mmWave BSs and reflectors to be placed, both our proposed solutions achieve a larger coverage compared to an existing approach where BSs and reflectors were placed sequentially.

Unmanned Aerial Vehicles (UAVs) are a potential platform for deploying mmWave BSs. One challenge that has to be addressed is the limited power onboard a UAV, which is used to hover and move the device, and of course, to transmit data. We deal with the deployment of UAVs with an aim to minimise their displacement in subsequent time instances. We take into consideration UE mobility, and propose LazyUAV, a set cover based geometric approach to minimise UAV displacement, while maintaining maximal coverage.



## List of publications from the content of the thesis

### In Journals :

1. **Subhojit Sarkar** and Sasthi C. Ghosh. “Relay selection in millimeter wave D2D communications through obstacle learning” in: *Ad Hoc Networks (Elsevier)*, Vol. 114, pp 102419, January 2021, ISSN 1570-8705, DOI: 10.1016/j.adhoc.2021.102419.

### In Conferences :

1. **Subhojit Sarkar** and Sasthi C. Ghosh. “Relay selection in millimeter wave D2D communications through obstacle learning” In: Proceedings of the *12th International Conference on COMMunication Systems & NETWORKS (COMSNETS 2020)*, IEEE, Bengaluru, India, January 7–11, 2020, pp. 468-475, DOI: 10.1109/COMSNETS48256.2020.9027458.
2. **Subhojit Sarkar**, Subhankar Ghosal, Subhadip Bandyopadhyay and Sasthi C. Ghosh. “A Stable Link Allocation Algorithm for 5G Millimeterwave Networks.” In: Proceedings of the *15th International Conference on COMMunication Systems & NETWORKS (COMSNETS 2023)* IEEE, Bengaluru, India, January 3-8, 2023, pp. 674-681, DOI: 10.1109/COMSNETS56262.2023.10041333.
3. **Subhojit Sarkar** and Sasthi C. Ghosh. “Mobility Aware Path Selection for Millimeterwave 5G Networks in the Presence of Obstacles”. In: Proceedings of the *3rd International Conference on Computer and Communication Engineering (CCCE 2023)*, Stockholm, Sweden, March 10-12, 2023 pp. 67-80, Communications in Computer and Information Science, vol 1823. Springer, Cham., DOI: 10.1007/978-3-031-35299-7\_6.
4. Rathindra Nath Dutta, **Subhojit Sarkar** and Sasthi C. Ghosh. “Joint Base Station and Reflector Placement in an urban mmWave Network”. In: *3rd International Mediterranean Conference on Communications and Networking (MeditCom 2023)*, IEEE, Dubrovnik, Croatia, September 4–7 2023, DOI: 10.1109/MEDITCOM58224.2023.10266650.
5. **Subhojit Sarkar**, Rathindra Nath Dutta and Sasthi C. Ghosh. “LazyUAV: A Minimal Displacement Coverage Strategy for Multi-UAV mmWave Networks” In: *3rd International Mediterranean Conference on Communications and Networking (MeditCom 2023)*, IEEE, Dubrovnik, Croatia, September 4–7 2023, DOI: 10.1109/MEDITCOM58224.2023.10266593.

### Under preparation :

1. Rathindra Nath Dutta, **Subhojit Sarkar** and Sasthi C. Ghosh. “Doing Away With Cameras: A Dynamic Obstacle Tracking Strategy for Proactive Handoffs in mmWave Networks”. To be communicated soon. (Draft available at : <https://arxiv.org/pdf/2305.00429.pdf>)



# Contents

<b>Acknowledgements</b>	<b>v</b>
<b>Abstract</b>	<b>vii</b>
<b>Contents</b>	<b>ix</b>
<b>List of Figures</b>	<b>xiv</b>
<b>List of Tables</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background and Fundamentals . . . . .	1
1.2 mmWave Communication: The Pros and Cons . . . . .	6
1.3 Some Approaches in mmWave Domain . . . . .	9
1.4 Research Gap and Motivation . . . . .	15
1.5 Contribution of the Thesis . . . . .	15
1.6 Outline of the Thesis . . . . .	20
<b>2 Literature Review</b>	<b>21</b>
2.1 Obstacle Aware Communication . . . . .	21
2.2 Infrastructure Deployment . . . . .	25
2.3 Research Gap and Our Contribution . . . . .	28
<b>I All about Obstacles</b>	<b>31</b>
<b>3 Link Selection in mmWave Networks through Obstacle Learning</b>	<b>33</b>
3.1 Overview . . . . .	33
3.2 System Model and Assumptions . . . . .	34
3.3 Learning Based Relay Selection Algorithm . . . . .	39
3.4 Simulation Results . . . . .	48
3.5 Conclusion . . . . .	55
<b>4 Tracking Dynamic Obstacles using Historical Link Failures</b>	<b>57</b>
4.1 Overview . . . . .	57
4.2 System Model and Assumptions . . . . .	58

4.3	Tracking A Single Dynamic Obstacle . . . . .	60
4.4	Tracking Multiple Dynamic Obstacles . . . . .	65
4.5	Problem Formulation . . . . .	66
4.6	Discovery and Implementation Algorithms . . . . .	70
4.7	Simulation Experiments . . . . .	74
4.8	Conclusion . . . . .	78
<b>5</b>	<b>Nearest is not the best! Towards Stable Link Allocation</b>	<b>81</b>
5.1	System Model . . . . .	83
5.2	Problem Formulation . . . . .	84
5.3	Proposed Stable Path Finding Algorithm . . . . .	88
5.4	Simulation Results . . . . .	91
5.5	Conclusion . . . . .	92
<b>II</b>	<b>Efficient Resource Placement</b>	<b>95</b>
<b>6</b>	<b>Joint Placement of Base Stations and Reflectors in Urban mmWave Networks</b>	<b>97</b>
6.1	Overview . . . . .	97
6.2	System Model . . . . .	99
6.3	Problem Formulation . . . . .	101
6.4	Joint Placement Algorithms . . . . .	104
6.5	Simulation Results . . . . .	108
6.6	Conclusion . . . . .	111
<b>7</b>	<b>LazyUAV: A Minimal Displacement Coverage Strategy for Multi-UAV mmWave Networks</b>	<b>113</b>
7.1	Overview . . . . .	113
7.2	System Model . . . . .	115
7.3	Proposed Approach: LazyUAV . . . . .	116
7.4	Simulation Experiments . . . . .	125
7.5	Conclusion . . . . .	127
<b>8</b>	<b>Conclusion and Future Directions</b>	<b>129</b>

# Acronyms

**5G** fifth generation.

**AR** augmented reality.

**BS** base station.

**D2D** device to device.

**FANET** flying ad hoc network.

**GPS** global positioning system.

**ILP** integer linear program.

**IRS** intelligent reflecting surface.

**LOS** line of sight.

**LP** linear program.

**LTE** long term evolution.

**MIMO** multi input multi output.

**mmWave** millimeter-wave.

**NLOS** non line of sight.

**PMR** passive metallic reflector.

**RSS** received signal strength.

**SINR** signal to noise plus interference ratio.

**UAV** unmanned aerial vehicle.

**UDN** ultra dense network.

**UE** user equipment.

**V2X** vehicle to everything.

**VR** virtual reality.

# List of Figures

1.1	Electromagnetic Spectrum. . . . .	2
1.2	A Typical mmWave Network. . . . .	4
1.3	Using a Reflector to Bypass an Obstacle. . . . .	13
1.4	A Sample Multi-UAV Network. . . . .	14
3.1	Considered mmWave Architecture. . . . .	35
3.2	Different Types of Obstacles Hampering LOS Transmission. . . . .	38
3.3	LOS Between UE and mmWave BS Despite Presence of an Obstacle. . . . .	41
3.4	An Example Demonstrating Computation of $S$ . . . . .	43
3.5	An Example to Demonstrate our Path Selection Algorithm For $k = 2$ . . . . .	48
3.6	Learning Phase for Varying Number of Static Obstacles. . . . .	50
3.7	Increase in Learning Rate with Filtering and Trimming. . . . .	51
3.8	Learning Phase for Dynamic Obstacles. . . . .	52
3.9	Comparison with an Existing Algorithm. . . . .	53
3.10	Number of Sectors to be Checked versus the UE-UE Distance, for Varying GPS Accuracies. . . . .	53
3.11	Effect of GPS Resolution on Inaccuracy. . . . .	54
3.12	Performance in a Real Life Scenario. . . . .	54
4.1	A Time Epoch $T$ . . . . .	60
4.2	Estimating Possible Trajectories. . . . .	61
4.3	(a) Possible Trajectory Cone of Dynamic Obstacle, and (b) Possible Zone of Obstruction Due to Dynamic Obstacle. . . . .	63
4.4	Obstacle Tracking Accuracy. . . . .	65
4.5	$L''$ is a Candidate Trajectory Line that Intersects All the Blocked Links. . . . .	72
4.6	Coverage by Typical RGB-D Cameras. . . . .	76
4.7	Handoff Performance vs RGB-D Camera Count. . . . .	76
4.8	Accuracy, Sensitivity and Precision vs. Discovery Time $\tau$ for Varying Number of Dynamic Obstacles and Link Requests. . . . .	79
5.1	Toy Example Demonstrating Link Failure. . . . .	82
5.2	Further BS is Better Due to (a) UE Mobility, and (b) Static Obstacle. . . . .	83
5.3	A Toy Example Demonstrating Stable Path Finding. . . . .	85
5.4	Estimating Link Active Time. . . . .	88
5.5	Max. UE Velocity vs $T_{active}$ . . . . .	91

5.6	No. of mmWave BS vs $\mathbb{T}_{active}$ .	92
5.7	No. of Static Obstacles vs $\mathbb{T}_{active}$ .	93
6.1	Example Demonstrating Utility of Reflecting Devices.	98
6.2	Sub-optimal Placement Failing to Achieve Maximum Coverage.	99
6.3	Optimal Placement Maximizing Coverage.	99
6.4	Effect of Reflector Count on Coverage for a Small Instance.	109
6.5	Effect of Reflector Count on Coverage, for $N_G = 50$ .	109
6.6	Effect of gNB Count on Coverage, for $N_R = 50$ .	110
6.7	Effect of Obstacle Percentage on Coverage.	110
6.8	Inaccuracy Due to Grid Resolution.	111
6.9	Effect of PMR Size on Coverage.	111
7.1	A Few of the Possible Covers for a Set of Points.	117
7.2	The Replica $q'$ of a Point $q$ About the Point $p$ .	118
7.3	Possible Cases for Optimal UAV Displacement.	121
7.4	Effect of $K$ on Coverage.	126
7.5	Effect of $r$ on Coverage.	126
7.6	Effect of $K$ on Average Displacement.	127
7.7	Effect of $r$ on Average Displacement.	127



# List of Algorithms

3.1	Obstacle Learning. . . . .	42
3.2	Discretizing the Angular Neighbourhood. . . . .	46
3.3	Creating a Visibility Graph. . . . .	46
3.4	Path Selection Algorithm. . . . .	47
4.1	Trajectory Cone Estimation of a Dynamic Obstacle. . . . .	63
4.2	Set Cover based DOTP. . . . .	73
4.3	Proactive Handoff Algorithm. . . . .	74
5.1	Stable Path Allocation Algorithm. . . . .	90
6.1	Joint Placement Using Set Cover. . . . .	105
6.2	Joint Placement Using LP Relaxation. . . . .	107



# List of Tables

1.1	Attenuation of Different Obstacles for 60 GHz mmWaves. . . . .	8
3.1	Notation . . . . .	39
3.2	Simulation Parameters. . . . .	49
4.1	Notation. . . . .	60
4.2	Signal Partitioning. . . . .	62
4.3	Confusion Matrices. . . . .	75
6.1	Notation. . . . .	102



# Chapter 1

## Introduction

This thesis deals with establishment of efficient [millimeter-wave \(mmWave\)](#) communication links over short, unobstructed paths. We do so by detecting potential obstructions, placing reflectors to bypass obstacles, and deploying [base stations \(BSs\)](#) on board [unmanned aerial vehicles \(UAVs\)](#). However, before diving into the technical details, for want of completeness, we give a brief background of wireless communications, and of [mmWaves](#) in particular.

### 1.1 Background and Fundamentals

The global mobile network traffic jumped from 55 exabytes (EB) per month in 2020 to 108 EB in 2022, a rise of almost 100% in just 2 years. The numbers are even more staggering if we look a little further back; a 20 fold increase over the last 7 years. It is estimated that the average smartphone data usage in the world will reach 46 gigabytes (GB) per month in 2028 (69% of it being served by 5G networks), compared to 19 GB in 2023 [1]. This estimate may even fall short if technologies such as [virtual reality \(VR\)](#) and [augmented reality \(AR\)](#) see ubiquitous popularity in the next 5 years. Most of these applications require high data rates; for example, uncompressed high resolution video streaming requires a mandatory data rate of 1.78 - 3.56 Gbps [2]. This sharp, unending rise in wireless

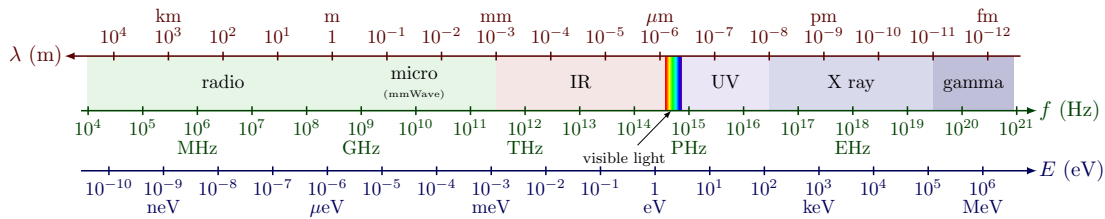


Figure 1.1: The electromagnetic spectrum<sup>1</sup>

traffic has led to depletion of the traditional microwave frequencies [3], most of which have been contained in a narrow band of the electromagnetic spectrum with frequencies from 300 MHz to 3 GHz. It is no surprise that the past decade has seen a meteoric rise in the interest shown in the “el Dorado” of communication spectrum, the **mmWave** realm [4–9].

**mmWaves**, as their name suggests, correspond to the band of radio frequencies whose wavelengths are in the range of millimeters (see Figure 1.1). Their frequencies typically range from 30 GHz to 300 GHz, of which up to 252 GHz are expected to be suitable for mobile communications [4]. Even if only 40% of this unused spectrum is allocated, it will provide up to two orders of magnitude expansion [4] over the currently deployed spectrum! Apart from providing spectrum expansion, communication in the **mmWave** domain provides another crucial advantage, enhanced data rate, in the rate of gigabits per second (Gbps). These carrier frequencies permit allocation of larger bandwidths, which in turn translate directly to higher data transfer rates. Besides the Gbps data rate, the latency for wireless traffic has decreased in **mmWave** communication [10]. Data intensive applications such as **VR**, **AR**, **vehicle to everything (V2X)**, can possibly not be served by the microwave spectrum, which typically provide a data rate in megabits per second.

The drawbacks of **mmWaves** are their poor propagation characteristics, and their acute sensitivity to blockages. Their poor penetration capacity has thrown up intriguing challenges that were absent in microwave communication. Their propagation loss through free space, however, which was once considered to be a disadvantage, has now been turned to their advantage; their fast attenuation

<sup>1</sup>Modified from [https://tikz.net/electromagnetic\\_spectrum/](https://tikz.net/electromagnetic_spectrum/)

over even modest distances allow aggressive frequency reuse, with low interference. Additionally, narrow beams by highly directional antennas actually provide additional security via eavesdropping immunity [11,12]. The consensus in the community thus is that to be scalable, **mmWave** communication should incorporate short, unobstructed links [3]. We discuss about **mmWave** propagation in more detail in [Section 1.2.1](#).

The advancement in low cost gigahertz capable integrated circuit design [13–15] has accelerated the transition into the **mmWave** realm. Indeed, devices like low noise power amplifiers coupled with low cost CMOS technology that performs well in **mmWave** frequency bands will make ubiquitous deployment of gigahertz communication a rapid reality [15–17]. Several standards exist for wireless networks around the 60 GHz spectrum, like ECMA-387 [18], IEEE 802.15.3c [19], and IEEE 802.11ad [20].

We give a few use case scenarios of short range, **line of sight (LOS)** communication in [Figure 1.2](#). There is usually a central **long term evolution (LTE) BS**, and several small cell **mmWave** enabled **BSs** inside the coverage area. These small cells can operate only when a **LOS** exists with a nearby communicating device. If there are obstacles on transmission path, a multi-hop **LOS** path can be established via a nearby idle **user equipment (UE)**, which directs the beam towards a destination device bypassing obstacles. Additionally, these devices can also amplify and forward a received signal [21], thereby effectively increasing the coverage range. Lastly, two nearby **UEs** that want to communicate with each other can do so directly without involvement of the **BS**, a scheme known as **device to device (D2D)** communication.

We now move on to some of the technical terms that are relevant to this thesis, and describe each of them briefly.

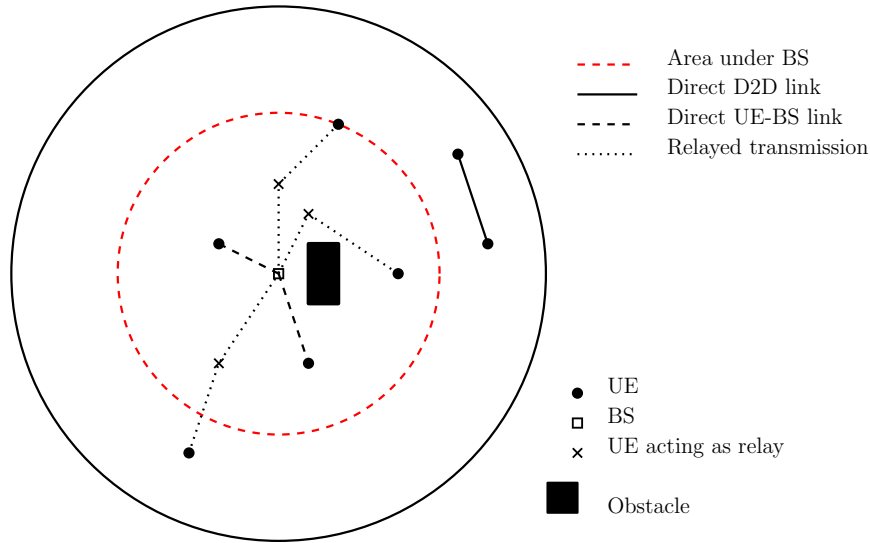


Figure 1.2: A Typical mmWave Network.

## Path Loss

Path loss can be informally defined as the decay in the energy of a transmitted signal as it passes through a medium; it depends on the transmission distance, the carrier frequency, and other location specific parameters. Friis [22] gave the following equation to model path loss way back in 1946

$$P_2 = P_1 G_1 G_2 \left( \frac{\lambda}{4\pi d} \right)^2, \quad (1.1.1)$$

where  $P_2$  is the received power at the receiver, and  $P_1$  is the transmitted power, both given in absolute linear units (eg., Watts);  $G_2$  and  $G_1$  correspond to the receiver and transmitter gains relative to an isotropic antenna,  $\lambda$  is the wavelength of the transmitted signal, and  $d$  is the transmission distance. Path loss is actually captured by the reciprocal of the term  $\left(\frac{\lambda}{4\pi d}\right)$ , indicating that path loss will increase as the separation is increased, or as the wavelength is lessened. In other words, for a fixed transmission distance between two communicating devices having fixed gains, the path loss varies as the square of the carrier frequency. By generalizing the log-distance slope in the far field, a better fit to path loss measurements can be made using the log-distance path loss model as follows,

$$P_2 = P_1 K \left( \frac{d_0}{d} \right)^\alpha, \quad d \geq d_0 \quad (1.1.2)$$



where  $d_0 \gg \lambda$  represents a close-in free space path loss reference distance in the far field,  $K$  is a dimensionless constant, and path loss exponent  $\alpha$  are adjusted to fit field measurements [23]. Taking the decibel values, (1.1.2) transforms into

$$P_2[dBm] = P_1[dBm] + 10 \log_{10} K - 10 \alpha \log_{10}(d/d_0), \quad d \geq d_0 \quad (1.1.3)$$

## SINR

The quality of a wireless link is typically expressed as the **signal to noise plus interference ratio (SINR)** at the receiver. Broadly speaking, the SINR at a receiver is given by

$$\text{SINR} = \frac{P}{I + \eta_0}, \quad (1.1.4)$$

where  $P$  is the power of the signal of interest at the receiver,  $I$  is the interference caused by nearby interfering signals, and  $\eta_0$  is the background noise of the medium. Let us consider a more general scenario of  $k$  transmitters, each of which communicates with its corresponding receiver. Then the SINR received at receiver  $j$  from transmitter  $i$  is given by

$$\text{SINR}_j = \frac{G_i G_j PL(d_{i,j})}{\sum_{k:k \neq i} PL(d_{k,j}) + \eta_0}, \quad (1.1.5)$$

where  $d_{i,j}$  is the Euclidean distance between the device pair  $(i, j)$ , and  $PL(d_{i,j})$  is the corresponding path loss between them. Usually, there is a threshold SINR below which the received signal cannot be decoded correctly; similarly, an application may require a minimum SINR, below which it cannot function properly. If the SINR between a UE and a BS falls below this threshold, a mechanism called a *handoff* is triggered. In this process, alternative BSs are explored for favourable SINR, and if available, one of them is selected to serve the UE.

## Shanon Capacity

The maximum rate at which information may be transferred via a wireless channel depends on its SINR, and is called its Shannon capacity. The Shannon capacity of a channel is given by

$$C = B \log_2(1 + \text{SINR}), \quad (1.1.6)$$

$C$  being the channel capacity in bits/s, and  $B$  the channel bandwidth in Hz.

## 1.2 mmWave Communication: The Pros and Cons

In this section, we describe some of the advantages, and challenges that come with communication in the [mmWave](#) realm. However, departing from the usual convention, we describe the disadvantages first.

### 1.2.1 The Cons

The disadvantages of [mmWaves](#) mainly stem from their propagation characteristics, both through free space, and through obstacles.

#### 1.2.1.1 Propagation Loss

There have been several studies on the propagation characteristics of [mmWaves](#) [24–28]. These waves suffer from much higher propagation loss as compared to their low frequencies microwave counterparts. The propagation loss in free space is proportional to the square of the carrier frequency. If we consider communication at 60 GHz, it corresponds to a wavelength of about 5 mm; on the other hand, the typical WiFi band at 2.4 GHz corresponds to a wavelength of 12.5 cm. For omnidirectional transmission and reception, [mmWaves](#) incur an astounding 625 times (28 dB) worse propagation loss as compared to WiFi [29]. Attenuation due to atmospheric Oxygen absorption tops in the 60 GHz range at 15 dB/km, while

that due to water vapour reaches 2 dB/km in the 24 GHz range [30]. However, since most of today's cell sizes in the urban environment are in the range of 200 meters, this atmospheric attenuation can effectively be neglected [5]; for example, rainfall of 25.4 mm/hr causes an attenuation of only 1.4 dB over 200 m in the 28 GHz range. Both transmitter and receiver have to be equipped with directional antennas to combat severe propagation loss.

### 1.2.1.2 Penetration Loss

Since electromagnetic waves can weakly diffract around obstacles of sizes larger than the wavelengths, mmWaves with their small wavelengths are extremely sensitive to blockage by obstacles. For example, common masonry items like bricks, concrete, and drywall all induce enough attenuation of mmWaves to discourage their usage without LOS. For example, brick pillars cause 28.3 dB penetration loss at 28 GHz [5], making building penetration of mmWaves difficult. Even humans cause intermittent attenuation of mmWave links. It is reported in [31] that upto 20-30 dB of penalization occurs due to human mobility. For a room with 1 to 5 persons, a mmWave channel is blocked for 1 to 2% of the time [32]. This can lead to time-varying network topology, necessitating stable links for delay sensitive applications like high definition live streaming. Foliage also causes non-negligible attenuation of mmWaves. For example, penetration of 10 m of foliage causes around 19 dB of attenuation at 40 GHz [30]. Since foliage is often not effectively captured by satellite imagery, novel practices have to be harnessed to determine their locations, and avoid them while transmitting data over the mmWave spectrum. We compile a list of common obstructing obstacles, and their corresponding attenuation in Table 1.1, where most of the data has been sourced from [11]. It is apparent that both static, as well as dynamic obstacles cause significant attenuation of mmWaves, and novel approaches are required to avoid them on the transmission paths.

Object	Thickness [cm]	Attenuation [dB]
Wooden panels	1.2	3.4-7.6
Drywall	4.8	5.2
Brick	11	16.9
Concrete	5	$\geq 30$
Clear Glass	1	4.3
Double-pane Tinted Glass	$0.4+1.5(\text{gap})+0.4$	$\geq 30$
Foliage	50-100	16-27
Heavy Hailstorm		26
Human	28	20-40
Vehicle		15-40

Table 1.1: Attenuation of Different Obstacles for 60 GHz mmWaves.

## 1.2.2 The Pros

Now, let us turn towards the positives. We will see that some of the inherent disadvantages of [mmWaves](#) will actually be turned into advantages.

### 1.2.2.1 High Data Rate

Shannon's formula (1.1.6) dictates that providing higher data rates will require sufficiently large transmission bandwidth. The good news is that the [mmWave](#) spectrum has abundant frequencies, allowing bandwidth in the range of gigahertz. The data rate provided by [mmWaves](#) is typically in the range of gigabits per second, thus aptly naming the communication "pseudo wired" [2].

### 1.2.2.2 Interference Mitigation

The small wavelengths allow deployment of smaller antennas, which provide very efficient focussing of transmission beams towards a narrow area, thereby effectively mitigating inter cell interference. We describe this in detail in [Section 1.3.1](#).

## 1.3 Some Approaches in mmWave Domain

As described in [Section 1.2.1.2](#), [mmWaves](#) have extremely poor penetrative properties. Hence, it is of paramount importance that the transmission links that are assigned are visible from each other. Also, the high attenuation through free space has to be countered too, by application of highly directional beamforming antennas. In this section, we describe some of the methods that are usually employed to allocate [LOS](#) paths.

### 1.3.1 Massive MIMO and Directional Beamforming

The microwave cellular spectrum currently in use typically transfers data over carrier frequencies having large wavelengths, which requires larger antenna sizes; this in turn limits the number of antennas on a device. The small wavelengths of [mmWaves](#) will allow incorporating smaller antennas, thereby paving way for massive [multi input multi output \(MIMO\)](#). If we consider a 38 GHz system, a half-wavelength spaced antenna array can fit more than 600 antennas in an area of just 1 square decimeter [\[33\]](#). Massive [MIMO](#) extends the capabilities of traditional [MIMO](#) by deploying a very large number of antennas in a compact form factor on a device. Of late, Facebook has reported base stations with 96 antennas as part of a massive [MIMO](#) network [\[34\]](#). The use of large number of antenna elements corresponds to using large array gains, which are vital to compensate for the high propagation loss of [mmWaves](#). These antenna arrays are different from traditional horn antennas; their radiation pattern is extremely flexible, depending on the number, type and orientation of antenna elements. Several arrangements are available, like uniform linear array, uniform rectangular array, and uniform circular array. This “massive” number of antennas are used to focus highly directional energy beams towards a narrow zone (also known as *beamforming*), bringing about drastic improvements in throughput and spectral efficiency [\[35\]](#).

Beamforming is a technique of focusing a wireless signal towards a chosen direction, rather than radiating to a large area. All the antenna elements emit the same signal, but with added phase shifts. These multiple phase shifted signals interfere with each other, constructively in some directions, and destructively in others. The result is that the signal is effectively *steered* towards a chosen direction, while the other directions receive little to no power. Without such large number of antenna elements to achieve beamforming, a base station would consume a lot of energy to send data in the **mmWave** realm [36, 37]. Furthermore, with advent of 3D beamforming, dynamic coverage for **UEs** in motion is also possible [38], improving user experience. Lastly, these so called “pencil beams” [39] also help in mitigating interference by not allowing the signal to travel towards unintended recipients, allowing deployment of **ultra dense networks (UDNs)**.

### 1.3.2 Ultra Dense Networks

Pencil beams emitted off massive **MIMO** antenna arrays allow for deployment of a large number of small range **BSs** in a small area. These can be deployed on trees, building walls, or even on top of lamp posts, usually in traffic hotspots. Analysis in [40] showed that we can aggressively reuse frequencies, and place **BSs** as close as 20 m apart. These **UDNs** can be defined as networks having more **BSs** than active users [41, 42]. A more quantitative definition is given in [43], which considered a network to be ultra dense if the number of deployed **BSs** is in the range of several thousands per square kilometer. Bandwidth hungry, data intensive applications of the near future can be served in localised environments by such **UDNs**. They are expected to perform well in the **mmWave** spectrum [44, 45], relying heavily on high gain, highly directional beamforming to minimise interference and attenuation. Additionally, numerous small range **BSs** can help in reducing the amount of blind spots in a coverage area, with the poor propagation characteristics of **mmWaves** actually helping to avoid inter cell interference. As is obvious though, we cannot go on increasing the infrastructure density indefinitely.

System parameters would reach a saturation point, post which adding more cells would add no benefit. A comprehensive survey on [UDNs](#) is given in [\[46\]](#).

### 1.3.3 Device to Device Communication

[D2D](#) communication can be formally defined as direct communication between two devices, without involving a base station or network [\[47\]](#). This sort of [BS](#) independent communication via directional [mmWave](#) links can be harnessed to increase network capacity [\[2\]](#), and reduce end to end delay. Indeed, if two [UEs](#) close to each other want to communicate with each other, there seems no justification to route the data through a [BS](#), which would increase network congestion. Such [D2D](#) communication can be non-orthogonal (where multiple [D2D](#) pairs use the same frequencies co-ordinated by the [BS](#)), or orthogonal (where there is no spatial reuse); the former leads to increased spectral efficiency, albeit at the cost of interference, while the latter has reduced spectral efficiency, but no interference. [D2D mmWave](#) transmission paths can be multi-hop as well; if two [UEs](#) do not have [LOS](#), other idle [UEs](#) in the vicinity can act as relays [\[48, 49\]](#), and effectively set up a multi-hop [D2D](#) path. Establishment of [LOS](#) paths, interference management, scheduling, and energy efficiency are all inherent challenges in this domain.

### 1.3.4 Tracking Obstacles

Locations of static obstacles are usually stored at the [BS](#) after being obtained from satellite imagery, or from open source maps [\[50\]](#). Avoiding static obstacles is easier, while dynamic obstacles present an altogether different challenge. Intermittent breaks in links cause degradation in user experience, while a number of unnecessary handovers are often introduced because of dynamic obstacles [\[51\]](#), which further add to system overhead. One way to deal with dynamic obstacles is via multi-connectivity [\[52\]](#). Here, multiple [BSs](#) have to be involved, some kept as backup as a precautionary measure for possible link failure, a process which

wastes precious resources. To track dynamic obstacles requires additional dedicated tracking hardware like LiDARs [53], cameras [54], and lasers [55]. The primary drawback of deploying such additional hardware is the considerable cost overhead that has to be borne by the service providers, and subsequently by the end users. Indeed, hardware such as radars are quite expensive and are sometimes prohibitive for ubiquitous deployment. Cameras on the other hand introduce privacy concerns [56], along with considerable image processing overhead. After tracking, handoffs may be triggered pre-emptively, without waiting for the link to actually break, thereby improving user experience [54].

### 1.3.5 Reflecting Devices

Due to the presence of obstacles in the coverage area, and subsequent **LOS** blockages, the number of **BSs** required for complete coverage can grow large, resulting in a high deployment expenditure. In this context, reflecting devices hold a lot of promise. They can be used to reflect mmWave signals incident on it from transmitter, towards the receiver, effectively establishing a 2-hop LOS path. These devices can be passive reflectors as in [57] or **intelligent reflecting surfaces (IRSs)** [58, 59]. While **passive metallic reflectors (PMRs)** simply reflect off incident signals according to Snell's laws, **IRSs** can steer these signals towards their intended destination. These **IRSs** work by tuning numerous low cost reflecting elements (printed dipoles), which adds a controlled amount of phase shift in every reflected signal. Much like massive **MIMO**, this causes interference among reflected signals, effectively achieving directional beamforming. In the outdoor environment, **IRSs** can be fixed on building facades, while in indoors they can be fixed on walls, ceilings, and even furniture. These low cost devices will play a vital role in deploying a sustainable **mmWave** enabled communication network in the near future [60]. Challenges in this field include phase shift optimisation, channel estimation, and spatial placement of these devices to augment the transmission environment. In **Figure 1.3**, we show the simple use case of a reflecting device. We see that the **UE** does not possess an **LOS** with the **BS**, and hence is unable to communicate



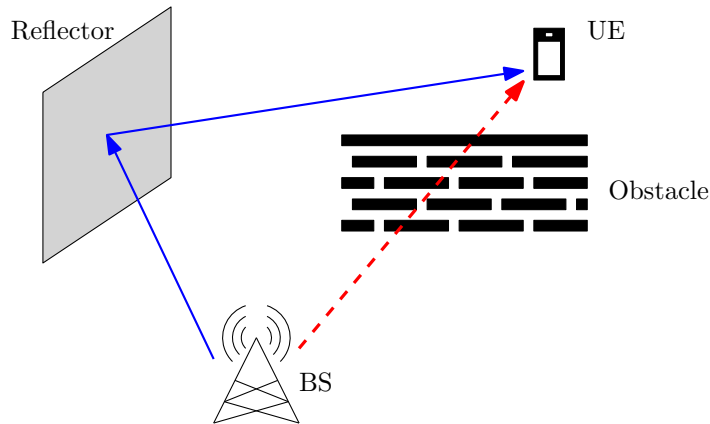


Figure 1.3: Using a Reflector to Bypass an Obstacle.

using the [mmWave](#) spectrum. A nearby reflector comes to the aid, by reflecting the signal transmitted by the [BS](#) towards the intended recipient.

### 1.3.6 Unmanned Aerial Vehicles (UAVs)

One of the more innovative promises of [fifth generation \(5G\)](#) networks is deployment of mobile, aerial [BSs](#) [61]. Though [UAVs](#) were first envisioned primarily for military use, their usage for civilian communication is possibly a matter of time. They will find applications in two pretty important scenarios, the first of them being traffic offloading from terrestrial infrastructure [62, 63]. For example, consider an area which has seen a sudden spike in traffic demand. In such a case, these [UAVs](#) with on board [BSs](#) can be deployed very fast to relieve the pressure on terrestrial infrastructure. A second, more serious application scenario of [UAVs](#) is when terrestrial infrastructure is destroyed during natural disasters [64, 65]. These vehicles can be deployed very quickly for disaster management groups, and provide faster relief. A network of multiple [UAVs](#) called [flying ad hoc network \(FANET\)](#) can work independently but in tandem with each other to provide coverage extension in infrastructure deficient areas [66, 67]. Incorporating [mmWave](#) communication on [UAVs](#) can be advantageous due to their high altitude above the ground which increases [LOS](#) probability with an end [UE](#). In other words, their higher degrees of freedom thanks to their highly controllable mobility make them ideal for establishing short, [LOS](#) transmission paths. We show a sample

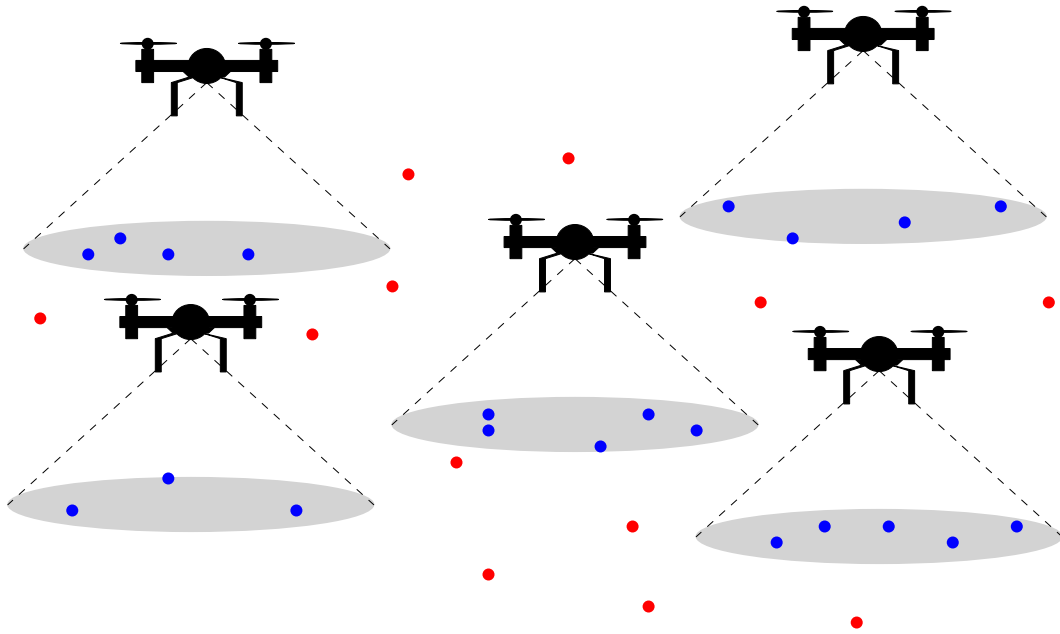


Figure 1.4: A Sample Multi-UAV Network.

multi-UAV BS network in [Figure 1.4](#), where there are 5 UAVs deployed, each having a coverage area shown in grey. This coverage area depends on a variety of factors, like transmit power, UAV height, and the azimuth angle of the antenna. Note that the blue disks represent UEs who are within the transmission range of a UAV, while the red disks represents those which are not.

Despite their inherent advantages, UAVs come with their own share of challenges. Unlike their terrestrial counterparts, UAVs do not possess a stable source of power. Their range is limited by their on board battery backup, which has to be utilised to hover, change position, and transmit power. Heavier battery packs increase the weight, which decreases the range, and a tradeoff has to be found. These constraints are popularly called size, weight, and power (SWAP) requirements [68], and give rise to interesting challenges, of which trajectory optimisation of UAVs deployed to serve as BSs is an important one. Indeed, the conflicting optimisation targets of maximising the number of UEs served, while minimising energy expenditure so as to maximise flight time remains a challenging problem. On top of that, the strict propagation constraints of mmWaves often require spatial repositioning of these UAVs, opening up another direction of research.

## 1.4 Research Gap and Motivation

In this section, we very briefly address the research questions that we try to answer in this thesis. We start off by asking if a priori knowledge of the coverage area is an absolute requirement before we start allocating links. Or, whether it can somehow be learnt post deployment? Indeed, spatial coverage environment may not always be available every time, and even if available, may not capture every single detail regarding the propagation path. Additionally, via long term network performance parameters, can we get a sense of where the zones of dynamic congestion are? Next we tackle the question of online tracking of dynamic obstacles; the usual approach is by deploying additional dedicated hardware, which is not at all an economic approach. Can we do something without resorting to such hardware? We then ask if the nearest BS is always the best choice, especially for UEs in motion? Indeed, an allocated link may break soon after allocation due to UE motion, which may either cause loss of LOS, or make the link too long to transmit data using the mmWave spectrum. We then shift attention to the problem of infrastructure deployment. We start off by asking if joint deployment of BSs and reflectors would give better coverage than a sequential approach? Finally, we address the problem of UAV deployment in mmWave networks. Since UAVs are highly power constrained, can we assign them to serve mobile terrestrial UEs such that their displacement is minimum, while ensuring coverage? These are the questions we raise, and attempt to answer in this thesis.

## 1.5 Contribution of the Thesis

The contribution of this thesis can be broadly divided into two parts. In the first part, we deal with detecting potential obstacles on mmWave transmission paths, and avoiding them. In the latter part of the thesis, we shift focus to deployment of mmWave enabling infrastructure to facilitate smoother communication.

## 1.5.1 Obstacle Detection for Stable Link Allocation

### 1.5.1.1 Learning obstacles without satellite imagery

In [Chapter 3](#), we consider the problem of learning the spatial and temporal transmission characteristics of the transmission environment, without any access to satellite imagery. The presence of obstacles was not a problem in microwave networks, as microwaves have good propagation characteristics, and can penetrate obstacles such as walls. However, [mmWaves](#) with their poor penetration nature, need obstacle free [LOS](#) paths to effectively transmit data. Spatial information can be gathered a priori from satellite imagery, or government records. However such information, apart from raising proprietary concerns, is often incomplete; for example, small obstacles like foliage are often not captured accurately. In this scenario, we make the following contributions:

**Contribution 1** : We propose a simple, learning based, deterministic algorithm to learn the location and size of static obstacles within the coverage area, without resorting to satellite imagery or government records. We do so by studying long term historical link failure data.

**Contribution 2** : The approach is extended to determine zones of dynamic congestion, that is, zones that usually have high density of obstacles that can obstruct data transmission.

**Contribution 3** : Finally, we give a path selection algorithm that has a high probability of successful transmission. Performance of our approach is measured against an existing approach, with respect to link failures.

### 1.5.1.2 Tracking dynamic obstacles from short term link failure data

While in [Chapter 3](#), we deal with zones of high dynamic congestion, we do not track such obstacles online, rather only focussing on learning the zones where high density of dynamic obstacles have been found in the past. This approach

may lead to overestimation of dynamic obstacle presence, and possible under-utilization of good links. The classical way to actively track dynamic obstacles is by deploying additional hardware, an expensive approach. In [Chapter 4](#), we propose an approach to the problem of tracking dynamic obstacles in an mmWave network, without using any dedicated tracking hardware such as camera, radar, or LiDAR. Given usual baseline infrastructure and short term link failure information, we attempt to obtain the trajectories of dynamic obstacles, and use them to perform proactive handoffs for at-risk links. We divide the chapter in two parts, the first one dealing with a single dynamic obstacle, while the second one considers multiple such obstacles. More formally, our contributions in [Chapter 4](#) are as follows:

**Contribution 1** : The single dynamic obstacle scenario is easy to handle, the proposed solution being extremely trivial. We use a signal space partitioning scheme, and Euclidean geometry in a [UDN](#), to extract the location, and velocity of the said obstacle. We show the effectiveness of the proposed approach to track a single dynamic obstacle.

**Contribution 2** : The real challenge lies in tracking multiple dynamic obstacles from short term link failure data, a problem which we model using a [integer linear program \(ILP\)](#), and prove to be NP-complete.

**Contribution 3** : We provide a greedy set cover based algorithm to obtain the trajectories of dynamic obstacles.

**Contribution 4** : We compare our proposed approach with an RGB-D camera based approach. We show that for low to moderate camera coverage, our approach achieves better handoff performance. We emphasize that since ours is merely a predictive approach, tracking through complete camera coverage will definitely outperform our method. However, such ubiquitous tracking would no doubt be accompanied by excessively high expenses, which would possibly make it infeasible in practice. We also validate our proposed tracking approach through simulation using the San Francisco taxi dataset [\[69\]](#).

### 1.5.1.3 Stable link allocation for avoiding unnecessary handoffs

In [Chapter 5](#), we argue that selecting the transmission path which promises the highest [received signal strength \(RSS\)](#) at the current time instance, might not always be the most beneficial approach. Indeed, with user mobility considered, a link which promises a high [RSS](#) at time  $t_1$  might deteriorate rapidly at time  $t_2$  due to the presence of obstacles, or due to the transmission distance being increased. Hence, it might be better to allocate a sub-optimal link, which might not give the best data rate at the current instant, but will remain active for a long time. More formally, the contribution in this chapter can be formalised as follows:

**Contribution 1** : We formulate the problem of allocating stable paths as a [ILP](#), and prove that it is an NP-complete problem.

**Contribution 2** : We select expected link active time as our metric, and assign those paths which are likely to remain active for the longest time, using a greedy approach.

**Contribution 3** : Via simulation, we show that the proposed approach allocates more stable links than the usual [RSS](#) based allocation.

## 1.5.2 Infrastructure Placement for Coverage Enhancement

### 1.5.2.1 Joint placement of BS and reflectors for coverage improvement

The [BS](#) deployment for wireless communication is an old, and well-studied problem. However, [mmWaves](#) have stringent [LOS](#) requirements for data transmission, which led to the concept of deploying a high number of base stations in [UDNs](#). Base stations being a costly resource, researchers have explored the idea of reflecting [mmWaves](#) off specially deployed, low cost reflectors, in an effort to bypass obstacles. In [Chapter 6](#), we argue that placing base stations and reflectors sequentially may lead to sub-optimal coverage. More formally, our contribution in [Chapter 6](#) is as follows:

**Contribution 1** : We consider the joint placement problem of **BSs** and reflectors to attain maximum coverage of the service area, and we provide a **ILP** for the same.

**Contribution 2** : Given the hardness of the joint placement problem, we first provide a greedy solution based on Max-Cover with an approximation bound. Thereafter, we develop an **linear program (LP)** relaxation based solution, and show that it achieves better coverage than the Max-Cover based solution.

**Contribution 4** : We perform extensive simulations to demonstrate the superiority of both our solutions over an existing two step approach.

### 1.5.2.2 Coverage ensured minimum displacement UAV deployment

Recently, there has been growing interest in deploying BSs on board unmanned aerial vehicles to facilitate better network performance. The spatial flexibility of **UAVs** provides crucial utility for two important scenarios; when terrestrial BSs are saturated and require traffic offloading, and when the same are destroyed due to natural calamities. However, the limited power on board a **UAV** plays a crucial role in its effectiveness. In **Chapter 7**, we study the problem of deploying **UAV BSs** in an **mmWave** network, with an aim to minimise their displacement in subsequent time, while maintaining coverage of mobile **UEs**. More formally, our contribution is as follows:

**Contribution 1** : Although the **UAV** placement problem is NP-Complete [70] in general, we show that the optimal placement of a single **UAV** for static **UEs** can be efficiently solved in polynomial time using a geometric approach.

**Contribution 2** : We then proceed to solve the multi **UAV** version of the same static scenario; since it is NP-complete, we use a Max-Cover based greedy approximation scheme.

**Contribution 3** : Taking into consideration **UE** mobility, we devise a polynomial time geometric algorithm that efficiently finds optimal placement of a single mobile **UAV** ensuring maximum coverage with minimum displacement.

**Contribution 4** : Finally, we solve our main objective which is to achieve maximal coverage by placing multiple **UAVs** with minimum displacements, using an approximate algorithm.

**Contribution 5** : Via extensive simulation, we show that our proposed LazyUAV algorithm achieves greater coverage with lower **UAV** mobility as compared to two baseline approaches.

## 1.6 Outline of the Thesis

The rest of the thesis is arranged as follows. In **Chapter 2**, we review some of the recent works in the field of **mmWave** communications. Thereafter, we dive into the contributory chapters, which are divided into **Part I**, and **Part II**. **Part I** deals mostly with obstacles and line of sight, with **Chapter 3** learning the locations of static obstacles, and zones of high dynamic congestion. We proceed to **Chapter 4**, where we track dynamic obstacles online, without any dedicated tracking hardware. In **Chapter 5**, we argue that selecting farther away base stations may sometimes be beneficial for long term stability of allocated links. In **Part II**, we shift focus towards efficient network infrastructure placement in **mmWave** networks. In **Chapter 6**, we deal with the joint deployment of **mmWave** base stations and reflectors for coverage improvement. We develop a **UAV** deployment strategy in **Chapter 7**, where we strive to serve maximal number of users, with minimum **UAV** displacement. We conclude in **Chapter 8** and give a few possible directions for future extension.



# Chapter 2

## Literature Review

In this chapter, we summarize the state of the art in the problems considered in this thesis, and identify the research gaps. Keeping up with the theme of the thesis, this review is also divided into two parts as follows.

### 2.1 Obstacle Aware Communication

The poor transmission characteristics of **mmWaves** provide unique challenges that were absent in traditional networks, and therefore call for newer ideas. A lot of the literature deals with establishment of **LOS** transmission paths of these waves, via relays. We first deal with the case of static obstacles.

#### 2.1.1 Handling Static Obstacles

The effect of static obstacles on relay selection in **mmWave** networks is a well-studied subject [71], and a variety of approaches exist. For example, authors in [72] proposed that transmission over multiple short hops would actually provide higher throughput than over a single long hop. The effect of obstacles on **mmWave** transmission in an indoor scenario was studied in [73], where the authors concluded that in the absence of an **LOS** path between transmission devices, reflected waves

cannot alone provide robust connectivity. The idea of deploying **mmWave** relays in an outdoor network was proposed in [74], and it was shown that the deployment of relays can lead to better coverage, and transmission capacity of **mmWave** networks. Several schemes for achieving multiple blockage avoidance were provided in [75], wherein the authors considered **BSs** and relays to be deployed along roads, as also considered in [76, 77]. Authors in [78] proposed a probabilistic relay selection mechanism for choosing the relay that had the highest expected data rate. A mobility aware relay selection mechanism was proposed in [79], wherein a greedy metric namely connectivity factor was introduced; the approach reported significant reduction in end-to-end delay, and packet loss. A multi-hop, polynomial time heuristic routing algorithm was developed in [80] to maximize the sum quality of the uncompressed HD video applications. Authors in [81] proposed a relay probing based two hop transmission path selection algorithm with an optimal threshold, reaching which the probing stops. A multi-hop relaying transmission scheme was presented in [82] to deal with the blockage problem of **mmWaves**, that steered blocked flows around obstacles by establishing multi-hop relay paths. A relay selection algorithm was proposed in [83], where several pre-planned relays were used in the area under a **BS**. The building topology of the area was assumed to be known, and a greedy algorithm was proposed that found an **LOS** of **UEs** with the help of close relays. Using a 3D model for buildings targeted at urban environments [83] investigates a relay selection and scheduling algorithm to support high end-to-end throughput in **mmWave** relay-assisted backhaul networks. However, this relay selection framework assumes that it has prior information regarding the location and sizes of static obstacles inside the coverage area. It also does not take into account the obstruction due to dynamic obstacles.

The idea of using nearby idle UEs to relay data between communicating devices, known popularly as *device relaying* was proposed in [48, 49]. It was a dramatic departure from the traditional network architecture, where the source and destination could select their own transmission paths, with limited or no intervention from the base station. One of the main advantages of this approach was that

of BS range extension via a relaying UE. Multi-hop transmissions using mobile phones was proposed in [84] to deal with non line of sight (NLOS), and achieve maximization of video quality. D2D relaying has been used in [85] to offload traffic from BSs. In that paper, authors jointly dealt with route selection, resource allocation, power control, and link scheduling with an aim to minimise transmit power.

However, most of these works focus on the immediate data rate maximization. For UEs in motion, if the nearest relay is chosen every time, there is a chance of frequent link failures, due to the fragile nature of mmWaves (both due to distance, and obstacles). A link failure is followed by a handoff requirement, which necessitates subsequent searching. In mmWave scenario, the handoff interval can be in the range of several seconds [86], which hinders user experience. Hence, the number of handoffs should be minimised, to improve system throughput, and subsequently quality of service. A path that remains active for a longer period of time can effectively reduce the number of handoffs.

### 2.1.2 Handling Dynamic Obstacles

Dynamic obstacles like pedestrians and vehicles, on the other hand, pose a harder problem. It has been found out that human blockages cause upto a 20-30 dB attenuation [54], while for a tinted car window the attenuation is 30-35 dB [87]. Such obstruction causes intermittent outages, decay in system throughput, and degrades overall user experience [54]. Additionally, a number of unnecessary handovers are often introduced because of dynamic obstacles [51], which further add to system overhead. As such, dealing with dynamic obstacles in an mmWave network forms a key challenge. The classical way to actively track dynamic obstacles is by deploying additional hardware. In fact, hardware dependent dynamic obstacle tracking is well-studied. Usual hardware used for tracking include LiDARs [53, 88, 89], cameras [54, 90–97], and lasers [55, 98, 99]. The primary drawback of deploying such additional hardware is the considerable cost overhead that has to be borne by the service providers, and subsequently by the end users. Indeed,

hardware such as radars are quite expensive and are sometimes prohibitive for ubiquitous deployment.

Obstacle induced link failures are typically dealt in one of two ways. In the reactive approach as in [100, 101], the handoff takes place only after the blocking has taken place, while in the proactive approach, link failures are predicted beforehand, and corrective measures taken accordingly. The proactive approach usually involves deploying dedicated tracking hardware, like RGB-D cameras, radars, or LiDARs. Authors [102] used an RGB-D camera to predict the location and mobility of human obstacles, and implemented a traffic mechanism that stops communication on the soon-to-be-blocked path, thereby freeing up that channel for use elsewhere. Authors in [103] leveraged camera imagery and convolutional long short-term memory (LSTM) based machine learning to achieve proactive handoffs. More recently, [92] proposed a machine learning based framework that deployed RGB cameras at the BSs to track dynamic obstacles, and achieve proactive handoffs preventing link blockages. Authors in [104] also uses RGB-D camera images to localise pedestrians and estimate their mobility to predict link blockages; subsequently, proactive handoffs are carried out before the human can block the link, thereby avoiding link failure. More recently, authors in [105] demonstrated the link switching in an mmWave environment, by tracking obstacles using a stereo camera for well-lit environment, and a LiDAR for dark environment. A reinforcement learning based handover mechanism was proposed in [54], which used a dedicated human tracking module. The problem with hardware like RGB-D cameras is that they typically have ranges of a few meters (0.5 m to 3.5 m) [106]; indeed, authors in [107] demonstrate an obstacle tracking approach using an RGB-D camera, where the obstacle distance is in the range of few meters. As such, their extensive deployment throughout the coverage area remains a big question. There have been works [104] that assume complete knowledge of the mobility of obstacles in the coverage area, an assumption which infeasible in real life. Similarly, authors in [103] consider a model where the communication path is always within the field of view of an RGB-D camera, which might not always be the case. Moreover, learning based approaches often require prohibitively large training overhead (for

example, [54] uses  $10^{10}$  tuples just for a  $4 \text{ m} \times 5 \text{ m}$  area, and two BSs), which might be unscalable in practice.

A slightly different way of dealing with dynamic obstacle induced link failures is via multi-connectivity [108, 109], an extension of the dual connectivity [52] that has already been proposed in LTE. Here, multiple BSs are associated with the same UE; if one link fails due to an obstacle, communication continues via the others. Another way that uses multiple beams was proposed recently in [110]. It used an extra guard beam to predict incoming obstructions, in order to protect the main communication beam. However, while multi-connectivity for all users has been recently reported to decrease network throughput [111], using a separate guide beam would decrease spectral efficiency and increase energy demands. An altogether different approach is therefore needed to deal with dynamic obstacles.

## 2.2 Infrastructure Deployment

Now we turn our attention towards the second part of the thesis, efficient deployment of network infrastructure to facilitate mmWave coverage. These can be fixed terrestrial ones like BSs and reflectors, or aerial ones like BS-equipped UAVs. The terrestrial ones are placed once, the target usually being to achieve a maximal cover given a fixed amount of resources. Deployment of UAVs on the other hand can have multiple optimization targets, including trajectory optimization, power optimization, amongst others. We first describe the recent works on deployment of fixed terrestrial infrastructure, and then move on to the aerial ones.

### 2.2.1 Terrestrial Infrastructure

Terrestrial infrastructure includes BSs and reflectors, which have to be deployed by the service provider based on spatial environment, and user demand. As is obvious, one would want to serve the maximum number of users with the minimum deployment cost, i.e., with the minimum number of BSs and reflectors. This can

be seen as a variant of the well-known Max-Cover problem with a fixed number of resources.

Base station deployment for wireless communications is an old, and well-studied problem. However, mmWaves BSs have stringent LOS requirements for data transmission which were absent in the traditional networks. With UDNs fast becoming a reality, dense deployment of small range BSs is a challenge that has to be addressed. Authors in [112] use computational geometry and LP relaxation to maximize the LOS coverage for a given number of BSs. A decomposition based approach was used in [113] to obtain an efficient solution to the problem of BS placement under outage constraints. A multi-arm bandit approach was used in [114] for deployment of such infrastructure outdoors, while swarm optimisation was used in [115] to minimally achieve LOS coverage in an indoor setting. A minimum cost, low complexity mmWave BS deployment algorithm was proposed in [116] that guaranteed network connectivity. A computational geometry based algorithm to automatically place large numbers of mmWave enabled BSs in an urban area with highly irregular buildings was given in [117]. A cooperative stochastic approximation based approximation framework was developed in [118] to deploy infrastructure with an aim to improve long term outage probability. Authors in [119] presented an outage guaranteed BS deployment technique in urban areas, using a scenario sampling approach to obtain a small scale deployment problem and optimally solving it by an iterative algorithm.

Due to the presence of obstacles in the coverage area, and subsequent LOS blockages, the number of BSs required for complete coverage can grow large, resulting in a high deployment expenditure. In this context, reflecting devices hold a lot of promise [58, 120, 121]. They can be used to reflect mmWave signals incident on it from transmitter, towards the receiver, effectively establishing a 2-hop LOS path. They have been used in [122] to improve the average received signal to noise ratio. Location of a single reflector was optimized in [123] to assist in data transmission from one BS and achieve maximum coverage, while a 2 reflector indoor scenario was considered in [124]. Authors in [125] showed that a reflector has to be placed nearer to the receiver compared to the transmitter to maximize

the signal to noise ratio. Authors in [121] reported an increase of 19 dB in received power by using a reflector in a 28 GHz setting. Passive and active repeaters were used in [126] for coverage enhancement, and improving received signal strength. Authors in [127] proposed an adaptive differential evolution algorithm to jointly optimize the number, locations, and phase shift coefficients of multiple IRSs.

Simultaneous placement of BSs and reflectors, however, has not been studied as much. Authors in [57] proposed a two step approach where they first solved a ILP to optimally deploy a given number of BSs, and subsequently solved another ILP to achieve increased coverage by optimally placing a fixed number of reflectors. However, such a sequential approach may lead to sub-optimal deployment. Hence, a combined approach is needed to deploy mmWave BSs and reflectors to increase the coverage area.

### 2.2.2 Aerial Infrastructure

The usage of UAVs in traditional networks is a well-studied subject [128], with UAV placement optimization at its very core [129–131]. Additionally, due to the power constraint, UAV trajectory design problems [132] have also received a lot of interest. There have been a lot of work involving efficient trajectory design to achieve a variety of targets. Authors in [133] provide a trajectory design of a UAV to efficiently achieve wireless power transfer. A throughput maximization scheme was presented in [134] by alternately optimizing power allocation and UAV trajectories. Authors in [135] provide a reinforcement learning based approach to establish UAV trajectories to maximise the expected uplink sum rate. A trajectory design algorithm was proposed in [136] that minimizes mission completion time, while ensuring data transfer with a high probability. Yet another trajectory planning approach was proposed in [137] that used a reinforcement learning based approach without any information about the UEs .

Deployment of mmWave BSs on board UAVs, on the other hand, presents unique challenges that were absent in traditional sub-6 GHz networks, since mmWaves

attenuate sharply through obstacles, and over moderate distances even in free space. A comprehensive analysis of the challenges, and existing solutions for **UAV** aided **mmWave** communications is given in [33, 138, 139]. The blockage effect of **UAV** rotors on **mmWaves** was first studied in [140]. A joint optimisation of **UAV-BS** positioning, **UE** assignment, and beamforming for maximizing the sum rate of the **UEs** was done in [141], and a sub-optimal algorithm for the non-convex problem was proposed. A reinforcement learning based trajectory design approach was presented in [135], where the target was to to maximize the expected uplink sum rate. Yet another reinforcement learning based approach was given in [142] which aimed at maximizing the coverage for an unknown environment. A lot of the works in literature [70, 143] deal with a one time deployment of **mmWave** enabled **UAVs**, allocating links to satisfy the immediate need. The problem is usually dealt with a variation of the K-means algorithm [62, 70]. Due to the inherent mobile nature of **UEs**, the transmission distance may change over time, possibly leading to link failures. Thus, **UAV** mobility certainly cannot be ruled out in order to maintain coverage. Since **UAV** mobility is to be fueled by limited on board power, initial placement and assignment become important parameters to maximise the up time.

## 2.3 Research Gap and Our Contribution

Majority of the static obstacle avoidance approaches mentioned in this review require a priori knowledge about the deployment area, which may not always be feasible. Indeed, satellite imagery can often fail to capture smaller obstacles like foliage, bushes, and billboards, which also obstruct **mmWave** communication. In **Chapter 3**, we propose a method to learn the locations and dimensions of static obstacles without any satellite imagery. We extend this notion to learn the zones of dynamic congestion, and provide a path selection algorithm that avoids these obstacles. We consider the problem of dynamic obstacle tracking in **Chapter 4** without any dedicated tracking hardware. Using short term link failure information, we aim to localise dynamic obstacles, and thereafter use the information to



---

achieve proactive handoffs even before links are actually broken. In [Chapter 5](#), we argue that for mobile [UEs](#), it might not always be advisable to allocate the nearest [BS](#). Taking into consideration [UE](#) mobility, and static obstacles, we allocate paths that will be unobstructed for the longest possible time. We show via simulations that this approach leads to longer link active times, as compared to the [RSS](#) based approach. We treat the problem of deploying [BSs](#) and reflectors in [Chapter 6](#), in a joint fashion. For a given amount of infrastructure, we aim to maximise the coverage area. Finally, in [Chapter 7](#), we take into consideration [UE](#) mobility and deal with the deployment of [UAVs](#) with on board [BSs](#), with an aim to minimising total displacement.



# Part I

## All about Obstacles



# Chapter 3

## Link Selection in mmWave Networks through Obstacle Learning<sup>1</sup>

### 3.1 Overview

As outlined in **Chapter 1**, **mmWave** communications can only bring about some noticeable change in data transmission rate if the two communicating devices are *near* and *visible* to each other. By visible we mean that the corresponding communication devices exhibit an **LOS** transmission path between them. By near we mean that the distance between the communicating devices is less than a threshold, above which **mmWave** communication does not provide a marked high data rate, even in the presence of **LOS**. However, it might not always be possible to know the propagation characteristics of the coverage area a priori. One way

---

<sup>1</sup>This chapter is based on the following papers:

**Subhojit Sarkar** and Sasthi C. Ghosh. “Relay selection in millimeter wave D2D communications through obstacle learning” In: Proceedings of the *12th International Conference on COMmunication Systems & NETworkS (COMSNETS 2020)*, IEEE, Bengaluru, India, January 7–11, 2020, pp. 468-475, DOI: 10.1109/COMSNETS48256.2020.9027458.

**Subhojit Sarkar** and Sasthi C. Ghosh. “Relay selection in millimeter wave D2D communications through obstacle learning” in: *Ad Hoc Networks (Elsevier)*, Vol. 114, pp 102419, January 2021, ISSN 1570-8705, DOI: 10.1016/j.adhoc.2021.102419.

to obtain such data is via using satellite imagery, open source maps [50], or by using government records. However, there may still be small obstacles that are not captured by satellites; indeed, obstacles such as bushes, billboards, cannot always be captured by satellite data. Similar problems arise at locations that do not have a clear view of the sky; for example, satellites cannot usually capture spatial details of regions beneath a bridge. For efficient mmWave communication, we need to obtain spatial details of potential obstacles in order to avoid them while assigning links. In this chapter, we consider the problem of *learning* the presence of obstacles in a coverage area served by a mmWave network, without resorting to any pre-obtained terrain data. The main contributions of this chapter are as follows:

- We develop a simple, learning based, deterministic algorithm to find the locations and sizes of static obstacles within the coverage area, without resorting to satellite imagery or maps.
- We extend this approach to determine the spatial zones with high vehicular traffic congestion, with an aim to avoid such zones while allocating links.
- We then create a *visibility graph* using the knowledge about the spatial environment that we have learnt in the primary learning phase.
- We use this notion of visibility graph to allocate links which are less likely to be obstructed. Via extensive simulations, we show how our algorithm performs with respect to the locations and sizes of obstacles. We compare our approach against a baseline link allocation approach, and show that though our approach performs poorly in the initial phases, its performance improves over time, ultimately outperforming the existing approach.

## 3.2 System Model and Assumptions

We now describe the system model considered, and the various assumptions made in this chapter.

**Network architecture :** We consider a deployment architecture which is similar to that proposed in [2]. As shown in Figure 3.1, there is a central LTE BS (shown by the square), and 6 mmWave BSs (shown by circles) arranged symmetrically on a regular hexagon, at the periphery of the coverage area of the LTE BS. These mmWave BSs are connected to the LTE BS, and thus to each other, by a high speed, wired backhaul network. The UEs (shown by crosses) are distributed uniformly at random in the coverage area of the LTE BS. All the BSs ( mmWave BS and LTE BS) have electronically steerable directional antennas (Liu et al. [6]) and mmWave transreceivers, to compensate for the high attenuation of mmWaves. For ease of analysis, we assume that all mmWave BSs are placed at a fixed height from the ground. A UE can directly communicate with an mmWave BS or with another UE if they are near and visible to each other. A UE can also reach an mmWave BS or another UE via other idle UEs which are willing to act as relays, provided each such hop satisfies the near and visible conditions. If a UE cannot establish such a path with an mmWave BS or the UE to which it wants to communicate, the UE has to use the LTE BS for traditional communication.

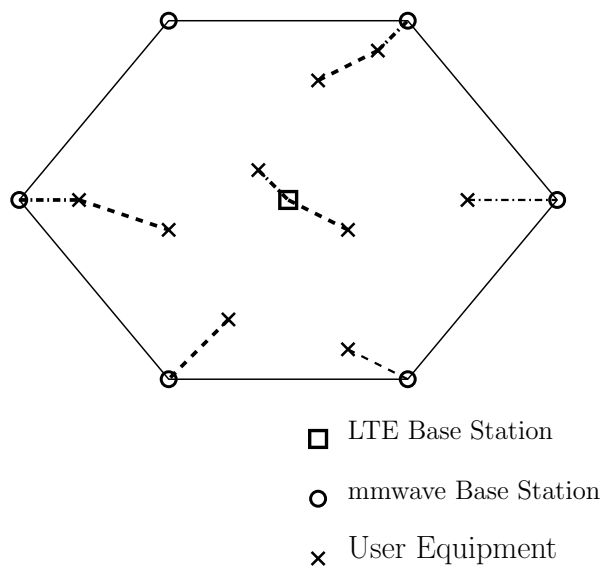


Figure 3.1: Considered mmWave Architecture.

**UE Hardware:** Each UE is equipped with a superaccurate chip [144] that can report its location to the LTE BS with accuracy of upto 30cm. They also have electronically steerable directional antennas (Liu et al. [6]) and mmWave

transreceivers, to compensate for the high attenuation of [mmWaves](#). Each [UE](#) can support a maximum of one pair of incoming and outgoing signals. The height of each [UE](#) is considered fixed, for ease of analysis.

**Coverage Area:** We divide up the area under an [LTE BS](#) into square grids, the smallest resolution of which can be the resolution of the [global positioning system \(GPS\)](#) accuracy (which can be as low as 30 cm) , though such a small resolution is hardly necessary. The smaller the resolution, the better the accuracy (and greater the computation). We assume that the [LTE BS](#) knows the locations of the [UEs](#) within its coverage area and hence can assign them to their corresponding grids. The [LTE BS](#) forwards such information to the relevant [mmWave BSs](#).

**Time Epoch:** The speed of transmission being in the range of gigabits per second, a lot of information can be transferred within a short time, thereby allowing small time epochs. For example, assuming pedestrian speeds being 2-4 meters/second and [GPS](#) resolution being 30 cm, [UEs](#) will remain within the same grid for at least 0.075 second. To avoid the high computation time involved with such a small resolution, we can assume a grid size of 1 meter and time epoch of 0.25 second; i.e., an [UE](#) remains within a grid of 1m for at least 0.25 second. We also discretize the user location into the centre of these grids, so that the exact location of the user within the grid is immaterial. Note that every time a link is assigned, alignment of transmitter-receiver antenna pair needs to be done. However, this alignment overhead is in the order of hundreds of microseconds even for [mmWaves](#) with extremely narrow beamwidths as mentioned and validated in Congiu et al. [145]. Thus we can safely ignore the overhead as the time epoch is several order of magnitude larger.

**Propagation Model:** The path loss and [SINR](#) are modelled using equations described in [Chapter 1](#). The relevant parameters which mostly have been taken from Bai and Heath [146], are included in [Table 3.2](#) in the simulation section.

At a particular time epoch, we can define the quality of a link between two communicating devices as *good* or *bad*, as follows. If the *measured SINR* is almost



equal to the *maximum SINR* for a pair of communicating devices, the difference being less than a particular threshold  $\nu$ , we call the link good. When the difference is above the threshold, we call the link bad. Here measured SINR (**A\_SINR**) means the actual SINR obtained after the link allocation, whereas *maximum SINR* (**MAX\_SINR**) means the theoretical maximum SINR between two communicating devices, assuming an **LOS** path between them.

**Obstacle modelling:** We consider two types of obstacles, *static* and *dynamic*. Some of the static obstacles can be known from satellite images, open source maps, and government records. However, satellite imagery cannot acquire all possible static obstacles, especially the smaller ones like bushes, trees and signboards. Also, as described in a following section, presence of a building in the transmission path between an **mmWave BS** and a **UE** does not guarantee obstruction. Obstruction depends on the location and size of the obstacle with respect to the heights of the **mmWave BS** and the **UE**. There might be a short obstruction, which is big enough to be captured by satellite imagery, but fails to obstruct the **LOS** between an **mmWave BS** and a **UE**. Motivated by this fact, we prefer to *learn* the presence of static as well as dynamic obstacles, *without* resorting to satellite imagery. Detailed learning procedure is elaborated in the next section.

*Static obstacles:* The static obstacles involve bigger barriers like buildings, bridges and towers, and smaller ones like bushes, trees and signboards. We use a simple training algorithm to *store* the probable locations of these obstacles in arrays, hereafter referred to as look-up tables. Simply put, we assume that there are no obstacles (unless it is present in the look-up table) and if, after link allocation, the *measured SINR* of a link is found to be much less than the *maximum SINR all the time*, we can come to a reasonable conclusion that there is an unknown static obstacle in the path of transmission, and transmission has taken place via **NLOS** paths. The obstacles can subsequently be added to the look-up table.

*Dynamic obstacles:* These include mobile obstacles like vehicles and humans. These hamper transmission, but we can say that in most cases, there is a good probability that in the subsequent time epochs, the obstacle will move away,

thereby providing theoretical maximum data rates. Thus, if a link has measured SINR comparable to the maximum SINR *even once*, we assume that any subsequent obstacle on that path is bound to be *dynamic*. Our subsequent path allocation algorithm avoids those links that are *more likely* to have a dynamic obstacle in their transmission paths. For example, two UEs on opposite footpaths at a busy intersection are more likely to have poor data rates due to the high chance of one or more dynamic obstacles being present, and would not be allocated a link if there are better options available.

Note that signal degradation can happen due to multiple reasons, like interference from neighbouring devices, antenna misalignment, along with obstacles presence on transmission path. There have been approaches proposed that can detect abrupt changes in signal strength caused due to the presence of obstacles. For example, authors in [147] propose a change point detection test for identifying the times where blockage happens, and ends.

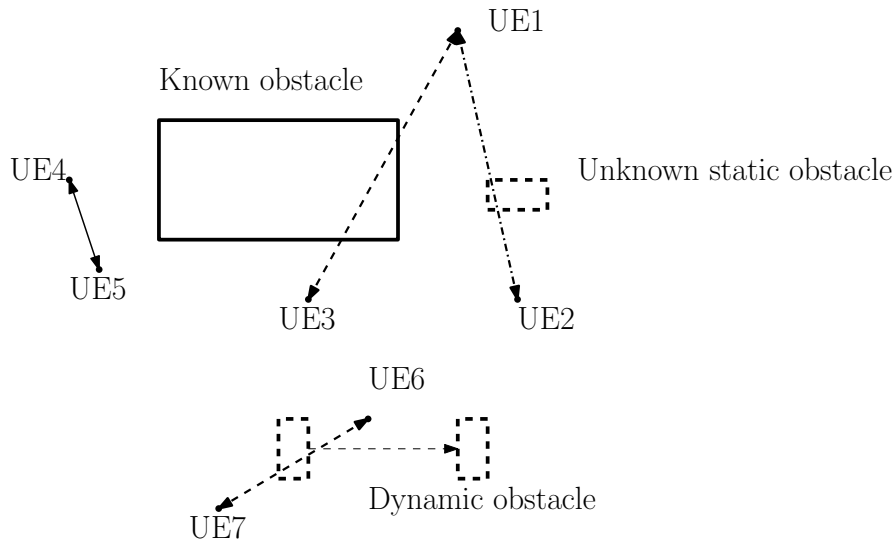


Figure 3.2: Different Types of Obstacles Hampering LOS Transmission.

We explain our proposed approach by an example. In Figure 3.2, we see that UE4 and UE5 have a good link quality. Hence, we deduce that there are no static obstacles between them. The link UE1 and UE3 is known to be bad because of a previously known obstacle, and hence it is never active in our algorithm. However, when it comes to link between UE1 and UE2, we initially believe that

there are no static obstacles in between them and hence, can possibly assign it as a link. However, we will soon find out that *every* time the link is active, the quality is bad. Thus, we will deduce that there exists a previously unknown static obstacle on the transmission path and update our look-up table accordingly. For UE6 and UE7, we will notice that the link is bad at one time epoch, but in the subsequent epoch, the link quality becomes good. This signifies that the initial obstruction was due to a dynamic obstacle, and there are no unknown static obstacles between these two locations. We point out here that even though UE1 may receive some signal from UE2 via reflections and multipath, it will undoubtedly have far lesser SINR as compared to a dominant LOS path, if present. Hence, it would be easy to check if the dominant path is indeed blocked by any obstacle. For example, authors in [147] propose a change point detection test for identifying the times where blockage happens, and ends.

The notation used in this chapter are described in Table 3.1 for completeness.

Symbol	Interpretation
$d_{i,j}$	Euclidean distance between $i$ and $j$
$\nu$	SINR threshold
$\mathbb{S}$	Static obstacle look-up table
$\mathbb{D}$	Dynamic obstacle look-up table
$\mathbb{C}$	Array storing number of times a grid has been part of transmission path
$\mathbb{B}_{i,j}$	Set of grids on line joining $i$ and $j$
$\mathcal{N}_{i,j}$	Subset of $\mathbb{B}_{i,j}$ not been <i>cleared</i>

Table 3.1: Notation

### 3.3 Learning Based Relay Selection Algorithm

Using the above architecture and model assumptions, we *learn* the presence of both static and dynamic obstacles in the transmission path between a pair of communicating devices. We use this information to create a visibility graph, which captures the quality of the links between the communicating devices, residing at their respective grids. Then using the visibility graph, we smartly assign links for relay selection.

### 3.3.1 Learning Approach

Now as pointed out, our algorithm has a learning phase. Initially, the LTE BSs assign links to UEs arbitrarily. A record of the quality of the assigned links are maintained. If the measured SINR of such a link is much less than the maximum SINR (the difference being above a threshold  $\nu$ ) *all the time*, we can reasonably begin to guess that there is a static obstacle there. However, if the difference between the measured SINR and the maximum SINR is less than the threshold value *even once*, we will know that the obstacle was a dynamic obstacle, like a vehicle.

There is a subtle point to be noted here. The obstructing power of an obstacle actually depends on whether the link is UE-UE or UE-mmWave BS, and we need to deal with them separately. As shown in Figure 3.3, the UE has an LOS with the mmWave BS, despite the “presence” of an obstacle in the path. In other words, a good link between two devices does not automatically rule out a presence of static obstacles on the transmission path if one of them is a mmWave BS. There might be a short obstruction, which is big enough to be captured by satellite imagery, but fails to obstruct the LOS between a mmWave BS and a UE. Similarly, the presence of an obstacle on a 2-D map does not necessarily mean that in 3-D that obstacle will block transmission. This further motivates us to consider a learning based approach rather than relying on satellite imagery and subsequently going for the learning method. Hence we determine whether an LOS exists between two communicating devices depending on whether one such device is an mmWave BS. In Algorithm 3.1, we describe the procedure for *static* and *dynamic* obstacles for the case of UE-UE transmission. The other cases can be handled similarly.

We take a link represented by a UE pair  $(u_1, u_2)$ , and compute the maximum SINR (assuming LOS) and measure the actual SINR obtained. We then check if the difference is less than the threshold  $\nu$ , which is an input parameter. If there is an obstacle on the transmission path, we assume that the corresponding attenuation is at least  $\nu = 25$  dB. This 25 dB is chosen because a tinted glass window has

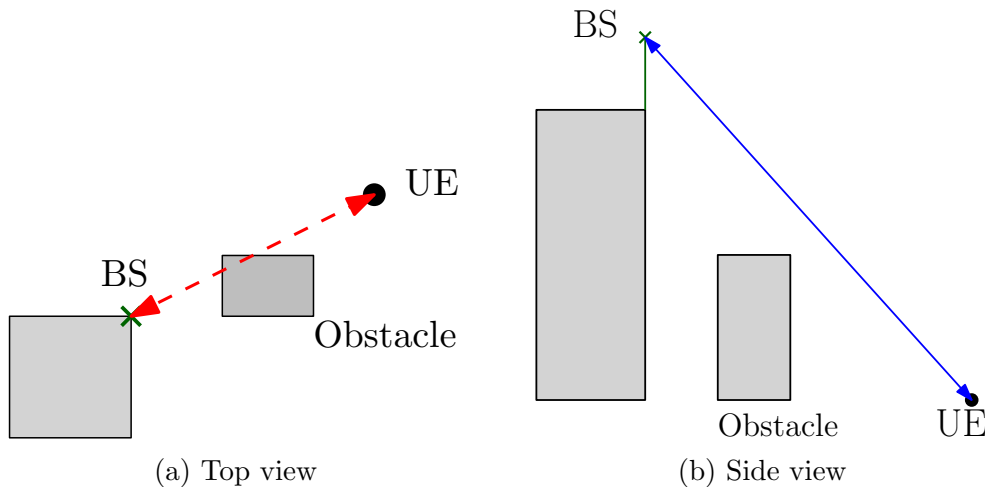


Figure 3.3: LOS Between UE and mmWave BS Despite Presence of an Obstacle.

the same loss as pointed out in [4]. We assume that a tinted window offers a minimal obstruction to any link. We take Bresenham’s algorithm (Bresenham [148]) and modify it slightly to compute  $\mathbb{B}_{u_1u_2}$ , the set of squares which lie on the path between  $u_1$  and  $u_2$ . It is a well-known algorithm in computer graphics that takes input two grids, and avoiding any floating point computations, computes the intermediate grids in between the two input grids, such that they form a close approximation of a straight line. We modify it to include *all* the grids that lie on the transmission path. We also compute  $\mathcal{N}_{u_1u_2}$ , a subset of  $\mathbb{B}_{u_1u_2}$ , which is the set of all intermediate grids between  $u_1$  and  $u_2$ , which have not yet being *cleared* of the possibility of having static obstacles. We use 2 arrays,  $\mathbb{S}$  and  $\mathbb{D}$ , to store the probabilities of each grid obstructing transmission due to static and dynamic obstacles respectively. We need an additional array,  $\mathbb{C}$  which keeps a record of the number of times a grid has been a part of a transmission path. Initially, all three arrays are initialized with zeroes. For  $\mathbb{S}$  and  $\mathbb{D}$  this implies that there are no obstacles (static or dynamic) initially. The value of  $\mathbb{C}$  being zero implies that the corresponding grid has not yet been a part of any transmission path. We mention here that the value of the arrays  $\mathbb{S}$  and  $\mathbb{D}$  will be interpreted as the probabilities of a grid square obstructing transmission, only if the corresponding value in array  $\mathbb{C}$  is greater than zero, i.e., has been part of a transmission.

For a link represented by  $(u_1, u_2)$ , if the difference between `A_SINR` and `MAX_SINR`

**Algorithm 3.1:** Obstacle Learning.

---

**Data:**  $(u_1, u_2)$ ,  $A\_SINR_{u_1, u_2}$ ,  $\nu$ ,  $\mathbb{S}$ ,  $\mathbb{D}$ ,  $\mathbb{C}$   
**Result:**  $\mathbb{S}$ ,  $\mathbb{D}$ ,  $\mathbb{C}$

- 1  $MAX\_SINR_{u_1 u_2} \leftarrow$  maximum SINR between  $u_1$  and  $u_2$  assuming LOS using Friis free space equation;
- 2  $\mathbb{B}_{u_1 u_2} \leftarrow$  Set of squares lying between  $u_1$  and  $u_2$  found by Bresenham's algorithm;
- 3  $\mathcal{N}_{u_1 u_2} \leftarrow \{b : b \in \mathbb{B}_{u_1 u_2} \mid C_b = 0 \text{ or } S_b \neq 0\}$ ;
- 4 **if**  $|MAX\_SINR_{u_1, u_2} - A\_SINR_{u_1, u_2}| < \nu$  **then**
- 5 **for**  $b \in \mathbb{B}_{u_1 u_2}$  **do**
- 6  $S_b \leftarrow 0$ ;
- 7  $\mathbb{D}_b \leftarrow \frac{\mathcal{D}_b \times C_b + 0}{C_b + 1}$ ;
- 8  $\mathbb{C}_b \leftarrow C_b + 1$ ;
- 9 **end**
- 10 **else**
- 11 **for**  $b \in \mathcal{N}_{u_1 u_2}$  **do**
- 12  $S_b \leftarrow 1 - (1 - S_b)(1 - \frac{1}{|\mathcal{N}_{u_1 u_2}|})$ ;
- 13 **end**
- 14 **for**  $b \in \mathbb{B}_{u_1 u_2}$  **do**
- 15  $\mathbb{D}_b \leftarrow \frac{\mathcal{D}_b \times C_b + \frac{1}{|\mathbb{B}_{u_1 u_2}|}}{C_b + 1}$ ;
- 16  $\mathbb{C}_b \leftarrow C_b + 1$ ;
- 17 **end**
- 18 **end**

---

is below  $\nu$ , in our array  $\mathbb{S}$  for static obstacles, we *clear* all the intermediate grid squares in  $\mathcal{N}_{u_1 u_2}$ , of the possibility of having static obstacles. That is,  $S_b = 0$  for all  $b \in \mathbb{B}_{u_1 u_2}$ . In the subsequent cases, any obstruction from these grids are guaranteed to be from dynamic obstacles only (unless new construction takes place). Similarly, in our array  $\mathbb{D}$  for dynamic obstacles, we update the previous values of all the intermediate grid squares  $b \in \mathbb{B}_{u_1 u_2}$ , as follows:

$$\mathcal{D}_b = \frac{\mathcal{D}_b \times \mathbb{C}_b + P_b^c}{\mathbb{C}_b + 1}. \quad (3.3.1)$$

where  $P_b^c$  is the current probability that grid point  $b \in \mathbb{B}_{u_1 u_2}$  has obstructed the transmission due to the presence of dynamic obstacles in it. Clearly, in this case,  $P_b^c = 0$  for all  $b \in \mathbb{B}_{u_1 u_2}$ . Therefore,  $\mathbb{D}$  stores the updated probability, computed by considering the past probability  $\mathbb{D}$  with the current probability  $P^c$ .

On the other hand, for two **UEs**  $u_1$  and  $u_2$ , if the difference between  $A\_SINR$

and  $\text{MAX\_SINR}$  is above  $\nu$ , we conclude that there is an obstacle on the transmission path. We assume that there is an *equal* likelihood of the presence of an obstacle in each of the intermediate squares and without further information, we cannot say which grid square is actually obstructing the **LOS** (or, indeed if all of them are). More specifically, if the obstacle is static, we assume that it can only be present in  $\mathcal{N}_{u_1u_2}$ , the set of intermediate grid squares which have not yet been cleared of the presence of static obstacles. For each grid square  $b \in \mathcal{N}_{u_1u_2}$ , we update the  $\mathbb{S}$  array as follows

$$S_b = 1 - (1 - S_b)\left(1 - \frac{1}{|\mathcal{N}_{u_1u_2}|}\right), \quad (3.3.2)$$

as we have assumed equiprobable presence of obstacles in  $\mathcal{N}_{u_1u_2}$ . We update  $\mathbb{D}$  according to equation (3.3.1), by taking  $P_b^c$  to be  $\frac{1}{|B_{u_1u_2}|}$  for all  $b \in B_{u_1u_2}$ . This is done because the clearing a grid of the possibility of having static obstacles does not ensure it will have no dynamic obstacle. After a sufficient number of iterations, the look-up tables  $\mathbb{S}$  and  $\mathbb{D}$  contain the information about *all* the discovered obstacles. We illustrate this process with an example.

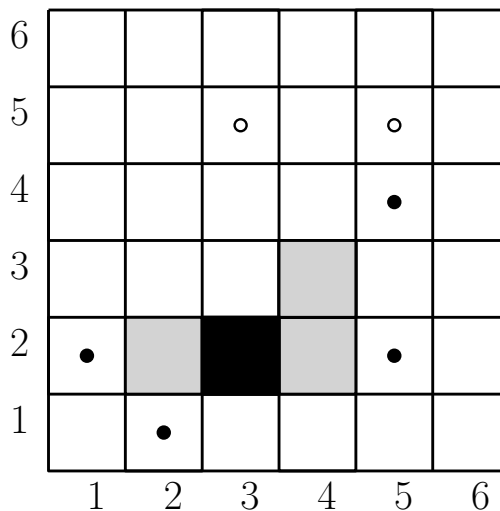


Figure 3.4: An Example Demonstrating Computation of  $\mathbb{S}$ .

Consider an empty  $6 \times 6$  array  $\mathbb{S}$  in Figure 3.4. From the past data, we see that **UEs** in (3, 5) and (5, 5) had good link quality, leading us to conclude that (4, 5) has no *static* obstacle. However, **UEs** in (1, 2) and (5, 2) have reported much lesser than maximum **SINR**, when they were allocated a **D2D** link. Therefore, there

is/was an obstacle on the transmission path between the communicating grids. We find the *questionable* grid squares between (1, 2) and (5, 2), and they being 3 in number, update the corresponding grid squares in  $\mathbb{S}$  using equation (3.3.2) with the value  $\frac{1}{3}$  ( $|\mathcal{N}_{(1,2)(5,2)}|$  being 3). A similar incident occurs when UEs in (2, 1) and (5, 4) tries to communicate. As earlier, we find the *questionable* grids between (2, 1) and (5, 4), in this case  $|\mathcal{N}_{(2,1)(5,4)}|$  being 2. Finally, from equation (3.3.2),  $\mathbb{S}_{3,2}$  stores the value  $1 - (1 - \frac{1}{3})(1 - \frac{1}{2}) = \frac{2}{3}$ , while  $\mathbb{S}_{2,2}$  and  $\mathbb{S}_{4,2}$  both store  $\frac{1}{3}$  and (4, 3) stores  $\frac{1}{2}$ . Thus,  $\mathbb{S}_{3,2}$  stores the highest probability of a static obstacle. The dynamic obstacles can be handled similarly.

After the locations of static obstacles, and the zones of congestion due to dynamic obstacles are learnt, the lookup tables  $\mathbb{S}$  and  $\mathbb{D}$  are combined into one array  $\mathbb{A}$ , post thresholding. Algorithm 3.2 subsequently divides up the area around a grid into equiangular sectors, and stores its distance from the nearest probable obstacle along each sector. The corresponding mmWave BS then uses this new lookup table to create a visibility graph, as shown in Algorithm 3.3 and subsequently the greedy Algorithm 3.4 is used to assign paths. Note here that the tables can be updated as and when new link status information is available. Hence, the table contains most recent information when it is being used.

We formally describe the process of sectoring in Algorithm 3.2. The input consists of the combined  $\mathbb{SD}^{Th}$  array after thresholding (this stores only boolean values), the maximum distance  $d_{max}$  till which D2D mmWave communication can be allowed, and the discrete angle  $\phi$ . The output is the obstacle array  $\mathbb{A}$ . Each element in the array consists of a list of nearest distances to obstacles, in each disjoint sector  $j$  subtending an angle  $\frac{360^\circ}{\phi}$  at the corresponding grid point. Typically the value of  $d_{max}$  is in the range of 100-150 meters (Hu and Blough [83]); to incorporate this in our model, for each grid point  $i$ , we assign a virtual obstacle in each sector  $j$ , at a distance  $d_{max}$  m. If an obstacle is found at a distance  $d$  (less



than  $d_{max}$ ) from grid  $i$  along sector  $j$ , we update the array  $\mathbb{A}$ . Thus,

$$\mathbb{A}(i, j) = \begin{cases} d_{max}, & d \geq d_{max} \\ d, & \text{otherwise.} \end{cases} \quad (3.3.3)$$

Here,  $\mathbb{A}(i, j)$  stores the nearest obstacle (real, or virtual) from grid point  $i$  along sector  $j$ . This algorithm is an exhaustive search. However, this needs to be run only once, when the learning phase is over. One point to note here that if  $\phi$  is large, we might miss obstacles. On the other hand a small value of  $\phi$  would lead to higher memory overhead. This value of  $\phi$  depends on the obstacle size, and the maximum transmission distance, and can be chosen accordingly. The effect of  $\phi$  on accuracy is showed in the simulation section.

### 3.3.2 Visibility Graph Creation

We subsequently create a visibility graph  $G_m = (V_m, E_m)$  at each **mmWave BS**  $m$ , using  $\mathbb{A}$ . The **LTE BS** sends the set  $\mathcal{U}$  of devices and their location information to the respective **mmWave BSs**, where **UEs** are identified by their corresponding grids. Using this information, each **mmWave BS** constructs a visibility graph, with the devices themselves forming the vertices. The graph formations are done at the **mmWave BSs** themselves, to keep the graph size manageable. The visibility graph creation is formally described in [Algorithm 3.3](#). For each pair of devices  $(u_1, u_2)$ , we first find the sector  $s$  on which  $u_2$  lies with respect to  $u_1$ . If the distance between two devices  $u_1$  and  $u_2$  is less than the smallest distance of an obstacle from  $u_1$  along  $s$ , the `addEdge()` function forms an edge between  $u_1$  and  $u_2$ . The edge weight  $w$  is the calculated maximum **SINR** of the  $u_1$ - $u_2$  path, which depends on the distance between them (assuming free space transmission). The construction of visibility graph could have been done in a decentralized manner, each **UE** sending out a handshaking signal and all others within this range responding. However, since we expect to compute this frequently, in order to reduce battery drainage, we prefer to do it centrally at the **mmWave BS**.

**Algorithm 3.2:** Discretizing the Angular Neighbourhood.

---

**Data:**  $\mathbb{SD}^{Th}$ ,  $d_{max}$ ,  $\phi$   
**Result:**  $\mathbb{A}$

```

1 for  $i = 1$  to  $N \times N$  do
2   for  $j = 1$  to  $\frac{360}{\phi}$  do
3      $\mathbb{A}(i, j) \leftarrow 0$ 
4      $H_i \leftarrow$  all grids in sector  $j$ , within  $d_{max}$  distance from  $i$  arranged in
       increasing order of distance;
5     for  $h \in H_i$  do
6       if  $\mathbb{SD}_h^{Th} = 1$  then
7          $\mathbb{A}(i, j) \leftarrow d_{ih}$ 
8         break
9       end
10    end
11    if  $\mathbb{A}(i, j) = 0$  then
12       $\mathbb{A}(i, j) \leftarrow d_{max}$ ;
13    end
14  end
15 end

```

---

**Algorithm 3.3:** Creating a Visibility Graph.

---

**Data:**  $\mathbb{A}$ ,  $\mathcal{U}$   
**Result:**  $G_m = (V_m, E_m)$

```

1  $G_m \leftarrow \phi$ 
2  $V_m \leftarrow \mathcal{D}$ 
3 foreach  $pair(u_1, u_2) \in \mathcal{U}$  do
4   Find the sector  $s$  in which  $u_2$  lies with respect to  $u_1$  ;
5   if  $\mathbb{A}(u_1, s) > dist(u_1, u_2)$  then
6      $w \leftarrow \text{SINR}(dist)$ ;
7     addEdge( $G_m, u_1, u_2, w$ )
8   end
9 end

```

---

### 3.3.3 Path Selection

Suppose device  $u_1$  wants to establish a path to device  $u_2$ . We have  $G_m$  at our disposal. Starting from  $u_1$ , we start a hop limited breadth first search for  $u_2$ . The maximum hop count  $k$  limits the depth till which breadth first search algorithm runs.  $\mathcal{P}$  stores the set of all paths with short, visible hops, the hop count being limited by  $k$ . The `hopCount()` function computes the number of hops in a path. If a 1-hop path exists in  $\mathcal{P}$ , it signifies that  $u_1$  and  $u_2$  have a short **LOS** path

between them. If the only paths in  $\mathcal{P}$  are multi-hop paths, we return the path having the highest SINR. For a multi-hop path, the SINR is taken as the SINR of the weakest link; i.e., the smallest SINR in a hop of a multi-hop path. We formally describe this in [Algorithm 3.4](#). If no  $k$ -hop restricted path is returned, the communication has to happen via the [LTE BS](#). In [Figure 3.5](#), we demonstrate execution of our algorithm for finding obstacle free paths, taking  $k$  as 2. The circles denote mmWave base stations, and the parallelograms are static obstacles. A demanding user is allocated links to a base station either directly, or via a nearby idle user (i.e.,  $k = 2$ ).

---

**Algorithm 3.4:** Path Selection Algorithm.

---

**Data:**  $d_1, d_2, k, G_m$   
**Result:**  $p^*$

- 1  $\mathcal{P} \leftarrow \text{hopLimitedBFS}(d_1, d_2, k, G_m)$ ;
- 2  $max \leftarrow 0$ ;
- 3  $p^* \leftarrow \phi$ ;
- 4 **for**  $p \in \mathcal{P}$  **do**
- 5     **if**  $\text{HopCount}(p) = 1$  **then**
- 6          $p^* \leftarrow p$ ;
- 7         **break**;
- 8     **end**
- 9     **if**  $\text{SINR}(p) > max$  **then**
- 10          $max \leftarrow \text{SINR}(p)$ ;
- 11          $p^* \leftarrow p$ ;
- 12     **end**
- 13 **end**
- 14 **return**  $p^*$ ;

---

### 3.3.4 Bringing It All Together

Now for want of clarity, we very briefly reiterate the flow of algorithms in this chapter. The static obstacles and the zones of probable dynamic congestion are stored in  $\mathbb{S}$  and  $\mathbb{D}$  respectively ([Algorithm 3.1](#)). In [Algorithm 3.2](#), we store the nearest obstacle in each sector with angle  $\phi$ , for each grid point, and use it to create the visibility graph in [Algorithm 3.3](#). We make use of [Algorithm 3.4](#) to find the best path between a source-destination pair  $(u_1, u_2)$ . If a 1-hop path is not

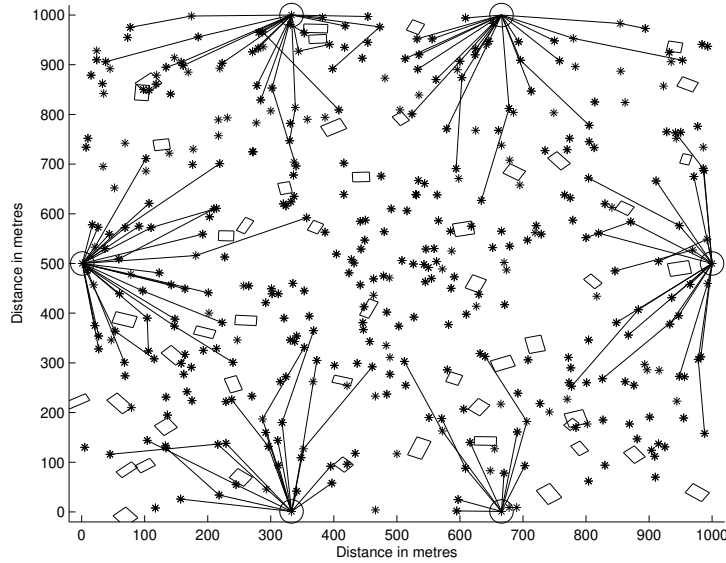


Figure 3.5: An Example to Demonstrate our Path Selection Algorithm For  $k = 2$ .

present we try to find  $k$ -hop restricted paths. If even this fails, the communication is handled by the [LTE](#) network.

One point to note here is that areas where no links were assigned during the learning phase, remain largely unexplored by our algorithm. However, as the learning algorithm always runs in the background, to quickly find new obstacles that might crop up in near future.

### 3.4 Simulation Results

The simulation parameters are mostly taken from [\[149\]](#) and [\[150\]](#). The path loss parameters are  $\alpha = 75.85$  and  $\beta = 3.73$ . The signal bandwidth is 20 MHz, and the maximum transmit power of a [UE](#) is 24 dBm, while the thermal noise density is  $-174$  dBm/Hz. We assume a square of size  $1000 \times 1000$   $m^2$  as the coverage area under the [LTE BS](#). The [LTE BS](#) is positioned at the center, and 6 peripheral [mmWave BSs](#) are arranged on a regular hexagon inside the coverage area, as shown in [Figure 3.1](#). There are some axes parallel static square obstacles having lengths and breadths both 15 m, distributed uniformly at random inside the coverage area. Some more smaller ( $1 \times 1$   $m^2$ ) static obstacles are also placed

Parameter	Value
Frequency	60 GHz
Bandwidth	20 MHz
Noise power spectral density ( $K_B T_e$ )	-174 dBm/Hz
Max. transmit power (UE)	24 dBm
Max. distance between one D2D pair	150 m
Min. attenuation by solid obstruction	25 dB
Path loss exponent for LOS	2
Path loss exponent for NLOS	4.49
No. of mmWave BS	6
$\alpha$	75.85
$\beta$	3.73

Table 3.2: Simulation Parameters.

uniformly at random inside the coverage area. These smaller obstacles represent trees, bushes, signboard-like obstacles which cannot be captured accurately by satellite imagery. Locations and dimensions of no static obstacle is known beforehand by our algorithm. We assume there are 2000 pedestrians with handheld **UEs** inside the area. Inside the area, there are two straight roads, intersecting each other at a right angle. The roads handle bidirectional traffic in two separate lanes, with car speeds chosen uniformly at random from 0 m/s to 15 m/s, the maximum speed slowing down to 7.5 m/s near the intersection. We assume that there are 50 cars on the roads at any time epoch. The vehicles are of size 2 m  $\times$  3 m each. The vehicles follow Manhattan mobility model; they travel in a straight line and when they arrive at the intersection, they continue along the same path with probability 0.5, turn either left or right with probability 0.2 each, and make a u-turn with probability 0.1. We assume that at one time, half of the **UEs** have to transmit or receive data. The rest are willing to act as relays. The parameters are listed in [Table 3.2](#). If there is an obstacle on the transmission path, we assume that the corresponding attenuation is at least  $\nu = 25$  dB. This 25 dB is chosen because a tinted glass window has the same loss as pointed out in [4]. We assume that a tinted window offers a minimal obstruction to any link.

It is assumed that an oracle possessed the information about the locations of obstacles while performing the simulation. We use the information provided by the oracle, to measure the actual **SINR** in our simulation. In our proposed algorithm,

we use only the link status information to find the locations of obstacles. Later, we verify the accuracy of our algorithm with the information provided by the oracle.

It would be pertinent to point out here that there is no direct way of determining if an obstacle obstructing a link is dynamic or static. It is only when we study long term link status information, regarding the blockage contribution of a grid, that we can reasonably begin to guess the nature of obstruction (if present) in a grid. More specifically, if a link passing through a grid has high SINR even once, it is guaranteed to be free of static obstacles; in that case, any subsequent obstruction was surely due to a dynamic obstacle.

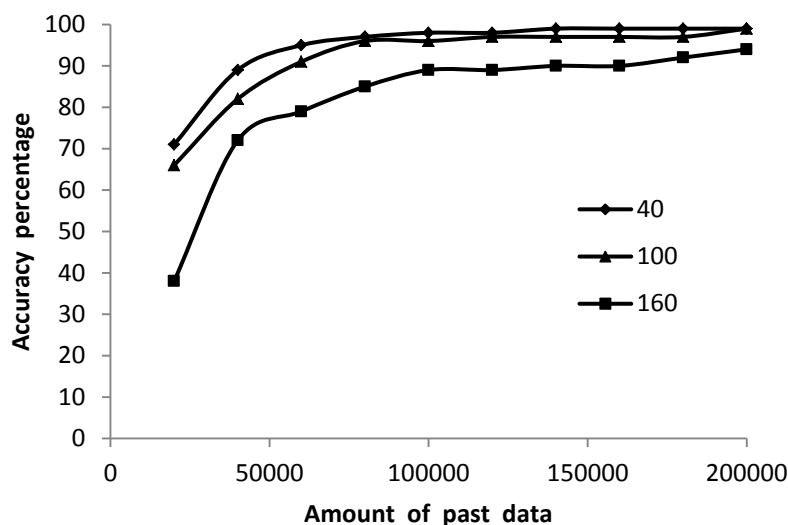


Figure 3.6: Learning Phase for Varying Number of Static Obstacles.

We plot the accuracy of our learning algorithm as the amount of past data increases. For the static obstacles, we define accuracy of the algorithm as the percentage of grid squares correctly identified to have static obstacles. Array  $\mathbb{S}$  stores the probabilities of grid squares having static obstacles, we carry out normalization, followed by some thresholding. Figure 3.6 shows the accuracy of our algorithm (threshold being 0.2) for different number of unknown static obstacles (160, 100 and 40), in the presence of dynamic obstacles as described earlier. As is expected, a higher number of obstacles need a longer learning time. After about 200,000 cycles of learning, the algorithm is able to detect almost all the static

obstacles. If we are reasonably certain regarding the sizes of static obstacles in the deployment area, i.e., if we are certain that obstruction is mainly due to large buildings, we can remove the smaller islands of 1's by filtering. Similarly, if we have an idea about the shape of obstacles, we can *trim* the edges of 1's in  $\mathbb{S}$ . This speeds up the learning rate as shown in Figure 3.7.

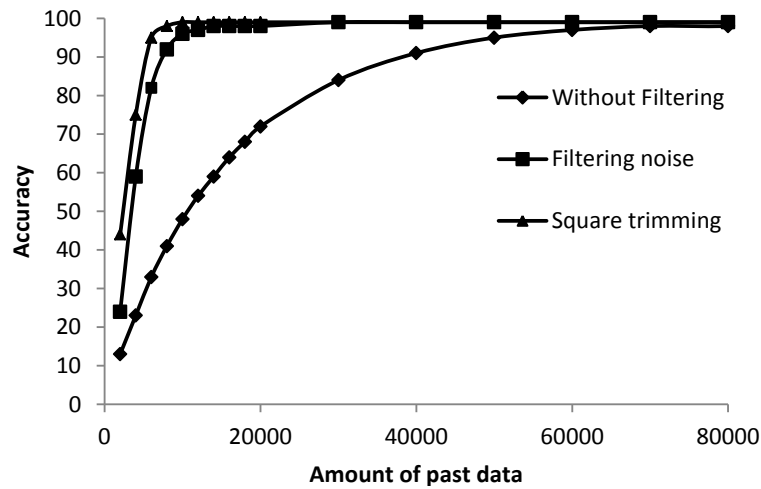


Figure 3.7: Increase in Learning Rate with Filtering and Trimming.

For the dynamic obstacles, we define the accuracy as the percentage of dynamic obstacles lying in the zone identified to have a high probability of dynamic obstacles at the concerned time epoch. For the dynamic case as well, similar normalization and thresholding are carried out as for the static case. Figure 3.8 shows the corresponding accuracy for dynamic obstacles (threshold being 0.2), in the presence of 50 static obstacles. Note that while some dynamic obstacles move arbitrarily without having any regular pattern, some others like cars follow regular movement patterns. The latter can be identified, but the former is hard to learn within a specified number of cycles. Thus, the accuracy of the algorithm for identifying dynamic obstacles is far from hundred percent even after 1,000,000 cycles. In comparison, the accuracy for identifying static obstacles is very close to hundred percent, just after 200,000 cycles.

We compare the path allocation performance of our learning based algorithm

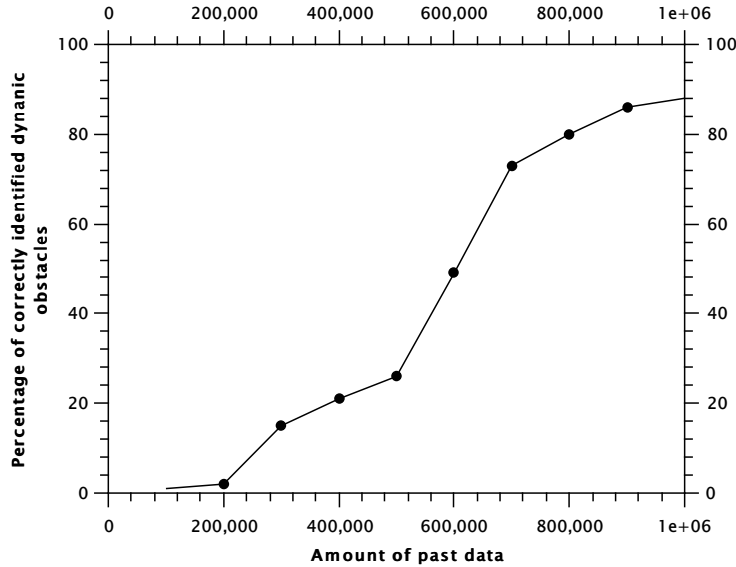


Figure 3.8: Learning Phase for Dynamic Obstacles.

against an existing solution [83], which assumed prior knowledge regarding the dimensions and locations of the larger static obstacles from satellite imagery. However, smaller static obstacles remained unidentified from satellite imagery. The effect of dynamic obstacles had also not been considered. We test our algorithm at various stages of learning by assigning links based on  $\mathbb{S}$  and  $\mathbb{D}$ , and compare it with the existing solution. Figure 3.9 shows the percentage difference from the *optimal link allocation* for the two algorithms, plotted along various stages of learning. We see that our algorithm gives much worse results as compared to the existing solution at the earlier stages of learning. However, as learning progresses, the performance of our algorithm gets much better. The optimal solution is derived by assuming the location of all static and dynamic obstacles (at each time epoch) to be known.

In Figure 3.10 we plot the maximum number of sectors (each  $36^\circ$ ) needed to be explored to check for LOS between a UE-UE pair, as the GPS resolution is varied. Simulation is performed by varying transmission distances from 10 m to 150 m, for GPS accuracies ranging from 1 m to 20 m. For an inaccurate UE location, and a small transmission distance, the algorithm has to explore a large number of sectors in the database in the neighbourhood of the UE.



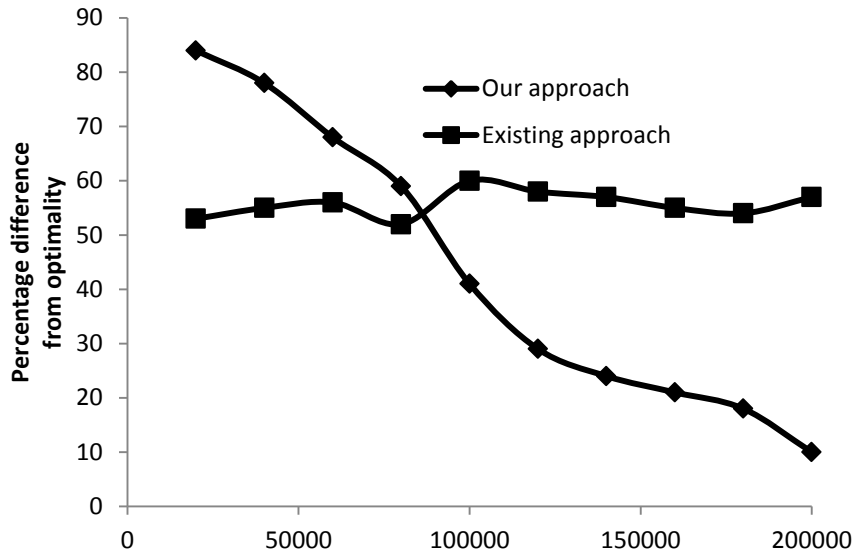


Figure 3.9: Comparison with an Existing Algorithm.

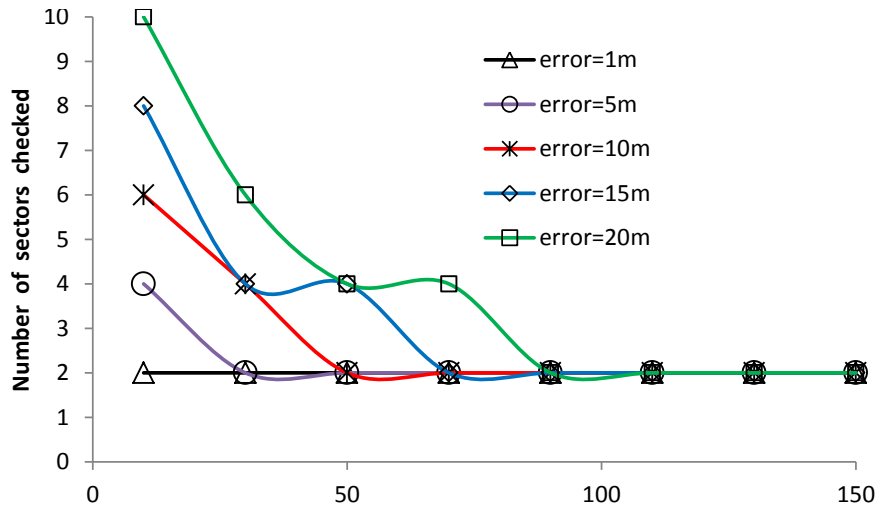


Figure 3.10: Number of Sectors to be Checked versus the UE-UE Distance, for Varying GPS Accuracies.

In [Figure 3.11](#), we plot the inaccuracy of our algorithm as the grid resolution is varied from 10m to 20m. Since our learning algorithm uses a modified version of Bresenham's Algorithm, when an [LOS](#) path exists between two grid points, all the large grid squares (having length 10 m and above) corresponding to each of the smaller ones (as used previously) will be set to zero. This gives us pretty inaccurate results, especially as the resolution worsens.

Finally, to check the validity of the proposed method in a real life scenario, we capture a small portion (100 m  $\times$  100 m) of Kolkata, India from OpenStreetMap

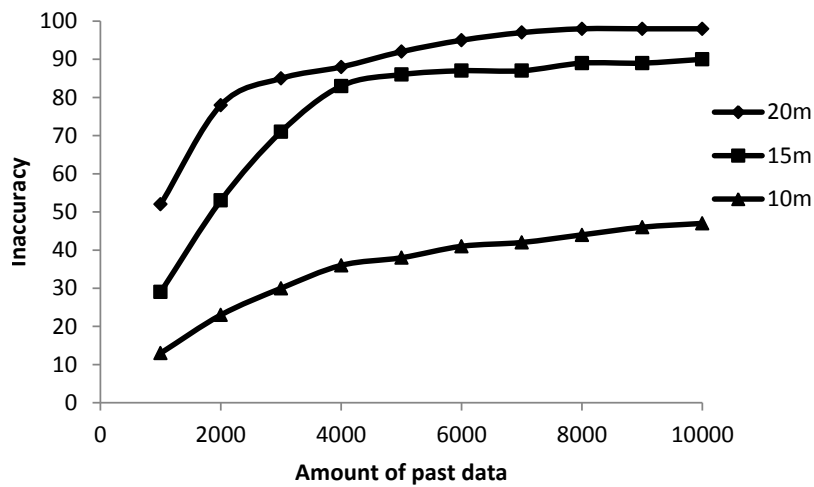


Figure 3.11: Effect of GPS Resolution on Inaccuracy.

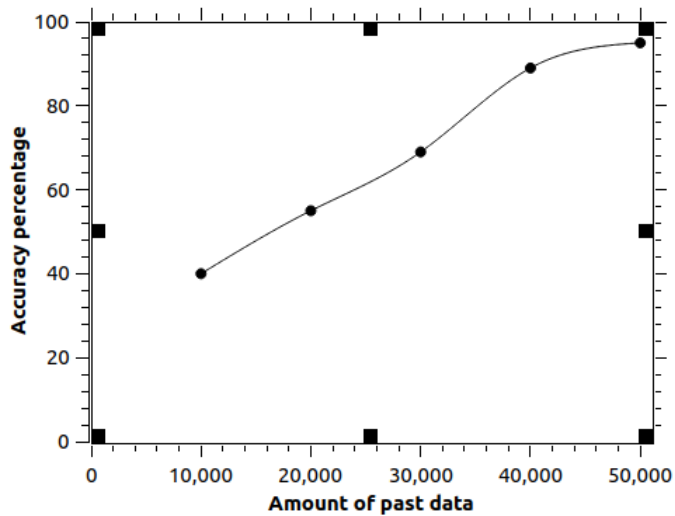


Figure 3.12: Performance in a Real Life Scenario.

[50] and run the static obstacle detection algorithm. The results are plotted in Fig. 3.12.

## 3.5 Conclusion

In this chapter, we propose a simple learning based algorithm for path selection using millimeter wave communication in 5G cellular network. It does not need any sort of satellite or terrain mapping beforehand; as a result, this directly bypasses the problem of detecting smaller obstacles by satellite imagery. Even zones of high dynamic congestion, like busy intersections, parking zones, can be detected by this approach. Subsequently, a path allocation algorithm is used to assign smart links, with less probability of obstruction.



# Chapter 4

## Tracking Dynamic Obstacles using Historical Link Failures<sup>1</sup>

### 4.1 Overview

In [Chapter 3](#), we dealt with locations of static obstacles, and zones of high dynamic congestion. However, we did not track dynamic obstacles online, rather only focusing on learning the zones with high density of dynamic obstacles from long term historical link failure data. However, this may lead to overestimation of dynamic obstacle presence, and possible under-utilization of good links. Indeed, ‘live’ tracking of dynamic obstacles would perhaps be better in avoiding prone-to-failure links. This is the problem we focus on in this chapter. The classical way to actively track dynamic obstacles is by deploying additional hardware. In fact, hardware dependent dynamic obstacle tracking is quite a well-studied domain. Usual hardware used for tracking include LiDARs [53], cameras [54, 90–97], and lasers [55, 98, 99]. The primary drawback of deploying such additional hardware is the considerable cost overhead that has to be borne by the service providers, and

---

<sup>1</sup>First half of the chapter is primarily based on the following paper:  
**Subhojit Sarkar** and Sasthi C. Ghosh. “Mobility Aware Path Selection for Millimeterwave 5G Networks in the Presence of Obstacles”. In: Proceedings of the *3rd International Conference on Computer and Communication Engineering (CCCE 2023)*, Stockholm, Sweden, March 10-12, 2023 pp. 67-80, Communications in Computer and Information Science, vol 1823. Springer, Cham. , DOI: 10.1007/978-3-031-35299-7\_6.

subsequently by the end users. Hardware such as radars are quite expensive and are sometimes prohibitive for ubiquitous deployment. Cameras on the other hand introduce privacy concerns [56], along with considerable image/video processing overhead.

In this chapter, we aim to tackle the problem of tracking dynamic obstacles in an mmWave network, without using any additional hardware such as camera, radar, or LiDAR. Given usual baseline infrastructure, and historical link failure information, we attempt to obtain the trajectories of dynamic obstacles. We divide up the chapter into two sections, the first one dealing with a single dynamic obstacle, while the second one considers multiple such obstacles; but first, we describe the system model considered, and the various assumptions made.

## 4.2 System Model and Assumptions

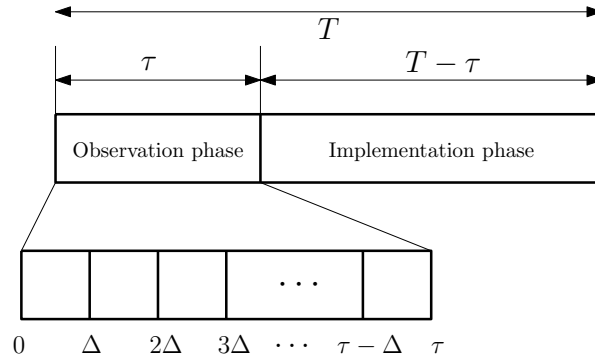
**Network architecture:** We consider an ultra dense [44], urban deployment scenario. A central LTE BS provides ubiquitous coverage. There is a set  $\mathcal{B} = \{b_1, b_2, \dots, b_B\}$  of mmWave BSs distributed uniformly at random inside the coverage area. Each such BS is connected to the central LTE BS, as well as to each other by a high speed backhaul network. We assume that the maximum mmWave transmission distance under LOS conditions is  $d_{max}$ ; above this threshold distance, the attenuation is so high, that the usage of mmWave bands is not justified. We discretize the time into slots  $0, \Delta, 2\Delta, \dots$ , where each slot is of duration  $\Delta$ .

**UE modelling:** There is a set of mmWave enabled UEs inside the coverage area. All UEs have electronically steerable directional antennas [6] and mmWave transreceivers, to compensate for the high attenuation of mmWaves. Each UE has a superaccurate GPS chip [144], and as part of the location update process, communicates its location to the LTE BS as part of the location update process. Each UE can communicate with an mmWave BS if there is an LOS path between the two, and the free space path loss is lesser than a threshold. Though

NLOS mmWave communication has been reported in some works like [151], the corresponding path loss exponents double as compared to LOS [11]; hence, we assume communication can happen over LOS paths only. If an LOS does not exist between a UE and any one of the mmWave BSs, the LTE BS steps in to provide traditional sub 6 GHz service. Each mmWave link can be represented by a line segment  $\{u - b\}$ , the end points being the positions of the UE  $u$  and the mmWave BS  $b$  respectively. The path loss between two nodes is computed using equations from Section 1.1. We use  $u_1 \longleftrightarrow u_2$  to denote that the devices  $u_1$  and  $u_2$  can communicate with each other via mmWaves.

**Obstacle Modelling:** There are a number of dynamic obstacles inside the coverage area, each travelling along a straight line for a small time epoch  $T$  (elaborated in the next paragraph). This is a reasonable assumption, since for example, a car changes directions rather infrequently, and  $T$  is a small period in the order of seconds. Indeed, intersections in roadways form a small percentage of the total road network. Unlike the UEs, the dynamic obstacles are not connected to any of the BSs. As such, the LTE BS has no information regarding the position, velocity, and past trajectory of the said obstacles. Since primary focus of this chapter is to track the dynamic obstacles from past link failures, we assume that unanticipated link failures occur due to the presence of dynamic obstacles only. We ignore the other possible causes of link failures such as imperfect channel state information, or high interference from neighbouring UEs. If there is an abrupt degradation of signal quality of a link, it implies that there is a dynamic obstacle on the transmission path. There have been existing approaches [147] that can detect abrupt changes in signal strength, and thus detect obstacles.

**Discovery and Implementation Phases:** We divide up the time into epochs of duration  $T$ . Each time epoch is further divided up into two phases, namely the *discovery* phase of duration  $\tau$ , and the *implementation* phase of remaining duration  $T - \tau$ , as shown in Fig. 4.1. The information obtained during the discovery phase is used during the implementation phase. After each time epoch  $T$ , the process starts afresh. As mentioned in the previous paragraph, the dynamic obstacles do not change their direction over an epoch. For completeness,

Figure 4.1: A Time Epoch  $T$ .

we summarize the notations used in this chapter in [Table 4.1](#).

Symbol	Interpretation
$T$	Length of a time epoch
$\Delta$	Length of a time slot
$\tau$	Duration of discovery phase
$\mathcal{B}$	Set of mmWave BSs
$u_1 \longleftrightarrow u_2$	$u_1$ and $u_2$ can communicate over mmWaves
$d_{max}$	Maximum transmission distance
$K$	Maximum number of dynamic obstacles

Table 4.1: Notation.

### 4.3 Tracking A Single Dynamic Obstacle

We first deal with tracking a single dynamic obstacle without additional hardware, i.e., in this section, we consider  $K = 1$ . We achieve this by a very simple processing of link failure data. The coverage area under consideration is discretized into small square grids, whose resolution is limited only by [GPS](#) accuracy. Our idea uses a modified version of the signal space partitioning scheme proposed in [\[51\]](#). There, the authors estimated the spatial location of a [UE](#) by processing the set of [BSs](#) from which it receives signals, without resorting to pinpointing the exact geographical location. Our idea involves both signal partitioning and accurate geographical locations to get an idea regarding the presence of a dynamic obstacle on a transmission path. In the toy example shown in [Figure 4.2](#), the crosses ( $\times$ ) enumerated with letters are the [mmWave BSs](#), and the disks ( $\bullet$ ) enumerated



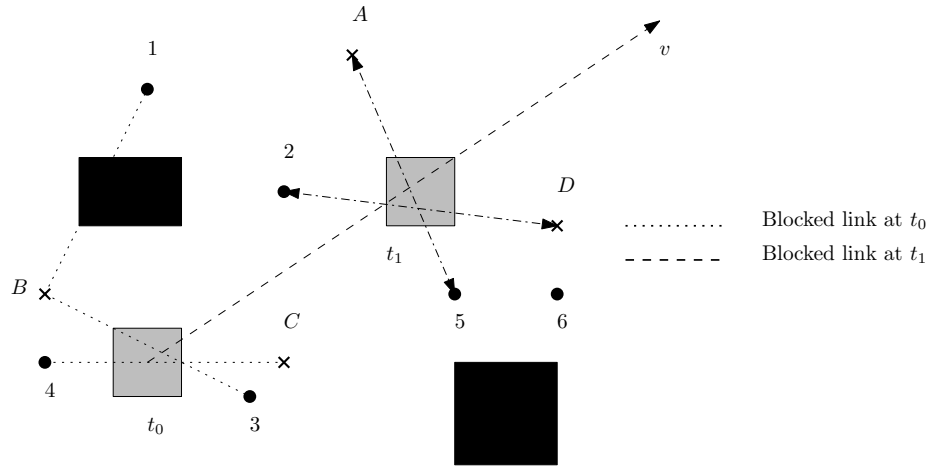


Figure 4.2: Estimating Possible Trajectories.

with numbers are the **UEs**. The dark black squares represent the known static obstacles, and the light grey squares represent the positions of a single dynamic obstacles at time slots  $t_0$  and  $t_1$ . The dynamic obstacle is moving in the shown direction with a velocity  $v$ . For the location of a **UE**  $i$ , the **LTE BS** is aware of  $L_i$ , the list of all **mmWave BSs** from which it is supposed to receive signals (under the absence of any dynamic obstacle). At a time  $t$ , an idle **UE** receives signals from the set  $L'_i(t)$  of all **mmWave BSs** that are within its close **LOS** range, and transmits the same to the **LTE BS**. Comparing this list with  $L_i$ , we get an idea regarding the presence of a dynamic obstacle. In **Figure 4.2**, we see that links  $3 - B$  and  $4 - C$  are blocked at time  $t_0$ . This tells us there is a dynamic obstacle on their spatial intersection position (as we have considered a single dynamic obstacle). We note that at time  $t_0$ , even link  $1 - B$  is blocked; however, since the location of the static obstacle is known beforehand, we do not take this as a marker for dynamic obstacle. Similarly at time  $t_1$ , we find that links  $2 - D$  and  $5 - A$  are blocked, which tells us the updated location of the dynamic obstacle. We summarize this in **Table 4.2**. Using the two intersecting locations, and the time interval  $[t_1, t_0]$ , we estimate the trajectory of the dynamic obstacle. We point out here that it may very well happen that the dynamic obstacle will not obstruct any link at multiple time instances, thereby making position extraction impossible. However for **UDNs**, this happens in rare cases, as validated in the simulation section.

UE ( $i$ )	$L_i$	$L'_i(t_0)$	$L'_i(t_1)$
1	A	A	A
2	A, B, C, D	A, B, C, D	A, B, C
3	B, C	C	B, C
4	B, C	B	B, C
5	A, C, D	A, C, D	C, D
6	C, D	C, D	C, D

Table 4.2: Signal Partitioning.

In the pre-processing stage the **LTE BS** computes the set of all transmissible **mmWave BSs** from each grid location. The following array based implementation is done to generate  $\mathbb{L}(i, j)$ , with  $i$  representing a grid point, and  $j$  being a **mmWave BS** in  $\mathcal{B}$ .

$$\mathbb{L}(i, j) = \begin{cases} 1, & \text{if } i \longleftrightarrow j \\ 0, & \text{otherwise} \end{cases} \quad (4.3.1)$$

After deployment, each idle **UE**  $i$  sends the list of **mmWave BSs** from which it is receiving signals, to the **LTE BS**. Using this data over a the discovery phase  $\{\Delta, 2\Delta, \dots, \tau\}$ , the **LTE BS** generates  $\mathbb{L}'_i(t)$ . In other words,  $\mathbb{L}'_i(t)$  stores the list of **mmWave BSs** from which  $i$  had an unobstructed, short **LOS** at time  $t$ . For a given  $\tau$ ,  $L_i$  and  $\mathbb{L}'_i(t)$  become the input of Algorithm 4.1,  $\mathbb{L}_i(t)$  being the subset of  $L_i$  corresponding to the grid locations of  $i$  at time  $t$ . We calculate the exclusive OR (X-OR) of the two input arrays for each time slot  $t$  and **UE**  $i$ , and store it in  $Y_i^t$ . This gives us an efficient measure of those links that are blocked due to a dynamic obstacle. For a pair of blocked **UEs** ( $u_1, u_2$ ) at time  $t$ , the logical OR of  $Y_{u_1}^t$  and  $Y_{u_2}^t$  gives us the spatial intersection points of the two links, and is subsequently used as a point in the trajectory  $\mathbb{T}$  of the dynamic obstacle in the `intersect()` function. Repeating this step over the discovery phase  $\tau$ , we get a set of points along with the corresponding time slots. Using  $\mathbb{T}$  and  $W$ , we can estimate *trajectory cone*  $\vec{C}$  and velocity  $\vec{v}$  of the dynamic obstacle. We define trajectory cone as the set of grids that encompasses all the possible grid locations of the dynamic obstacle in the future.

**Algorithm 4.1:** Trajectory Cone Estimation of a Dynamic Obstacle.

---

**Data:**  $\mathbb{L}, \mathbb{L}'[\Delta, 2\Delta, \dots, \tau]$   
**Result:**  $\vec{C}, \vec{v}$

- 1 **for**  $t \leftarrow 1 : \tau$  **do**
- 2     **foreach** user  $i$  **do**
- 3          $\mathbb{Y}_i^t \leftarrow \mathbb{L}_i \oplus \mathbb{L}'_i(t)$
- 4     **foreach**  $(u_1, u_2) \mid u_1 \neq u_2$  **do**
- 5         **if**  $\mathbb{Y}_{u_1}^t \cap \mathbb{Y}_{u_2}^t \neq \emptyset$  **then**
- 6              $\mathbb{T}^i \leftarrow \text{intersect}(\mathbb{Y}_{u_1}^t, \mathbb{Y}_{u_2}^t, u_1, u_2)$
- 7             **break**

8 Using  $\mathbb{T}$  and  $\tau$ , estimate  $\vec{C}$  and  $\vec{v}$   
9 **return**  $\vec{C}, \vec{v}$

---

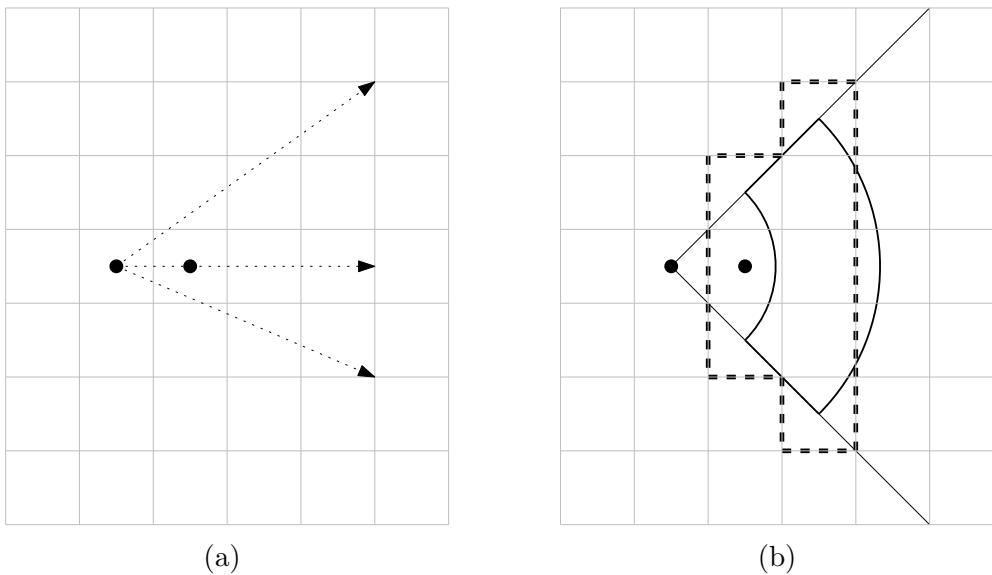


Figure 4.3: (a) Possible Trajectory Cone of Dynamic Obstacle, and (b) Possible Zone of Obstruction Due to Dynamic Obstacle.

*Remark 1 :* There is however, a subtle additional point that has to be taken care of. The finite grid resolution introduces inaccuracy in the predicted path. In other words, trajectories at different angles may very well have same intersecting grids. In Figure 4.3(a), the positions of the dynamic obstacle at time  $t_0$  and  $t_1$  are shown by disks. As is evident, the finite grid resolution maps multiple possible trajectories to the same points. The larger the resolution, the larger is the estimated *trajectory cone*. In other words, the trajectory cone is the set of grids that contains all possible obstacle trajectories. To deal with this as shown in Figure 4.3(b), we enumerate the maximal set of possible trajectories of the

dynamic obstacle. If we have more and more information regarding the positions of the dynamic obstacle, this cone shrinks. In other words, the larger the  $\tau$ , the greater the accuracy; however, too long a discovery phase decreases the duration of the implementation phase. This decreases the usefulness of tracking since the dynamic obstacle may change its velocity after a time epoch  $T$ .

*Remark II* : We point out here that this approach cannot be applied very easily for multiple obstacle scenario. This is because in such a case, there is no way of deducing which of the obstacles obstructed the transmission (or indeed, if multiple of them did). In [Figure 4.2](#), the links 2 –  $D$  and 5 –  $A$  may have been blocked at time  $t_1$  by an entirely different obstacle that obstructed link 4 –  $C$  at time  $t_0$ . We would have no way of knowing which obstacle(s) obstructed the links; in fact, each link might even have been blocked by multiple obstacles on a transmission path. In such cases, we would not have been able to distinguish between them using this approach.

## Results On Single Obstacle Tracking

The modelling parameters are mostly adapted from [\[152\]](#). We consider a square area of size 200 m  $\times$  200 m as the coverage area under the [LTE BS](#), the grid resolution being 1m. An [LTE BS](#) provides ubiquitous coverage, while some [mmWave BSs](#) provide high speed, short range services. There are some known static obstacles inside the coverage area, size of each being 5 m  $\times$  5 m. There is a single dynamic obstacle moving in a straight line, over a time epoch. We define the accuracy of our approach as the percentage of time the dynamic obstacle is present in the reported trajectory cone, and plot it in [Figure 4.4](#). Our approach would not have been very successful in sparsely deployed networks, with the trajectory being identified in only 20% of the cases for 10 [mmWave BSs](#) and 50 relay [UEs](#). However, as we move towards ultra dense networks and heavy user density, we see that the trajectory of the dynamic obstacle can be obtained accurately in upto 90% of the cases.

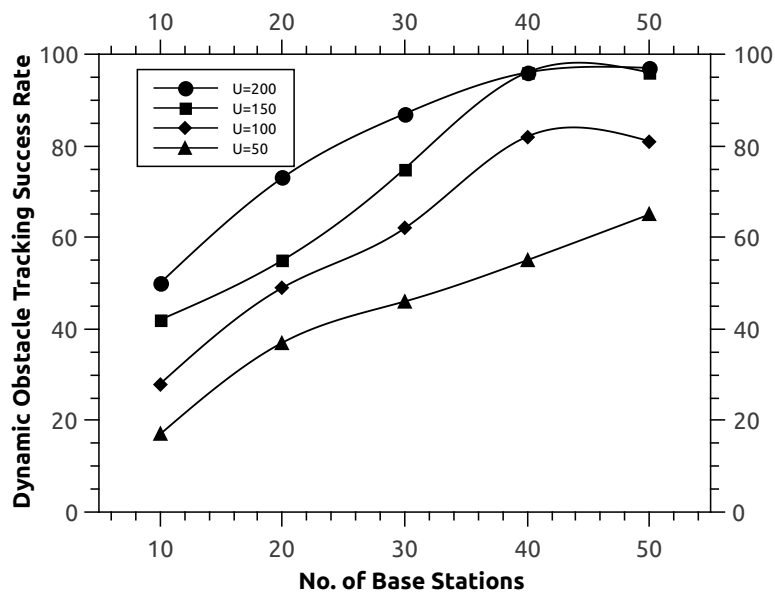


Figure 4.4: Obstacle Tracking Accuracy.

In this half of the chapter, we have proposed a simple algorithm to track a dynamic obstacle inside an [mmWave](#) transmission environment without using any additional hardware. It should be pointed out here that this approach requires a large number of [mmWave BSs](#) to track the obstacle, which might be impractical. However, the target of this section was primarily to be a proof of concept of the fact that a dynamic obstacle can be tracked without cameras. Next, we will consider the more challenging problem of tracking multiple dynamic obstacles, without resorting to any additional hardware.

## 4.4 Tracking Multiple Dynamic Obstacles

We now turn our attention towards the much harder, multiple dynamic obstacle tracking challenge. As in the previous section, we will not use any additional dedicated tracking hardware like radars, LiDARs, or cameras. Instead, we will rely solely on short term link failure data. Formally, our main contribution in this section can be summarised as follows:

- We have embarked upon, what we believe, is the first attempt to track multiple dynamic obstacles that obstruct LOS transmission in an mmWave network, without resorting to any dedicated tracking hardware like RGB-D cameras.
- A ILP is developed for the dynamic obstacle tracking problem (DOTP), and the problem is proved to be NP-complete.
- Using short term historical link failure information, we provide a greedy set cover based algorithm to obtain the trajectories of dynamic obstacles, and use them to achieve proactive handoffs before links are actually disrupted.
- We compare our proposed approach with an RGB-D camera based approach. We show that for low to moderate camera coverage, our approach produces better obstacle tracking, and subsequently manages to avoid more link failures. We emphasize that since ours is merely a predictive approach, tracking through complete camera coverage will definitely outperform our method. However, such ubiquitous tracking would no doubt be accompanied by excessively high expenses, which would possibly make it infeasible in practice.

It would be pertinent to point out here that the aim of this section is primarily to demonstrate the viability of using link failure information to track dynamic obstacles, without the need for any tracking hardware. The approaches presented here are very basic, and we hope that this idea sparks interest in hardware independent obstacle tracking approach, which can possibly help in lowering infrastructure costs in next generation networks.

## 4.5 Problem Formulation

Let  $\mathcal{L}$  be the set of all blocked links associated with an mmWave BS after the discovery phase of a certain epoch. Let us assume that the upper bound on the number of dynamic obstacles is  $K$ .

In order to write the considered DOTP in a mathematical form, we introduce a set of binary indicator variables  $X_k \in \{0, 1\}$  for all  $0 \leq k \leq K$  defined as follows:

$$X_k = \begin{cases} 1 & \text{when obstacle } k \text{ causes at least one link failure} \\ 0 & \text{otherwise} \end{cases}$$

Now, our objective is to find the minimum number of obstacles accounting for all link failures. Thus, the objective function can be given by:

$$\text{minimize} : \sum_{k=1}^K X_k \quad (4.5.1)$$

Now let us introduce another set of binary indicator variables  $Y_{k,l} \in \{0, 1\}$  for all  $1 \leq k \leq K$  and  $1 \leq l \leq |\mathcal{L}|$  defined as follows:

$$Y_{k,l} = \begin{cases} 1 & \text{when } l\text{-th link is failed due to } k\text{-th obstacle} \\ 0 & \text{otherwise} \end{cases}$$

Clearly we need to have,

$$\sum_{k=1}^K Y_{k,l} \geq 1 \quad \forall l \quad (4.5.2)$$

At time instant  $t$ , a link  $l$  is failed due to an obstacle  $k$ , only if the trajectory line of obstacle  $k$  intersects the line segment representing the link. For notational brevity, from now onward we refer to the trajectory of obstacle  $k$ , and the line segment representing the link  $l$ , simply by the indices  $k$  and  $l$  respectively. A link  $l$  is defined by its two end points  $l_b$  and  $l_e$ . Now an obstacle  $k$  intersects a link  $l$  at point  $P_{k,l}$  if and only if  $k$  partitions the line segment  $l$  in the ratio  $\alpha_{k,l}$  and  $1 - \alpha_{k,l}$ , where  $0 \leq \alpha_{k,l} \leq 1$ . The coordinate of  $P_{k,l}$  can be computed from  $l$  and  $\alpha_{k,l}$  as follows.

$$P_{k,l}[x] = \alpha_{k,l}l_b[x] + (1 - \alpha_{k,l})l_e[x] \quad (4.5.3)$$

$$P_{k,l}[y] = \alpha_{k,l}l_b[y] + (1 - \alpha_{k,l})l_e[y] \quad (4.5.4)$$

$$P_{k,l}[t] = t \text{ such that } l \in \mathcal{L}_t \quad (4.5.5)$$

Here for a point  $P$ ,  $(P[x], P[y])$  denotes its spatial coordinates on the Cartesian plane, and  $P[t]$  denotes the time when this point on the link  $l$  is being considered and  $\mathcal{L}_t$  is the set of all such links which are blocked at time  $t$ . Similarly, for points  $l_b$  and  $l_e$ , the spatial coordinates on the Cartesian plane are given by  $(l_b[x], l_b[y])$  and  $(l_e[x], l_e[y])$  respectively. Thus  $\mathcal{L} = \cup_{1 \leq t \leq \tau} \mathcal{L}_t$ . Now the intersection point  $P_{k,l}$  must also lie on the trajectory line  $k$ , and it must therefore satisfy the equation of the line  $k$  given by

$$\frac{x - x_k}{\delta x_k} = \frac{y - y_k}{\delta y_k} = \frac{t - 0}{\delta t_k}$$

Here  $(x_k, y_k, 0)$  is the initial point on the line  $k$  at time  $t = 0$ . The parameters  $\delta x_k, \delta y_k$  and  $\delta t_k$  are the intercept values with  $x, y$  and  $t$  axes respectively. Thus, we have  $x = x_k + t \frac{\delta x_k}{\delta t_k}$  and  $y = y_k + t \frac{\delta y_k}{\delta t_k}$ . Since for a line,  $\frac{\delta x_k}{\delta t_k}$  and  $\frac{\delta y_k}{\delta t_k}$  are constants, they can be dealt with only two variables, namely  $A_k$  and  $B_k$  respectively, in our mathematical program. Thus, whether the point  $P_{k,l}$  lies on the line  $k$  or not, can be encoded by the following linear constraints  $\forall k, l$ :

$$P_{k,l}[x] = x_k + P_{k,l}[t] A_k \quad (4.5.6)$$

$$P_{k,l}[y] = y_k + P_{k,l}[t] B_k \quad (4.5.7)$$

Note that here,  $P_{k,l}[t] = t$ , is a constant specified by the link  $l \in \mathcal{L}_t$ . Now combining (4.5.3) with (4.5.6), and (4.5.4) with (4.5.7) we have the following two linear constraints:

$$\alpha_{k,l} l_b[x] + (1 - \alpha_{k,l}) l_e[x] = x_k + P_{k,l}[t] A_k \quad (4.5.8)$$

$$\alpha_{k,l} l_b[y] + (1 - \alpha_{k,l}) l_e[y] = y_k + P_{k,l}[t] B_k \quad (4.5.9)$$

Note here  $\alpha_{k,l}, x_k, y_k, A_k, B_k$  all are optimization variables and the rest are constants. We must also ensure  $0 \leq \alpha_{k,l} \leq 1$  whenever  $Y_{k,l} = 1$ . This can be encoded as a linear constraint by introducing a large positive constant  $M$ , denoting positive infinity as follows:

$$-(1 - Y_{k,l})M \leq \alpha_{k,l} \leq 1 + (1 - Y_{k,l})M \quad (4.5.10)$$



If  $M$  is infinity, it can be verified that whenever  $Y_{k,l} = 1$ ,  $0 \leq \alpha_{k,l} \leq 1$ , and  $-\infty < \alpha_{k,l} < \infty$  when  $Y_{k,l} = 0$ .

The integrality constraints are given by

$$X_k, Y_{k,l} \in \{0, 1\} \quad (4.5.11)$$

Thus the ILP is given by the objective function (4.5.1) and constraints (4.5.2), (4.5.8), (4.5.9), (4.5.10) and (4.5.11).

We next prove that the problem of detecting multiple obstacle trajectories is NP-complete.

**Lemma 4.1.** *DOTP is NP-complete.*

*Proof.* To show that DOTP is NP-complete, we choose the point-line-cover (PLC) [153] problem as the candidate for reduction. In PLC, given  $n$  points and an integer  $K$ , we need to decide whether there exists  $K$  straight lines such that all points are covered. Here, a line  $L$  is said to cover a point  $p$  if and only if  $p$  lies on  $L$ . PLC is a known NP-complete problem [153]. Given an instance  $I = (P, K)$ , where  $P = \{p_1, p_2, \dots, p_n\}$ , of PLC, we apply the following reduction. For every point  $p_i \in P$ , we create two points  $p'_i$  and  $p''_i$ , both having coordinates same as that of  $p_i$ . Moreover,  $(p'_i, p''_i)$  forms a link of zero length at time  $t_i$  for our DOTP. Now suppose, there exists a deterministic polynomial time algorithm  $\mathcal{D}$  for DOTP. Then for a given instance  $I' = (\mathcal{L}, K)$  of DOTP, where  $\mathcal{L}_{t_i} = \{(p'_i, p''_i) \mid p_i \in P\}$ , and  $\mathcal{L} = \bigcup_{i=1}^n \mathcal{L}_{t_i}$ ,  $\mathcal{D}$  decides in polynomial time whether there exist  $K$  lines such that all links are intersected by at least one of these lines. Now by construction, if a line intersects a link  $(p'_i, p''_i)$ , it must also cover the original point  $p_i$  in PLC, and vice versa. This means we have essentially solved instance  $I$  of PLC in polynomial time. This is a contradiction. Therefore, DOTP is NP-hard, and such an algorithm  $\mathcal{D}$  cannot exist unless  $P=NP$ .

Now to show DOTP is also in NP, consider the following. Given an obstacle trajectory and a link, we can check in constant time whether the link intersects

the trajectory. Thus, for  $K$  trajectories, and  $n$  links, we can check whether all links are blocked in  $O(nK)$  time. Thus, DOTP is NP-complete.

□

## 4.6 Discovery and Implementation Algorithms

As mentioned in Section 6.2, our approach works in two phases. While in the discovery phase we try to obtain the possible trajectories of the dynamic obstacles, in the implementation phase we apply this knowledge to achieve proactive handoffs with an aim to avert link failures. We now present a greedy algorithm to obtain the set of obstacle trajectories  $\mathcal{O}$ , and follow it up with a simple handoff scheme to avert link failures.

### 4.6.1 Dynamic Obstacle Tracking Algorithm

We can model DOTP as a set cover problem as follows. Suppose we are given some candidate trajectory lines, each covering a subset of the universe  $\mathcal{L}$ . A trajectory line  $k$  covers a link  $l \in \mathcal{L}$  if  $k$  intersects  $l$ . Then an optimal (minimum) set cover of this universe, essentially gives us the required solution of the DOTP. Recall that the set cover problem is a well known APX-hard problem [154]. A greedy solution to set cover can be obtained by repeatedly selecting the set covering maximum number of yet-to-be-covered elements. This approach has an approximation ratio of  $\log n$ , where  $n$  is the number of candidate sets. Thus given a set of possible trajectory lines, we already have an approximation algorithm that returns a minimal subset of trajectory lines covering all links. The main challenge is to get this set of candidate trajectory lines, as there can be infinitely many possible lines intersecting just two links! However, using the following lemma, we show that we can always generate a finite set of such candidate lines.

**Observation 1.** *For a set of links associated with a single  $BS$  and blocked by a common obstacle, the links (line segments), the obstacle trajectory (line), and the  $BS$  (point), lie on the same plane.*

The intersection point of a blocked link and an obstacle trajectory is actually a point in 3-D, where the third dimension is the time instant at which the link is blocked by the obstacle trajectory. Let us consider a single base station, and a set of links which are blocked by a common obstacle trajectory. Thus, if we consider a 3-D environment, with time being the third axis, the  $BS$ , the said blocked links, and the obstacle trajectory must lie on a single plane. Now, for such a plane, we have the following lemma.

**Lemma 4.2.** *If a line  $L$  intersects the set of  $m$  links  $\mathcal{L} = \{\{u_1 - b\}, \{u_2 - b\}, \dots, \{u_m - b\}\}$  associated with a  $BS$   $b$ , there exists at least one  $(u_i, u_j)$  pair ( $1 \leq i, j \leq m, i \neq j$ ) such that the line passing through  $u_i$  and  $u_j$  must also intersect all of these links.*

*Proof.* Let us assume that an oracle has provided us with the actual trajectory line  $L$  of the obstacle which causes the  $m$  links to fail. We can always translate the given trajectory line parallel to itself till it reaches the location of any one  $UE$ , say  $i$ ; let the new line parallel to  $L$  and passing through  $i$  be called  $L'$ . Thereafter, keeping  $i$  as a pivot, we can rotate  $L'$  till it touches another  $UE$ , say  $j$ . Let this line obtained after rotating  $L'$  about  $i$  be called  $L''$ . Notice that  $L''$  is an extension of the line segment formed by joining  $i$  and  $j$ . Furthermore, the translation and rotation operations have been done ensuring that all links are touched by the line  $L''$ , i.e., none of the links become uncovered. Hence  $L''$  is a candidate trajectory.  $\square$

In Fig. 4.5,  $L$  represents the actual trajectory line.  $L'$  is the translated line that passes through  $u_4$  and parallel to  $L$ . After rotating  $L'$ , about  $u_4$ , we get  $L''$  which passes through both  $u_2$  and  $u_4$ , and intersects all the links in  $\mathcal{L} = \{\{u_1 - b\}, \{u_2 - b\}, \{u_3 - b\}, \text{ and } \{u_4 - b\}\}$ .

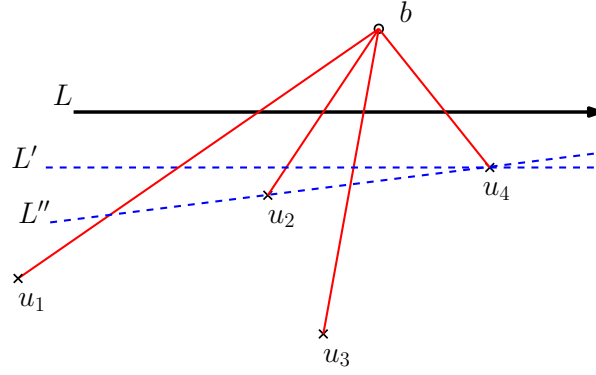


Figure 4.5:  $L''$  is a Candidate Trajectory Line that Intersects All the Blocked Links.

Remember that  $\mathcal{L}$  represented a set of blocked links which had been blocked by a single obstacle. Let us denote  $\hat{\mathcal{L}}$  as the set of all blocked links associated with  $b$ . If there are  $k$  obstacles that block all the links in  $\hat{\mathcal{L}}$ , there are essentially  $k$  planes as described above. However, we have no idea regarding the number of obstacles that were involved in blocking. Using the above lemma, we can essentially find  $\binom{|\hat{\mathcal{L}}|}{2}$  planes (each containing  $b$ , and a pair of UEs). This actually gives us a way to obtain a finite candidate trajectory set. If we consider the set  $\mathcal{C}$  of all lines passing through the all possible pairs of UEs associated with a BS, the required trajectory lines must be a subset of  $\mathcal{C}$ . As there can be  $\binom{|\hat{\mathcal{L}}|}{2}$  possible UE pairs,  $|\mathcal{C}| = O(|\hat{\mathcal{L}}|^2)$ . Moreover, the optimal solution must be the minimum subset of  $\mathcal{C}$  covering entire  $\hat{\mathcal{L}}$ .

Given a candidate line  $c \in \mathcal{C}$  joining two end points of two links from  $\mathcal{L}$ , we scan the entire  $\mathcal{L}$ , to find how many links are intersected by this line  $k$  (taking into account the respective lines), which gives us the set  $\mathcal{C}_k \subseteq \mathcal{L}$ , the set of links covered by this line. This can be computed in linear time with respect to  $\mathcal{L}$ . Once we obtain all such  $\mathcal{C}_k$ , we follow the footsteps of greedy set cover solution described above. This will return us at most  $OPT \times \log(|\mathcal{L}|)$  many trajectories where  $OPT$  is the size of optimal set cover solution. This process is formalized into Algorithm 4.2. Note that the `intersection()` procedure has to perform the checking in 3-D, time being the third dimension.

Algorithm 4.2 takes input the set  $\mathcal{L}$  after the time  $\tau$ , and outputs the set  $\mathcal{O}$  of

**Algorithm 4.2:** Set Cover based DOTP.

---

```

Data:  $\mathcal{L}$ 
Result:  $\mathcal{O}$ 
1 Set  $\mathcal{U} \leftarrow \mathcal{L}$  // universe to cover
2 Set  $\mathcal{C} \leftarrow \emptyset$  // candidate trajectories
3 Set  $\mathcal{O} \leftarrow \emptyset$  // output trajectories which covers  $\mathcal{U}$ 
  /* generating candidate lines */
4 for  $\{u - b\} \in \mathcal{L}$  do
5   for  $\{u' - b\} \in \mathcal{L}$  and  $u' \neq u$  do
6      $c \leftarrow (u, u')$  // candidate line
7      $\mathcal{C} \leftarrow \mathcal{C} \cup \{c\}$ 
  /* apply greedy set cover */
8 while  $\mathcal{U} \neq \emptyset$  do
9   for  $c \in \mathcal{C}$  do
10     $I_c \leftarrow \{u \in \mathcal{U} \mid \text{intersection}(c, u) = 1\}$ 
11     $k \leftarrow \underset{c' \in \mathcal{C}}{\text{argmax}} \{I_{c'}\}$ 
12     $\mathcal{U} \leftarrow \mathcal{U} \setminus I_k$ 
13     $\mathcal{O} \leftarrow \mathcal{O} \cup \{k\}$ 
14 Return  $\mathcal{O}$ 

```

---

trajectories that intersects all links in  $\mathcal{L}$ . Since  $|\mathcal{C}| = O(|\mathcal{L}|^2)$ , and `intersection()` takes  $O(|\mathcal{L}|^2)$ , the overall running time of Algorithm 4.2 is  $O(|\mathcal{L}|^4)$ .

### 4.6.2 Proactive Handoff Algorithm

Armed with the set  $\mathcal{O}$  of reported trajectory lines, and the currently active set of links  $\mathcal{A}$ , we can now proceed towards triggering handoffs for obstacle prone links in the implementation phase. The algorithm begins off by taking input  $\mathcal{O}$  and  $\mathcal{A}$ , and outputs a set of new links  $\mathcal{L}'$ . Here, an active link  $a \in \mathcal{A}$  is represented by a pair  $\{u - b\}$ . For each such active link  $a$ , we call the `intersection()` subroutine, and check for any possible intersection with trajectories in  $\mathcal{O}$  in future; if such a possibility exists, we trigger a handoff. For the UE  $u$ , we check whether an mmWave BS  $b' \neq b$  exists, such that  $\{u - b'\}$  is not intersected by any of the obstacle trajectories in  $\mathcal{O}$  (in future). If such an mmWave BS  $b'$  exists, we update the set  $\mathcal{L}'$  with the new link  $\{u - b'\}$ . If no such mmWave BS exists, the LTE

BS steps in to provide sub-6 GHZ service, and we update the set  $\mathcal{L}'$  with the link  $\{u - \text{LTE BS}\}$ . This process is formalized into Algorithm 4.3.

---

**Algorithm 4.3:** Proactive Handoff Algorithm.
 

---

**Data:**  $\mathcal{A}, \mathcal{O}, \mathcal{B}$   
**Result:**  $\mathcal{L}'$

```

1 Set  $\mathcal{L}' \leftarrow \emptyset$  // new links
2 for  $a = \{u - b\} \in \mathcal{A}$  do
3   for  $o \in \mathcal{O}$  do
4     if  $\text{intersection}(a, o) = 1$  then // trigger handoff
5       Set  $\text{flag} \leftarrow 0$ 
6       for  $b' \in \mathcal{B} \mid b' \neq b$  do
7         for  $o' \in \mathcal{O}$  do
8           if  $\text{intersection}(\{u - b'\}, o') = 0$  then
9              $\mathcal{L}' \leftarrow \mathcal{L}' \cup \{u - b'\}$ 
10            Set  $\text{flag} \leftarrow 1$ 
11            break;
12       if  $\text{flag} = 0$  then
13          $\mathcal{L}' \leftarrow \mathcal{L}' \cup \{u - \text{LTE BS}\}$ 
14    $\mathcal{A} \leftarrow \mathcal{A} \setminus a$ 
15 Return  $\mathcal{L}'$ 

```

---

## 4.7 Simulation Experiments

To demonstrate the handoff performance of our algorithm and compare it with an RGB-D camera based method [90], we first select a smaller simulation setup as follows. We consider a service area of size  $100 \text{ m} \times 100 \text{ m}$ . There are 2 **mmWave BSs** which can provide coverage to the **UEs** having **LOS** with any of them, and an **LTE BS** which provides ubiquitous coverage. There are a few obstacles of size  $1 \text{ m} \times 1 \text{ m}$  which are moving independently inside with a velocity chosen uniformly at random from  $[0, 10] \text{ m/s}$ . The obstacle can change its direction after an epoch  $T$  of duration 5 second. There are several **UEs** communicating with the **mmWave BSs**. The obstacles cause link failures, and the same is reported at the corresponding **mmWave BS** over the discovery phase. After the end of time  $\tau = 3$  second, each **mmWave BS** runs Algorithm 4.2, and determines a

set of trajectories  $\mathcal{O}$ . Thereafter, Algorithm 4.3 is run in a centralised manner at the LTE BS, and the set of links that are at risk of breaking due to the obstacles is determined. Handoff is triggered at such UEs in an attempt to provide uninterrupted service. As is obvious, scarce UE density will lead to

		Actual Obstruction				Actual Obstruction	
		+	-			+	-
Predicted Obstruction	+	19.7	29.8	+ <td style="border: 1px solid black; text-align: center;">40.2</td> <td style="border: 1px solid black; text-align: center;">7.3</td>	40.2	7.3	
	-	20.8	29.7	- <td style="border: 1px solid black; text-align: center;">7.5</td> <td style="border: 1px solid black; text-align: center;">42</td>	7.5	42	
a For 10 links.				b For 100 links.			

Table 4.3: Confusion Matrices.

low number of link failures, leading to insufficient information, and inaccurate predictions. We demonstrate the effect of UE density in two confusion matrices; Table 4.3a shows the effect for 10 failed links, while Table 4.3b shows the effect for 100 failed links. It is evident that the algorithm performs poorly when the UE density is low, giving correct handoff requirements only 50% of the time. However, the performance improves to above 80% when the number of failed links is 100. We point out here though, that achieving complete accuracy appears unlikely without dedicated tracking hardware. Indeed, the same obstacle may obstruct links associated with multiple mmWave BSs, which our algorithm would report as multiple trajectories, thereby introducing inherent false positives. In other words, in the absence of sufficient link failure information, the algorithm performs poorly.

Next, we show the tracking performance of an RGB-D camera based approach. As in [90, 102], we consider Microsoft Kinect cameras [106] to be the tracking devices. These cameras typically have a field of view of  $90^\circ$ , and range around few meters (0.5 to 3.5 m) [106]. We vary the camera count, and deploy them uniformly at random inside the coverage area. We assume that obstacles whose trajectories lie fully within the coverage area of the cameras, are successfully tracked. As shown

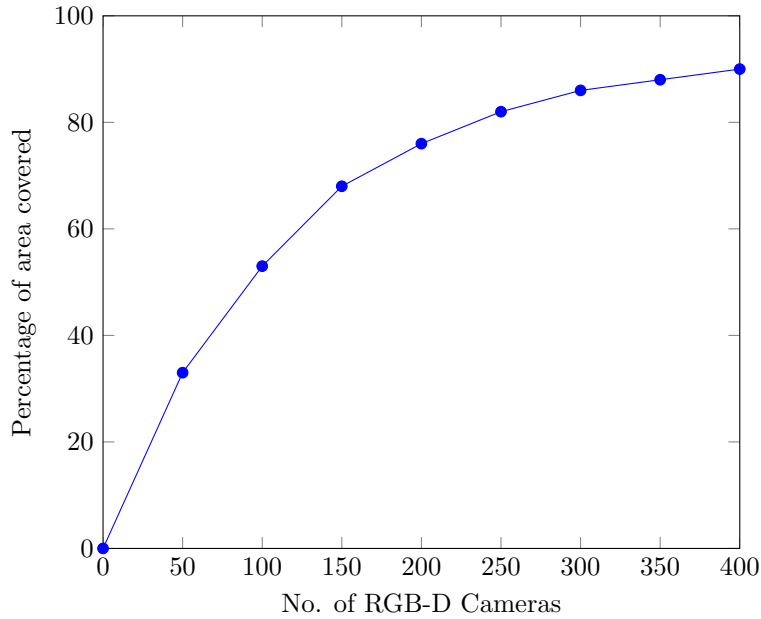


Figure 4.6: Coverage by Typical RGB-D Cameras.

in Fig. 4.6, it takes hundreds of RGB-D cameras are deployed for the considered  $100 \text{ m} \times 100 \text{ m}$  service area to achieve complete tracking capability.

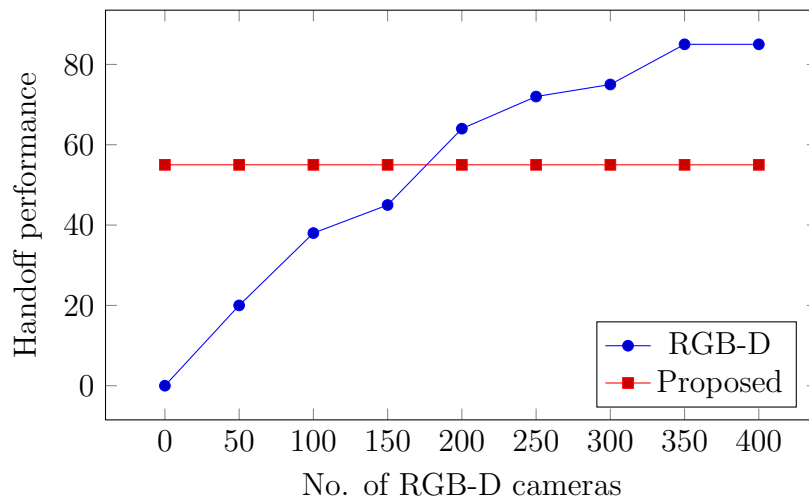


Figure 4.7: Handoff Performance vs RGB-D Camera Count.

The comparison of handoff performance between our method, and an RGB-D camera based method [90] is given in Fig. 4.7. We define handoff performance as the percentage of links actually required handoffs, and were handed over to unobstructed mmWave BSs. For low camera count, the performance is worse since



handoff requires tracking information at two stages; one to determine the at risk links, and the other to predict whether the newly allocated link will be obstructed. If the number of deployed RGB-D cameras is below 150 for the considered coverage area of size  $100 \text{ m} \times 100 \text{ m}$ , our proposed approach clearly outperforms the camera based approach. If the camera density is huge (more than 150 in this case), then the camera based approach performs better than ours. Since ours is simply a prediction algorithm based on historical link failure data alone, sufficient coverage obtained by means of deploying a large number of RGB-D cameras will certainly outperform our approach. In real life, achieving complete RGB-D camera coverage will seldom be the case; this is because of their small ranges (few meters), and all obstacles may not have line of sight with the deployed cameras. To achieve this practically, a huge number of cameras are required to be deployed in the coverage area, with marked increase in deployment costs.

To validate the performance of our proposed tracking approach in a real life scenario, we run simulations using the CRAWDDAD taxi dataset [69], which contains GPS taxi traces in San Francisco Bay Area, USA. We randomly select 15 taxis from this dataset as our dynamic obstacles, and run our experiments considering a service area of size  $5000 \text{ m} \times 5000 \text{ m}$ . We consider a set of link requests generated uniformly at random over the service area, for the considered time period epoch  $T = 12$  second. We execute Algorithm 4.2 on the blocked links generated upto time  $\tau$ . We measure the performance of our proposed scheme based on three metrics, namely *accuracy*, *precision*, and *sensitivity*, which are defined as follows.

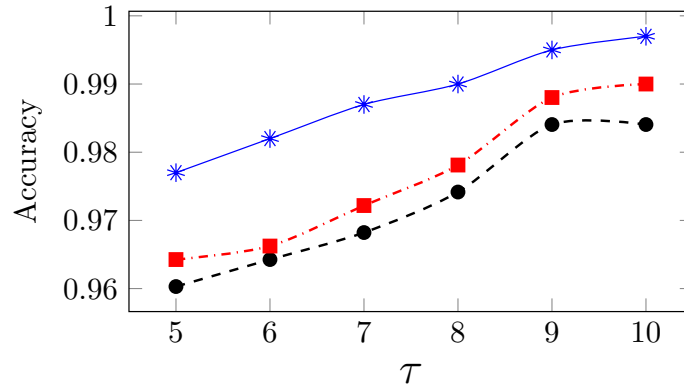
- *accuracy*: the ratio of number of links correctly predicted according to their blocking status, to the total number of links under consideration.
- *sensitivity*: ratio of the number of links correctly predicted to be “blocked”, to the total number of actually blocked links.
- *precision*: ratio of the number of links correctly predicted as “blocked”, to the total number of links predicted as “blocked” (both correctly, as well as incorrectly).

We vary the duration of discovery phase  $\tau$  from 5 to 10 seconds, and plot these three metrics in Fig. 4.8 for different number of obstacles and broken links. As is intuitive, all three metrics see an improvement with increase in discovery phase time  $\tau$ . With increase in  $\tau$ , we get more information about the blocked links, leading to improved accuracy, sensitivity, and precision. However, we cannot indefinitely increase  $\tau$ , since as  $\tau$  increases, duration of implementation phase decreases, leading to less utility of the learnt information. In other words, if we increase  $\tau$  indefinitely, the duration of the implementation phase shortens, thereby reducing the time for which the output of our tracking algorithm can be used for achieving uninterrupted service.

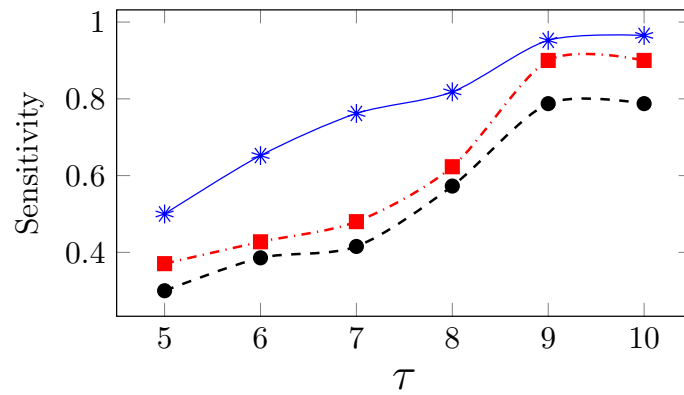
## 4.8 Conclusion

In this chapter, we have taken a rather ambitious attempt at tracking multiple dynamic obstacles in an mmWave enabled coverage area, without deploying any sort of dedicated tracking hardware. The aim of this work was to demonstrate the viability of such an approach, and achieve subsequent proactive handoffs in an attempt to lower link failures. We proved the hardness of the problem, and modelled it as a version of the classical set cover problem, which we solved using the usual greedy approximation approach. Given the range constraints of RGB-D cameras, we show that unless near complete coverage is provided with huge number of such hardware, our approach performs better obstacle tracking, and subsequent handoffs. We further show that the performance of our predictive approach improves with high UE density (more link failure information), hypothesising that in sparse networks, we might need augmentation with some additional hardware to obtain satisfactory prediction, which can lead to new optimization problems. Since our tracking algorithm runs independently at each mmWave BS, in the implementation phase we can end up having more reported trajectories than the number of actual obstacles. This is because, the same obstacle which obstructed links associated with multiple mmWave BSs may be reported as multiple obstacles in our approach. One seemingly interesting way to improve the approaches in this

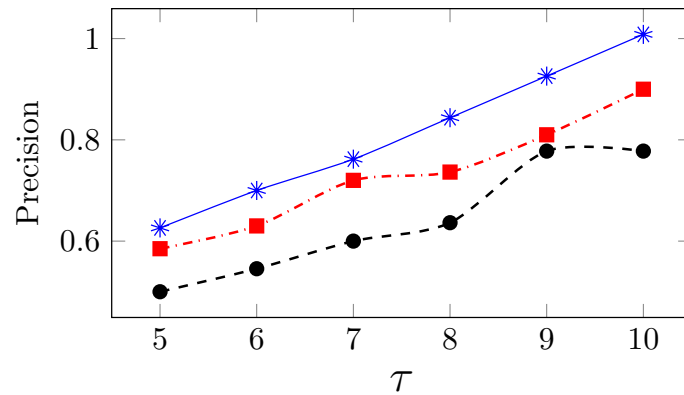
-●- #Obstacles=5,#Link requests=500    -■- #Obstacles=10,#Link requests=500    -\* - #Obstacles=10,#Link requests=1000



(a) Accuracy.



(b) Sensitivity.



(c) Precision.

Figure 4.8: Accuracy, Sensitivity and Precision vs. Discovery Time  $\tau$  for Varying Number of Dynamic Obstacles and Link Requests.

work is by considering dedicated roads and sidewalks along which the dynamic obstacles are constrained to move upon. We have not assumed any such paths; it

seems logical that such constraints will in fact, improve prediction performance. Such avenues will be explored in our future work.

## Chapter 5

# Nearest is not the best! Towards Stable Link Allocation<sup>1</sup>

In [Chapter 3](#), we provided a path selection algorithm that assigned the path with the highest calculated [SINR](#), without any consideration for [UE](#) mobility. However, that might not be the most prudent approach. Indeed, a link with high [SINR](#) at a time may degrade rapidly at the next time instant due to the introduction of obstacles on the transmission path brought about by [UE](#) mobility. We illustrate this by an example in [Figure 5.1\(b\)](#). Let us consider a [D2D](#) link between two [UEs](#)  $U_A$  and  $U_B$  moving in the directions as shown, with velocities  $v_A$  and  $v_B$  respectively. At time  $t_0$ , the two [UEs](#) had an unobstructed [LOS](#), and were assigned a link. However, there is a nearby static obstacle, which at time  $t_2$ , will cause obstruction of the said link. Depending on [UE](#) velocity and static obstacle density, such obstruction may soon trigger handoff requests, which will not only decrease network throughput, but also degrade user experience. A similar incident will occur if the transmission distance between two devices increases rapidly due to their mobility. We illustrate this in [Figure 5.1\(a\)](#). The two [UEs](#)

---

<sup>1</sup>This chapter is primarily based on the following paper:  
**Subhojit Sarkar**, Subhankar Ghosal, Subhadip Bandyopadhyay and Sasthi C. Ghosh. “A Stable Link Allocation Algorithm for 5G Millimeterwave Networks.” In: *Proceedings of the 15th International Conference on COMmunication Systems & NETworkS (COMSNETS 2023)* IEEE, Bengaluru, India, January 3-8, 2023, pp. 674-681, DOI: 10.1109/COMSNETS56262.2023.10041333.

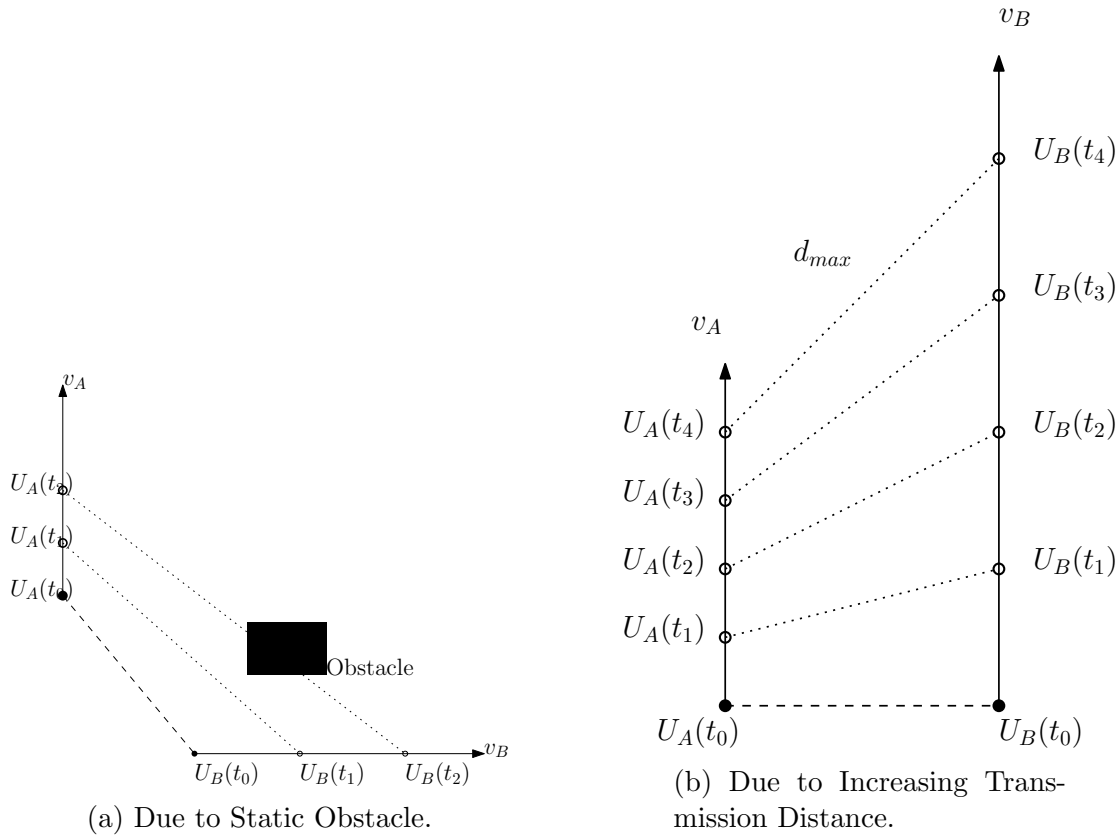


Figure 5.1: Toy Example Demonstrating Link Failure.

were close to each other at time  $t_0$ , and hence a good **D2D** link was established. However, due to the high relative velocity between them, the transmission distance increases rapidly, and at time  $t_4$ , exceeds the threshold distance  $d_{max}$ , above which **mmWave** communication is not viable. This again leads to link degradation, and potentially link failure.

In this chapter, we take the notion of *link stability* as our metric, and try to assign transmission paths that are likely to remain active for a long time. In **Figure 5.2(a)**, the **BS**  $B_1$  is closer to the **UE** than  $B_2$ ; however, due to its mobility, the **UE** is likely to spend more time in the transmission range of  $B_2$ . Thus counter-intuitively, the **UE** is allocated  $B_2$ . A similar thing happens due to static obstacles. In **Figure 5.2(b)**, the closer  $B_1$  is likely to lose **LOS** with the **UE** due to the static obstacle, unlike the farther away  $B_2$ . In other words, it might not be advisable to always assign the **mmWave** link with the highest **SINR** (and consequently highest **RSS**), based on current network parameters, but rather assign them looking towards the future, with an aim to avoid frequent link

failures. This is the main motivation of this chapter. Note that the same notion can be extended to relaying **UEs** as well; in that case, we would not always allocate the shortest **D2D** link, rather study on how long the possible links are likely to maintain the propagation requirements of **mmWaves**.

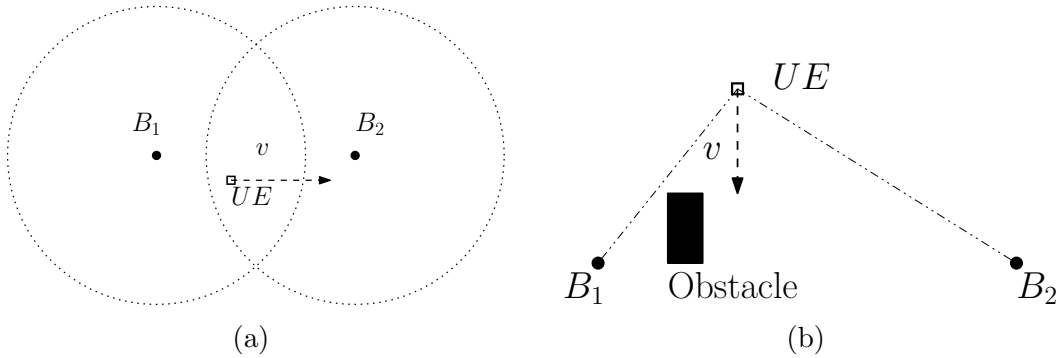


Figure 5.2: Further BS is Better Due to (a) UE Mobility, and (b) Static Obstacle.

More formally, the contribution in this chapter are as follows:

1. We give a mathematical formulation of the the problem of assigning stable **mmWave** links, and show that the problem is NP-complete.
2. Assuming a known **UE** mobility model, we provide a link allocation algorithm that allocates links that have the longest calculated link active time.
3. Via simulations, we show that this approach increases the average link active time ( $\mathbb{T}_{active}$ ).

## 5.1 System Model

There is a central **LTE BS** operating at sub-6GHz, providing ubiquitous coverage. A set  $\mathcal{B} = \{b_1, b_2, \dots, b_B\}$  of **mmWave BSs** is distributed uniformly at random inside the coverage area. These **BSs** are connected to the **LTE BS**, as well as among themselves by a high speed backhaul network. There are some known static obstacles inside the coverage area, which block **mmWave** signals. Some

mobile **UEs** are moving independently inside the coverage area; they can include pedestrians with handheld mobile phones, bicyclists, or vehicular **UEs** with in-built **mmWave**-enabled devices. All the **UEs** move following the random waypoint mobility model [155]. More specifically, a **UE** chooses a direction uniformly at random from  $[0, 2\pi]$ , and a velocity uniformly at random from  $[v_{min}, v_{max}]$ , and continues in said direction for a specific amount of time, after which the process repeats. All the **UEs** have electronically steerable directional antennas [6] and **mmWave** transreceivers, to compensate for the high attenuation of **mmWaves**. Each **UE** has a superaccurate **GPS** chip [144], an inbuilt compass, and an accelerometer, which it uses to communicate its location and velocity to the **LTE BS** as a part of the location update process. Two devices can communicate over the **mmWave** spectrum only if the distance between them is within  $d_{max}$ , and they have an **LOS** between them. Furthermore, to limit network congestion, the maximum allowable hops in a transmission path is constrained to  $h_{max}$ .

## 5.2 Problem Formulation

Let us consider that at time  $t$ , there is a set  $\mathcal{U}(t)$  of demanding **UEs**, and a set  $\mathcal{W}(t)$  of willing-to-relay **UEs**. Additionally, there is a time invariant set  $\mathcal{B}$  of small cell **mmWave** BSs. There is a set  $\mathcal{S}$  of known static obstacles, whose locations and dimensions are known at the **LTE BS**. An idle **UE**  $w \in \mathcal{W}(t)$  can act as a relay for only one demanding **UE**  $u \in \mathcal{U}(t)$ . We consider a graph  $\mathcal{G}(t) = (V, E)$  where there is a vertex corresponding to each element in  $\mathcal{U}(t) \cup \mathcal{W}(t) \cup \mathcal{B}$ , and there is an edge between a pair of vertices if the two devices have an **LOS** path from one another, and their distance is within a threshold  $d_{max}$ .

Note that there may exist multiple potential paths from each  $u_i \in \mathcal{U}(t)$  towards each BS  $b_k$  in  $\mathcal{B}$ . Let  $p_{ik}$  be the set of all such paths from  $u_i$  to  $b_k$ . Let  $P_i = \bigcup_{k \in \mathcal{B}} p_{ik}$  be the set of all paths from  $u_i$  to any of the BSs. For a given **UE**  $u_i$ , our task is to allocate the most stable path in  $P_i$ . The stability of a path is defined in the next



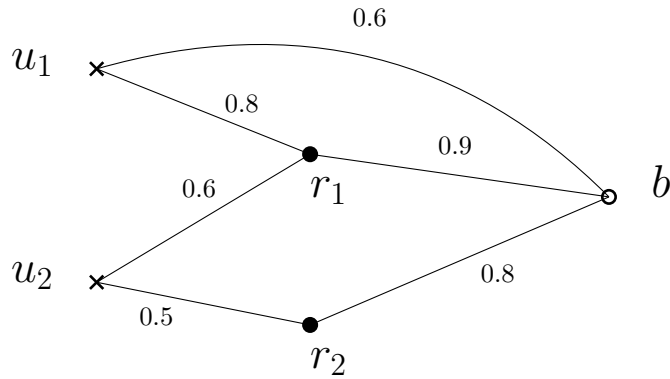


Figure 5.3: A Toy Example Demonstrating Stable Path Finding.

paragraph. If there is no such path from a UE, i.e.,  $|P_i| = 0$ , data transmission takes place over traditional sub-6GHz network.

Since UEs are mobile and network topology is time varying, a path active at time  $t$  may not remain active at next time instant  $t + 1$ . Re-assigning a path in place of a broken one would involve substantial searching, which would incur an overhead on the communication. Thus, we need to minimize the number of path switches. A way to achieve this is by selecting paths which are less likely to fail in the near future. A path would fail if any one of its corresponding link fails. Now, we define the stability of a link  $\vec{yz}$  at time  $t$  as the probability  $e_{yz}(t)$  that the two devices  $y$  and  $z$  can communicate using mmWaves at time  $t$ , i.e.,  $y$  and  $z$  can communicate if  $\text{dist}(y,z) \leq d_{max}$  and there is an LOS between them at time  $t$ . Since the UEs are assumed to move independently of each other, the stability of each link is assumed to be independent. Thus for a multi-hop path, the stability of the entire path is defined as the product of the individual stabilities of all the links that form the path. That is, the stability of a path originating at  $u_i \in \mathcal{U}(t)$ , going through the willing relays  $w_1, w_2, \dots, w_n \in \mathcal{W}(t)$  and terminating at  $b_k \in \mathcal{B}$  is defined as  $S^i = e_{iw_1} e_{w_1 w_2} \dots e_{w_n b_k}$ . Our aim is to find a stable path for each UE to any BS. In other words, our objective is to find a set of most stable relay disjoint paths for all  $u_i \in \mathcal{U}(t)$  to some BS in  $\mathcal{B}$ . Here two paths are disjoint if there are no common willing relays between them; this is because each relaying UE can handle a maximum of one incoming/outgoing traffic at any given time instant. Note however, that multiple paths can terminate at the same mmWave BS.

We illustrate this using a simple example in Fig. 5.3. There are two UEs  $u_1$  and  $u_2$  which want to communicate with a BS  $b$ . There are 2 nearby idle UEs  $r_1$ , and  $r_2$  which can act as relays. The edge weights denote the stabilities of the corresponding links. Note that a relaying UE can relay the data from at most one demanding user. Thus the corresponding stabilities of the paths  $u_1 - b$ ,  $u_1 - r_1 - b$ ,  $u_2 - r_1 - b$ , and  $u_2 - r_2 - b$  are 0.6, 0.72, 0.54, and 0.4 respectively. The problem basically boils down to allocating *stable* multi-source, multi-destination, edge-disjoint paths. We call this problem stable path selection problem (SPSP).

Let us consider an indicator variable  $X_{yz}^i$ , for  $y \in \mathcal{U}(t) \cup \mathcal{W}(t)$ ,  $z \in \mathcal{W}(t) \cup \mathcal{B}$ , and  $u_i \in \mathcal{U}(t)$ , such that

$$X_{yz}^i = \begin{cases} 1, & \text{if } \vec{yz} \text{ is assigned for } u_i \\ 0, & \text{otherwise.} \end{cases} \quad (5.2.1)$$

We can now formally present the problem as a mathematical program as follows:

$$\text{maximize } \sum_i \left( S^i \sum_z X_{iz}^i \right) \quad (5.2.2)$$

subject to the following constraints

$$S^i = \prod_y \prod_z \left( X_{yz}^i * e_{yz} + (1 - X_{yz}^i) \right) \forall u_i \quad (5.2.3)$$

$$\sum_y X_{y\alpha}^i - \sum_z X_{\alpha z}^i = 0 \forall \alpha \in \mathcal{W}(t) \ \& \ \forall u_i \quad (5.2.4)$$

$$\sum_y X_{y\alpha}^i \leq 1 \forall \alpha \in \mathcal{W}(t) \ \& \ \forall u_i \quad (5.2.5)$$

$$\sum_i X_{yz}^i \leq 1 \forall y, z \quad (5.2.6)$$

$$\sum_y \sum_z X_{yz}^i \leq h_{max} \forall u_i \quad (5.2.7)$$

$$X_{yz}^i \in \{0, 1\} \quad (5.2.8)$$

We intend to maximize the sum of stabilities of the allocated paths (5.2.2). The stability of the path for each requesting UE is computed by equation (5.2.3).

Here if a link  $\vec{yz}$  is allocated to the path originating at  $u_i$  (i.e.,  $X_{yz}^i = 1$ ), we get  $X_{yz}^i * e_{yz} + (1 - X_{yz}^i) = e_{yz}$ ; otherwise, we get 1 when the link is not allocated to the path (i.e.,  $X_{yz}^i = 0$ ). Flow constraint (5.2.4) signifies that the number of incoming links is exactly equal to the number of outgoing links at all relay nodes on a given path. Constraint (5.2.5) together with (5.2.4) ensures that each relay node appears at most once in a particular path. Furthermore, a relay device can be part of at most one such path, which is ensured by the constraint (5.2.6). The maximum hop count constraint (5.2.7) puts an upper bound on the number of links in an allotted path. The  $X_{yz}^i$ 's are binary variables as given by (5.2.8). The objective function  $(S^i \sum_z X_{iz}^i) = S^i$  when a path from UE  $i$  to any BS in  $\mathcal{B}$  exists, and 0 otherwise.

We now prove that our problem is NP-complete.

**Lemma 5.1.** *SPSP is NP-complete.*

*Proof.* Consider a special case of our problem where every edge weight  $e_{yz}$  of  $\mathcal{G}(t)$  has the value 1. Now we construct a flow graph  $\mathcal{G}'(t) = (V', E')$  from  $\mathcal{G}(t) = (V, E)$  as follows; we designate  $\mathcal{U}(t)$  to be the set of source vertices and introduce  $|\mathcal{U}(t)|$  many sink vertices, whose set is denoted as  $\mathcal{S}$ . Thus  $V' = V \cup \mathcal{S} = \mathcal{U}(t) \cup \mathcal{R}(t) \cup \mathcal{B} \cup \mathcal{S}$ . In addition to the edges in  $\mathcal{G}(t)$ , we introduce a few extra edges (from each  $b \in \mathcal{B}$  to each  $s \in \mathcal{S}$ ) each with unit weights in  $\mathcal{G}'(t)$  such that  $\mathcal{B} \cup \mathcal{S}$  forms a complete bipartite graph. Thus  $E' = E \cup \{(b, s) | b \in \mathcal{B}, s \in \mathcal{S}\}$ . The edge weights in  $\mathcal{G}'(t)$  denotes the flow capacity through that edge. Thus  $\mathcal{G}'(t)$  essentially denotes a multi-commodity flow network. Now any algorithm that solves SPSP for the instance  $\mathcal{G}(t)$ , essentially obtains a optimal solution of the multi-commodity flow problem for the instance  $\mathcal{G}'(t)$  by selecting appropriate edges between  $\mathcal{B}$  and  $\mathcal{S}$ . Since multi-commodity flow problem is NP-Complete even for unit capacity values [156], SPSP is also NP-Complete.  $\square$

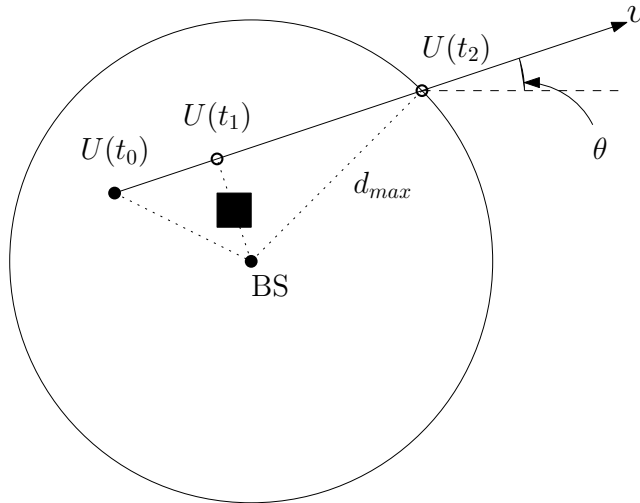


Figure 5.4: Estimating Link Active Time.

### 5.3 Proposed Stable Path Finding Algorithm<sup>2</sup>

In this section, we propose a greedy solution to SPSP. We consider that over a time epoch, a UE continues along a straight line, with no change in velocity. This is not unjustified, since in real life, obstacles usually do not change directions frequently. We illustrate the UE-BS scenario in detail, while the UE-UE case can be similarly approached. In Figure 5.4, there is a single static mmWave BS at  $(b_x, b_y)$ , a static obstacle at position  $(o_x, o_y)$ , and a mobile UE. At time  $t_0$ , the location of the UE is at  $U(t_0) = (x_0, y_0)$ , its velocity is  $v$ , and it is moving at an angle  $\theta$  with the horizontal axis. From high school geometry, the position of the UE at time  $t$  will be

$$x_t = x_0 + v \times \cos(\theta) \times t \quad (5.3.1)$$

$$y_t = y_0 + v \times \sin(\theta) \times t \quad (5.3.2)$$

According to our mobility pattern, we can find the time varying equation of the link between the UE and the BS as:

$$y - y_t = \frac{b_y - y_t}{b_x - x_t} \times (x - x_t) \quad (5.3.3)$$

<sup>2</sup>This algorithm is drawn from the following paper:  
Subhojit Sarkar, Sasthi C. Ghosh: *Mobility Aware Path Selection for Millimeterwave 5G Networks in the Presence of Obstacles*. CCCE 2023: 67-80

Plugging in the position of the static obstacle  $(o_x, o_y)$  in Equation (5.3.3), we get the calculated time  $t_1$  at which the link will be broken due to the static obstacle lying on its transmission path. Note that an additional checking has to be done to ensure that the obstacle lies on the line segment joining the two devices, and not merely on the extended line. Also, it might very well be the case that  $t_1$  is negative; it simply implies that in such a case, the said obstacle will never block the link under consideration as long as the velocity is unchanged. A similar checking has to be done to calculate  $t_2$ , the time for which the UE will be within the maximum transmission distance  $d_{max}$ , from the BS. That is,

$$\sqrt{(x_t - b_x)^2 + (y_t - b_y)^2} \leq d_{max} \quad (5.3.4)$$

Plugging Eqns. (5.3.1) and (5.3.2) in Eq. (5.3.4), we get a second degree equation in  $t$  which gives us  $t_2$ . For non-zero  $v$ , we can get one of the following scenarios:

- two imaginary values, indicating that said UE is never within transmission range of the BS
- two unequal real values, indicating UE will lie inside the transmission range of the BS within this time interval. In this case, we take the least positive value as  $t_2$ , the time after which the UE is likely to move out of the transmission range. In case both values are negative, we can safely ignore the considered obstacle.
- two equal real values, indicating the UE is moving tangentially to the coverage area of the BS.

The final calculated transmission time for the link (considering static obstacles and UE mobility) is the minimum of the two values  $t_1$  and  $t_2$ . The UE-UE case can be handled exactly similarly. Armed with the locations of static obstacles, and the UE trajectories, we can now formally describe the proposed algorithm.

At time  $t$ , Algorithm 5.1 greedily allocates paths to all demanding UE, i.e.,  $\mathcal{U}(t)$ . We create a visibility graph  $\mathbb{G} = (V, E)$  where the nodes are communicating

**Algorithm 5.1:** Stable Path Allocation Algorithm.

---

```

Data:  $\mathcal{U}, \mathcal{W}, \mathcal{B}, \mathcal{S}, d_{max}$ 
Result:  $\mathbb{P}$ 
// Creating graph
1 for  $i \in \mathcal{U} \cup \mathcal{W}$  do
2   for  $j \in \mathcal{W} \cup \mathcal{B} \mid i \neq j$  do
3      $\mathbb{T}_{ij}^1 \leftarrow \text{obstacle}(i, j, \mathcal{S})$ 
4      $\mathbb{T}_{ij}^2 \leftarrow \text{transmit}(i, j, d_{max})$ 
5      $\mathbb{T}_{cal} \leftarrow \min(\mathbb{T}_{ij}^1, \mathbb{T}_{ij}^2)$ 
6     if  $\mathbb{T}_{cal} > 0$  then
7        $\text{addEdge}(\mathbb{G}, i, j, \mathbb{T}_{cal})$ 
// Creating supernode
8 for  $b \in \mathcal{B}$  do
9    $\text{addEdge}(\mathbb{G}, b, \mathbb{B}, \text{INF})$ 
// Greedily run widest path algorithm
10 for  $u \in \mathcal{U}$  do
11    $P_u \leftarrow \text{widestPath}(\mathbb{G}, u, \mathbb{B})$ 
12   if  $P_u \neq \phi$  then
13      $\text{Add } P_u \text{ to } \mathbb{P}$ 
14      $\text{Remove non-BS nodes in } P_u \text{ from } \mathbb{G}$ 
15   else
16      $\text{Serve } u \text{ via sub-6GHz band}$ 
17      $\text{Remove } u \text{ from } \mathbb{G}$ 
18 return  $\mathbb{P}$ 

```

---

devices (all **UEs** and all **mmWave BSs**). There is an edge between two nodes only if the calculated link active time ( $\mathbb{T}_{active}$ ) between the two devices is strictly positive. The `obstacle()` function calculates the time to failure of a link due to a static obstacle in  $\mathcal{S}$ , while the `transmit()` function calculates the time for which a link will be active due to the maximum transmission range criterion. The weight of an edge in  $\mathbb{G}$  is the minimum of the two calculates. We run a widest path algorithm `widestPath()` on  $\mathbb{G}$ , starting from each  $d \in \mathcal{U}$ , and terminating at the supernode  $\mathbb{B} = \bigcup_B f$ . This is a modified version of Dijkstra's shortest path algorithm as suggested in [157]. It returns  $P_u$ , the path beginning at  $u$  and ending at any  $b \in \mathcal{B}$ , and having the longest calculated active time. The calculated active time for a path is defined as the minimum calculated active time for a link on that path. Thus, `widestPath()` basically maximises the minimum calculated

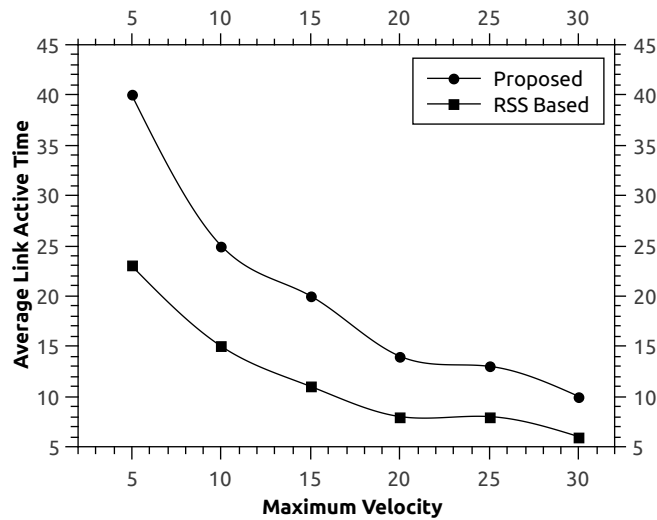


Figure 5.5: Max. UE Velocity vs  $T_{active}$ .

path active time. If the `widestPath()` function finds a path, the same is added to  $\mathbb{P}$ ; the participating non-BS nodes are removed from  $\mathbb{G}$ . In case  $P_d$  is empty, data transmission takes place over sub-6GHz service.

## 5.4 Simulation Results

The modelling parameters are mostly adapted from [152]. We consider a square area of size  $100 \text{ m} \times 100 \text{ m}$  as the coverage area under the LTE BS, which provides ubiquitous coverage. Some mmWave BSs are distributed uniformly at random inside, and provide high speed, short range services. There are users with mmWave enabled devices moving around inside with speeds ranging from  $[0, 10 \text{ m/s}]$ ; 50% of the UEs require a high speed link to any mmWave BS, while the rest are willing to act as relays. There are some known static obstacles inside the coverage area, size of each being  $5 \text{ m} \times 5 \text{ m}$ . We now define *average link active time*,  $T_{active}$  as the average time to failure of all allocated paths, which may occur due to either obstacles, or UE mobility. In Figure 5.5, we plot the effect of the maximum velocity ( $V_{max}$ ) of the UEs on the average link active time. The maximum velocity is varied from 5 m/s to 30 m/s. We see that for low speeds, the proposed approach outperforms the traditional RSS based approach by a significant amount. As  $V_{max}$

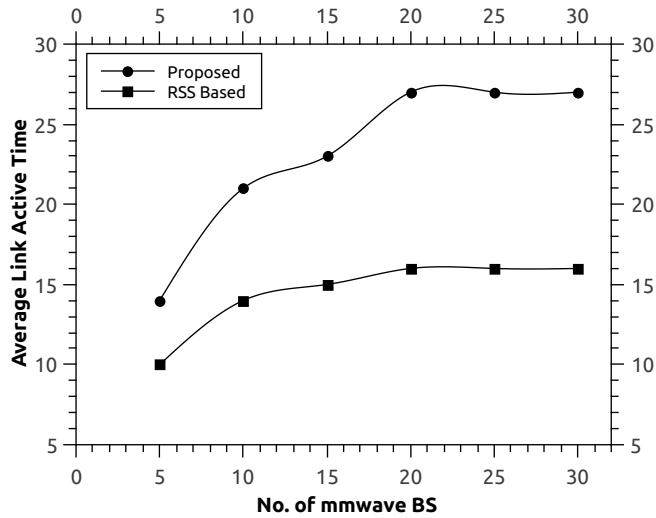


Figure 5.6: No. of mmWave BS vs  $T_{active}$ .

increases, the difference becomes smaller, although the proposed method continues to outperform the [RSS](#) based method. In [Figure 5.6](#), we plot the effect of the [mmWave BS](#) density on the average link active time. We vary the [mmWave BS](#) count from 5 to 30, and see that the average link active time stabilizes around BS count of 20, and the proposed algorithm continues to provide more stable links than the [RSS](#) based approach. The effect of the number of static obstacles inside the coverage area on the average link active time is shown in [Figure 5.7](#). The obstacle count is varied from 2 to 10; since each obstacle is of size  $25m^2$ , the number of grids covered is obtained by multiplying the obstacle count by 25. As is obvious, with increasing number of static obstacles, the average link active time falls for both the methods, though the proposed method continues to outperform the traditional method.

## 5.5 Conclusion

In this chapter, we allocate [mmWave](#) transmission paths which have the longest calculated time to failure, by taking into consideration static obstacles, and [UE](#) mobility. We show via simulation that our approach provides higher average link active times than the usual [RSS](#)-based approach. The obvious drawback of this



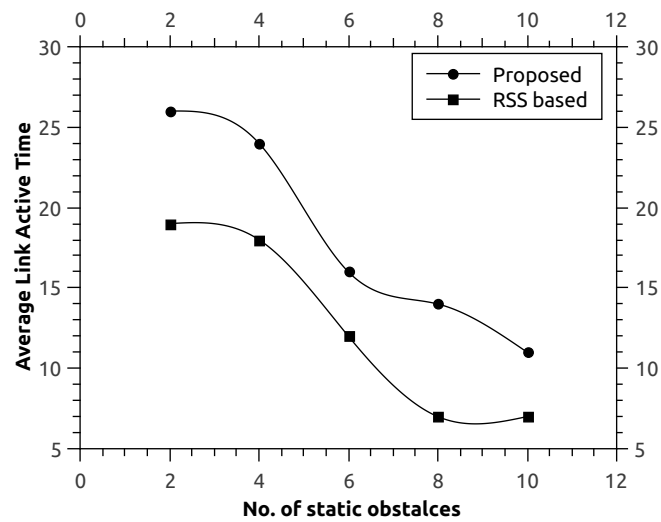


Figure 5.7: No. of Static Obstacles vs  $\mathbb{T}_{active}$ .

approach is the simple mobility model considered. In real life, UE traffic is highly correlated with the traffic environment, both spatially and temporally. One way to bypass this is by using real life mobility traces, and satellite maps to extract actual UE mobility, and use that data to obtain more accurate link failure times.



## Part II

# Efficient Resource Placement



# Chapter 6

## Joint Placement of Base Stations and Reflectors in Urban mmWave Networks<sup>1</sup>

### 6.1 Overview

Obstacle free, strict [LOS](#) requirement is one of the primary hurdles that has to be dealt with before deploying [mmWave](#) networks. One way to deal with this problem is to densely deploy small range base stations (also called gNBs), to overcome outage due to obstacles. However, these base stations are costly resources, and their dense deployment may not always be feasible. Reflectors have been proposed to augment the transmission environment, and reflect [mmWaves](#) bypassing the obstacles. Evidently, they can be used to provide coverage in area that do not have LOS with the available [BSs](#), as demonstrated in [Figure 6.1](#). One deployment approach is to place [BSs](#) and reflectors sequentially, i.e., optimally placing the available [BSs](#) first, and subsequently deploying the reflectors at locations to

---

<sup>1</sup>This chapter is based on the following paper:  
Rathindra Nath Dutta, **Subhojit Sarkar** and Sasthi C. Ghosh. “Joint Base Station and Reflector Placement in an urban mmWave Network”. In: *3rd International Mediterranean Conference on Communications and Networking (MeditCom 2023)*, IEEE, Dubrovnik, Croatia, September 4–7 2023 (to appear)

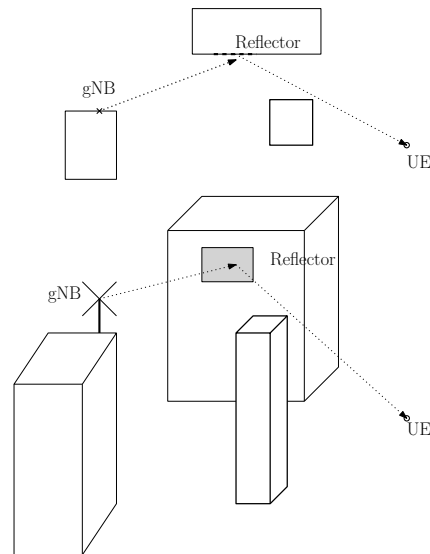


Figure 6.1: Example Demonstrating Utility of Reflecting Devices.

maximally cover the uncovered area. In a two step approach, even if optimal placement of BSs is achieved, and followed by optimal reflector placement, it is not guaranteed that the result will achieve global optimum. We demonstrate this by a toy example in Figure 6.2. We have a single BS, and only one reflector, and want to achieve maximum coverage of the service area. In case we are to place the BS without considering the reflector, the optimal location of the BS would have been as shown in Figure 6.2. Subsequently we place the reflector  $R_1$  as shown in Figure 6.2 to maximally cover the remaining area that do not have an LOS with the deployed BS. It can be observed that the area shaded in grey is still uncovered. However, as shown in Figure 6.3, we can achieve complete coverage by using only one reflector  $R_1$  and one BS by considering their placement jointly. In this chapter, we jointly deal with the problem of deploying a given number of BSs and reflectors in a mmWave environment. In other words, we try to achieve maximum coverage for a given amount of network infrastructure (BSs and reflectors). Our contribution in this chapter can be summarized as follows:

- We consider the joint placement problem of BSs and reflectors to attain maximum coverage of the service area, and we provide a ILP for the same.
- Given the hardness of the joint placement problem, we first provide a greedy solution based on set cover with an approximation bound.

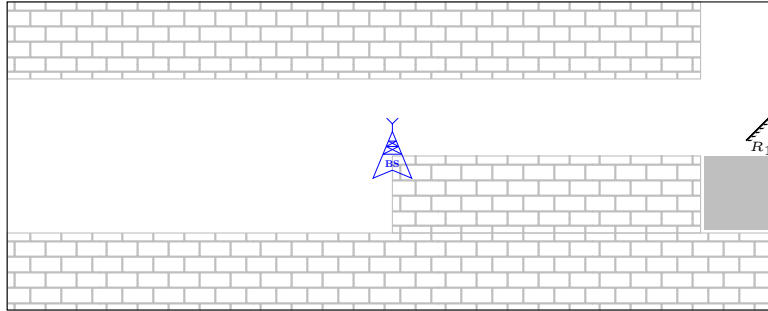


Figure 6.2: Sub-optimal Placement Failing to Achieve Maximum Coverage.

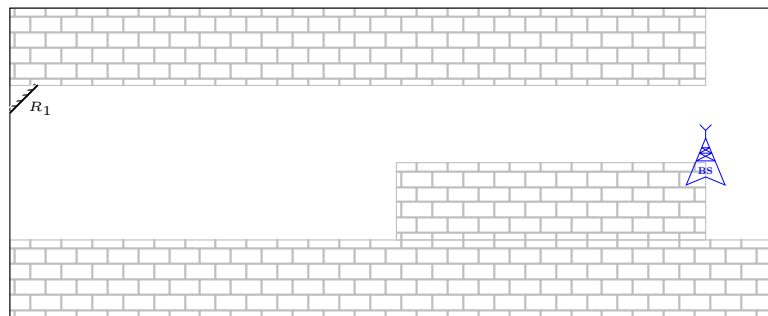


Figure 6.3: Optimal Placement Maximizing Coverage.

- Furthermore, we develop an LP relaxation based solution and show that it works better than the set cover based solution.
- We perform extensive simulations to demonstrate the superiority of our proposed joint solution over the existing two step approach.

## 6.2 System Model

We consider an urban deployment area and divide it up into grids of equal size as in [158], each grid being represented by its centre point. The resolution of the grids can be as small as possible, at the cost of high computational overhead. We further consider the following for modelling our system.

**Obstacles:** The spatial environment of this considered area is known beforehand. This data can be obtained by using OpenStreetMap [50] in the pre-processing stage. The environment consists of a set  $\mathcal{S}$  of static obstacles like buildings whose positions and all 3 dimensions (length, breadth, and height) are

known to a good accuracy. For tractability, we consider the buildings to be axes parallel cuboids.

**Base stations:** We consider small cell base stations, with effective transmission distance of upto  $d_{max}$ , beyond which the free space propagation is too large to provide an effective data rate. These base stations, equivalently called gNBs, can only be deployed on the rooftops of the buildings. The set of potential BS locations, i.e., the grids containing rooftop edges of the buildings are denoted by  $\mathcal{G}$ . We consider that there are  $N_G$  base stations to be deployed.

**Reflectors:** These reflecting devices can be of two types, namely IRSs [159], and PMRs [57]. IRSs are comparatively costlier, and can reflect radio waves towards a particular direction using a controllable phase shift. PMRs are basically metallic sheets and have the advantages of low cost, zero energy consumption (unlike IRSs), but can reflect incident signals towards only one fixed direction [cite twoStep]. Reflectors are allowed to be deployed on the building facades. The building facades are also discretized up into grids similar to the coverage area. The potential reflector locations on the building facades are denoted by  $\mathcal{R}$ . We consider that there are  $N_R$  reflectors to be deployed.

**Demand vector:** Since some of the grids are occupied by static obstacles, the rest form a set  $\mathcal{C}$  of serviceable grids. We are provided with a demand vector  $\mathcal{W} = \{W_1, W_2, \dots, W_{|\mathcal{C}|}\}$  which gives the normalised demand weights of each grid in  $\mathcal{C}$ . This data can be obtained by historical demand records available with service providers. This indirectly captures the effect of dynamic obstacles, since operators can provide appropriate weights depending on the congestion in the service area. For example, a busy crossing may have a high demand, while a secluded alley may have a low demand.

**LOS criteria:** We consider that data transmission between UE and BS (directly or via a reflector), takes place in the far-field region. The pathloss  $PL(a, b)$  between two grids  $a$  and  $b$  can be calculated using the 3GPP pathloss models [160]. If the the distance  $d_{a,b}$  between the grids is less than  $d_{max}$ , and subsequently the pathloss is lesser than a threshold, we assume that devices placed at these two



grids can communicate with each other. For each location  $i \in \mathcal{C}$ , we precompute the *LOS* sets as follows:

$$LOS(i) = \{j \mid d_{i,j} < d_{max} \text{ and } i\text{-}j \text{ have LOS}\}$$

**Reflection criteria:** The visibility of a potential **BS** location  $j$  with a grid location  $i$  via reflection off a reflector placed at  $k$  and oriented at  $\hat{o}$  can be computed by a function  $\mathcal{A}$  as follows, using ray tracing and Snell's Laws. Therefore,  $\mathcal{A}(i, j, k, \hat{o}) = 1$  if a reflector placed at a location  $k \in \mathcal{R}$  with orientation  $\hat{o} \in \mathcal{O}$  can reflect a signal coming from **BS** placed at  $j \in \mathcal{G}$  to a location  $i \in \mathcal{C}$ , otherwise  $\mathcal{A}(i, j, k, \hat{o}) = 0$ . Thus we define the **REFLECT** sets as follows:

$$\begin{aligned} \text{REFLECT}(i) = \{ & (j, k, \hat{o}) \mid k \in LOS(i), j \in LOS(k) \\ & \text{and } \mathcal{A}(i, j, k, \hat{o}) = 1 \} \end{aligned}$$

In case the reflector being considered is a IRS, adherence to Snell's Law is not required, since the programmable phase shifts of the reflecting elements can reflect a signal towards a desired direction using constructive and destructive interference. In such cases,  $\mathcal{A}(i, j, k, \hat{o})$  is always treated as 1, and the **REFLECT** sets are defined as follows:

$$\text{REFLECT}(i) = \{(j, k, \hat{o}) \mid k \in LOS(i), j \in LOS(k)\}$$

The notations used throughout this chapter are listed in Table 6.1.

## 6.3 Problem Formulation

To help us formulate the problem in an optimization framework, we introduce four sets of binary indicator variables  $C_i$ ,  $G_j$  and  $R_{k,\hat{o}}$  where  $i \in \mathcal{C}$ ,  $j \in \mathcal{G}$ ,  $k \in \mathcal{R}$ , and

Symbol	Interpretation
$\mathcal{S}$	Set of static obstacles
$\mathcal{G}$	Set of potential gNB locations
$\mathcal{R}$	Set of potential reflector locations
$\mathcal{C}$	Set of grids to be covered
$\mathcal{W}$	Demand vector
$d_{max}$	Maximum transmission distance
$N_G$	No. of gNB to be deployed
$N_R$	No. of reflectors to be deployed
$G_{pos}$	gNB deployment locations
$R_{pos}$	Reflector deployment locations

Table 6.1: Notation.

$\hat{o} \in \mathcal{O}$ , and define them as follows:

$$C_i = \begin{cases} 1 & \text{if } i\text{-th grid is covered} \\ 0 & \text{otherwise} \end{cases}$$

$$G_j = \begin{cases} 1 & \text{if } j\text{-th grid has a gNB} \\ 0 & \text{otherwise} \end{cases}$$

$$R_{k,\hat{o}} = \begin{cases} 1 & \text{if } k\text{-th grid has a reflector oriented at } \hat{o} \\ 0 & \text{otherwise} \end{cases}$$

We approximate a grid by its centre. As such, we assume a grid to be covered if its centre has an LOS and is closer than a threshold distance, from either a base station, or a reflector. The objective of the optimization is to obtain a maximum weighted coverage of the service area formulated as follows:

$$\text{maximize} \quad \sum_{i \in \mathcal{C}} C_i W_i \tag{6.3.1}$$

subject to the following constraints:

$$C_i \leq \sum_{j \in LOS(i)} G_j + \sum_{(j,k,\hat{o}) \in REFLECT(i)} G_j R_{k,\hat{o}} \quad \forall i \quad (6.3.2)$$

$$\sum_{j \in \mathcal{G}} G_j \leq N_G \quad (6.3.3)$$

$$\sum_k \sum_{\hat{o}} R_{k,\hat{o}} \leq N_R \quad (6.3.4)$$

$$\sum_{\hat{o}} R_{k,\hat{o}} \leq 1 \quad \forall k \quad (6.3.5)$$

$$C_i, G_j, R_{k,\hat{o}} \in \{0, 1\} \quad \forall i, j, k, \hat{o} \quad (6.3.6)$$

The objective function given in expression (6.3.1) tries to maximize the weighted sum of the covered area with the given number of BSs and reflectors. Constraint (6.3.2) forces all  $C_i$ s to 0 whenever  $i$ -th grid is not covered directly by any BS or indirectly via any reflector coupled with a BS. Constraints (6.3.3) and (6.3.4) state the upper bound as per the available BSs and reflectors. Constraint (6.3.5) ensures that a reflector having a unique orientation is associated with a single BS. Finally, the 0/1-integrality constraints are given in expression (6.3.6).

Recall that for three binary variables  $X, A, B$ , the expression  $X = AB$  can be linearized as  $X \leq A, X \leq B$  and  $X \geq A+B-1$ . Now notice that constraint (6.3.2) is nonlinear as it involves multiplication of two binary variables. This can be linearized by introducing a set of new binary variables  $X_{j,k,\hat{o}}$ , where  $X_{j,k,\hat{o}} = G_j R_{k,\hat{o}}$ . Now we can replace the constraint (6.3.2) with the following new constraints:

$$C_i \leq \sum_{j \in LOS(i)} G_j + \sum_{(j,k,\hat{o}) \in REFLECT(i)} X_{j,k,\hat{o}} \quad \forall i \quad (6.3.7)$$

$$X_{j,k,\hat{o}} \leq G_j \quad \forall j, k, \hat{o} \quad (6.3.8)$$

$$X_{j,k,\hat{o}} \leq R_{k,\hat{o}} \quad \forall j, k, \hat{o} \quad (6.3.9)$$

$$X_{j,k,\hat{o}} \geq G_j + R_{k,\hat{o}} - 1 \quad \forall j, k, \hat{o} \quad (6.3.10)$$

$$X_{j,k,\hat{o}} \in \{0, 1\} \quad \forall j, k, \hat{o} \quad (6.3.11)$$

Thus the final **ILP** is given by the objective function (6.3.1) and the constraints (6.3.3) through (6.3.11).

## 6.4 Joint Placement Algorithms

In this section, we present two greedy approaches for the considered joint placement problem. The first one is based on the classical set-cover problem, while the second one uses deterministic rounding with **LP** relaxation.

### 6.4.1 Joint Placement Using Set Cover

Recall that given a set  $S = \{S_1, S_2, \dots, S_n\}$  with each  $S_i$  being a subset of some universe  $U$ , for a given integer  $k$  the objective of the max-cover problem [161, 162] is to choose  $k$  sets from  $S$  such that their union is maximized. Now observe that, we can pose the joint placement problem as a set cover instance as follows. Let  $\mathcal{C}$  be the universe that we want to maximally cover. Each **BS** or a reflector associated with a **BS** has a fixed coverage area and thus forms the sets  $S_i$ . Now we apply an iterative greedy selection procedure, where in each iteration we select a **BS** (or a reflector) location that maximally covers the (remaining) area. Note that a new **BS** can only be placed if the number of placed **BSs** is less than  $N_G$ , while a reflector can only be placed if the number of placed reflectors is less than  $N_R$  and a **BS** is there for its association. After placing a **BS** (or a reflector) we update the remaining coverage area. We continue this process until we run out of resources, or the entire area is covered. This process is formalized into [Algorithm 6.1](#).

In [Algorithm 6.1](#),  $cov_j^G$  and  $cov_{k,\hat{\delta}}^R$  denote the set of locations covered by the **BS** placed at location  $j$ , and the reflector placed at location  $k$  with orientation  $\hat{\delta}$ , respectively. Here  $G_{pos}$  and  $R_{pos}$  are the set of selected **BS** locations and reflector positions/orientations respectively. Initially both  $G_{pos}$  and  $R_{pos}$  are empty sets and at each iteration of the while loop, either a **BS** or a reflector is selected, and the corresponding set is updated. We update the set of uncovered locations  $U$

**Algorithm 6.1:** Joint Placement Using Set Cover.

---

**Data:**  $\mathcal{C}$ , LOS, REFLECT,  $N_G$ ,  $N_R$   
**Result:**  $G_{\text{pos}}$ ,  $R_{\text{pos}}$

- 1 Initialize  $U \leftarrow \mathcal{C}$ ,  $\ell \leftarrow 0$ ,  $G_{\text{pos}} \leftarrow \emptyset$  and  $R_{\text{pos}} \leftarrow \emptyset$   
// Locations covered by each gNB/reflector
- 2  $\text{cov}_j^G = \{i \in \mathcal{C} \mid j \in \text{LOS}(i)\} \quad \forall j \in \mathcal{G}$
- 3  $\text{cov}_{k,\hat{\delta}}^R = \{i \in \mathcal{C} \mid \exists j, (j, k, \hat{\delta}) \in \text{REFLECT}(i)\} \quad \forall k, \hat{\delta}$
- 4 **while**  $\ell < N_G + N_R$  **do**
- 5      $j \leftarrow \underset{j'}{\text{argmax}}\{\text{cov}_{j'}^G \cap U\}$
- 6      $k, \hat{\delta} \leftarrow \underset{k', \hat{\delta}'}{\text{argmax}}\{\text{cov}_{k', \hat{\delta}'}^R \cap U\}$
- 7     **if**  $|\text{cov}_j^G \cap U| \geq |\text{cov}_{k, \hat{\delta}}^R \cap U|$  and gNBs already placed  $< N_G$  **then**
- 8         // Finalise a gNB at j
- 8          $G_{\text{pos}} \leftarrow G_{\text{pos}} \cup \{j\}$
- 9         Update  $U \leftarrow U \setminus \text{cov}_j^G$
- 10    **else**
- 11         // Finalise a reflector at k
- 11          $R_{\text{pos}} \leftarrow R_{\text{pos}} \cup \{(k, \hat{\delta})\}$
- 12         Update  $U \leftarrow U \setminus \text{cov}_{k, \hat{\delta}}^R$
- 13    **if**  $U = \emptyset$  **then**
- 14         **break**
- 15     $\ell \leftarrow \ell + 1$
- 16 **return**  $G_{\text{pos}}$  and  $R_{\text{pos}}$

---

accordingly. If we achieve complete coverage, i.e.,  $U = \emptyset$ , we terminate the loop; otherwise, the loop continues till the resources (BSs and reflectors) are exhausted.

With efficient hashing, the set intersection and set minus operations can be implemented efficiently that runs in linear time with respect to the cardinality of the sets. Here the size of  $\text{cov}$  sets can be at most  $|\mathcal{C}|$ . Therefore, each iteration requires at most  $O(|\mathcal{C}|(|\mathcal{G}| + |\mathcal{R}||\mathcal{O}|^2))$  time. Since we iterate at most  $(N_G + N_R)$  times the overall running time of Algorithm 6.1 is  $O(|\mathcal{C}|(|\mathcal{G}| + |\mathcal{R}||\mathcal{O}|^2)(N_G + N_R))$ . It is known that the greedy max-cover algorithm has a tight approximation ratio of  $1 - \frac{1}{e}$  [161, 162]. Since our Algorithm 6.1 follows the same strategy of greedy max-cover it must also provide the similar approximation guarantee. Thus, the locations covered by Algorithm 6.1 should be at least  $(1 - \frac{1}{e})OPT$ , where  $OPT$  is the optimum number of locations that could have been covered by an optimal algorithm.

## 6.4.2 Joint Placement Using LP Relaxation

Although the previous algorithm returns a solution with some guarantee, we present another solution to the same joint placement problem which we experimentally find to be better than the previous solution. More specifically, here we present an LP relaxation based joint solution for the considered base station and reflector placement problem. Note that while solving a ILP is computationally hard for large instances, the corresponding relaxed LP can be solved in polynomial time. Here, relaxed LP corresponding to a ILP implies that the integrality constraints (6.3.6) and (6.3.11) are relaxed so that the variable can take any real value in the closed interval  $[0, 1]$ . Now solving a relaxed LP in no way guarantees the variables to attain integral values which is required in an actual solution of the problem. One can round-off all the variables to closest integers but it may not satisfy all the constraints of the original ILP, making the solution invalid. Here we apply an iterative rounding-off scheme where in each iteration, we round-up a single variable either a  $G_j$  or a  $R_{k,\delta}$  which is closest to 1 maintaining feasibility, that is satisfying all the constraints. This essentially means that we fix placement of either a BS or a reflector in each iteration thus solving the placement problem jointly. Given  $G_j = 1$  (or  $R_{k,\delta} = 1$ ) as an additional constraint, we solve the relaxed LP once again and repeat the same process. Note that, here we are adding the constraint  $G_j = 1$  (or  $R_{k,\delta} = 1$ ) one variable at a time. Thus to avoid infeasibility, the only thing we need to take care of are constraints given in (6.3.3) and (6.3.4), the rest are automatically satisfied as per our design. Whenever a full coverage is attained, we may terminate the loop early. Otherwise we continue until the available BS and reflectors are exhausted. This procedure is formalized into Algorithm 6.2, which we briefly describe now.

Here  $G_{pos}$  and  $R_{pos}$  are the sets of BS locations and reflector positions/orientations respectively, as before. Let  $LP^{(\ell)}$  denotes the LP at iteration  $\ell$ . Initially  $LP^{(0)}$  is the relaxed version of the original ILP formulated in Section 6.3. At each iteration of the while loop, we solve  $LP^{(\ell)}$  and select either a BS or a reflector having the highest fractional value satisfying the constraints (6.3.3) and (6.3.4).

**Algorithm 6.2:** Joint Placement Using LP Relaxation.

---

**Data:**  $\mathcal{C}$ , LOS, REFLECT,  $N_G$ ,  $N_R$   
**Data:**  $G_{\text{pos}}$ ,  $R_{\text{pos}}$

- 1 Initialize  $i \leftarrow 0$ ,  $G_{\text{pos}} \leftarrow \emptyset$  and  $R_{\text{pos}} \leftarrow \emptyset$
- 2 Set  $\text{LP}^{(0)} \leftarrow$  relax the ILP given in Section 6.3
- 3 **while**  $i < N_G + N_R$  **do**
- 4     Solve  $\text{LP}^{(i)}$  optimally
- 5      $j \leftarrow \underset{j'}{\text{argmax}}\{G_{j'}\}$
- 6      $k, \hat{\delta} \leftarrow \underset{k', \hat{\delta}'}{\text{argmax}}\{R_{k', \hat{\delta}'}\}$
- 7     **if**  $G_j \geq R_{k, \hat{\delta}}$  and gNBs already placed  $< N_G$  **then**
- 8         // Finalise a gNB at  $j$   
 $G_{\text{pos}} \leftarrow G_{\text{pos}} \cup \{j\}$   
       // Add constraint  $G_j = 1$  in LP
- 9          $\text{LP}^{(i+1)} \leftarrow \text{LP}^{(i)} \cup \{G_j = 1\}$
- 10    **else**
- 11         // Finalise a reflector at  $k$   
 $R_{\text{pos}} \leftarrow R_{\text{pos}} \cup \{(k, \hat{\delta})\}$   
       // Add constraint  $R_k = 1$  in LP
- 12          $\text{LP}^{(i+1)} \leftarrow \text{LP}^{(i)} \cup \{R_{k, \hat{\delta}} = 1\}$
- 13    **if**  $\sum C_i = |\mathcal{C}|$  **then**
- 14         **break**
- 15     $i \leftarrow i + 1$
- 16 **return**  $G_{\text{pos}}$  and  $R_{\text{pos}}$

---

We update the corresponding set  $G_{\text{pos}}$  or  $R_{\text{pos}}$  accordingly, and a corresponding constraint added to the LP. If we achieve complete coverage, i.e.,  $\sum_i C_i$  becomes equal to the number of serviceable grids, we terminate the loop early; otherwise, the loop continues till the resources (BSs and reflectors) are exhausted.

Observe that in each iteration, we place either one BS or a reflector. Thus the loop is repeated at most  $(N_G + N_R)$  times. In each iteration, the dominant task is solving the relaxed LP. Here the relaxed LP can have at most  $n = |\mathcal{C}| + |\mathcal{G}| + |\mathcal{R}||\mathcal{O}|^2 + |\mathcal{G}||\mathcal{R}||\mathcal{O}|^2$  number of variables with  $O(n)$  many constraints and requires  $O(n^{3.5})$  time to solve using the Karmarkar's algorithm [163]. Thus overall running time of Algorithm 6.2 is  $(N_G + N_R)n^{3.5}$ .

## 6.5 Simulation Results

We consider a rectangular service area of size  $1000 \text{ m} \times 1000 \text{ m}$ , discretized into grids of size  $1 \text{ m}$ . There are some buildings inside the area, that have rectangular cross-sectional area, and rise vertically up. The percentage of area covered by these obstacles are varied from 5% to 20% in our simulations. The maximum height of the buildings where a **BS** can be placed is considered to be  $10 \text{ m}$ , and the building facades are also discretized into grids of size  $1 \text{ m}$ . The maximum transmission distance  $d_{max}$ , and UE height is taken to be  $100 \text{ m}$  and  $1.5 \text{ m}$  respectively. The maximum transmitted power at a **BS** is  $49 \text{ dBm}$ , and the antenna gains at a **BS** and **UE** are  $21.5 \text{ dBi}$  and  $5.5 \text{ dBi}$  respectively, as in [57]. For modelling the path loss, we follow the 3GPP [160] standard models for omni-directional Urban Micro **LOS** and **NLOS** path loss. The number of **BSs** and reflectors are varied in our experiments, and the coverage noted as performance parameter.

We refer to our joint placement solution based on Max-Cover given in Algorithm 6.1 as **JP-SC** and the one based on **LP** relaxation given in Algorithm 6.2 as **JP-LPR**. We denote the sequential deployment scheme as proposed in [57] as **Two Step** and compare our proposed two approaches with this. To check for optimality, we also solve the **ILP** for a small instance using Gurobi solver [164]. We denote the optimal solution as **Optimal**.

Our main comparison metric is the *coverage*, which we define as the percentage of the total number of grids that is covered by a given deployment scheme. We demonstrate the coverage performance of the aforesaid four approaches in Figure 6.4. Note that here we consider a smaller instance ( $N_G = 20$ ), as solving a **ILP** for larger instances is intractable. We see that our **JP-LPR** approach follows the optimal one closely, outperforms both the **JP-SC** and the **Two-Step** methods, with the **Two-Step** approach performing the worst. For the rest of the test cases with large instances, we do not solve the **ILP** but give a comparative study of the performance of the three other methods.



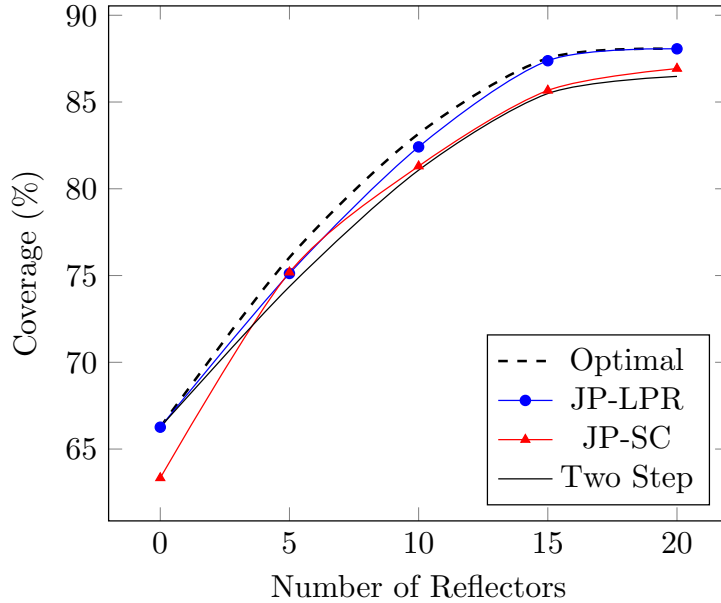


Figure 6.4: Effect of Reflector Count on Coverage for a Small Instance.

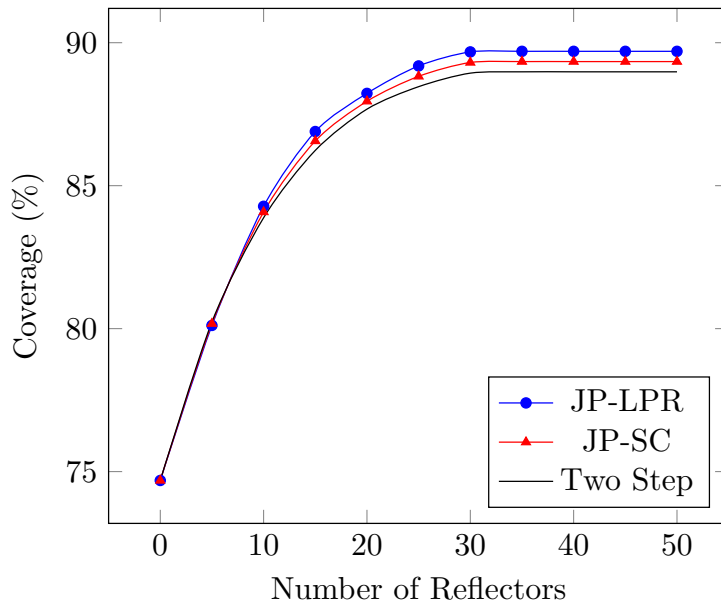


Figure 6.5: Effect of Reflector Count on Coverage, for  $N_G = 50$ .

We see that the trends in Figure 6.4 are maintained in Figures 6.5 and 6.6 as well. In Figure 6.5, we fix the number of BSs to 50, and vary the reflector count, whereas in Figure 6.6, we fix the available reflectors at 50 and vary the number of BSs. In both cases, the increase in coverage percentage with the number of BSs or reflectors saturates after a point. This is because no further serviceable area can be covered by placing any more resources. The percentage of obstacles present naturally plays a role in the coverage obtained for a given amount of

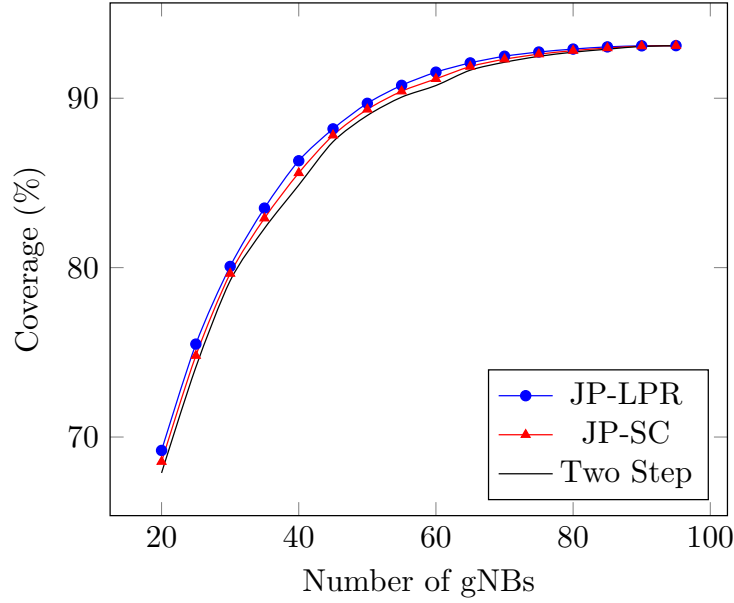


Figure 6.6: Effect of gNB Count on Coverage, for  $N_R = 50$ .

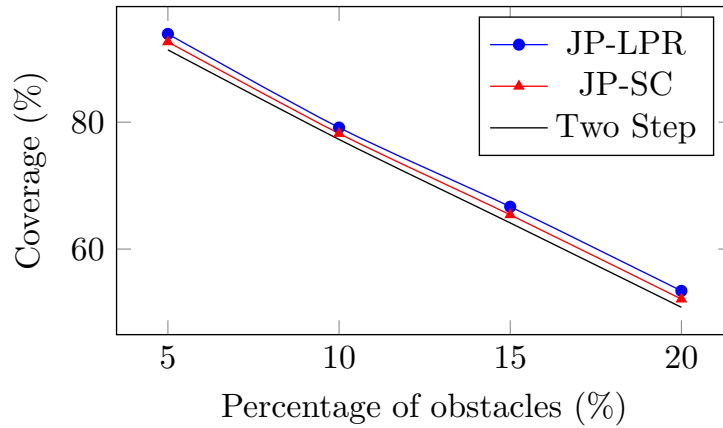


Figure 6.7: Effect of Obstacle Percentage on Coverage.

resources. We demonstrate this in Figure 6.7, where for a given 50 BSs and 50 reflectors, the percentage of grids achieving coverage decreases with increase in obstacle percentage, as expected. The relative performance among JP-LPR, JP-SC and Two-Step is also preserved here. The discretization of the service area introduces inherent inaccuracies in the results. We show the effect of the grid resolution (in meters) on the inaccuracy. We define *inaccuracy* percentage as the relative number of grids incorrectly considered to be covered as compared to the total number of grids. The plot in Fig. 6.8 shows that as the grid resolution is made larger, more and more inaccuracy creeps in. It is pertinent to mention here that a smaller grid size leads to heavier computational overhead, and a balance

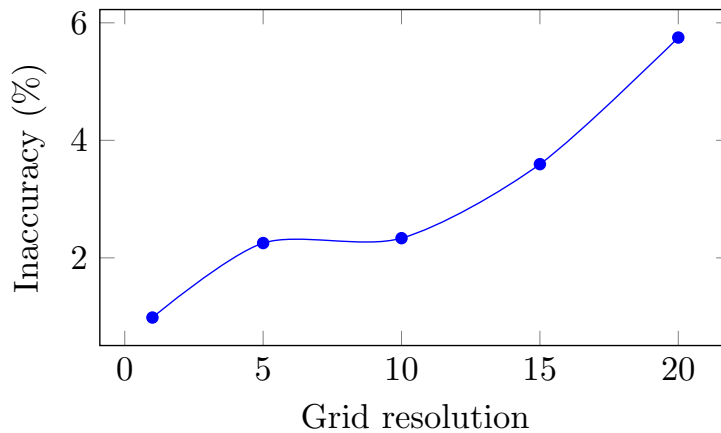


Figure 6.8: Inaccuracy Due to Grid Resolution.

usually has to be found. To analyse the effect of reflector size on the deployment strategies, we run the simulations for a  $N_G = 30$ ,  $N_R = 10$  and varying the reflector sizes as  $1 \text{ m} \times 1 \text{ m}$ ,  $3 \text{ m} \times 3 \text{ m}$ , and  $5 \text{ m} \times 5 \text{ m}$ . The results are plotted in Fig. 6.9. As is obvious, larger reflector size achieves greater coverage as compared to smaller ones, however the effect saturates at  $3 \text{ m} \times 3 \text{ m}$  only.

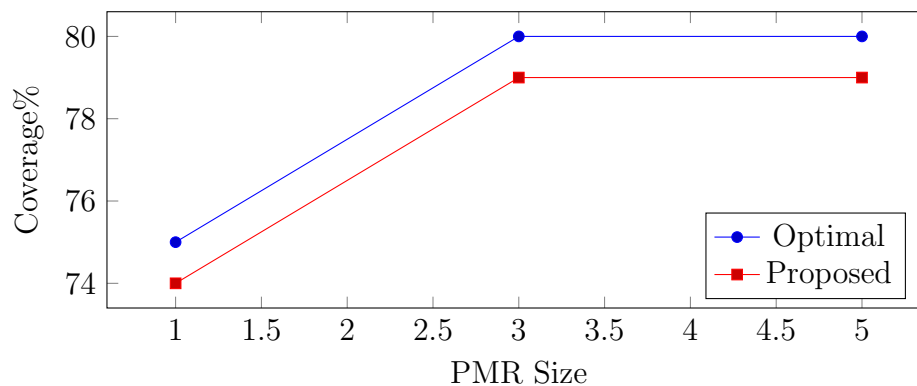


Figure 6.9: Effect of PMR Size on Coverage.

## 6.6 Conclusion

In this chapter, we jointly deal with the problem of **BS** and reflector placement in an outdoor **mmWave** network. We model the problem as a **ILP**, which is hard to solve. We then provide a Max-Cover based greedy solution, and an **LP** relaxation based solution for the same. We demonstrate via simulation that both approaches provide larger coverage than a sequential approach.



# Chapter 7

## LazyUAV: A Minimal Displacement Coverage Strategy for Multi-UAV mmWave Networks<sup>1</sup>

### 7.1 Overview

Recently, there has been growing interest in deploying BSs on board UAVs [138] to facilitate better network performance. The spatial flexibility of UAVs provides crucial utility for two important scenarios; when terrestrial BSs are saturated and require traffic offloading [62, 63], and when the same are destroyed due to disasters like natural calamities [64, 65]. However, the downside of UAVs is their limited onboard power supply, which is used to hover, shift, and transmit, altogether a daunting task [165]. Increasing the battery capacity increases the payload weight, which in turn increases energy demand, decreasing flight time;

---

<sup>1</sup>This chapter is based on the following paper:  
**Subhojit Sarkar**, Rathindra Nath Dutta and Sasthi C. Ghosh. “LazyUAV: A Minimal Displacement Coverage Strategy for Multi-UAV mmWave Networks” In: *3rd International Mediterranean Conference on Communications and Networking (MeditCom 2023)*, IEEE, Dubrovnik, Croatia, September 4-7 2023 (to appear)

hence, it is inadvisable to keep on increasing battery capacity. Harnessing other techniques like trajectory optimization would be a better approach. When it comes to the **mmWave** realm, the stringent propagation requirements pose yet another challenge. Consequently, **mmWave BS** deployment on **UAVs** is an exciting area of current research [33, 139]. The mobility of **UEs** plays a key role in how long an allotted link can remain active, without the **UAV** itself having to move; the poor propagation characteristics of **mmWaves** necessitate this. In this chapter, we consider the problem of deploying multiple **UAVs** with an aim to minimising their displacements in subsequent time. This is done to ensure that minimum energy is utilized in displacing a **UAV** from one point to another to ensure coverage. We take into consideration user mobility, and propose **LazyUAV**, a Set-Cover based geometric approach to minimise **UAV** displacement, while maintaining maximal coverage. The main contributions of this chapter are summarised as follows:

- Although the **UAV** placement problem is NP-Complete [70] in general, we show that the optimal placement of a single **UAV** for a static scenario can be efficiently solved in polynomial time using a geometric approach.
- We then proceed to solve the multi **UAV** version, which is NP-hard, using a Max-Cover based greedy approximation scheme.
- Taking into consideration **UE** mobility, we devise a polynomial time geometric algorithm that efficiently finds optimal placement of a single **UAV** ensuring maximum coverage, with minimum displacement.
- We solve the multi- **UAV** deployment problem, ensuring minimum displacement while maintaining maximal coverage by a greedy max-cover based approach.
- Via extensive simulation, we show that our proposed approach achieves greater coverage with lower **UAV** mobility as compared to two baseline approaches.

## 7.2 System Model

Let us consider a service area under a central **BS**. There is a set of  $N$  terrestrial **UEs** inside, and over a short time interval  $\Delta t$ , they move along a straight line with constant velocity.  $\Delta t$  depends on the traffic mobility model, and the available routes in the coverage area. For example, if it is a highway with mostly straight roads,  $\delta t$  can be quite large, as there is low chance of users abruptly changing direction. In an urban environment, there are frequent intersections, which allows users to change directions frequently. The value of  $\Delta t$  can thus be chosen depending on the deployment scenario. Each **UE** communicates its position and velocity to the central **BS** as part of the location update process. The high bandwidth demand of the **UE** is served by a set of  $K$  **mmWave** enabled **UAVs**, whose heights are fixed at  $H$ . A **UAV** can serve a **UE** if the transmission distance is within a threshold,  $d_{max}$ . **UAVs** can usually change their azimuth angle to achieve better beamforming; however to keep things simple in this chapter, we consider the azimuth angle of each antenna to be  $0^\circ$ , i.e., the coverage area of each **UAV** is circular. Due to the fixed height and transmit power considered, the 2-D projection of the coverage area of an individual **UAV** becomes a circle with centre at the location of the **UAV**, and having fixed radius  $r$ , where  $r = \sqrt{d_{max}^2 - H^2}$ . The **UAVs** can travel in straight lines towards their destination. This assumption holds true since with the absence of very high altitude buildings, there are effectively no restricted zones for the **UAVs**. We assume that there are enough frequency channels available so that nearby links do not interfere with each other. A **UE** is said to be *covered* by a **UAV**, if it lies within a circle of radius  $r$  centered around that **UAV**. Note that **mmWave** links can fail both due to distance, as well as obstacles. However, in this paper we focus only on the distance criterion, leaving the problem of handling obstacles for possible future work.

## 7.3 Proposed Approach: LazyUAV

Our proposed LazyUAV solution is built up in four stages, where we first discuss the optimal placement of a single UAV for a static scenario in Section 7.3.1, which is then extended to placing multiple UAVs in Section 7.3.2. We then consider mobility of the UEs in Section 7.3.3, where a single UAV is placed and displaced optimally. Finally, we generalize our solution for placing of multiple UAVs for mobile UEs in Section 7.3.4.

### 7.3.1 Covering Static UEs with a Single UAV

Initially we consider that each UE has a fixed location, and we have a single UAV that needs to be placed optimally. This can be thought of as *covering* maximum number of points using a disk of radius  $r$ . A point is said to be covered by a disk of radius  $r$ , if and only if the point lies on the periphery, or inside the disk. In other words, a point is covered only if its distance from the center of a disk is lesser than or equal to  $r$ . From here onward, we will use the words ‘UAV’ and ‘disk’, interchangeably as convenient, and they both have the same coverage radius  $r$ . Moreover, we also interchangeably use the terms ‘UE’ and ‘point’ as per convenience. We can now make the following observation, which then leads to Lemma 7.1.

**Observation 2.** *If a set of points  $P$  can be covered by a disk of radius  $r$ , there might be infinitely many placements of the disk as depicted in Figure 7.1. However, note that all of them can be considered equivalent in terms of covering  $P$ . Indeed, the infinite number of possible disk locations that cover  $P$  are all equivalent with respect to their coverage. Furthermore, consider an arbitrary placement of the disk covering all points in  $P$ . One can always suitably displace this disk, such that at least two points in  $P$  end up on its periphery. This new placement is equivalent to the original one with respect to the coverage.*

**Lemma 7.1.** *Given  $N$  points, there are at most  $2\binom{N}{2}$  distinct maximal covers using a disk of radius  $r$ .*



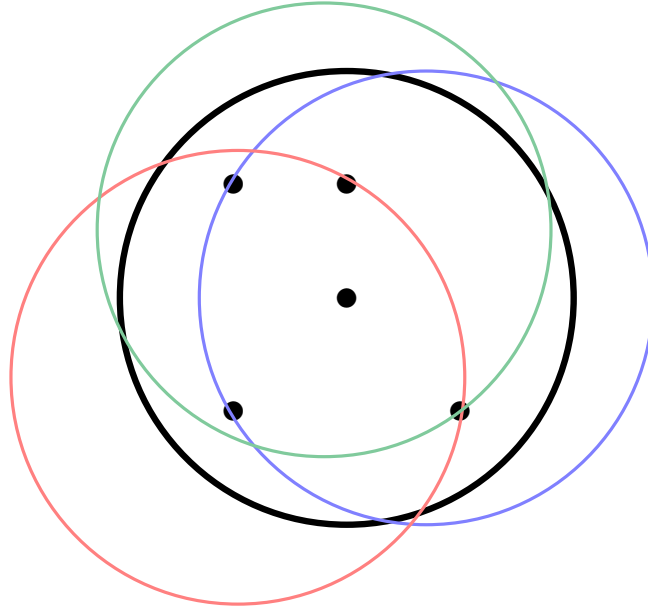


Figure 7.1: A Few of the Possible Covers for a Set of Points.

*Proof.* Assume on the contrary, that there are more than  $2\binom{N}{2}$  distinct maximal covers. Now by above observation, we can place a disk for each cover such that it touches at least two of the covered points. Now using pigeonhole principle, there must be at least a pair of points for which there exists more than two distinct covers. This is a contradiction, since there exists at most two circles of fixed radius  $r$ , which passes through two arbitrarily given points. Hence the proof.  $\square$

One can evaluate each of these  $O(N^2)$  distinct covers, and take the one having maximum coverage which will result into a  $O(N^3)$  solution. Instead, an optimal placement of the UAV can be efficiently obtained in  $O(N^2 \log N)$  time using an angular-sweep method, where  $N$  is the total number of points (UEs) considered. The mechanism of angular-sweep is now briefly described for completeness. Note that, if two points are more than  $2r$  distance apart, both of them cannot be covered by a single disk of radius  $r$ . Let us fix a point  $p$ , and for all the remaining  $N - 1$  points  $q$ , which are within  $2r$  distance from  $p$ , we compute the angle  $\alpha_{p,q}$ . Here  $\alpha_{p,q}$  is the angle of  $q$  about  $p$  with respect to the positive  $x$ -axis as shown in Figure 7.2. We sort the points based on  $\alpha_{p,q}$ , which takes  $O(N \log N)$  time.

We then place the periphery of the disk on  $p$ , and rotate the disk about  $p$  and observe the covered set of points. Here we consider only discrete rotations of the

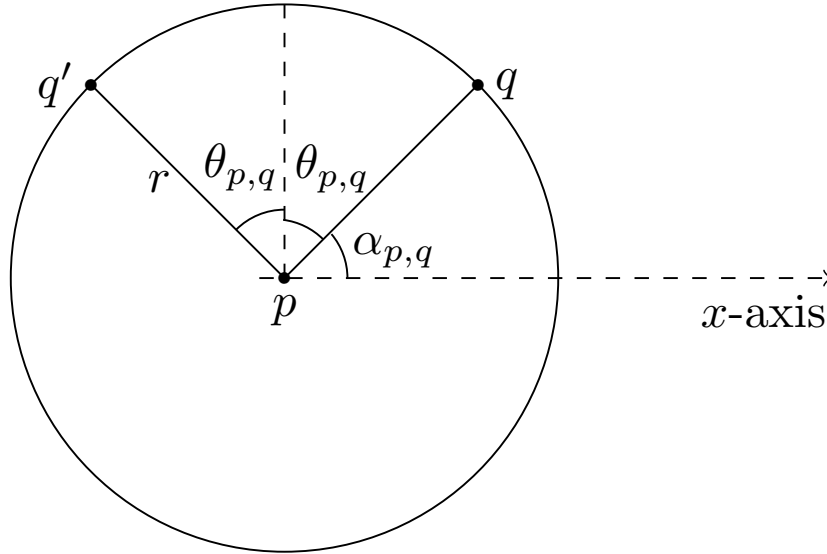


Figure 7.2: The Replica  $q'$  of a Point  $q$  About the Point  $p$ .

circle, such that each rotation denotes an event: either a new point enters into the circle or an already covered point leaves the circle. Although, at each orientation of the circle, calculating the coverage from scratch requires  $O(N)$  time; this can be amortized into  $O(1)$  when we consider a complete  $360^\circ$  rotation of the circle. For implementing this efficiently and get the amortized cost of  $O(1)$  per rotation, we replicate each of those  $N - 1$  points. For a point  $q$ , its *replica* is rotated about  $p$  with an angle  $2\theta_{p,q}$  where  $\theta_{p,q} = 90^\circ - \alpha_{p,q}$  as depicted in Figure 7.2. Furthermore, we tag the original points as ‘start’ events, while the replicas are tagged as ‘end’ events. Now we sort these  $2(N - 1)$  points, and then process them linearly as per the sorted order. If we keep track of number of covered points for each rotation, we can report in  $O(N)$  time, when it had covered maximum number of points among all rotations. Thus the overall time required is still  $O(N \log N)$ . Now repeating this angular-sweep process for each of the  $N$  points, and taking the overall maximum actually gives us the required cover. Thus total time complexity is  $O(N^2 \log N)$ .

### 7.3.2 Multiple Static UAV Placement using Set-Cover

Since each UAV has same coverage radius  $r$  (can be treated as unit), the problem reduces to a Unit-Disk-Cover problem [166], which is basically the geometric version of the classical Set-Cover problem. In particular, since  $K$ , the number of UAVs, is a constant, we are interested in the Max-Cover problem where given an integer  $K$  and a collection of sets  $S_i \subseteq \mathcal{U}$ , we want to maximally cover the elements (UEs) from the universe  $\mathcal{U}$ . Given a set of possible UAV placements, we can obtain the UEs covered by each such placement, which gives us the sets  $S_i$ . Here, we apply a simple greedy strategy [162] as explained below. We pick a UAV position that covers maximum number of uncovered UEs. Removing these covered UEs from the universe, we iteratively run the process again, till all UEs are covered, or all UAVs are used up. This results into an approximation ratio of  $1 - \frac{1}{e}$  [162].

Now the main hurdle is constructing a finite number of such candidate sets  $S_i$ . On the surface, it may seem like a impossible task, as there can be infinitely many placement of a unit-disk each of which covers same set of points as depicted in Figure 7.1. Fortunately by establishing an equivalence of all such covers, we get only a finitely many candidate sets by Lemma 7.1. Now given  $K$ , we can readily apply the  $(1 - \frac{1}{e})$ -approximation algorithm for the Max-Cover problem. Here the number of candidate sets is  $O(N^2)$ , and to check for coverage of an individual disk, and update the residual coverage in each iteration of the Max-Cover solution, we get a worse case complexity of  $O(N^3 f)$ . Here  $f$  is the maximum possible number of sets that can contain a point.

### 7.3.3 Covering Mobile UEs with a Single UAV

Since the UEs are assumed to move independently of each other, there is no certainty that a set of UEs covered by a single UAV at time  $t$ , can still be covered by the same UAV at time  $t + \Delta t$ . In this case, we would prefer to maintain a maximum possible coverage, by allowing the UAV to change its position. Here

our objective is to move the UAV in such a way, that its displacement from time  $t$  to time  $t + \Delta t$  is also minimized. Let us first focus only on the coverage at time points  $t$  and  $t + \Delta t$ . Note that, solving the UAV placement problem independently for the two time slots, even using the approach given in Section 7.3.1, may produce two entirely non-overlapping coverage solutions. Additionally, it does not provide any reasonable bound on the UAV displacement. We therefore must consider the UAV placement at two time points jointly. Suppose a set of points  $P$  is covered by a UAV at time  $t$ , now at time  $t + \Delta t$  they move to some new positions  $P'$ . We can apply the same angular-sweep method to obtain a maximum coverage of  $P'$  with a disk of radius  $r$  in  $O(m^2 \log m)$  time, where  $m = |P| = |P'|$ . Suppose only  $m' \leq m$  points can be covered at time  $t + \Delta t$ . We will use this  $m'$  as the selection metric. Therefore while doing the angular-sweep at time  $t$  we select the placement having maximum  $m'$ . If there is a tie, we consider the one requiring minimum displacement of the UAV. The optimality of this selection process guaranteed by the following lemma.

**Lemma 7.2.** *Maximum number of mobile UEs covered by a single UAV, can be found in polynomial time.*

*Proof.* As discussed earlier, there are only  $O(N^2)$  many possible coverings at time  $t$ . All the corresponding coverage sets  $S_i$  can be obtained in  $O(N^3)$  time. For each  $S_i$ , let  $S'_i \subseteq S_i$  be the maximum set of UEs that can still be covered at time  $t + \Delta t$ . Let  $m_i = |S_i|$  and  $m'_i = |S'_i|$ , clearly we have  $m'_i \leq m_i$ . Using the angular-sweep method discussed earlier, each  $S'_i$  can be obtained in  $O(m_i^2 \log m_i)$ , making the total required time  $O(\sum_i m_i^2 \log m_i) = O(N^2 f^2 \log(Nf))$ . Finally we take the coverage set  $S_{i^*}$  where  $i^* = \operatorname{argmax}\{m'_i\}$ . This takes at most  $O(N^2)$  time. Now by construction,  $S_{i^*}$  denotes the maximum possible cover by a single UAV considering two time points  $t$  and  $t + \Delta t$ .  $\square$

Now recall that, for the optimal coverage set  $S_{i^*}$ , a single disk may have infinitely many placements for each time point  $t$  and  $t + \Delta t$ . We now place the disk at  $t$  and  $t + \Delta t$  in such a way that their displacement is minimized. Recall that we

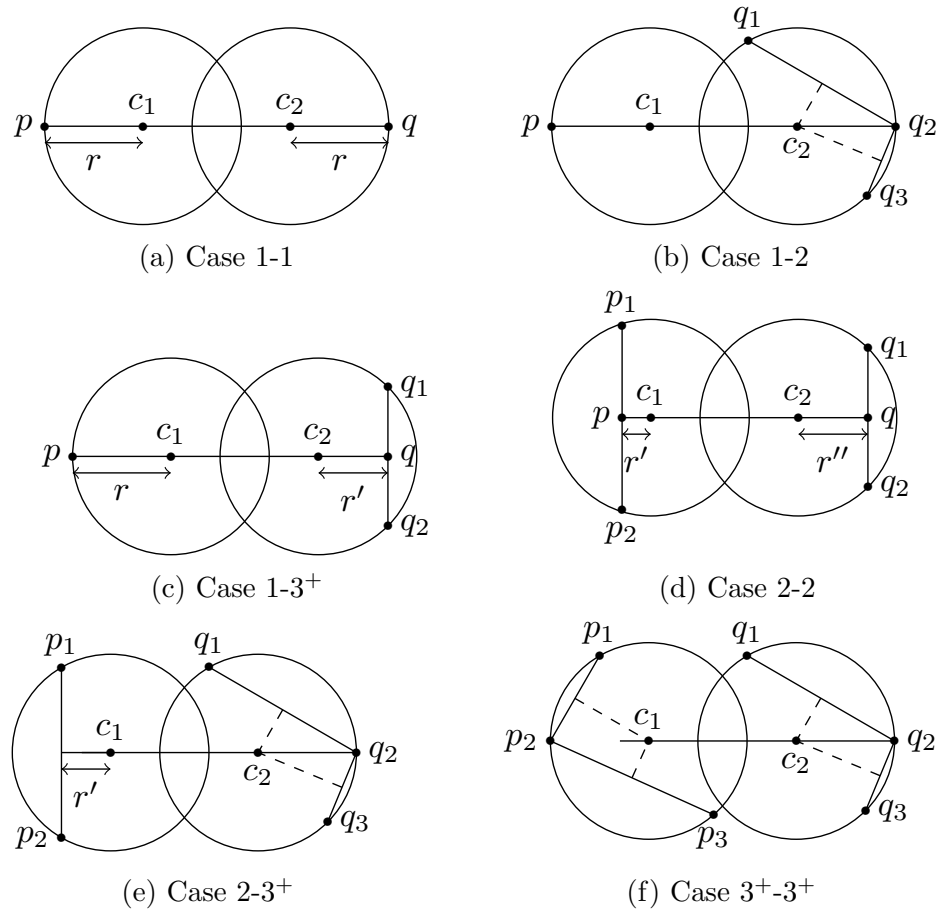


Figure 7.3: Possible Cases for Optimal UAV Displacement.

can only cover  $S'_{i^*} \subseteq S_{i^*}$  at time  $t + \Delta t$ , thus we focus only on the **UEs** in  $S'_{i^*}$ . Let  $P$  be the set of coordinates of the **UEs** in  $S'_{i^*}$  at time  $t$ . Now they move to a new place at time  $t + \Delta t$ , and let  $P'$  be set of new coordinates. By construction, both  $P$  and  $P'$  can independently be covered by a disk of radius  $r$ . Consider a 2D plane containing the points from both  $P$  and  $P'$ . Now suppose we place a circle  $C_1$  of radius  $r$  around  $P$  at any arbitrary position, such that all points in  $P$  is covered. Similarly another circle  $C_2$  is placed around  $P'$ . For our minimum **UAV** displacement, the center of these two circles  $C_1$  and  $C_2$  must be as close as possible. In optimal placement of  $C_1$  and  $C_2$  one of the following cases must occur:

**Case 1-1 :** Let  $C_1$  touches a only single point  $p \in P$  and  $C_2$  also touches a single point  $q \in P'$ . In this case the both the centres of  $C_1$  and  $C_2$  must lie on the line joining  $p$  and  $q$  as depicted in **Figure 7.3(a)**. Let  $(x_p, y_p)$  and  $(x_q, y_q)$  be the coordinates of points  $p$  and  $q$  respectively; then centers of  $C_1$  and  $C_2$

will respectively be at  $(x_p + \Delta x, y_p + \Delta y)$  and  $(x_q - \Delta x, y_q - \Delta y)$ , where  $\Delta x = r(x_q - x_p)/dist(p, q)$  and  $\Delta y = r(y_q - y_p)/dist(p, q)$ . Here  $dist(p, q)$  is the Euclidean distance between the points  $p$  and  $q$ .

**Case 1-2 :** Let  $C_1$  touches only a single point  $p \in P$ , while  $C_2$  touches two points  $q_1, q_2 \in P'$ . Let  $q$  be the mid point of  $q_1$  and  $q_2$ . In this case the two centres lie on the line joining  $p$  and  $q$  as depicted in [Figure 7.3\(b\)](#). Moreover, the line  $\overline{pq}$  is the perpendicular bisector of  $\overline{q_1q_2}$ . The centre of  $C_1$  can be determined similarly as above, while the centre of  $C_2$  is given by  $(x_q - r'(x_q - x_p)/dist(p, q), y_q - r'(y_q - y_p)/dist(p, q))$ , where  $r' = \sqrt{r^2 - dist(q_1, q_2)^2}$ .

**Case 1-3<sup>+</sup> :** Let  $C_1$  touches only a single point  $p \in P$ , while  $C_2$  touches three or more points from  $P'$ . In this case there is only one circle that covers  $P'$ , and its centre is uniquely determined by the (at least) three points it touches. Let  $c_2$  be the centre of circle  $C_2$ . Now the centre of  $C_1$  must lie on the line joining  $p$  and  $c_2$ , and be obtained similarly as above. This is depicted in [Figure 7.3\(c\)](#).

**Case 2-2 :** Let  $C_1$  touches two points  $p_1, p_2 \in P$  and  $C_2$  also touches two points  $q_1, q_2 \in P'$ . Let  $p$  and  $q$  be the mid points of the line segments  $\overline{p_1p_2}$  and  $\overline{q_1q_2}$  respectively. Then the centres of  $C_1$  and  $C_2$  must lie on the line  $\overline{pq}$ . Moreover,  $\overline{pq}$  is perpendicular bisector of both  $\overline{p_1p_2}$  and  $\overline{q_1q_2}$ . This scenario is depicted in [Figure 7.3\(d\)](#). The centres of  $C_1$  and  $C_2$  can be obtained similarly as discussed above.

**Case 2-3<sup>+</sup> :** Let  $C_1$  touches two points  $p_1, p_2 \in P$ , while  $C_2$  touches three or more points from  $P'$ . Here also the centre  $c_2$  of circle  $C_2$  is uniquely determined. Now the center of  $C_1$  must lie of the line  $\overline{pc_2}$ , where  $p$  is the mid point of of the line segment  $\overline{p_1p_2}$ , and can be similarly computed. This case is depicted in [Figure 7.3\(e\)](#).

**Case 3<sup>+</sup>-3<sup>+</sup> :** Let both  $C_1$  and  $C_2$  touches three or more points from  $P$  and  $P'$  respectively. Here the both circles are uniquely determined by its three peripheral points as depicted in [Figure 7.3\(f\)](#).

All the remaining cases are basically symmetrical to these listed ones, and thus can be handled in a similar fashion.

Now to obtain the optimal placement of the disk covering  $P$  at time  $t$  and  $P'$  at time  $t + \Delta t$  with minimum displacement, we can exhaustively check for the above cases and take the one giving optimum displacement. To do this efficiently, we first make the following observations.

**Observation 3.** *A set of points  $P$  is covered by a disk of radius  $r$ , only if the hull points of  $P$  are covered by the disk.*

**Observation 4.** *If a disk covering  $P$  touches two or more points from  $P$ , the points must be adjacent hull points.*

Let the convex hulls of the point sets  $P$  and  $P'$  be  $CH_P$  and  $CH_{P'}$  respectively. Therefore, we can only focus on these hull points and try out all possible combinations of them as per the aforementioned cases. For each hull point in  $CH_P$  (or  $CH_{P'}$ ) we only observe a finite number of combinations, which takes constant  $O(1)$  time. Recall that, the convex hull of  $n$  points can be obtained in  $O(n \log h)$  time, where  $h$  is the number of hull points [167]. Thus the total time to exhaustively search for an optimal placement of each UAV considering both time points, is upper bounded by  $O(m_{i^*} \log h) = O(N \log N)$ .

Notice that we have solved the problem of placing a single UAV covering mobile UEs. For this we considered the coverage at two discrete time points  $t$  and  $t + \Delta t$ , but not in between. In the following lemma we guarantee that our discrete coverage solution ensures a continuous coverage.

**Lemma 7.3.** *For a UE and a UAV moving along two arbitrary straight lines, if the UAV covers the UE at two time instances  $t_1$  and  $t_2$ , it covers the UE for all  $t$  in the interval  $[t_1, t_2]$ .*

*Proof.* Let the positions of the UE at times  $t_1$  and  $t_2$  be  $\hat{\mathbf{u}}_1$  and  $\hat{\mathbf{u}}_2$  respectively. Similarly, the positions of the UAV at times  $t_1$  and  $t_2$  are  $\hat{\mathbf{v}}_1$  and  $\hat{\mathbf{v}}_2$  respectively. Note that since the UE moves along a straight line, all its positions within  $(t_1, t_2)$

can be effectively written as  $\hat{\mathbf{u}} = \alpha\hat{\mathbf{u}}_1 + (1 - \alpha)\hat{\mathbf{u}}_2$ , where  $0 < \alpha < 1$ . Similarly, all positions of the UAV in  $(t_1, t_2)$  can be effectively written as  $\hat{\mathbf{v}} = \beta\hat{\mathbf{v}}_1 + (1 - \beta)\hat{\mathbf{v}}_2$ , where  $0 < \beta < 1$ . Now, since the time period of travel for both devices is the same, it follows trivially that  $\alpha = \beta$ . Also, we are given that the UAV covers the UE at both  $t_1$  and  $t_2$ , i.e.,  $\|\hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1\| \leq r$ , and  $\|\hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2\| \leq r$ , where  $\|\cdot\|$  denotes the Euclidean ( $l_2$ ) norm. Hence, it suffices to prove that  $\|\hat{\mathbf{u}} - \hat{\mathbf{v}}\| \leq r$ . Now,

$$\begin{aligned} \|\hat{\mathbf{u}} - \hat{\mathbf{v}}\|^2 &= \|\alpha(\hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1) + (1 - \alpha)(\hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2)\|^2 \\ &= \alpha^2\|\hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1\|^2 + (1 - \alpha)^2\|\hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2\|^2 \\ &\quad + 2\alpha(1 - \alpha)|\langle \hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1, \hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2 \rangle| \end{aligned}$$

Here,  $\langle \cdot, \cdot \rangle$  denotes the inner product. Using the Cauchy-Schwarz inequality we have,

$$|\langle \hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1, \hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2 \rangle| \leq \|\hat{\mathbf{u}}_1 - \hat{\mathbf{v}}_1\| \|\hat{\mathbf{u}}_2 - \hat{\mathbf{v}}_2\| \leq r^2.$$

Plugging this into the above inequality we have:

$$\|\hat{\mathbf{u}} - \hat{\mathbf{v}}\|^2 \leq \alpha^2 r^2 + (1 - \alpha)^2 r^2 + 2\alpha(1 - \alpha)r^2 = r^2. \quad \square$$

### 7.3.4 Multiple Mobile UAV Placement using Set-Cover

Similar as before, we consider each candidate coverage set  $S_i$  at time  $t$  and construct the maximal coverage  $S'_i$  for time  $t + \Delta t$  using the angular-sweep method as discussed in Section 7.3.3. Now following the method given in Section 7.3.2, we similarly obtain a Max-Cover solution using these  $S'_i$  as the candidate sets. Let  $\mathcal{S}$  be the set of coverage sets obtained through the Max-Cover algorithm. For each  $S'_i \in \mathcal{S}$ , we independently fix the UAV positions considering minimum displacement. Thus sum of the UAV displacement is minimized. Now by Lemma 7.3, each UAV maintains its assigned cover within the  $\Delta t$  interval. Thus using  $K$  UAVs, we maximize the number of UEs that are continuously covered within the  $\Delta t$  interval, with minimum UAV displacement. Note here that for a small  $\Delta t$ , the UAV displacement will obviously be small, but this entire operation of UAV



assignment has to be done frequently. For larger  $\Delta t$ , UAV displacement will be higher, but the algorithm has to be run less frequently.

## 7.4 Simulation Experiments

We consider a square service area of size  $1000 \text{ m} \times 1000 \text{ m}$ . There are 100 users distributed uniformly at random inside this area. Each of them travels in a straight line for a time period 10s with a velocity chosen uniformly at random from  $[0, 10 \text{ m/s}]$ . We compare our proposed LazyUAV algorithm against two baseline approaches, the first one being a Greedy approach where we try to cover the UEs maximally with the available UAVs without considering the mobility of the UEs. After the covers are decided, we observe the number of UEs are fully covered considering their mobility, and take that as the measured value. To implement this Greedy approach we utilize the scheme proposed in Section 7.3.2. Another is a deployment strategy based on line segment disk cover (LSDC) algorithm proposed in [168]. In LSDC, we aim to maximally cover the mobile UEs with zero displacement. In other words, both the end points of each covered mobile UE, are covered by a single static UAV. We consider two metrics, namely the number of mobile UEs covered and the average displacement of the UAVs, to test the performance of these three strategies.

In Figure 7.4, we show the effect of varying the number of UAVs on coverage. With increase in number of UAVs, the coverage starts to increase; however, LazyUAV outperforms the other two significantly. Note that, due to the lack of insight into the device mobility, the coverage of Greedy approach gets stuck around 30 after a point, and does not increase any more with  $K$ . The effect of the coverage radius  $r$  of each UAV on the number of UEs covered, is shown in Figure 7.5. As evident, the UE coverage increases with  $r$  increases, where LazyUAV performs the best.

Next we consider the displacement of the UAVs for each of the three approaches in Figures 7.6 and 7.7. Observe the in both these figures, the LSDC approach has

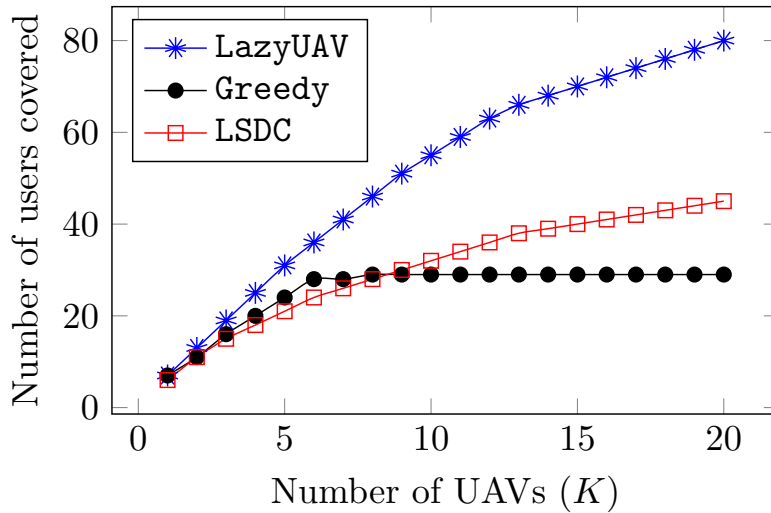


Figure 7.4: Effect of  $K$  on Coverage.

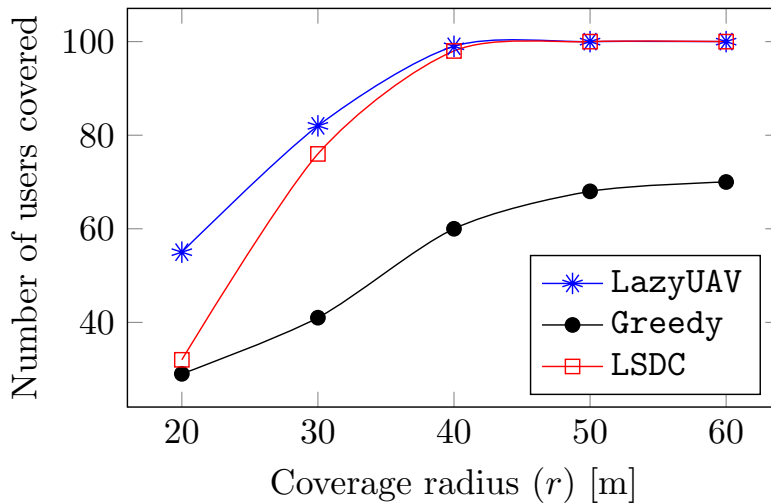
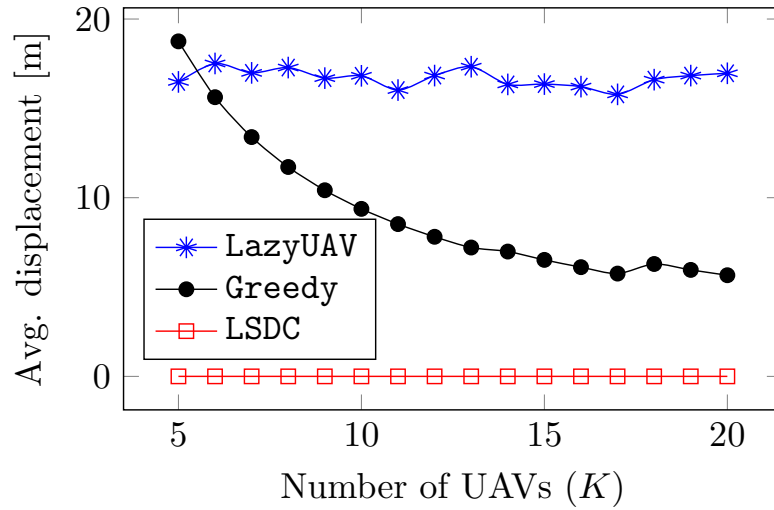
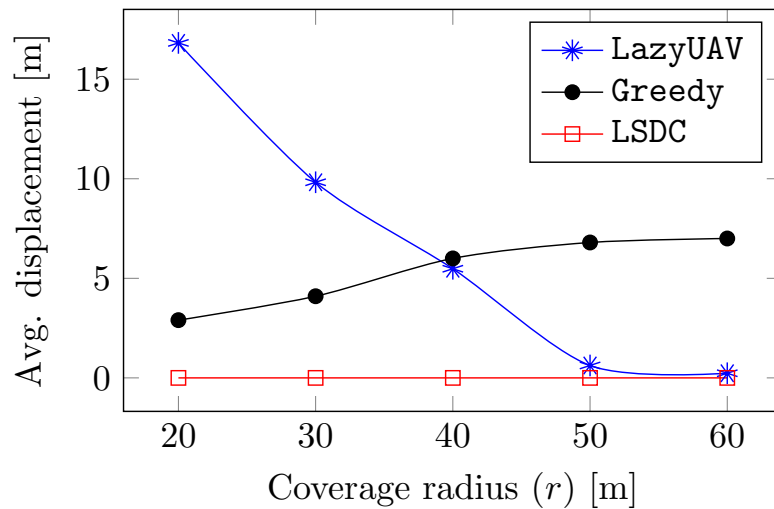


Figure 7.5: Effect of  $r$  on Coverage.

zero average displacement as one would expect. In Figure 7.6, it is evident that our LazyUAV approach has almost same average displacement while  $K$  is increased, while the displacement in Greedy approach decreases with  $K$ . This is because, Greedy only covers up to 30% of the UEs which leaves plenty of room for it to minimize the UAV displacements. Whereas, LazyUAV sacrifices in terms of UAV displacements to maximize its coverage. On the other hand in Figure 7.7, the displacement decreases for LazyUAV as  $r$  increases as expected, while for Greedy the displacement slightly increases with  $r$ . The reason for displacement increase for Greedy, is due its gradual increase in coverage with  $r$ .

Figure 7.6: Effect of  $K$  on Average Displacement.Figure 7.7: Effect of  $r$  on Average Displacement.

## 7.5 Conclusion

In this work, we considered minimising the displacements of UAVs while ensuring maximal coverage. We modelled the problem of deploying UAVs as a variant of unit disk cover problem, and used a greedy Max-Cover based geometric approach to provide a coverage scheme that ensures UAV mobility is minimised. Using simulations, we show that our approach outperforms the baseline approaches. The obvious drawback of this work is the obstacle free coverage area considered. Stringent propagation requirements of mmWaves also include the LOS component, along with the distance criterion. Moreover, the coverage provided by each UAV

has been considered to be equal. In reality, UAVs can change their heights, transmission powers, and azimuth angles to achieve a required coverage. In that case, the problem will no longer remain equivalent to the unit disk cover problem, posing interesting challenges. Finally, we have considered coverage with respect to the initially allocated UAV; i.e., handoffs have not been considered. Intuitively, the total UAV mobility would appear to decrease if we allow a UE to be covered by different UAVs at different points of time, which in itself is an even more interesting problem.

## Chapter 8

# Conclusion and Future Directions

In this thesis, we have broadly dealt with two problems in the field of [mmWave](#) communications, namely obstacle detection for obstruction-free, stable transmission path allocation, and efficient deployment of network infrastructure for achieving maximal coverage of the service area. The weakly penetrating nature of [mmWaves](#) necessitates [LOS](#) paths between the transmitter and receiver. Hence, achieving such [LOS](#) transmission paths by avoiding obstacles and efficiently deploying transmission hardware is of paramount importance for large scale migration to the [mmWave](#) communication spectrum.

In [Part I](#) of the thesis, we deal with obstacle detection, and subsequent stable path allocation. In [Chapter 3](#), we first detect locations of static obstacles without resorting to any a priori satellite imagery, based on historical link failure data alone. We extend this approach to detect zones of high dynamic congestion, and allocate paths avoiding such obstruction-prone zones. In [Chapter 4](#), we aim to track dynamic obstacles without resorting to additional tracking hardware like RGB-D cameras, and use that information to trigger handoffs for potentially at-risk links. We first consider the case of tracking a single dynamic obstacle in the coverage area, and propose a simple approach to track the same. Thereafter, we turn our attention towards the much harder problem of tracking multiple dynamic obstacles from short term link failure information alone. We show that the problem is NP-complete, and propose a greedy set cover based algorithm to solve the same.

Via simulations, we show that for low to moderate camera coverage, our approach achieves better handoff performance in comparison with an RGB-D camera based approach. We deal with the notion of link stability for mobile UEs, in Chapter 5. Taking into consideration the mobility of UEs, and static obstacles in the coverage area, we assign those links to mobile UEs that are likely to satisfy the mmWave transmission criteria for the longest possible time.

We shift focus to the hardware deployment problem in Part II of the thesis. We first deal with the joint placement of mmWave BSs and reflectors in Chapter 6, with an aim to maximise the coverage area for a given amount of infrastructure. We propose two approaches to solve the joint problem, one based on the classical set-cover approximation, and another based on LP relaxation. We demonstrate the superiority of the coverage obtained by treating the problem jointly, as compared to a two step sequential approach. Finally in Chapter 7, we consider the deployment of UAVs equipped with on board mmWave BSs. Taking into consideration UE mobility, we propose LazyUAV, an algorithm that attempts to deploy multiple mobile UAVs while ensuring minimum displacement, and maximal coverage. We show via simulations that LazyUAV achieves lesser displacement compared to two baseline approaches.

Next, we discuss some of the limitations of the works presented in this thesis. We begin off by stating that the path allocation algorithms in this thesis can only be applied in an outdoor scenario. Indeed, a bulk of the algorithms use accurate location of the users, which might not be available for indoor scenarios. Additionally, the joint base station and reflector problem does not consider zones of dynamic congestion, instead focusing only on static obstacles. It would be logical to utilise the tracking information obtained in Chapters 3 and 4 to better place the infrastructure. As for the LazyUAV algorithm, it does not consider any obstacles, just focusing merely on transmission range. Incorporating the information learnt from Chapters 3 and 4 would undoubtedly bring the problem closer to real life. Due to the lack of obstacles, the coverage area of each UAV is assumed to be circular, which is almost never the case in real life. We have also not considered changing the heights of the UAVs, or their corresponding azimuth angles to

modify the coverage area. Finally, in a broader aspect, we have not considered any interference in infrastructure placement problem, relying instead on the assumption that the number of channels available at a base station is sufficient to serve the demanding users. With massive surge in user demand, this may not be very realistic. Frequency has to be shared between demanding users, leading to interference and fairness concerns. Finally, UAVs may even incorporate handoff UEs to neighbouring UAVs along with self displacement. This approach has not been explored in the thesis.

Some of the directions of continuing the works in this thesis can be as follows. The dynamic obstacle tracking approaches may be improved by considering the fact that a single obstacle may block links associated with multiple mmWave BSs, especially for the case of fast moving dynamic obstacles. Apart from link failure information, link success information can also be encoded in the problem to obtain better obstacle tracking results. For the mmWave BS and reflector placement problem, we have considered a grid to be covered if it is served by one such BS (either directly, or via reflector) alone. However in real life,  $k$ -cover of the deployment area may be necessary, especially in places with high dynamic obstacle congestion. Additionally, incorporating the effect of dynamic obstacle directly into the infrastructure placement problem seems to be a challenging problem, which we would like to explore in the future. Finally, LazyUAV in its current form does not incorporate handoffs; such handoffs would undoubtedly reduce UAV mobility even more. We would like to explore these directions in future.





# Bibliography

- [1] “Ericsson mobility report data and forecasts.” <https://www.ericsson.com/en/reports-and-papers/mobility-report/dataforecasts/mobile-traffic-forecast>.
- [2] J. Qiao, X. S. Shen, J. W. Mark, Q. Shen, Y. He, and L. Lei, “Enabling device-to-device communications in millimeter-wave 5g cellular networks,” *IEEE Communications Magazine*, vol. 53, pp. 209–215, January 2015.
- [3] L. Wei, R. Q. Hu, Y. Qian, and G. Wu, “Key elements to enable millimeter wave communications for 5g wireless systems,” *IEEE Wireless Communications*, vol. 21, pp. 136–143, December 2014.
- [4] Z. Pi and F. Khan, “An introduction to millimeter-wave mobile broadband systems,” *IEEE Communications Magazine*, vol. 49, pp. 101–107, June 2011.
- [5] T. S. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. N. Wong, J. K. Schulz, M. Samimi, and F. Gutierrez, “Millimeter wave mobile communications for 5G cellular: It will work!,” *IEEE Access*, vol. 1, pp. 335–349, 2013.
- [6] J. Liu, N. Kato, J. Ma, and N. Kadowaki, “Device-to-device communication in lte-advanced networks: A survey,” *IEEE Communications Surveys Tutorials*, vol. 17, pp. 1923–1940, Fourthquarter 2015.
- [7] R. C. Daniels and R. W. Heath, “60 ghz wireless communications: Emerging requirements and design recommendations,” *IEEE Vehicular Technology Magazine*, vol. 2, pp. 41–50, Sep. 2007.

- 
- [8] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5g," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, 2014.
- [9] A. M. Al-samman, M. H. Azmi, and T. A. Rahman, "A survey of millimeter wave (mm-wave) communications for 5g: Channel measurement below and above 6 ghz," in *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International Conference of Reliable Information and Communication Technology (IRICT 2018)*, pp. 451–463, Springer, 2019.
- [10] R. Ford, M. Zhang, M. Mezzavilla, S. Dutta, S. Rangan, and M. Zorzi, "Achieving ultra-low latency in 5g millimeter wave cellular networks," *IEEE Communications Magazine*, vol. 55, pp. 196–203, March 2017.
- [11] T. S. Rappaport, R. W. Heath Jr, R. C. Daniels, and J. N. Murdock, *Millimeter wave wireless communications*. Pearson Education, 2015.
- [12] Q. Xue, P. Zhou, X. Fang, and M. Xiao, "Performance analysis of interference and eavesdropping immunity in narrow beam mmwave networks," *IEEE Access*, vol. 6, pp. 67611–67624, 2018.
- [13] C. Doan, S. Emami, A. Niknejad, and R. Brodersen, "Millimeter-wave cmos design," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 144–155, Jan 2005.
- [14] B. Razavi, "Design of millimeter-wave cmos radios: A tutorial," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, pp. 4–16, Jan 2009.
- [15] F. Gutierrez, S. Agarwal, K. Parrish, and T. S. Rappaport, "On-chip integrated antenna structures in cmos for 60 ghz wpan systems," *IEEE Journal on Selected Areas in Communications*, vol. 27, pp. 1367–1378, October 2009.
- [16] A. M. Niknejad, S. Emami, B. Heydari, M. Bohsali, and E. Adabi, "Nanoscale cmos for mm-wave applications," in *2007 IEEE Compound Semiconductor Integrated Circuits Symposium*, pp. 1–4, Oct 2007.

- [17] T. S. Rappaport, J. N. Murdock, and F. Gutierrez, “State of the art in 60-ghz integrated circuits and systems for wireless communications,” *Proceedings of the IEEE*, vol. 99, pp. 1390–1436, Aug 2011.
- [18] “Ecma-387.” <https://www.ecma-international.org/publications-and-standards/standards/ecma-387/>.
- [19] H. Singh, S.-K. Yong, J. Oh, and C. Ngo, “Principles of ieee 802.15.3c: Multi-gigabit millimeter-wave wireless pan,” in *2009 Proceedings of 18th International Conference on Computer Communications and Networks*, pp. 1–6, Aug 2009.
- [20] E. Perahia, C. Cordeiro, M. Park, and L. L. Yang, “Ieee 802.11ad: Defining the next generation multi-gbps wi-fi,” in *2010 7th IEEE Consumer Communications and Networking Conference*, pp. 1–5, Jan 2010.
- [21] L. Jiang and H. Jafarkhani, “mmwave amplify-and-forward mimo relay networks with hybrid precoding/combining design,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 1333–1346, 2020.
- [22] H. T. Friis, “A note on a simple transmission formula,” *Proceedings of the IRE*, vol. 34, pp. 254–256, May 1946.
- [23] T. Rappaport, *Wireless communications: Principles and practice*. Prentice Hall, 1996.
- [24] Hao Xu, V. Kukshya, and T. S. Rappaport, “Spatial and temporal characteristics of 60-ghz indoor channels,” *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 3, pp. 620–630, 2002.
- [25] S. Geng, J. Kivinen, X. Zhao, and P. Vainikainen, “Millimeter-wave propagation channel characterization for short-range wireless communications,” *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 3–13, Jan 2009.
- [26] A. Tharek and J. McGeehan, “Propagation and bit error rate measurements within buildings in the millimeter wave band about 60 ghz,” in *8th European*

- Conference on Electrotechnics, Conference Proceedings on Area Communication*, pp. 318–321, June 1988.
- [27] A. Hammoudeh and G. Allen, “Millimetric wavelengths radiowave propagation for line-of-sight indoor microcellular mobile communications,” *IEEE Transactions on Vehicular Technology*, vol. 44, pp. 449–460, Aug 1995.
- [28] N. Moraitis and P. Constantinou, “Measurements and characterization of wideband indoor radio channel at 60 ghz,” *IEEE Transactions on Wireless Communications*, vol. 5, pp. 880–889, April 2006.
- [29] S. Singh, R. Mudumbai, and U. Madhow, “Interference analysis for highly directional 60-ghz mesh networks: The case for rethinking medium access control,” *IEEE/ACM Transactions on Networking*, vol. 19, pp. 1513–1527, Oct 2011.
- [30] M. Marcus and B. Pattan, “Millimeter wave propagation: spectrum management implications,” *IEEE Microwave Magazine*, vol. 6, pp. 54–62, June 2005.
- [31] S. Singh, F. Ziliotto, U. Madhow, E. Belding, and M. Rodwell, “Blockage and directivity in 60 ghz wireless personal area networks: from cross-layer model to multihop mac design,” *IEEE Journal on Selected Areas in Communications*, vol. 27, pp. 1400–1413, October 2009.
- [32] S. Collonge, G. Zaharia, and G. Zein, “Influence of the human activity on wide-band characteristics of the 60 ghz indoor radio channel,” *IEEE Transactions on Wireless Communications*, vol. 3, pp. 2396–2406, Nov 2004.
- [33] Z. Xiao, L. Zhu, Y. Liu, P. Yi, R. Zhang, X.-G. Xia, and R. Schober, “A survey on millimeter-wave beamforming enabled uav communications and networking,” *IEEE Communications Surveys & Tutorials*, vol. 24, pp. 557–610, Firstquarter 2022.
- [34] “Facebook project aries.” <https://research.facebook.com/videos/connectivity-labs-project-aries/>.

- [35] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive mimo for next generation wireless systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186–195, 2014.
- [36] W. Roh, J.-Y. Seol, J. Park, B. Lee, J. Lee, Y. Kim, J. Cho, K. Cheun, and F. Aryanfar, "Millimeter-wave beamforming as an enabling technology for 5g cellular communications: theoretical feasibility and prototype results," *IEEE Communications Magazine*, vol. 52, pp. 106–113, February 2014.
- [37] S. Kutty and D. Sen, "Beamforming for millimeter wave communications: An inclusive survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 949–973, 2016.
- [38] Y. Hu, K. Kang, S. Majhi, and H. Qian, "Downlink beamforming design for mobile users in massive mimo system," *Digital Signal Processing*, vol. 130, p. 103716, 2022.
- [39] M. Shariat, M. Dianati, K. Seppänen, T. Suihko, J. Putkonen, and V. Frascolla, "Enabling wireless backhauling for next generation mmwave networks," in *2015 European Conference on Networks and Communications (EuCNC)*, pp. 164–168, June 2015.
- [40] Z. Marzi, U. Madhow, and H. Zheng, "Interference analysis for mm-wave picocells," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, Dec 2015.
- [41] D. López-Pérez, M. Ding, H. Claussen, and A. H. Jafari, "Towards 1 gbps/ue in cellular systems: Understanding ultra-dense small cell deployments," *IEEE Communications Surveys & Tutorials*, vol. 17, pp. 2078–2101, Fourthquarter 2015.
- [42] S. Stefanatos and A. Alexiou, "Access point density and bandwidth partitioning in ultra dense wireless networks," *IEEE transactions on communications*, vol. 62, no. 9, pp. 3376–3384, 2014.

- [43] M. Ding, D. Lopez-Perez, G. Mao, P. Wang, and Z. Lin, "Will the area spectral efficiency monotonically grow as small cells go dense?," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, Dec 2015.
- [44] R. Baldemair, T. Irnich, K. Balachandran, E. Dahlman, G. Mildh, Y. Selén, S. Parkvall, M. Meyer, and A. Osseiran, "Ultra-dense networks in millimeter-wave frequencies," *IEEE Communications Magazine*, vol. 53, no. 1, pp. 202–208, 2015.
- [45] Z. Gao, L. Dai, D. Mi, Z. Wang, M. A. Imran, and M. Z. Shakir, "Mmwave massive-mimo-based wireless backhaul for the 5g ultra-dense network," *IEEE Wireless Communications*, vol. 22, pp. 13–21, October 2015.
- [46] M. Kamel, W. Hamouda, and A. Youssef, "Ultra-dense networks: A survey," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 2522–2545, Fourthquarter 2016.
- [47] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Communications Surveys & Tutorials*, vol. 16, pp. 1801–1819, Fourthquarter 2014.
- [48] K. Vanganuru, S. Ferrante, and G. Sternberg, "System capacity and coverage of a cellular network with d2d mobile relays," in *MILCOM 2012 - 2012 IEEE Military Communications Conference*, pp. 1–6, 2012.
- [49] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-device communication in 5g cellular networks: challenges, solutions, and future directions," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 86–92, 2014.
- [50] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *IEEE Pervasive Computing*, vol. 7, no. 4, pp. 12–18, 2008.
- [51] L. Sun, J. Hou, and T. Shu, "Spatial and temporal contextual multi-armed bandit handovers in ultra-dense mmwave cellular networks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 12, pp. 3423–3438, 2021.

- [52] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, “Improved handover through dual connectivity in 5g mmwave mobile networks,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 9, pp. 2069–2084, 2017.
- [53] D. Marasinghe, N. Rajatheva, and M. Latva-aho, “Lidar aided human blockage prediction for 6g,” in *2021 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2021.
- [54] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, “Reinforcement learning based predictive handover for pedestrian-aware mmwave networks,” in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 692–697, 2018.
- [55] M. Häselich, B. Jöbgen, N. Wojke, J. Hedrich, and D. Paulus, “Confidence-based pedestrian tracking in unstructured environments using 3d laser distance measurements,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4118–4123, 2014.
- [56] A. T.-Y. Chen, M. Biglari-Abhari, and K. I.-K. Wang, “Context is king: Privacy perceptions of camera-based surveillance,” in *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, 2018.
- [57] C. K. Anjinappa, F. Erden, and I. Güvenç, “Base station and passive reflectors placement for urban mmwave networks,” *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3525–3539, 2021.
- [58] Q. Wu, S. Zhang, B. Zheng, C. You, and R. Zhang, “Intelligent reflecting surface-aided wireless communications: A tutorial,” *IEEE Transactions on Communications*, vol. 69, no. 5, pp. 3313–3351, 2021.
- [59] Q. Wu, X. Guan, and R. Zhang, “Intelligent reflecting surface-aided wireless energy and information transmission: An overview,” *Proceedings of the IEEE*, vol. 110, pp. 150–170, Jan 2022.

- [60] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Communications Magazine*, vol. 58, no. 1, pp. 106–112, 2020.
- [61] Y. Zeng, Q. Wu, and R. Zhang, "Accessing from the sky: A tutorial on uav communications for 5g and beyond," *Proceedings of the IEEE*, vol. 107, no. 12, pp. 2327–2375, 2019.
- [62] A. Omran, L. Sboui, M. Kadoch, Z. Chang, J. Lu, and R. Liu, "3d deployment of multiple uavs for emergent on-demand offloading," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 692–696, 2020.
- [63] N. Tafintsev, D. Moltchanov, S. Andreev, S.-p. Yeh, N. Himayat, Y. Koucheryavy, and M. Valkama, "Handling spontaneous traffic variations in 5g+ via offloading onto mmwave-capable uav "bridges"," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 9, pp. 10070–10084, 2020.
- [64] N. Jin, J. Gui, and X. Zhou, "Equalizing service probability in uav-assisted wireless powered mmwave networks for post-disaster rescue," *Computer Networks*, vol. 225, p. 109644, 2023.
- [65] T. Do-Duy, L. D. Nguyen, T. Q. Duong, S. R. Khosravirad, and H. Claussen, "Joint optimisation of real-time deployment and resource allocation for uav-aided disaster emergency communications," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 11, pp. 3411–3424, 2021.
- [66] M. A. Khan, A. Safi, I. M. Qureshi, and I. U. Khan, "Flying ad-hoc networks (fanets): A review of communication architectures, and routing protocols," in *2017 First International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT)*, pp. 1–9, 2017.
- [67] A. Chriki, H. Touati, H. Snoussi, and F. Kamoun, "Fanet: Communication, mobility models and security issues," *Computer Networks*, vol. 163, p. 106877, 2019.



- [68] K. Meng, Q. Wu, J. Xu, W. Chen, Z. Feng, R. Schober, and A. L. Swindlehurst, "Uav-enabled integrated sensing and communication: Opportunities and challenges," *IEEE Wireless Communications*, pp. 1–9, 2023.
- [69] M. Piorkowski, N. Sarafijanovic-Djukic, and M. Grossglauser, "Crawdad epfl/mobility," 2022.
- [70] F. Li, C. He, X. Li, J. Peng, and K. Yang, "Geometric analysis-based 3d anti-block uav deployment for mmwave communications," *IEEE Communications Letters*, vol. 26, no. 11, pp. 2799–2803, 2022.
- [71] K. Shamganth and M. J. N. Sibley, "A survey on relay selection in cooperative device-to-device (d2d) communication for 5g cellular networks," in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pp. 42–46, 2017.
- [72] J. Qiao, L. X. Cai, X. S. Shen, and J. W. Mark, "Enabling multi-hop concurrent transmissions in 60 GHz wireless personal area networks," *IEEE Transactions on Wireless Communications*, vol. 10, pp. 3824–3833, November 2011.
- [73] Z. Genc, U. H. Rizvi, E. Onur, and I. Niemegeers, "Robust 60 ghz indoor connectivity: Is it possible with reflections?," in *2010 IEEE 71st Vehicular Technology Conference*, pp. 1–5, 2010.
- [74] S. Biswas, S. Vuppala, J. Xue, and T. Ratnarajah, "On the performance of relay aided millimeter wave networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 576–588, 2016.
- [75] Y. Liu, Q. Hu, and D. M. Blough, "Blockage avoidance in relay paths for roadside mmwave backhaul networks," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp. 1–7, 2018.
- [76] J. Du, E. Onaran, D. Chizhik, S. Venkatesan, and R. A. Valenzuela, "Gbps user rates using mmwave relayed backhaul with high-gain antennas," *IEEE*

- Journal on Selected Areas in Communications*, vol. 35, no. 6, pp. 1363–1372, 2017.
- [77] Q. Hu and D. M. Blough, “Optimizing millimeter-wave backhaul networks in roadside environments,” in *2018 IEEE International Conference on Communications (ICC)*, pp. 1–7, 2018.
- [78] D. Singh and S. C. Ghosh, “Network-assisted d2d relay selection under the presence of dynamic obstacles,” in *2019 IEEE 44th Conference on Local Computer Networks (LCN)*, pp. 129–132, 2019.
- [79] D. Singh and S. C. Ghosh, “Mobility-aware relay selection in 5g d2d communication using stochastic model,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2837–2849, 2019.
- [80] N. Eshraghi, B. Maham, and V. Shah-Mansouri, “Millimeter-wave device-to-device multi-hop routing for multimedia applications,” in *2016 IEEE International Conference on Communications (ICC)*, pp. 1–6, May 2016.
- [81] N. Wei, X. Lin, and Z. Zhang, “Optimal relay probing in millimeter-wave cellular systems with device-to-device relaying,” *IEEE Transactions on Vehicular Technology*, vol. 65, pp. 10218–10222, Dec 2016.
- [82] Y. Niu, C. Gao, Y. Li, L. Su, and D. Jin, “Exploiting multi-hop relaying to overcome blockage in directional mmwave small cells,” *Journal of Communications and Networks*, vol. 18, no. 3, pp. 364–374, 2016.
- [83] Q. Hu and D. M. Blough, “Relay selection and scheduling for millimeter wave backhaul in urban environments,” in *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 206–214, Oct 2017.
- [84] J. Kim and A. F. Molisch, “Quality-aware millimeter-wave device-to-device multi-hop routing for 5g cellular networks,” in *2014 IEEE International Conference on Communications (ICC)*, pp. 5251–5256, 2014.

- [85] D. Ebrahimi, H. Elbiaze, and W. Ajib, "Device-to-device data transfer through multihop relay links underlying cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9669–9680, 2018.
- [86] A. Talukdar, M. Cudak, and A. Ghosh, "Handoff rates for millimeterwave 5g systems," in *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, 2014.
- [87] J. Jebramcik, J. Wagner, N. Pohl, I. Rolfes, and J. Barowski, "Millimeter wave material measurements for building entry loss models above 100 ghz," in *2021 15th European Conference on Antennas and Propagation (EuCAP)*, pp. 1–5, 2021.
- [88] D. Marasinghe, N. Jayaweera, N. Rajatheva, S. Hakola, T. Koskela, O. Tervo, J. Karjalainen, E. Tirola, and J. Hulkkonen, "Lidar aided wireless networks - beam prediction for 5g," in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, pp. 1–7, 2022.
- [89] D. Marasinghe, N. Rajatheva, and M. Latva-aho, "Lidar aided human blockage prediction for 6g," in *2021 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2021.
- [90] Y. Oguma, R. Arai, T. Nishio, K. Yamamoto, and M. Morikura, "Proactive base station selection based on human blockage prediction using rgb-d cameras for mmwave communications," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2015.
- [91] Y. Koda, K. Nakashima, K. Yamamoto, T. Nishio, and M. Morikura, "Cooperative sensing in deep rl-based image-to-decision proactive handover for mmwave networks," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–6, 2020.
- [92] G. Charan, M. Alrabeiah, and A. Alkhateeb, "Vision-aided 6g wireless communications: Blockage prediction and proactive handoff," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10193–10208, 2021.

- [93] M. A. L. Sarker, I. Orikumhi, J. Kang, H.-K. Jwa, J.-H. Na, and S. Kim, "Vision-aided beam allocation for indoor mmwave communications," in *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1403–1408, 2021.
- [94] Y. Oguma, T. Nishio, K. Yamamoto, and M. Morikura, "Performance modeling of camera-assisted proactive base station selection for human blockage problem in mmwave communications," in *2016 IEEE Wireless Communications and Networking Conference*, pp. 1–7, 2016.
- [95] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter wave base stations with cameras: Vision aided beam and blockage prediction," *CoRR*, vol. abs/1911.06255, 2019.
- [96] Y. Koda, K. Nakashima, K. Yamamoto, T. Nishio, and M. Morikura, "Cooperative sensing in deep rl-based image-to-decision proactive handover for mmwave networks," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–6, 2020.
- [97] T. Nishio, H. Okamoto, K. Nakashima, Y. Koda, K. Yamamoto, M. Morikura, Y. Asai, and R. Miyatake, "Proactive received power prediction using machine learning and depth images for mmwave networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2413–2427, 2019.
- [98] S. Gidel, P. Checchin, C. Blanc, T. Chateau, and L. Trassoudaine, "Pedestrian detection and tracking in an urban environment using a multilayer laser scanner," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 579–588, 2010.
- [99] A. Fod, A. Howard, and M. Mataric, "A laser-based people tracker," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 3, pp. 3024–3029 vol.3, 2002.

- [100] X. An, C.-S. Sum, R. V. Prasad, J. Wang, Z. Lan, J. Wang, R. Hekmat, H. Harada, and I. Niemegeers, “Beam switching support to resolve link-blockage problem in 60 ghz wpans,” in *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 390–394, IEEE, 2009.
- [101] Y. Kumar S and T. Ohtsuki, “Influence and mitigation of pedestrian blockage at mmwave cellular networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15442–15457, 2020.
- [102] T. Nishio, R. Arai, K. Yamamoto, and M. Morikura, “Proactive traffic control based on human blockage prediction using rgb-d cameras for millimeter-wave communications,” in *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, pp. 152–153, 2015.
- [103] T. Nishio, H. Okamoto, K. Nakashima, Y. Koda, K. Yamamoto, M. Morikura, Y. Asai, and R. Miyatake, “Proactive received power prediction using machine learning and depth images for mmwave networks,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2413–2427, 2019.
- [104] Y. Oguma, T. Nishio, K. Yamamoto, and M. Morikura, “Performance modeling of camera-assisted proactive base station selection for human blockage problem in mmwave communications,” in *2016 IEEE Wireless Communications and Networking Conference*, pp. 1–7, IEEE, 2016.
- [105] T. Zhang, J. Liu, and F. Gao, “Vision aided beam tracking and frequency handoff for mmwave communications,” in *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–2, 2022.
- [106] Microsoft, “Azure kinect dk hardware specifications.” <https://learn.microsoft.com/en-us/azure/kinect-dk/hardware-specification>.
- [107] H. Okamoto, T. Nishio, M. Morikura, K. Yamamoto, D. Murayama, and K. Nakahira, “Machine-learning-based throughput estimation using images

- for mmwave communications,” in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, pp. 1–6, IEEE, 2017.
- [108] F. B. Tesema, A. Awada, I. Viering, M. Simsek, and G. P. Fettweis, “Mobility modeling and performance evaluation of multi-connectivity in 5g intra-frequency networks,” in *2015 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, IEEE, 2015.
- [109] M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, “Multi-connectivity in 5g mmwave cellular networks,” in *2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pp. 1–7, 2016.
- [110] R. Hersyandika, Y. Miao, and S. Pollin, “Guard beam: Protecting mmwave communication through in-band early blockage prediction,” in *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pp. 4093–4098, IEEE, IEEE, 2022.
- [111] L. Weedage, C. Stegehuis, and S. Bayhan, “Impact of multi-connectivity on channel capacity and outage probability in wireless networks,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7973–7986, 2023.
- [112] N. Palizban, S. Szyszkowicz, and H. Yanikomeroglu, “Automation of millimeter wave network planning for outdoor coverage in dense urban areas using wall-mounted base stations,” *IEEE Wireless Communications Letters*, vol. 6, no. 2, pp. 206–209, 2017.
- [113] M. Dong, M. Cho, K. Lee, S. Yoon, and T. Kim, “Cost-optimal deployment of millimeter-wave base stations under outage requirement,” *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10544–10559, 2022.
- [114] F. Erden, C. K. Anjinappa, E. Ozturk, and I. Guvenc, “Outdoor mmwave base station placement: A multi-armed bandit learning approach,” 2020.
- [115] Y. Wang and X. Zhu, “A novel network planning algorithm of three-dimensional dense networks based on adaptive variable-length particle swarm optimization,” *IEEE Access*, vol. 7, pp. 45940–45950, 2019.

- [116] M. Dong, T. Kim, J. Wu, and E. W. M. Wong, “Cost-efficient millimeter wave base station deployment in manhattan-type geometry,” *IEEE Access*, vol. 7, pp. 149959–149970, 2019.
- [117] S. S. Szyszkowicz, A. Lou, and H. Yanikomeroglu, “Automated placement of individual millimeter-wave wall-mounted base stations for line-of-sight coverage of outdoor urban areas,” *IEEE Wireless Communications Letters*, vol. 5, pp. 316–319, June 2016.
- [118] Y. Zhang, L. Dai, and E. W. M. Wong, “Optimal bs deployment and user association for 5g millimeter wave communication networks,” *IEEE Transactions on Wireless Communications*, vol. 20, pp. 2776–2791, May 2021.
- [119] M. Dong, T. Kim, J. Wu, and E. W.-M. Wong, “Millimeter-wave base station deployment using the scenario sampling approach,” *IEEE Transactions on Vehicular Technology*, vol. 69, pp. 14013–14018, Nov 2020.
- [120] S. Gong, X. Lu, D. T. Hoang, D. Niyato, L. Shu, D. I. Kim, and Y.-C. Liang, “Toward smart wireless communications via intelligent reflecting surfaces: A contemporary survey,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2283–2314, 2020.
- [121] W. A. Gulzar Khawaja, O. Ozdemir, F. Erden, I. Guvenc, M. Ezuma, and Y. Kakishima, “Effect of passive reflectors for enhancing coverage of 28 ghz mmwave systems in an outdoor setting,” in *2019 IEEE Radio and Wireless Symposium (RWS)*, pp. 1–4, 2019.
- [122] W. Yan, X. Yuan, and X. Kuai, “Passive beamforming and information transfer via large intelligent surface,” *IEEE Wireless Communications Letters*, vol. 9, no. 4, pp. 533–537, 2020.
- [123] S. Zeng, H. Zhang, B. Di, Z. Han, and L. Song, “Reconfigurable intelligent surface (ris) assisted wireless coverage extension: Ris orientation and location optimization,” *IEEE Communications Letters*, vol. 25, no. 1, pp. 269–273, 2021.

- [124] M. Issa and H. Artail, "Using reflective intelligent surfaces for indoor scenarios: Channel modeling and ris placement," in *2021 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pp. 277–282, 2021.
- [125] K. Ntontin, D. Selimis, A.-A. A. Boulogeorgos, A. Alexandridis, A. Tsolis, V. Vlachodimitropoulos, and F. Lazarakis, "Optimal reconfigurable intelligent surface placement in millimeter-wave communications," in *2021 15th European Conference on Antennas and Propagation (EuCAP)*, pp. 1–5, 2021.
- [126] O. Ozdemir, F. Erden, I. Guvenc, T. Yekan, and T. Zarian, "28 ghz mmwave channel measurements: A comparison of horn and phased array antennas and coverage enhancement using passive and active repeaters," 2020.
- [127] P.-Q. Huang, Y. Zhou, K. Wang, and B.-C. Wang, "Placement optimization for multi-irs-aided wireless communications: An adaptive differential evolution algorithm," *IEEE Wireless Communications Letters*, vol. 11, pp. 942–946, May 2022.
- [128] 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*, vol. 21, no. 4, pp. 3417–3442, 2019.
- [129] 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.
- [130] E. Kalantari, H. Yanikomeroglu, and A. Yongacoglu, "On the number and 3d placement of drone base stations in wireless cellular networks," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, pp. 1–6, 2016.
- [131] Y. Liu, W. Huangfu, H. Zhou, H. Zhang, J. Liu, and K. Long, "Fair and energy-efficient coverage optimization for uav placement problem in the



- cellular network,” *IEEE Transactions on Communications*, vol. 70, no. 6, pp. 4222–4235, 2022.
- [132] Q. Wu, L. Liu, and R. Zhang, “Fundamental trade-offs in communication and trajectory design for uav-enabled wireless network,” *IEEE Wireless Communications*, vol. 26, no. 1, pp. 36–44, 2019.
- [133] J. Xu, Y. Zeng, and R. Zhang, “Uav-enabled wireless power transfer: Trajectory design and energy optimization,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 8, pp. 5092–5106, 2018.
- [134] Y. Zeng, R. Zhang, and T. J. Lim, “Throughput maximization for uav-enabled mobile relaying systems,” *IEEE Transactions on Communications*, vol. 64, no. 12, pp. 4983–4996, 2016.
- [135] S. Yin, S. Zhao, Y. Zhao, and F. R. Yu, “Intelligent trajectory design in uav-aided communications with reinforcement learning,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 8227–8231, 2019.
- [136] Y. Zeng, X. Xu, and R. Zhang, “Trajectory design for completion time minimization in uav-enabled multicasting,” *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2233–2246, 2018.
- [137] Y. Li, A. H. Aghvami, and D. Dong, “Intelligent trajectory planning in uav-mounted wireless networks: A quantum-inspired reinforcement learning perspective,” *IEEE Wireless Communications Letters*, vol. 10, no. 9, pp. 1994–1998, 2021.
- [138] Z. Xiao, P. Xia, and X.-g. Xia, “Enabling uav cellular with millimeter-wave communication: potentials and approaches,” *IEEE Communications Magazine*, vol. 54, no. 5, pp. 66–73, 2016.
- [139] C. Zhang, W. Zhang, W. Wang, L. Yang, and W. Zhang, “Research challenges and opportunities of uav millimeter-wave communications,” *IEEE Wireless Communications*, vol. 26, pp. 58–62, February 2019.

- [140] J. Bao, D. Sprinz, and H. Li, “Blockage of millimeter wave communications on rotor uavs: Demonstration and mitigation,” in *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*, pp. 768–774, 2017.
- [141] L. Zhu, J. Zhang, Z. Xiao, and R. Schober, “Optimization of multi-uavs aided millimeter-wave massive mimo networks,” in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1–6, 2020.
- [142] H. Vaezy, M. Salehi Heydar Abad, O. Ercetin, H. Yanikomeroglu, M. J. Omid, and M. M. Naghsh, “Beamforming for maximal coverage in mmwave drones: A reinforcement learning approach,” *IEEE Communications Letters*, vol. 24, no. 5, pp. 1033–1037, 2020.
- [143] J. Sabzehali, V. K. Shah, H. S. Dhillon, and J. H. Reed, “3d placement and orientation of mmwave-based uavs for guaranteed los coverage,” *IEEE Wireless Communications Letters*, vol. 10, no. 8, pp. 1662–1666, 2021.
- [144] S. K. Moore, “Super-accurate gps coming to smartphones in 2018 [news],” *IEEE Spectrum*, vol. 54, pp. 10–11, November 2017.
- [145] R. Congiu, H. Shokri-Ghadikolaei, C. Fischione, and F. Santucci, “On the relay-fallback tradeoff in millimeter wave wireless system,” in *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 622–627, 2016.
- [146] T. Bai and R. W. Heath, “Coverage and rate analysis for millimeter-wave cellular networks,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, pp. 1100–1114, 2015.
- [147] H. Xie, N. González-Prelcic, and T. Shimizu, “Blockage detection and channel tracking in wideband mmwave mimo systems,” in *ICC 2021 - IEEE International Conference on Communications*, pp. 1–6, 2021.
- [148] J. E. Bresenham, “Algorithm for computer control of a digital plotter,” *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965.

- [149] A. Al-Hourani, S. Chandrasekharan, and S. Kandeepan, “Path loss study for millimeter wave device-to-device communications in urban environment,” in *2014 IEEE International Conference on Communications Workshops (ICC)*, pp. 102–107, 2014.
- [150] J. Lianghai, A. Klein, N. Kuruvatti, and H. D. Schotten, “System capacity optimization algorithm for d2d underlay operation,” in *2014 IEEE International Conference on Communications Workshops (ICC)*, pp. 85–90, 2014.
- [151] S. Rajagopal, S. Abu-Surra, and M. Malmirchegini, “Channel feasibility for outdoor non-line-of-sight mmwave mobile communication,” in *2012 IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1–6, IEEE, 2012.
- [152] Y. Sun, G. Feng, S. Qin, Y.-C. Liang, and T.-S. P. Yum, “The smart handoff policy for millimeter wave heterogeneous cellular networks,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1456–1468, 2018.
- [153] S. Kratsch, G. Philip, and S. Ray, “Point line cover: The easy kernel is essentially tight,” *ACM Trans. Algorithms*, vol. 12, apr 2016.
- [154] N. Alon, D. Moshkovitz, and S. Safra, “Algorithmic construction of sets for  $k$ -restrictions,” *ACM Trans. Algorithms*, vol. 2, p. 153–177, apr 2006.
- [155] X. Lin, R. K. Ganti, P. J. Fleming, and J. G. Andrews, “Towards understanding the fundamentals of mobility in cellular networks,” *IEEE Transactions on Wireless Communications*, vol. 12, pp. 1686–1698, April 2013.
- [156] S. Even, A. Itai, and A. Shamir, “On the complexity of time table and multi-commodity flow problems,” in *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)*, pp. 184–193, 1975.
- [157] M. Pollack, “Letter to the editor—the maximum capacity through a network,” *Operations Research*, vol. 8, no. 5, pp. 733–736, 1960.

- [158] M. Kamenetsky and M. Unbehaun, “Coverage planning for outdoor wireless lan systems,” in *2002 International Zurich Seminar on Broadband Communications Access - Transmission - Networking (Cat. No.02TH8599)*, pp. 49–49, 2002.
- [159] Z. Li, O. A. Topal, Özlem Tuğfe Demir, E. Björnson, and C. Cavdar, “mmwave coverage extension using reconfigurable intelligent surfaces in indoor dense spaces,” 2023.
- [160] “Study on channel model for frequencies from 0.5 to 100 ghz (release 15).” [https://www.etsi.org/deliver/etsi\\_tr/138900\\_138999/138901/15.00.00\\_60/](https://www.etsi.org/deliver/etsi_tr/138900_138999/138901/15.00.00_60/), 2018.
- [161] D. S. Hochbaum, *Approximating Covering and Packing Problems: Set Cover, Vertex Cover, Independent Set, and Related Problems*, p. 94–143. USA: PWS Publishing Co., 1996.
- [162] U. Feige, “A threshold of  $\ln n$  for approximating set cover,” *J. ACM*, vol. 45, p. 634–652, jul 1998.
- [163] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” in *Proceedings of the Sixteenth Annual ACM Symposium on Theory of Computing*, STOC ’84, (New York, NY, USA), p. 302–311, Association for Computing Machinery, 1984.
- [164] Gurobi Optimization, LLC, “Gurobi Optimizer Reference Manual.” <https://www.gurobi.com>, 2022.
- [165] B. Uragun, “Energy efficiency for unmanned aerial vehicles,” in *2011 10th International Conference on Machine Learning and Applications and Workshops*, vol. 2, pp. 316–320, 2011.
- [166] M. Basappa, R. Acharyya, and G. K. Das, “Unit disk cover problem in 2d,” *Journal of Discrete Algorithms*, vol. 33, pp. 193–201, 2015.

- 
- [167] T. M. Chan, “Optimal output-sensitive convex hull algorithms in two and three dimensions,” *Discrete & Computational Geometry*, vol. 16, no. 4, p. 361–368, 1996.
- [168] M. Basappa, “Line segment disk cover,” *Discrete Applied Mathematics*, vol. 305, pp. 261–271, 2021.