TÉCNICO LISBOA

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

The Interplay Between Positioning and Beamforming in Millimeter Wave Communications

João Francisco Cardoso Gante

Supervisor: Doctor Leonel Augusto Pires Seabra de Sousa
Co-Supervisor: Doctor Gabriel Falcão Paiva Fernandes

Thesis approved in public session to obtain the PhD Degree in Electrical and Computer Engineering

Jury final classification: Pass with Distinction

Jury

Chairperson: Doctor João Manuel Lage de Miranda Lemos, Professor Catedrático do Instituto Superior Técnico da Universidade de Lisboa

Members of the Committee:

Doctor Susana Isabel Barreto de Miranda Sargento, Professora Catedrática da Universidade de Aveiro

Doctor Luís Filipe Barbosa de Almeida Alexandre, Professor Catedrático da Faculdade de Engenharia da Universidade da Beira Interior

Doctor Mário Alexandre Teles de Figueiredo, Professor Catedrático do Instituto Superior Técnico da Universidade de Lisboa

Doctor Mário Gonçalo Mestre Veríssimo Silveirinha, Professor Associado (com agregação) do Instituto Superior Técnico da Universidade de Lisboa

Doctor João Pedro Castilho Pereira Santos Gomes, Professor Associado do Instituto Superior Técnico da Universidade de Lisboa

Doctor Gabriel Falcão Paiva Fernandes, Professor Auxiliar da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Funding Institutions
Fundação para a Ciência e Tecnologia (FCT)

2020

Acknowledgments

Crafting a doctoral dissertation is a long process – one that is much easier to complete in right environment, surrounded by supportive colleagues, friends, and loved ones.

Directly contributing to the obtained degree, a very special acknowledgment goes to my supervisors, Leonel Sousa and Gabriel Falcão. As this project drifted away from their main fields of expertise, I never felt the pressure to reign its direction back in any way, as one often hears from other students. Instead, I found encouragement to navigate through these uncharted seas, especially when faced with resistance from external peers. From them, I have been privileged to learn how to do top-notch research, and to understand the subtleties of the academic world, for which I am extremely thankful.

During this journey, I often had a great time at lunch with colleagues from IST and INESC-ID, where we alternated between random banter and thoughtful discussions. Among them, I would like to highlight António Simões, with whom I spent countless hours dreaming about the possibilities that could derive from our engineering studies. António was a source of support, encouraged me to seek mathematical rigour, and helped me hone many of the contents of this thesis. He has my gratitude.

By the last third of this journey, I joined nPlan, back then a newborn startup. With their first nplanauts, Alan Mosca, Dev Amratia, and Vahan Hovannisyan, I had a crash course into the enterprise world, production-grade software development, and state-of-the-art machine learning concepts like embeddings and ensembles. Often seen as orthogonal to a doctoral dissertation, these skills were critical for the highly productive end of this project, where the majority of publications were written. Those skills were also responsible for the somewhat-decent public repository that enables the reproduction of this thesis' results, which greatly expanded the visibility of the work. Perhaps most importantly, with them I learned that exceptional work can only reliably happen in a supportive environment, were mistakes are seen as fantastic opportunities to learn. However, it is certainly not a mistake to say they have my heartfelt thanks.

A journey cannot happen without a purpose, and it would be impossible to talk about that purpose without mentioning my parents, Adélia Cardoso and João Gante. My parents always went above and beyond to provide the best education to their kids, sacrificing their personal life and savings to ensure we could have the best education they could provide. Naturally, in this setting, I continuously found that I did not have to place any bounds in my ambitions, that everything was within reach if I worked accordingly. Now, equipped with one of the best educations the world has to offer, I can set my eyes into making the world a better place. To them – obrigado.

Finally, the most important paragraph of these acknowledgments: Joana. The muse for all my work. The smile in yours truly's face. The wisdom behind these kind words. This journey was a rollercoaster with a few gloomy moments, and it would have been very difficult to complete without you. With you, I learned that balance is essential, that there is no point in working recklessly for weeks if it hinders my abilities for months. That frustration and anger are not to be repressed to fully experience moments of joy and happiness. That having others alongside me makes the journey a much more pleasant experience. To you, Joana, I am eternally indebted.

This work was supported by Fundação para a Ciência e a Tecnologia (FCT) through the PhD grant with reference SFRH / BD / 103960 / 2014.

Abstract

Whereas physical obstacles were mostly associated with signal attenuation, their presence in 5G's millimeter wave systems adds complex, non-linear phenomena, including reflections and scattering. The result is a multipath propagation environment, shaped by the obstacles encountered during transmission, indicating a strong relationship between a device's received radiation and its position. This Thesis addresses two key challenges brought by millimeter wave transmissions, the beamforming selection process and outdoor positioning. In fact, new techniques are proposed, where an interplay is observed: the knowledge of a device's position can aid the beamforming selection process, and the use of beamforming unlocks new sources of information to estimate a device's position. Regarding the former, often requiring either expensive channel measurements or a lengthy brute-force search, the proposed approach leverages the device position to narrow down the search, reducing the beamforming selection latency or, alternatively, enabling narrower beams. As for the latter, to untangle the information hidden in the received signal into a mobile device position, this Thesis proposes the employment of neural networks over beamformed transmissions, enabling low-power positioning with a single anchor. This positioning system can be further enhanced so as to track users, using short-term historical data and sequence learning approaches. The proposed system sets a new state-of-the-art for non-line-of-sight millimeter wave outdoor positioning accuracy, making it a very competitive and promising alternative in the field.

Keywords

Beamforming, Deep Learning, Millimeter Wave, Neural Networks, Outdoor Positioning

Resumo

Outrora predominantemente associada à atenuação de sinais, a presença de obstáculos físicos nos sistemas 5G de ondas milimétricas induz complexos fenómenos não-lineares, nomeadamente a reflexão e o espalhamento. Consequentemente, verifica-se uma presença significativa de propagação multi-caminho, propagação essa que é moldada pelos obstáculos encontrados durante a transmissão, indicando uma forte relação entre a radiação recebida por um dispositivo e a sua posição. Esta Tese visa dar resposta a dois desafios chave que advêm do uso de ondas milimétricas, o processo de formação de feixe e o posicionamento exterior. Novas técnicas são propostas, onde se observa um efeito recíproco: o conhecimento da posição de um dado dispositivo auxilia a formação do feixe, enquanto que o uso da formação de feixe desbloqueia novas fontes de informação para o processo de estimação da posição. Relativamente à primeira, que frequentemente requer dispendiosas medições de canal ou longas pesquisas exaustivas, a abordagem proposta apoia-se na posição do dispositivo para diminuir o leque de soluções possíveis, reduzindo a latência do processo de formação de feixe ou, alternativamente, possibilitando feixes mais estreitos. Quanto à segunda questão, esta Tese propõe o emprego de redes neuronais para desembaraçar a informação de posição latente recolhida através de transmissões com formação de feixe, permitindo uma abordagem de posicionamento com uma única âncora e alta eficiência energética. Este sistema pode ser aprimorado com o uso de dados históricos recentes e abordagens de aprendizagem para sequências, permitindo então que siga utilizadores. O sistema de posicionamento proposto estabelece um novo patamar para o estado da arte da precisão de posicionamento exterior com ondas milimétricas sem linha de vista, sendo uma alternativa competitiva e promissora na área.

Palavras Chave

Formação de Feixe, Aprendizagem Profunda, Ondas Milimétricas, Redes Neuronais, Posicionamento Exterior

Contents

List of Figures

List of Tables

List of Algorithms

List of Acronyms

A-GPS	Assisted Global Positioning System
ALM	Augmented Lagrangian Multiplier
AoA	Angle-of-Arrival
AoD	Angle-of-Departure
BF	Beamforming
BFF	Beamformed Fingerprint
BS	Base Station
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CS	Compressive Sensing
CSI	Channel State Information
DL	Deep Learning
DNN	Deep Neural Network
FPGA	Field-Programmable Gate Array
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPU	Graphics Processing Unit
HPBW	Half-Power Beam Width
IoT	Internet of Things
LOS	Line-of-Sight

LSTM Long Short-Term Memory

LTE Long Term Evolution

MIMO Multiple-Input Multiple-Output

ML Machine Learning

mmWave Millimeter Wave

MSA Mobile Station Assisted

MSB Mobile Station Based

MSE Mean-Square-Error

NLOS non-Line-of-Sight

NYU New York University

OTDoA Observed-Time-Difference-of-Arrival

PDP Power Delay Profile

RF Radio Frequency

RMSE Root-Mean-Square-Error

RNN Recurrent Neural Network

SNIR Signal-to-Noise-plus-Interference Ratio

SNR Signal-to-Noise Ratio

TCN Temporal Convolutional Network

ToA Time-of-Arrival

TTFF Time to First Fix

UHF Ultra High Frequency

Notation

Sets

$\mathbb{R}$	Set of real numbers.
$\mathbb{C}$	Set of complex numbers.
$S = \{a, b, c\}$	Set S, containing the elements a, b, and c.
$\bigcup_{i=1}^{n} S_i$	Finite union of sets S_1 through S_n.

Vectors, matrices, and related operations

$\mathbf{a} \in \mathbb{R}^{n \times 1}$	Column vector $\mathbf{a}$, containing n real elements.
$\mathbf{A} \in \mathbb{C}^{m \times n}$	Matrix $\mathbf{A}$, containing m rows and n columns with complex entries.
$\mathbf{A}^T$	Matrix $\mathbf{A}$'s transpose.
$\mathrm{trace}(\mathbf{A})$	Matrix $\mathbf{A}$'s trace, the sum of its diagonal entries.

Statistical operators

$\mathbf{E}\{\mathbf{x}\}$	Expected value of $\mathbf{x}$, where $\mathbf{x}$ is a random vector.
$p(a\|\mathbf{B})$	Conditional probability of the event a given $\mathbf{B}$.

Variables and functions

$\mathrm{Re}\{a\}$	Real component of the complex number a.
$\mathrm{Im}\{a\}$	Imaginary component of the complex number a.
$\sigma(\cdot)$	Sigmoid function.
$\tanh(\cdot)$	Hyperbolic tangent function.
$\odot$	Hadamard product.
$\|\|\cdot\|\|_*$	Nuclear norm.
$\lceil\cdot\rceil$	Ceiling operation.

Part I

Introduction

1
Introduction

Contents

The advent of 5G is expected to bring new wireless communication capabilities, yet at a cost of additional challenges. One of 5G's highlights is the introduction of Millimeter Wave (mmWave) communications, unlocking a significant block of untapped bandwidth [3]. However, with mmWave transmissions, the propagation properties change dramatically compared to previous mobile networks: the resulting radiation not only has severe path loss properties, but also reflects on most visible obstacles [4][5].

Beamforming (BF) is usually employed to counteract the aforementioned attributes in systems containing Multiple-Input Multiple-Output (MIMO) antennas, enabling a steerable and directive radiation pattern [6][7]. When Line-of-Sight (LOS) communications are unattainable, the focused beam can be aimed towards obstacles, such that its reflection reaches the desired target. With careful execution, spatial multiplexing is attainable by simultaneously transmitting multiple BF streams with reduced interference levels [5]. While the BF and the reflection concepts are straightforward, one question arises: how should the resulting beam be formed? Either the transmitter has some Channel State Information (CSI), and chooses the most beneficial BF weights according to that information [8], or it doesn't, and needs to perform some beam search algorithm beforehand [9]. Each approach has its own drawbacks: the former requires an expensive measurement, while the latter has a complex search problem to solve. The first part of this Thesis concerns the beam selection process, and proposes a new big data technique that can make use of the device position to enhance the beam search concept [10].

The recent focus in mmWave communications also led to the proposal of new positioning systems [11]. The accuracy achievable in certain conditions is remarkable, achieving sub-meter precision in indoor [12] and ultra-dense LOS outdoor scenarios [13]. Nevertheless, in order to be broadly applicable to outdoor localization, a mmWave positioning system must also be able to accurately locate with devices in non-Line-of-Sight (NLOS) conditions, using a limited number of Base Stations (BSs). These requirements, allied to multiple, often overlapping non-linear propagation phenomena such as reflections and diffractions, pose serious challenges to the traditional geometry-based positioning methods. In fact, the recent mmWave experimental work in [14] demonstrates that geometry-based methods cannot be directly applied to accurately locate NLOS targets, and thus new solutions are needed. The second part of this Thesis addresses this concern, starting with the following question: if the position of a device can aid the BF process, is the reverse relationship also true? As it will be shown, the properties of mmWave transmissions can be leveraged to create an information-rich fingerprint, which was coined as Beamformed Fingerprint (BFF) [15]. With the availability of fingerprint data, Deep Learning (DL) methods [15] and hierarchy techniques [16] are then proposed to infer accurate position estimates, achieving state-of-the-art results for single-point estimates (average error of 3.3 m), in a scenario containing mostly NLOS positions.

The goal of a positioning system is to estimate the position of a target, which is a direct consequence of its movement (or lack thereof). The movement of a user, in its turn, is limited by physical restrictions, such as velocity and acceleration, as well as human-made constraints, such as traffic rules. As a consequence, it is possible to gather additional sources of information if sequences of positions are considered, as opposed to single-point estimates. In [17], the use of sequence-based DL architectures when sequences of BFFs are available to the system was proposed, effectively enabling the system to track a mobile device. As the last results section will demonstrate, Temporal Convolutional Networks (TCNs) [18] achieve a state-of-the-art estimation error as low as 1.78m, even in the presence of heterogeneous movement types and NLOS positions, using BFFs from a single BS. Moreover, not only the obtained average error is smaller, but also significantly more stable: when compared to the single-point estimate system described in the previous paragraph, the Root-Mean-Square-Error (RMSE) can be 5.41× smaller.

To conclude the analysis of the proposed positioning system, its energy consumption was also assessed - after all, the main goal of a positioning system is to locate mobile devices, which often have limited energy sources. As it will be seen in this Thesis, the proposed BFF positioning system is also 47× and 85× more energy efficient per position fix (for continuous and sporadic fixes, respectively) than low-power Assisted Global Positioning System (A-GPS) implementations, while also having slightly lower estimation errors [19]. Therefore, the proposed method significantly reduces the energy required for precise positioning in the presence of millimeter wave networks, and thus is a contender to dethrone Global Navigation Satellite System (GNSS)-based methods as the default low-power commercial positioning system.

1.1 Contributions

This Thesis positions itself at the intersection of two major fields, telecommunications and machine learning. The domain knowledge from both fields enabled a new perspec-

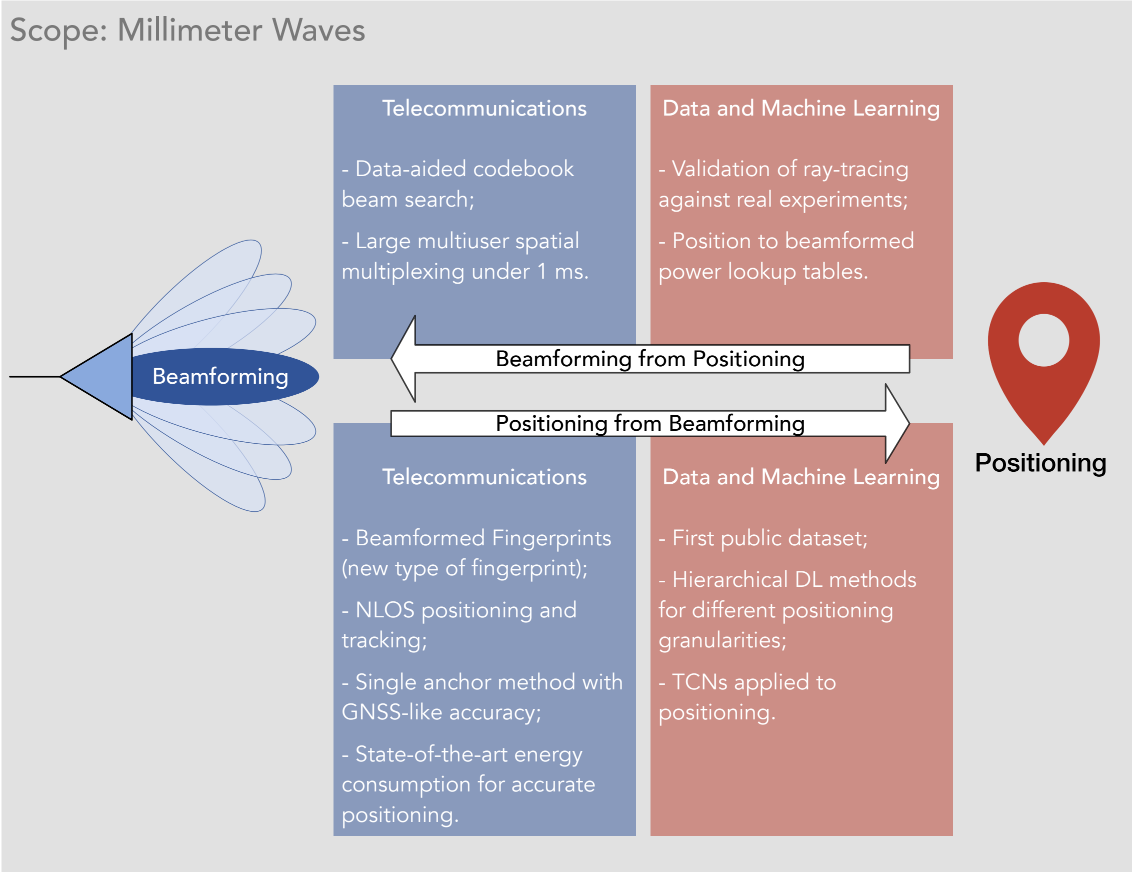

Scope: Millimeter Waves
Telecommunications
- Data-aided codebook beam search;
- Large multiuser spatial multiplexing under 1 ms.
Data and Machine Learning
- Validation of ray-tracing against real experiments;
- Position to beamformed power lookup tables.
Beamforming
Beamforming from Positioning
Positioning from Beamforming
Positioning
Telecommunications
- Beamformed Fingerprints (new type of fingerprint);
- NLOS positioning and tracking;
- Single anchor method with GNSS-like accuracy;
- State-of-the-art energy consumption for accurate positioning.
Data and Machine Learning
- First public dataset;
- Hierarchical DL methods for different positioning granularities;
- TCNs applied to positioning.

Figure 1.1: An overview of the contributions made in this Thesis, split by major field and by application.

tive over two current challenges in mmWave MIMO communications, beamforming and positioning. This Thesis also advances the study of the interplay between those two techniques, where the use of one can aid the other.

The main contributions of this document, in the area of mmWave communications, can then be split in two different directions: *i*) the improvement of the beamforming selection given the device position, and *ii*) the development of a new fingerprint-based positioning method that leverages the use of beamforming (the aforementioned *beamformed fingerprint*). Both directions are tackled using data-driven approaches, where the abundance of wireless communication data can be exploited, and a summary can be seen in fig. 1.1.

The most important contribution of this Thesis, the fingerprint-based positioning method, is also built upon recent advances in DL techniques, which have been under the spotlight for the past few years. The experiments behind this method required the creation of accurate ray-tracing simulation data, whose values were validated against real measurements. All data and code developed for this method is publicly available in an open-source repository, facilitating its reproduction and promoting further research. In particular, the data created is the first of its kind in the field, enabling simulations on a precise replica of a city (New York).

Each of the aforementioned main contributions was accomplished after a thorough review of the literature. Therefore, all the results obtained in the scope of these contributions were carefully assessed regarding the related state of the art, suggesting competitive figures both in terms of performance and resources consumption. In spite of the fact that all these contributions share a common guideline, the details referring to each one of them can be significantly different. Hence, in order to enhance clarity, their core concepts are introduced in the individual chapters referring to each one of these contributions.

1.2 Publications

The work developed throughout this Thesis resulted in two articles in international scientific journals and four papers in international scientific conferences' proceedings. The following is a list of these publications:

- International Scientific Journals:
  - J. Gante, L. Sousa and G. Falcão, "Dethroning GPS: Low-Power Accurate 5G Positioning Systems using Machine Learning", in IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 2020.
  - J. Gante, G. Falcão and L. Sousa, "Deep Learning Architectures for Accurate Millimeter Wave Positioning in 5G", in Neural Processing Letters, 2019.
- International Scientific Conference's Proceedings:
  - J. Gante, G. Falcão and L. Sousa, "Enhancing Beamformed Fingerprint Outdoor Positioning with Hierarchical Convolutional Neural Networks", in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Brighton, 2019, pp. 1473-1477.
  - J. Gante, G. Falcão and L. Sousa, "Beamformed Fingerprint Learning for Accurate Millimeter Wave Positioning", in IEEE 88th Vehicular Technology Conference (VTC2018-Fall). Chicago, 2018, pp. 1-5.
  - J. Gante, G. Falcão and L. Sousa, "Data-Aided Fast Beamforming Selection for 5G", in IEEE International Conference on Acoustics, Speech, and Signal

Processing (ICASSP). Calgary, 2018, pp. 1183-1187.

- H. Wang, J. Gante, M. Zhang, G. Falcão, L. Sousa and O. Sinnen, "High-Level Designs of Complex FIR Filters on FPGAs for the SKA", in IEEE 18th International Conference on High Performance Computing and Communications (HPCC). Sidney, 2016, pp. 797-804.

1.3 Organization

The remaining of the Thesis is organized as follows. The core concepts regarding mmWave MIMO communications, which are the basis of both contributions, are presented in chapter 2. Chapter 3 expands upon those concepts to introduce the developed beam search method, including its results. The theory behind positioning methods and the proposed beamformed fingerprint data format are characterized in chapter 4. Chapters 5 and 6 follow on with the created single-point estimate and tracking positioning systems, respectively. Finally, the conclusions are drawn in chapter 7, with the last part of this Thesis being devoted to future research directions.

Part II

Beamforming from Positioning

2

Millimeter Wave Wireless Systems

Contents

The advent of 5G related research reopened the door to the so called mmWave frequencies, unlocking a huge chunk of untapped bandwidth [3]. Although the mmWave band is formally defined as the spectrum ranging from 30 to 300 GHz, the industry and most recent scientific articles have loosely associated the term mmWave with the frequency band between 10 and 100 GHz. The renewed interest in the mmWave band is clear: while all of the world's cellphones currently operate in less than 1 GHz of bandwidth, the unlicensed 60 GHz band alone has over 5 GHz of bandwidth available [20].

At those frequencies, where the wavelengths are measured in millimeters, most objects in the physical environment are very large relative to the wavelength. Obstacles like lampposts, walls, or even people, which cause signal attenuation for traditional wireless communication frequencies (Ultra High Frequency (UHF)), are now responsible for pronounced propagation phenomena, including complete signal blockage. However, those propagation phenomena also include reflections and scattering, enabling communication links between the transmitter and the receiver even in the presence of obstructions that block the LOS path [4].

In the absence of obstacles, the short wavelength of mmWaves also exhibits another problem, as it boasts stronger interactions with the molecular constituents of the air [21]. The interactions result in a higher path loss, in excess to the well-known Friis distance-dependent free space loss, and they are particularly strong at 60 and 180 GHz. Those frequencies are then inadequate for long-range transmissions and well suited for unlicensed networks in (and around) homes and buildings, where the radiated signals will quickly decay and not interfere with other nearby networks. The optimal frequencies for long-range mmWave systems are often defined as the 200-280 GHz band and below 50 GHz [5], and this Thesis' experiments are performed on the latter.

In summary, the referred changes result in a series of technical challenges, which explains why mmWaves were left untouched for most wireless services. Some of these main challenges include counteracting these frequencies' severe path loss properties, and achieving NLOS communications, due to reflections on visible obstacles. Fortunately, the form factor of an antenna scales proportionally to the used wavelength, and thus highly directional multiple-element antennas, capable of being electrically steered, can be inexpensively integrated into mmWave systems. The remaining of this Chapter is split so as to elaborate on these three key elements of mmWave systems: open-space transmission, obstacle interaction, and the antenna techniques available for those systems.

2.1 Open-Space Transmission

The free space propagation of electromagnetic waves is usually the starting point for the evaluation of any potential wireless communications system. The propagation loss of the radiated signal power in free space is described as a function of the distance and the used antenna properties, and can be imagined as a fixed amount of power being stretched over the surface of a sphere expanding through space. The well-known Friis distance-dependent free space loss [22], originally conceived to describe an isotopic radiator, is written as

$$\frac{P_r}{P_t} = D_t D_r \left(\frac{\lambda}{4\pi d} \right)^2 , \tag{2.1}$$

where the left-hand side of the equation corresponds to the ratio between the received and transmitted power (P_r and P_t, respectively), D_t and D_r are the antenna directivities with respect to an isotropic radiator, λ is the wavelength, and d is the distance between the transmitter and the receiver antennas. Taking the analogy written above, the receiver antenna is able to capture power from its fixed-area surface, but the power flux density (measured in Watts per square meter) of the expanding sphere decreases with the distance to its center (the transmitter). Since the size and, therefore, the area of an antenna with equivalent electromagnetic properties scales linearly with the wavelength, a system employing equivalent antennas will see greater losses at higher frequencies. On the other hand, increasing the transmitted frequency without changing the antenna's size will increase its directivity [20], attenuating the effect described in the previous sentence.

To illustrate the effect of frequency on the path loss, let's consider two wireless communication systems, one operating at 2.4 GHz (typical WiFi connection) and another at 60 GHz (mmWave). Assuming that both are using omnidirectional antennas, with $D = 1$, the free space loss at a distance of 10 meters is -60dB for the former, and -88dB for the latter. This simple example shows that moving up to mmWave frequencies is no simple affair, if the same antenna technology is to be used. Fortunately, as it was mentioned above, the antenna size scales linearly with the used wavelength, and thus complex antenna arrays with higher directivities can be used with the same form factor as simple lower-frequency antennas. Furthermore, those arrays can be used in both the transmitter and the receiver, and the net gain of using said arrays on both communication ends fairly overpowers the additional loss that comes with increased frequencies [5] – at the

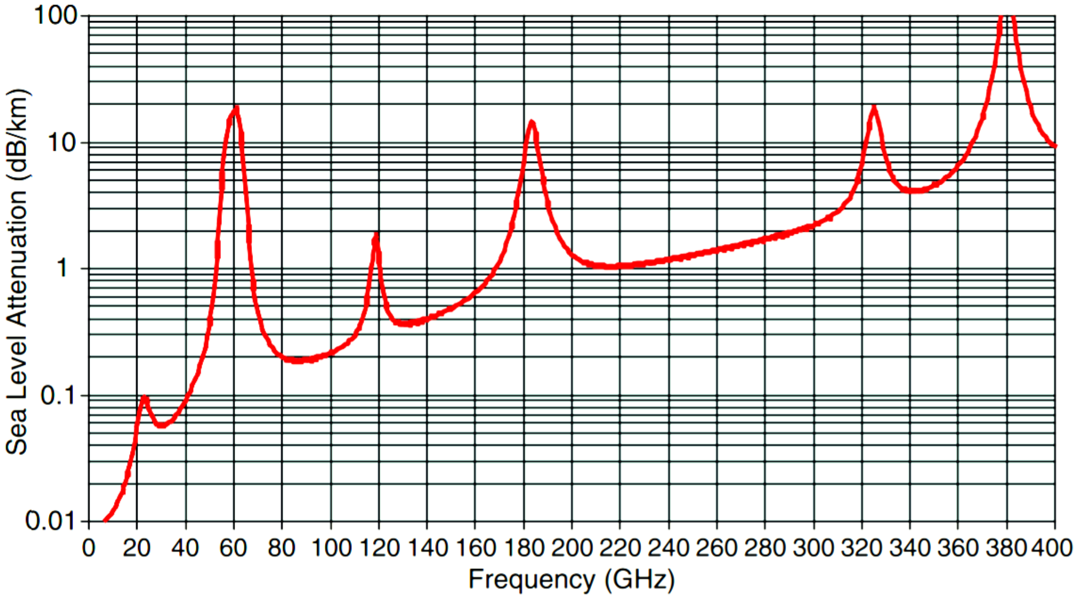


Figure 2.1: The attenuation (dB/km) in excess of free space propagation due to atmospheric absorption, at sea level [from [1]].

cost of substantially higher complexity. Returning to the sphere analogy, the use of a directive transmitter (which is facilitated in mmWave systems) corresponds to focusing the power flux density in the cross-section that will be captured by the receiver, leading to less wasteful communications.

In communications and propagation analysis, it is customary to represent propagation path loss using decibel values. This is due to the dynamic range of propagating signal powers, which change by orders of magnitude over small distances. Equation 2.1 is then often rewritten as

$$P_r[dBm] = P_t[dBm] + 10\log_{10} K - 10\alpha \log_{10} d, \tag{2.2}$$

where K is a constant for the selected system (defined from eq. (2.1)), and α is 2 in the free space propagation model discussed above. However, the free space propagation model does not always hold true in practice, with $\alpha \leq 2$ attainable when waveguiding is present (e.g. a hallway in a building), and $\alpha > 2$ in the presence of destructive interference. Nevertheless, the Friis model, associated with the appropriate excess attenuation, is often a sensible approach to model outdoor LOS propagation.

In this Chapter's introduction, a brief summary of the excess attenuation due to typical atmospheric conditions on mmWaves was laid out. In fact, for frequencies below 50 GHz, that excess attenuation does not exceed 1 dB/km, as opposed to 20 dB/km for 60

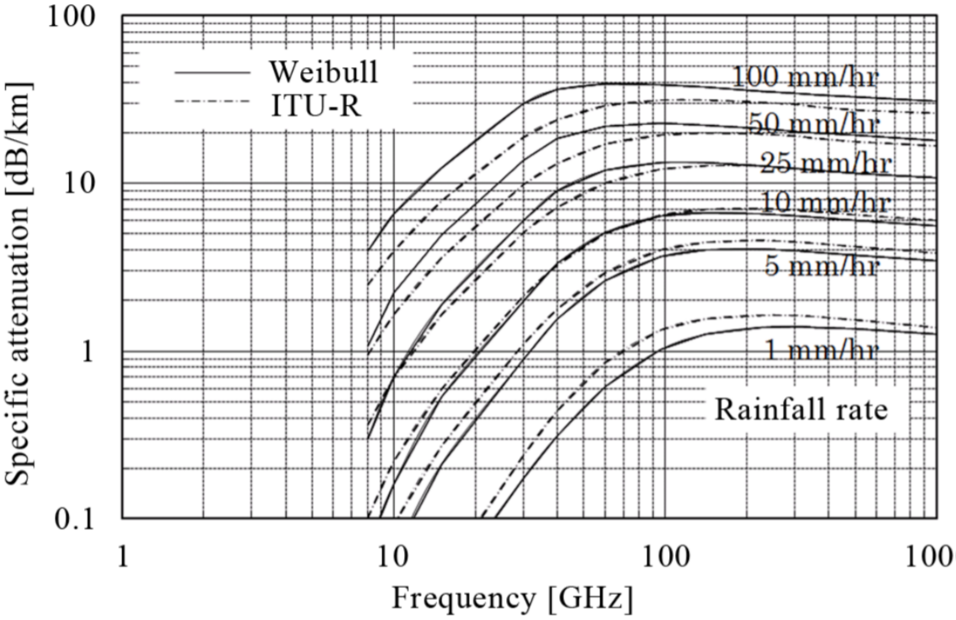


Figure 2.2: The attenuation (dB/km) in excess of free space propagation as a function of frequency and rain rate, for multi-GHz frequencies [from [2]].

GHz [1], as it is clear from fig. 2.1. However, these values do not account for additional weather effects, like rain, hail, or snow. With mmWaves, the physical dimensions of the particles resulting from those effects are on the same order of magnitude as the used wavelength, resulting in much stronger interactions (as it will be described in the following section). Rain, being the most common of these effects, is the most studied interaction, where detailed models to predict the excess attenuation given the amount of rainfall and the target transmission frequency were built in the 1980s [23]. For instance, at 28 GHz, the frequency that will be used in this Thesis' experiments, heavy rainfall (25 mm/hr) causes an excess attenuation of about 7 dB/km, as shown in fig. 2.2. While in practice 5G mmWave networks are expected to be deployed for communication ranges below 1 km in urban environments [5], strong weather effects can still cause non-negligible channel degradation.

2.2 Obstacle Interaction

Having discussed the open-space transmission for mmWaves, which share a significant background with traditional UHF systems, this section now focuses on the aspects that make mmWaves a radically different challenge: obstacle interaction. In LOS mmWave scenarios, where no obstacle is found during transmission, it is a matter of applying the models discussed in the previous section to determine the possibility of establishing a communication link. However, when an obstacle is found, a myriad of phenomena can happen, which may or may not lead to NLOS communication links. This section aims to clarify when should each phenomenon be considered, and when NLOS paths can be formed, laying a critical foundation for the remainder of this Thesis.

The most often discussed propagation phenomenon in the presence of obstacles is the *transmission* and *reflection* pair. Propagating electromagnetic waves across different media (e.g. from the air to an obstacle) are altered at the interface of both materials, due to the difference in wave impedance in the two media. At such interfaces, the electromagnetic wave is partially reflected and partially transmitted, and the ratio of energy that flows through each part is given by the absolute value of reflection coefficient ($|\mathbf{\Gamma}|$), also known as the reflectance. To understand the value of the reflectance for each scenario, one must look at the Fresnel equations [24], which can be derived from Maxwell's equations. Considering the two polarization components of a wave, s (electric field normal to the plane of incidence) and p (magnetic field normal to the plane of incidence), the two reflectance components are given by

$$|\mathbf{\Gamma}_s| = \left| \frac{Z_2\ cos\theta_i - Z_1\ cos\theta_t}{Z_2\ cos\theta_i + Z_1\ cos\theta_t} \right|^2 \tag{2.3}$$

and

$$|\mathbf{\Gamma}_p| = \left| \frac{Z_2\ cos\theta_t - Z_1\ cos\theta_i}{Z_2\ cos\theta_t + Z_1\ cos\theta_i} \right|^2 \quad , \tag{2.4}$$

where Z is the wave impedance of the material (subscripts 1 and 2 referring to the inbound and obstacle media, respectively) and θ is the angle with respect to the normal of the interface plane (subscripts i and t referring to the inbound and refracted wave, respectively). The final value of the reflectance is given by weighting in equations (2.3) and (2.4), according to the proportion of s and p polarized waves. The formula for a given medium's wave impedance is given by

$$Z = \sqrt{\frac{j\omega\mu}{\sigma + j\omega\epsilon}} \quad , \tag{2.5}$$

where ω is the angular frequency of the wave ($\omega = 2\pi f$), and μ, σ, and ϵ are the medium's magnetic permeability, conductivity, and electric permittivity, respectively (although these three medium properties are also often a function of ω). For mmWave transmissions, the experimental measurements determined that many indoor and outdoor objects have a reflectivity exceeding 0.7 [25], due to the higher impedance mismatch at those frequencies. This value suggests that NLOS paths due to reflections are a viable solution, which is not the case for UHF transmissions.

Let's now focus our attention on the other element of the pair, the transmitted wave, which will be attenuated as it traverses an obstacle. The attenuation distance Δ at which the amplitude of an electromagnetic wave falls by $1/e$ as it propagates through a medium is given by [26]

$$\Delta = \frac{-1}{\mathrm{Im}\left\{k_0\ \sqrt{\epsilon_r}\right\}} \quad , \tag{2.6}$$

where k_0 is the wave number $(2\pi/\lambda)$ in free space and ϵ_r is the relative electric permittivity. The relative electric permittivity is given by

$$\epsilon = \epsilon_0\ \epsilon_r \quad , \tag{2.7}$$

where ϵ_0 is the dielectric permittivity of free space. The interpretation of equation (2.6) is dependent on the complex value of ϵ_r, defined by the medium, and falls within a range delimited by two extremes. On one end of that range, we have dielectric materials, where the real component of ϵ_r outweighs its imaginary component. For dielectrics, Δ does not depend directly on the transmission frequency – it is dictated by the material's electromagnetic properties, which may change with the selected frequency. Manipulating equation (2.6) so as to be expressed as an attenuation (in dB/m), we obtain for the dielectric case [26]

$$A_{dielectric} = 1636 \frac{\sigma}{\sqrt{\mathrm{Re}\left\{\epsilon_r\right\}}} \quad . \tag{2.8}$$

On the other end of the range, we have conductors, where the imaginary component of ϵ_r is now the dominant part. For conductors, the simplification of equation (2.6) is often known as *skin depth*, and Δ scales with the inverse of the square root of the frequency. The attenuation (in dB/m) for the conductor case can be written as [26]

$$A_{conductor} = 545.8 \sqrt{\sigma\ f_{GHz}} \quad . \tag{2.9}$$

From this analysis it stems that the increased frequency of mmWaves results in significantly increased penetration losses for most materials. In practical terms, most visible obstacles will now block direct communication paths, which will also result in isolated indoor and outdoor communication environments [25].

A third phenomenon that is often observed in wireless communication systems is *diffraction*. Diffraction is the propagation of radio signals around an object and is the mechanism that traditionally supports communications when a mobile wireless device turns the corner and moves from LOS to NLOS. It is described by the Huygens-Fresnel principle, which explains how penumbra regions (transition from light to shadow) behind objects are formed. Although the exact loss profile and, therefore, shape of this penumbra region varies significantly with the shape of the obstacle [27], its size grows with the wavelength. This means that even though it is often relied upon in UHF systems [20], diffraction becomes very lossy with movements of just a few centimeters at frequencies in the mmWave bands [5], and cannot be relied upon for mmWave propagation.

The last phenomenon covered by this section is *scattering*, due to diffuse reflections. Scattering at mmWave frequencies is a non-negligible propagation mechanism, since physical objects such as walls, people, and lampposts are significantly larger than a wavelength. As such, the surface of many of those objects becomes rough when compared to the size of the wavelength, resulting in illuminated scatterers that are able to serve mobile devices in their vicinity. Although the scattered power is modeled to decay proportionally to d^4 [5] (as opposed to d^2 in the free space model), it has been shown that scattering can generate reliable NLOS links at mmWave frequencies [4].

2.3 Antennas and Beamforming

The use of higher frequencies, as elaborated in section 2.1, allows for significant advantages over their lower frequency counterparts, as smaller wavelengths require proportionally smaller antennas. One of those advantages is straightforward, as smaller antennas can be mounted on smaller devices. However, there is a far more consequential repercussion of using smaller antennas: antenna arrays can now be used in devices where typically only a single antenna would fit. The use of big antenna arrays, also called massive MIMO, unlocks several otherwise impossible signal processing techniques. One of those techniques, which will be extensively used throughout this Thesis, is beamforming. By varying the amplitude and/or phase on each individual antenna, it is possible not only to focus the transmitted signal, which is equivalent (in the far field) to having a single higher directivity antenna, but also to electrically steer it. This technique can also be applied at the receiver, by mixing the incoming radiation at the multiple receiving antennas appropriately.

Beamforming can thus be used to counter the potential issues referred throughout this Chapter, extending the possible transmission range and allowing to aim the electromagnetic signal. In fact, with meticulous planning, BF can both boost the received power at the desired target while minimizing the interference at the remaining users, enhancing the area spectral efficiency through spatial multiplexing. However, the use of BF also requires greater planning when deploying the BSs. While it does increase the transmission range, it is difficult to predict whether the antenna can aim the steered beam towards every desired location – phenomena like scattering or diffraction are very hard to reliably estimate.

Beamforming is intrinsically a spatial focusing operation, using a group of radiators to gather or emanate energy through a specific direction over its aperture. The result is an increased received/transmitted gain at the expense of a reduced covered space or, in other words, an increased directionality. There are three different approaches to achieve BF in an array with N antenna elements [8]:

- Analog BF: a set of N phase and/or amplitude weights are applied in the Radio Frequency (RF) domain, one per receive/transmit antenna, using a single RF chain[1];
- Digital BF: the beamforming weights are applied in the digital domain, requiring N RF chains (one per antenna element);
- Hybrid BF: the beamforming weights are applied in both the digital and RF domains, having $1 \leq n \leq N$ RF chains, each chain containing N phase/amplitude weights.

The analog BF, while having the lowest hardware requirements, has its transmissions limited to a single data stream at a time. A RF chain is required for each data stream, and so, if analog BF is used at a BS, spatial multiplexing becomes impossible to achieve in a traditional telecommunication system. On the other hand, through the possession of N RF chains, an antenna with digital BF is capable of transmitting up to N data streams. Nevertheless, a large number of RF chains is hard and expensive to implement, and results in a considerable amount of power dissipation. As a result, being a compromise between hardware requirements and spatial-multiplexing capabilities, the hybrid BF scheme is the most consensual among theoretical studies and practical implementations [8].

Regardless of the selected approach to achieve BF, a communication system must have a method to select the appropriate set of BF weights for each desired data stream, in order to maximize the its information transmitting capabilities. While a significant

[1] An RF chain includes (but is not limited to) analog-to-digital and digital-to-analog converters, power and low-noise amplifiers, mixers, etc.

amount of publications have investigated the possibility of a downlink beam selection based on partial or complete channel knowledge [5][28], most of the practical implementations do a beam search over a limited codebook [9, 29, 30], consisting of a predefined set of beam patterns.

Selecting the BF weights based on channel knowledge allows for optimal system performance [5]. Unfortunately, such methods require a complete or partial estimation of the channel matrix H (CSI), whose size grows with the number of used antennas (at both ends, transmitter and receiver). Furthermore, if there is a high path loss between the transmitter and the receiver, it might not be possible to execute said method without actually performing some kind of beam search to boost the Signal-to-Noise Ratio (SNR) [31], further crippling these methods' applicability. Thus, the complexity for this type of beam selection method grows with channel estimation, which can quickly become overwhelming as the number of antennas increase.

The beam search over a limited codebook, while significantly easier to implement, has a significant overhead for probing the channel before establishing a communication. The beam search method must be able to run in under a millisecond for both the transmitter and the receiver, to meet the demanding 5G targets. As such, the beam search method must have a very small codebook in order to be effective, limiting the beamforming capabilities. For instance, the tests performed in [29] require about 10 ms to search over a codebook with 8 different beam patterns and 20 ms over a 16 element codebook, for a single user beamforming. If multi-user spatial multiplexing is desired, the complexity grows exponentially with the number of simultaneously connected users, and thus the existing methods are inadequate for 5G's goals.

The latency figures mentioned for the beam search case, while obtained from a prototype, clearly indicate that big codebooks would be impractical with the existing search methods. In fact, the work in [32] argues that applying brute-force search defeats the point of massive MIMO mmWave systems. As the search time would increase linearly with the size of the codebook, and the codebook size would have to increase exponentially with the number of antennas to harness their benefits, it would be impossible to search over a massive MIMO codebook in practical time - by the time an answer is found, there is no guarantee that the channel properties have remained the same. In other words, it would be impossible to preserve the assumption of channel coherence throughout the complete duration of the procedure [32].

The simplicity of beam search also appears to be its downfall. Intuitively, with nothing to guide the search, blindly pointing radiation into unknown space and hoping to hit the target gets more time consuming as the transmitted cone of propagation narrows down. However, the search doesn't necessarily have to be blind. The properties of the received radiation in a point in space depend on the obstacles found between the transmitter and that point, and the behavior of that interaction with the obstacles has been subject of several studies - it is in fact predictable, as described in section 2.2. To face the beam search challenge, and to guide the process, the work developed in the following chapter leverages the aforementioned mmWave propagation properties.

2.4 Summary

This Chapter goes through the fundamentals of electromagnetic radiation propagation for mmWave transmissions, and how the standard assumptions for UHF propagation are no longer correct. The new propagation environment is now filled with a new set of challenges, which also opens the door for new techniques, such as this Thesis'.

In section 2.1, the importance of directive transmissions for mmWaves becomes instantly evident, due to the well known Friis free space loss equation. Even with directive transmissions, these higher frequencies are riddled with non-negligible sources of additional attenuation, in particular due to weather effects.

An inevitable consequence of adopting mmWaves transmissions is the emergence of complex phenomena that arise from obstacle interaction, as seen in section 2.2. Out of these phenomena, one stands out: most obstacles now reflect the majority of inbound radiation, which simultaneously block LOS transmissions and give rise to new reflection-based communication paths.

Although the new conditions are far more challenging, a large array of telecommunication-based applications stand to gain from mmWaves. Section 2.3 describes the most straightforward benefit: new antenna technologies. Using a smaller wavelength requires using proportionally smaller antennas, which equate to smaller form factors or, alternatively, larger antenna arrays. Larger arrays, in turn, unlock the use of beamforming, a powerful signal processing technique that allow us to dynamically steer and direct a transmission using a system with no moving parts. As with mmWaves, beamforming suffers from a 'no free lunch' effect - the potential benefits are massive, but hard to harness.

The remaining of this Thesis builds on the information laid out in this Chapter, particularly with respect to reflected transmissions and the use of beamforming, in the context of mmWaves. The following Chapter concerns the use of this information to build a new BF beam selection method given the position of the user, while the remaining Chapters focus on the more interesting inverse problem.

3

Fast Beamforming Selection

Contents

For 5G BSs, which are expected to be located in elevated positions in urban scenarios, most of the obstacles found during transmission are static for a significant amount of time, as they are primarily buildings. Therefore, given the properties of mmWaves, it is reasonable to assume that a receiver in a given position will predominantly measure a constant average received power for a specific BS beamformed transmission. Once found, the set of BF configurations that is most successfully able to communicate with a given position is expected to contain at least one viable solution until a significant change in the surrounding environment occurs (e.g. a new building).

This chapter proposes a technique that narrows down the BF beam search procedure, introduced in Section 2.3, through the suggestion of likely viable solutions to the problem at the receiver's position. With the search space reduction, larger codebooks are enabled, which combined with a proper design (e.g. [33]) can lead to simple, yet powerful, solutions. The original idea, which culminated in the paper in [10], stemmed from a suggestion in [5], where the authors claim that a significant part of large-scale properties of mmWave transmissions are static in urban environments, to the point that accurate ray-tracing simulations should be able to approximate them. It was the first paper to suggest the use of positioning together with those static properties, back in 2017 – by then, it was faced with some opposition, especially because the paper in [32] has strong arguments against the scalability of vanilla codebook-based beam search methods, as mentioned in Section 2.3. Regardless of that, the practical benefits of beam search methods far outweigh their downsides: as of 2020, the first commercial mmWave modems are being built with beam tracking techniques (e.g. [34]), and the research in this field is still quite active (e.g [35]).

Outdoor mmWave experiments are very expensive, and thus the results in this Thesis were assessed at simulation level. The exact description for electromagnetic radiation is contained in Maxwell's equations, resulting in the propagation phenomena described before. However, being a set of partial differential equations, the complexity associated with simulating real-life scenarios would be quite high. Since the mmWave radiation will reflect on most visible obstacles, it is possible to establish a parallelism with light propagation, and thus apply ray-tracing techniques [36–38]. By assigning each ray with power and phase values, it becomes possible to create a map with the power level for each position, resulting from all constructive and destructive ray interference. This Chapter aims to demonstrate not only the usefulness of the proposed beam selection method, but also the validity of using ray-tracing techniques on accurate 3D maps for mmWave simulations.

3.1 System Model and Problem Formulation

Let us consider a system containing a single stationary transmitter, with an antenna array made up of N_S elements, and a set of N_M mobile receivers, totaling N_R receiving antennas. In the frequency domain, the signal at the receiver devices, $\mathbf{r} \in \mathbb{C}^{N_M \times 1}$, can be written as

$$\mathbf{r} = \mathbf{WHFs} + \mathbf{Wz}, \tag{3.1}$$

where $\mathbf{W} \in \mathbb{C}^{N_M \times N_R}$ and $\mathbf{F} \in \mathbb{C}^{N_S \times N_M}$ are the beamforming matrices for the receivers and transmitter, respectively, $\mathbf{H} \in \mathbb{C}^{N_R \times N_S}$ is the channel matrix, $\mathbf{s} \in \mathbb{C}^{N_M \times 1}$ are the transmitted signals, and $\mathbf{z} \in \mathbb{C}^{N_R \times 1}$ is the noise. Ultimately, to achieve multi-user spatial-multiplexing, the diagonal elements of $\mathbf{WHF}$ should be maximized, while keeping the non-diagonal elements (unwanted interference) in check.

When the information regarding the receivers' antenna arrays is unavailable or inconvenient to obtain, such as in the establishment of a new connection, the system is unable to cooperate so as to jointly optimize both beamforming matrices. In such scenarios, the dimensions of $\mathbf{W}$ can be unknown, as the transmitter may not know N_R. As result, the best option left for the transmitter is to maximize the desired signals at the receivers' locations and, in that case, $\mathbf{W}$ is an N_M by N_M identity matrix ($\mathbf{I}$). Under those circumstances, equation 3.1 can be simplified to $\mathbf{r} = \mathbf{HFs} + \mathbf{z}$, where each element of the $\mathbf{HF}$ product corresponds to the j-th signal at the position of the i-th user.

Since $\mathbf{H}$ is sparse in the angular domain [39], it is then heavily influenced by the propagation path. In other words, only a few propagation angles can result in successful communications. The method introduced in this chapter indirectly estimates the channel matrix ($\hat{\mathbf{H}}$) through the measurement of its past instances, which should match the angular sparseness of $\mathbf{H}$ if the obstacle layout remains static. Through this method, $\hat{\mathbf{H}}$ has two major sources of unpredictable distortions: (*i*) incorrect device position estimation and (*ii*) obstacle layout changes (e.g. mobile objects).

When beamforming is defined by a finite codebook $\mathbf{C}_{Tx}$ (containing B_{Tx} entries), each column of $\mathbf{F}$, $\mathbf{F}_i$, must take the form of a codebook entry. We can thus formulate our codebook-based constrained optimization problem as follows:

$$
\begin{aligned}
\underset{\mathbf{F}}{\text{maximize}} \quad & \operatorname{trace}(|\hat{\mathbf{H}}\mathbf{F}|) \\
\text{subject to:} \quad & \mathbf{F}_i \in \mathbf{C}_{Tx}, \\
& |\hat{\mathbf{H}}\mathbf{F}|_{i,j} < I_{th},\ i \neq j, \\
& |\hat{\mathbf{H}}\mathbf{F}|_{i,j} \geq P_{th},\ i = j,
\end{aligned}
\tag{3.2}
$$

where I_{th} is the maximum allowable interference and P_{th} is the value corresponding to the minimum acceptable power for the communication. Note that this problem formulation is written in terms of the amplitude of the channel modified by the transmitter beamforming ($|\hat{\mathbf{H}}\mathbf{F}|$), but it is also valid in terms of its power ($|\hat{\mathbf{H}}\mathbf{F}|^2$) or any equivalent physical quantity.

3.2 Proposed Beam Search Method

The core of the proposed beam search method resides in keeping an estimate of the channel for all B_{Tx} propagation patterns in codebook $\mathbf{C}_{Tx}$, in a discrete spatial grid, henceforth called matrix $\mathbf{P}$. This discrete spatial grid represents the multiple positions that are served by a BS, and that can be occupied by a user. $\mathbf{P}$ is a physical quantity that represents the estimated channel for a specific transmitted beamforming pattern at a given position, and thus it can be formally described as

$$\mathbf{P} \propto \hat{\mathbf{H}}\mathbf{F}. \tag{3.3}$$

Since the used beam patterns are directive, each location ends up having just a few codebook entries above the minimum power threshold P_{th}. Therefore, in the data preprocessing stage, table $\mathbf{B}$, containing the S most suitable codebook entries for each location, is generated. Since the key task of the beam search method is to obtain the best codebook index for a given connection, the table entries for each location should be sorted (in descending order) according to the received power, aiming to minimize the required search. During the beam search process, feedback regarding the tested beam pattern suitability is obtained. Through that feedback, $\mathbf{P}$ can be updated using a moving average, allowing the proposed system to gradually cope with changes in the surrounding area. Additionally, if the BS has access to a 3D model of its surrounding area, it can bootstrap the initial estimate of $\mathbf{P}$ through ray-tracing, as depicted in fig. 3.1.

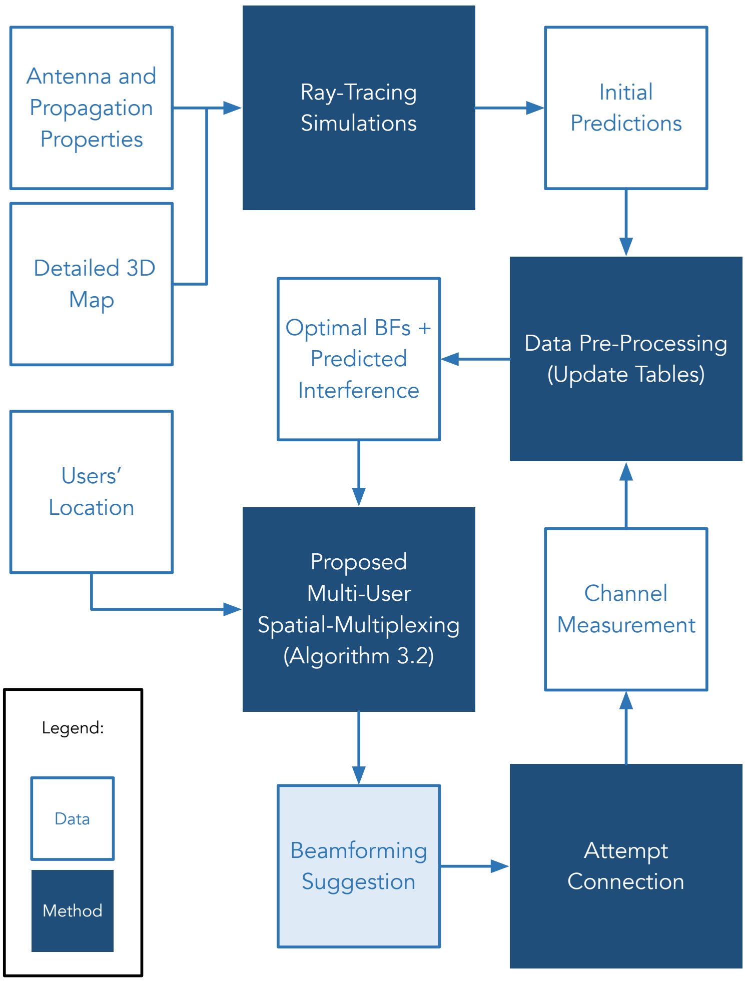

Antenna and Propagation Properties
Ray-Tracing Simulations
Initial Predictions
Detailed 3D Map
Optimal BFs + Predicted Interference
Data Pre-Processing (Update Tables)
Users' Location
Proposed Multi-User Spatial-Multiplexing (Algorithm 3.2)
Channel Measurement
Legend:
Data
Method
Beamforming Suggestion
Attempt Connection

Figure 3.1: Relationship between the used data and methods, resulting in the beamforming suggestions. This diagram includes how can ray-tracing simulations be integrated to bootstrap the system. Note that in the presence of a single user, algorithm 3.2 can be replaced by algorithm 3.1.

In case a single user per sub-channel is wanted, there is no spatial multiplexing involved, so the system doesn't need to worry about interference from its own signals. As such, the system just needs to scan table **B** until a match is found, in a process depicted in algorithm 3.1. However, in order to achieve spatial multiplexing, the transmitter must be confident that the radiation aimed at a target user does not cause too much interference to the remaining connected users, especially when probing a channel already in use. As a result, achieving spatial multiplexing is far more difficult than to perform BF for a single user, and the complexity grows quadratically with the number of connected users.

To select a codebook entry for a new user, two separate tasks must be executed: *(i)* assessing which beams might be capable of establishing a connection with the newcomer,

Algorithm 3.1 Single User Beamforming

```
$B_{Tx}$ = number of entries in the used codebook, $\mathbf{C}_{Tx}$;
$S$ = # of top codebook entries to be stored (per position);
$L$ = spatial grid size;
$U$ = # of users attempting to connect;

Load the channel estimate matrix $\mathbf{P}$, with $B_{Tx}$ by $L$ elements;
Load the sorted optimal beamforming matrix $\mathbf{B}$, with $L$ by $S$ elements;

for $u \leftarrow 0$ to $U - 1$ do
  Receive the location of user $u$;
  Convert the received location into the correspondent grid position $l$;
  for $s \leftarrow 0$ to $S - 1$ do
    if $B[l][s]$ is empty do
      Use other connection method (e.g. brute-force search);
    else do
      Attempt to connect user $u$ using the codebook entry $B[l][s]$;
      Update $\mathbf{P}$ with the result of the attempt, break loop if the connection is successful;
    end if
  end for
end for
```

knowing the existing interferences that it must be able to endure, and *(ii)* estimating the additional interference created by the new connection on the previously connected users. The connection can only be established after verifying that all the interference levels are below the target threshold.

The optimization formulation on section 3.1 covers all the aforementioned concerns. However, since the execution time of the whole beam search is critical and searching through all possible combinations of **F** would quickly exceed the timing requirements, as the number of users increases, an heuristic was designed. While adding sequentially new users, the system will expect that the existing users have locked their beamforming configuration, drastically reducing the required search space while keeping a near-optimal solution. The problem formulation for this heuristic can be formally written as in equation 3.4, and it is further described in algorithm 3.2.

$$\begin{aligned}
\underset{\mathbf{F}}{\text{maximize}} \quad & \text{trace}(|\hat{\mathbf{H}}(k)\mathbf{F}|) \\
\text{subject to:} \quad & \mathbf{F}_i \in \mathbf{C}_{Tx} \text{ if } i = k, \\
& \mathbf{F}_i = \mathbf{F}_i(k-1) \text{ if } i < k, k > 0 \\
& |\hat{\mathbf{H}}(k)\mathbf{F}|_{i,j} < I_{th},\ i \neq j, \\
& |\hat{\mathbf{H}}(k)\mathbf{F}|_{i,j} \geq P_{th},\ i = j,
\end{aligned} \tag{3.4}$$

Considering that the data pre-processing step sorted the codebook entries by the resulting received power (and that they keep being sorted as **P** is updated), the system

Algorithm 3.2 Multi-User Spatial Multiplexing

```
$B_{Tx}$ = number of entries in the used codebook, $\mathbf{C}_{Tx}$;
$I_{th}$ = maximum interference threshold;
$S$ = # of top codebook entries to be stored (per position);
$L$ = spatial grid size;
$U$ = # of users attempting to connect;

Load the channel estimate matrix $\mathbf{P}$, with $B_{Tx}$ by $L$ elements;
Load the sorted optimal beamforming matrix $\mathbf{B}$, with $L$ by $S$ elements;
Initialize table $T$ with the connected users' location and the respective codebook entry selected;

for $u \leftarrow 0$ to $U-1$ do
  Receive the location of user $u$;
  Convert the received location into the correspondent grid position $l$;
  for $s \leftarrow 0$ to $S-1$ do
    if $B[l][s]$ is empty do
      Break loop;
    else if $B[l][s] \in T$ do
      Skip this codebook entry;
    end if
    $I_{est} = max$ (estimated interference on users $\in \{T \cup u\}$) ;
    if $I_{est} \leq I_{th}$ do
      Attempt to connect user $u$ using the codebook entry $B[l][s]$;
      Update $\mathbf{P}$ with the result of the attempt, including the measured interferences;
      Add $B[l][s]$ to $T$ and break loop if the connection is successful;
    end if
  end for
  if user $u$ not connected do
    Repeat the process in another sub-band or use another connection method;
  end if
end for
```

can simply check the problem restrictions sequentially using the sorted entries, until it finds a suitable one[1]. This method, formally depicted in algorithm 3.2, can quickly find a solution for the aforementioned problem, provided that the number of saved codebook entries per location is large enough to include a viable combination (if it exists).

Since obtaining a small amount of data from the mobile user (*e.g.* its Global Positioning System (GPS) location) adds a very small delay [29], the resulting latency will be mostly due to the algorithm's execution in the BS. With a low execution time, the proposed method with the added heuristic is thus able to quickly produce a set of beamforming suggestions that might be tried off before the system resorts to heavier algorithms. Therefore, adding this method to a working beam search system should reduce its latency significantly and also lead to energy savings.

[1] As mentioned before, each element of the **HF** product corresponds to the j-th signal at the position of the i-th user, and thus the interferences can also be estimated through **P**.

3.3 Apparatus and Ray-Tracing Accuracy

Before addressing the results of the proposed algorithm, a major assumption must be verified regarding the reliability of ray-tracing simulations. To do so, we must first introduce ray-tracing algorithms, in the context of telecommunications. As the name suggests, it is a class of methods that is based on the actual *ray-tracing* techniques used to simulate light propagation in an environment with light sources, as well as reflective and absorbing surfaces. Referring back to section 2.2, we can see that they can capture all relevant obstacle interactions. The ray-tracing software packages used in telecommunications, such as the one used throughout this Thesis, add the excess free space loss to the ray propagation simulation, and have the ability to model directive propagation. Therefore, if the obstacle's 3D model used for simulations is precise, the outcome should be similar to the actual propagation in the modeled region.

To validate the assumptions written above, a recreation of the experimental results obtained in [40] is devised. In [40], directive horn antennas are used to measure the propagation characteristics of the 28 GHz frequency, near the New York University (NYU) campus. By aiming the transmitter in the direction with the strongest link and by physically rotating the receiver (both in the azimuth plane and in altitude), the authors in [40] were able to assess whether it was possible to establish a communication link with those transmitter-receiver location pairs.

Using the high precision open-source 3D map made available by the New York City Department of Information Technology & Telecommunications [41], as well as the specifications depicted in table 3.1, the measurements were recreated using Wireless InSite 3.0.0.1 [42]. To compute the received power, the rotating horn antenna receivers were swapped by isotropic receivers which, aside from the antenna gain, should yield similar

Table 3.1: Simulation Specifications

Parameter Name	Value
Carrier Frequency	28 GHz
Transmit Power	30 dBm
Max. Tx. Gain	24.5 dBi (horn antenna)
Half-Power Beam Width (HPBW)	10.9°
Downtilt	10°
Codebook Size	16 (150° arc with 10° between entries)
Saved BF	4 (per receiver location)
Receiver Grid Size	160801 (400 × 400 m, 1 m between receiver)
# of Executions	10^6

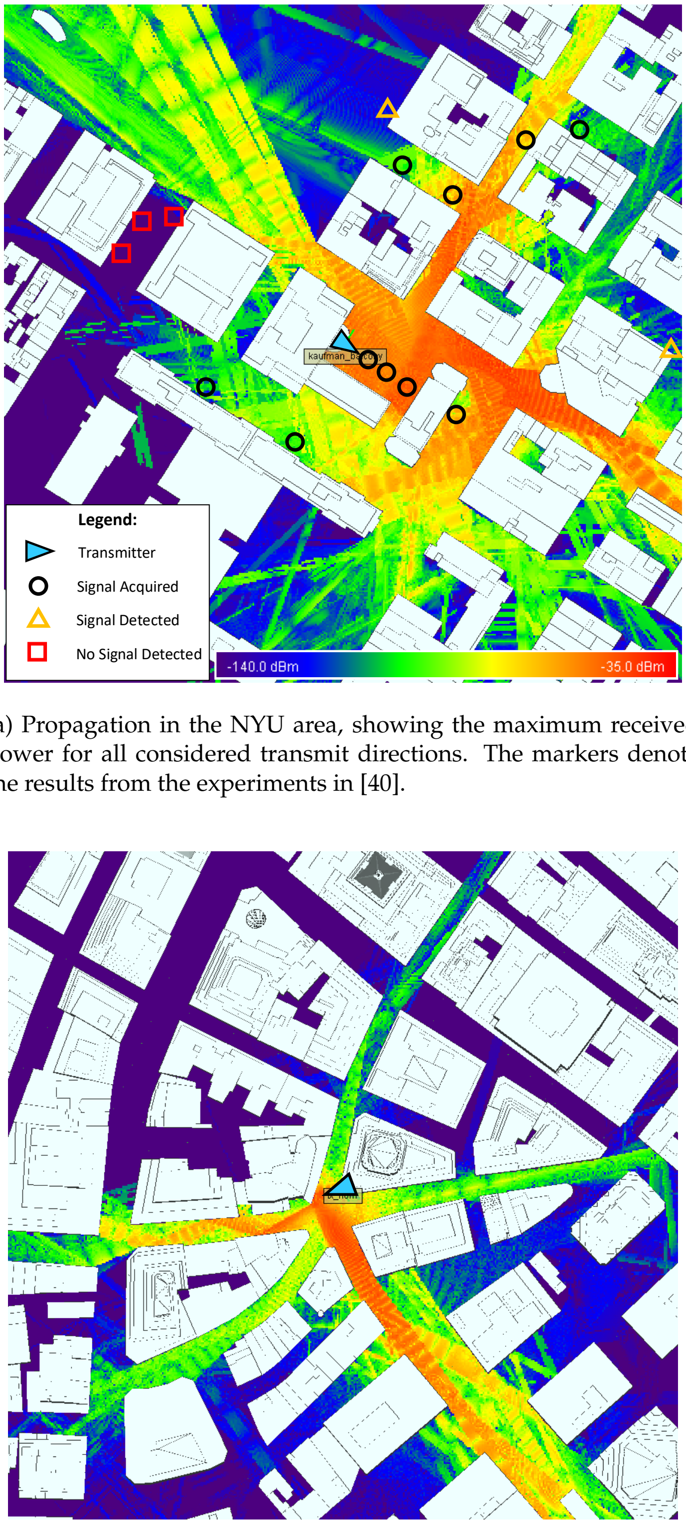
results. In the actual field measurements, using horn antennas as receivers, successfully received or detected signals correspond to a maximum path loss of 168 dB or 178 dB, respectively. Considering the aforementioned parameters, this corresponds to a minimum received power of -113.5 dBm or -123.5 dBm at the simulated isotropic receivers.

The recreation result can be observed in Fig. 3.2a, where the maximum received power for all possible transmit directions is shown. In the results shown, a grid of isotropic receivers was placed on the map, 1 m above the ground, with 1 m of separation between antennas. The colored markings represent the measurement locations in [40], and their respective results. As it can be observed inside the markings, the simulated received power matches the experimental measurements. Furthermore, similar results were also obtained in [43] (using a different ray-tracing software), and thus validating the proposed assumptions.

3.4 Simulation Results

Since the dynamic changes in the obstacles are hard to model, this results section focuses on the feasibility of the algorithm, as well as its execution time. Unfortunately, the used ray-tracing software doesn't fully support MIMO antennas with beamforming, and thus a physically rotating horn antenna (with the specifications depicted in table 3.1) is used as transmitter. Even though it is an imperfect representation, the used specifications are very similar to practical MIMO antennas - the transmitter's physical specifications were inherited from the antenna used in the experiments in [40], while the codebook size is inherited from [29]. The results described throughout this section were obtained using an Intel i7 3820 Central Processing Unit (CPU) and single-threaded simulation code.

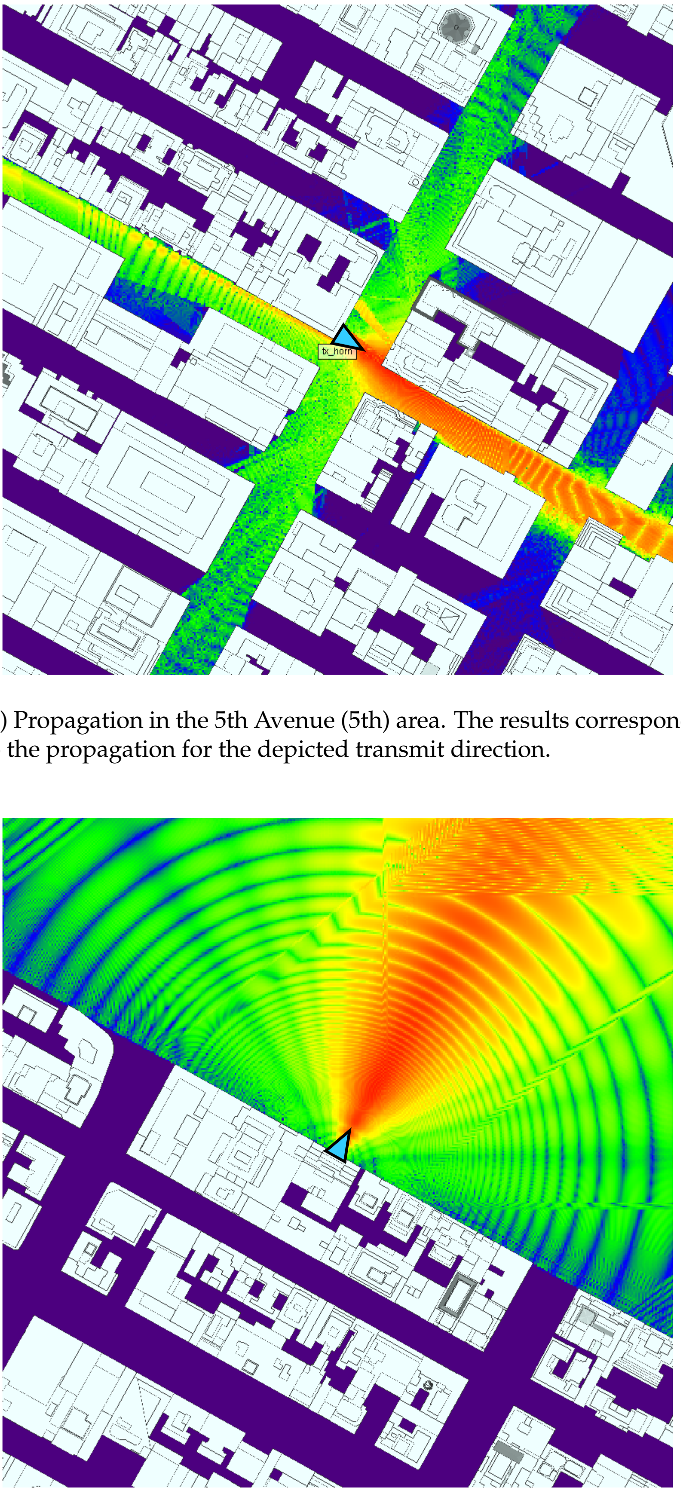
By predicting the propagation for each entry in the codebook through $\mathbf{P}$, data preprocessing can then be executed, resulting in a very compact data structure containing the most suitable codebook entries for each location ($\mathbf{B}$). With the aforementioned data structure and the realistic simulation specifications presented in table 3.1, the average execution time for a system with a single randomly positioned user running algorithm 3.1 was under 51 ns. Considering the obtained results, the proposed single user beamforming method's bottleneck would certainly be time required to attempt the connection given the codebook entry. The results shown do not change with the minimum received power threshold nor with the user locations.



(a) Propagation in the NYU area, showing the maximum received power for all considered transmit directions. The markers denote the results from the experiments in [40].

(b) Propagation in the Wall Street (WSt) area. The results correspond to the propagation for the depicted transmit direction.

Figure 3.2: Ray Tracing simulations in New York City, where it is observable the dependency on the layout of the area. The power scale ranges from -140 to -35 dBm.



(a) Propagation in the 5th Avenue (5th) area. The results correspond to the propagation for the depicted transmit direction.

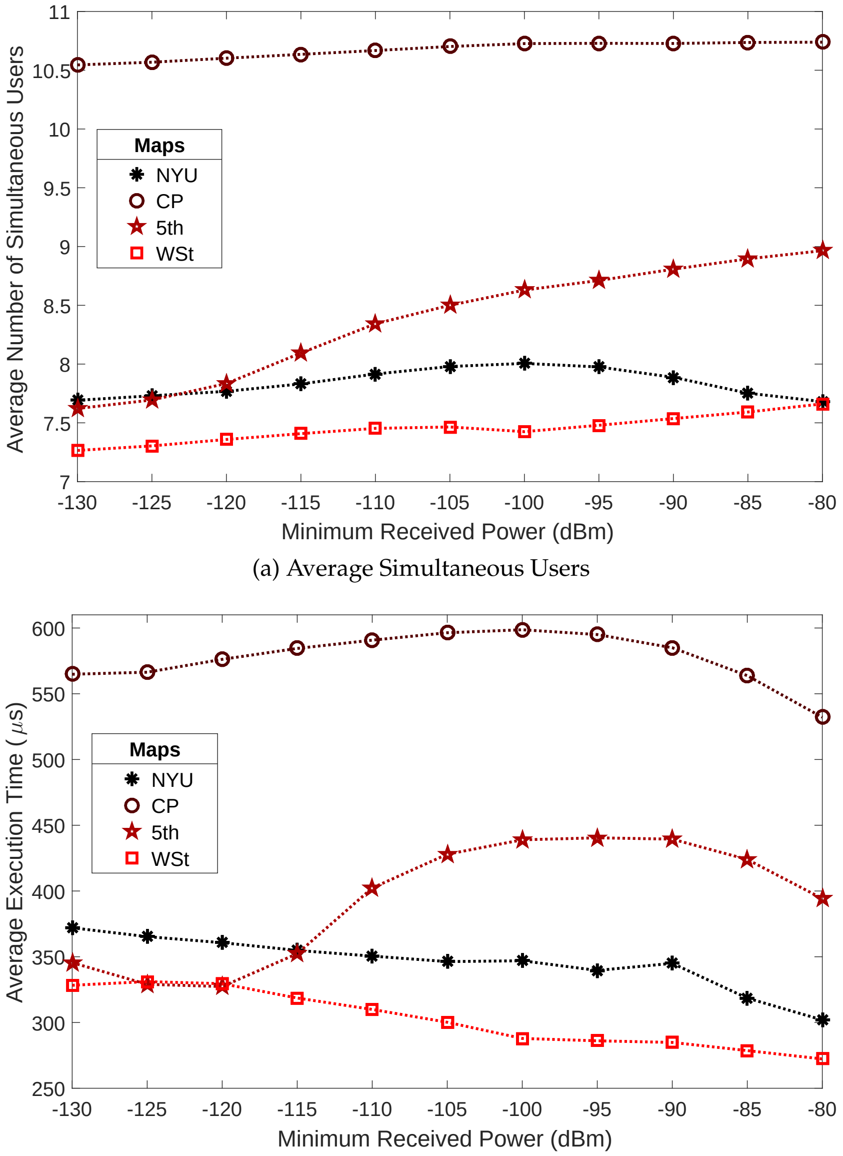
(b) Propagation in the Central Park (CP) area. The results correspond to the propagation for the depicted transmit direction.

Figure 3.3: Continuation of the previous figure, with Ray Tracing simulations for another two distinct zones in New York City.

When multi-user spatial multiplexing is preferred to single user beamforming, the algorithm's complexity increases quadratically with each subsequent user. Besides fetching the optimal codebook entries for each new user, the interference for the multiple users must now be computed, which also requires the full propagation data (contained in $\mathbf{P}$). As the number of connected users increases, not only the interference computing step gets more demanding, but also the likelihood that the new user is able to meet the interference threshold with its most suitable codebook entries reduces. As result, the execution time quickly increases with each simultaneously connected user. To mitigate that increasingly difficult interference computing step, algorithm 3.2 stops checking for the connected users' interferences as soon as it finds one user above the interference threshold, quickly discarding the potential user from the considered frequency sub-band if all S BF suggestions fail. This discarding step can be safely made under the assumption that there are enough available sub-bands, which can also be assessed in parallel so as to minimize the latency.

Considering the parameters in table 3.1, the average results for 100 randomly positioned users attempting to connect are shown in figs. 3.4, 3.5, and 3.6. These figures show two performance metrics for the proposed algorithm, the average number of achievable simultaneous connections and the average time it took to find the codebook entries that enable those simultaneous users, as the minimum received threshold shifts, for three maximum interference levels (-5, -10, and -15 dB, respectively). To eliminate any bias due to a specific obstacle layout, four different city zones were considered, as shown in figs. 3.2 and 3.3.

As previously mentioned, it is clear in those figures that the execution time increases considerably with the number of successful connections, jumping from 200μs for 5 simultaneous users (fig. 3.6) to almost 600μs for 10 simultaneous users (fig. 3.4). It is also possible to observe that, regarding the number of simultaneous connections, the maximum allowed interference has significantly more impact than the minimum received power threshold. In fact, increasing the minimum received power threshold often has a positive effect on the number of simultaneous users, as accepting low power connection makes the system very sensible to interference from other connections. The optimal maximum interference threshold depends on the Signal-to-Noise-plus-Interference Ratio (SNIR) targets for the system, but increasing its value from -15 to -5 dB makes the simulation system able to handle twice as many simultaneous users.



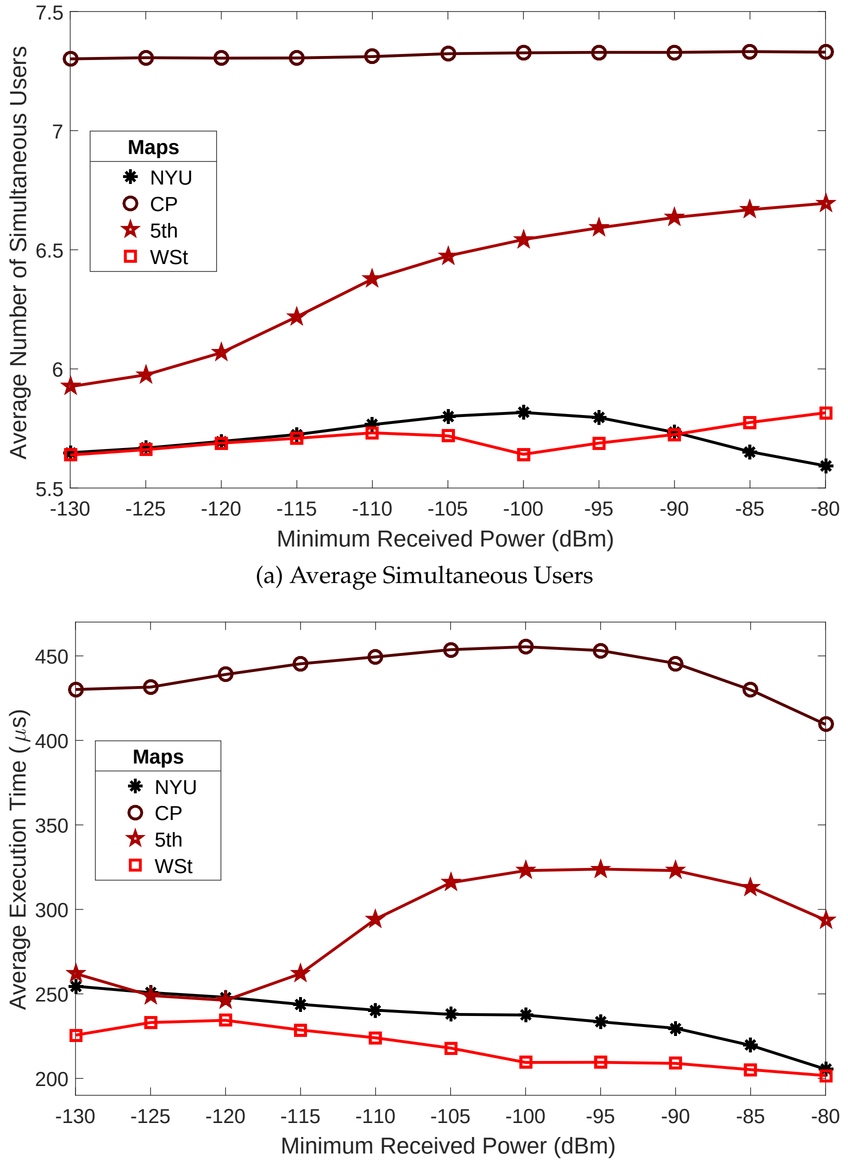
(a) Average Simultaneous Users

(b) Average Execution Time

Figure 3.4: Simulation results for 100 users attempting to connect, with a maximum interference of **-5 dB**. The maps in the figures' legend correspond to the areas shown in figs. 3.2 and 3.3.

Most zones had similar results, except for Central Park (the only zone consisting of mostly open space), which consistently allowed more simultaneously connected users. That increased multi-user capability can be explained due to the used codebook - the BF signals used in the experiments are sent with minimal initial overlap, due to their central direction and HPBW. Therefore, if no obstacles are to be found, the codebook entries are guaranteed to keep their high degree of separability, explaining the obtained results.

The simulations shown in this section do not consider BF at the receiver, which could be obtained from the received signal's Angle-of-Arrival (AoA). With BF at the receiver,

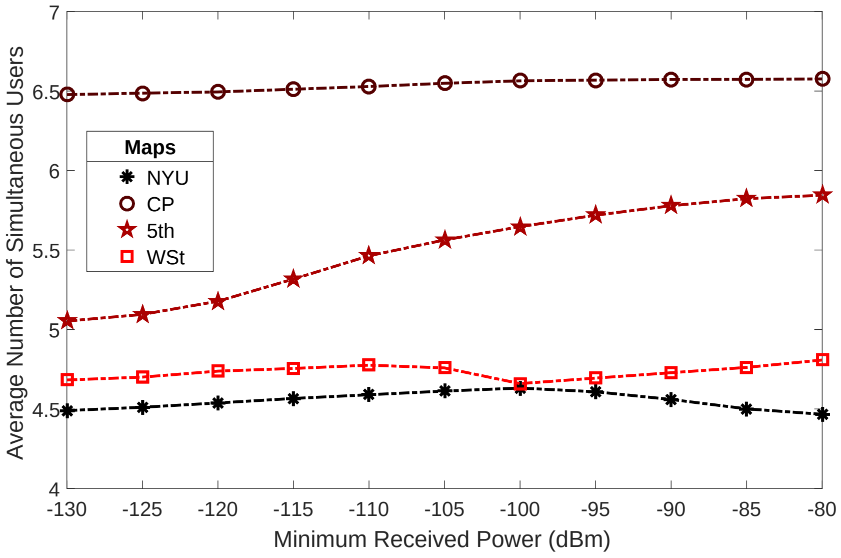


(a) Average Simultaneous Users

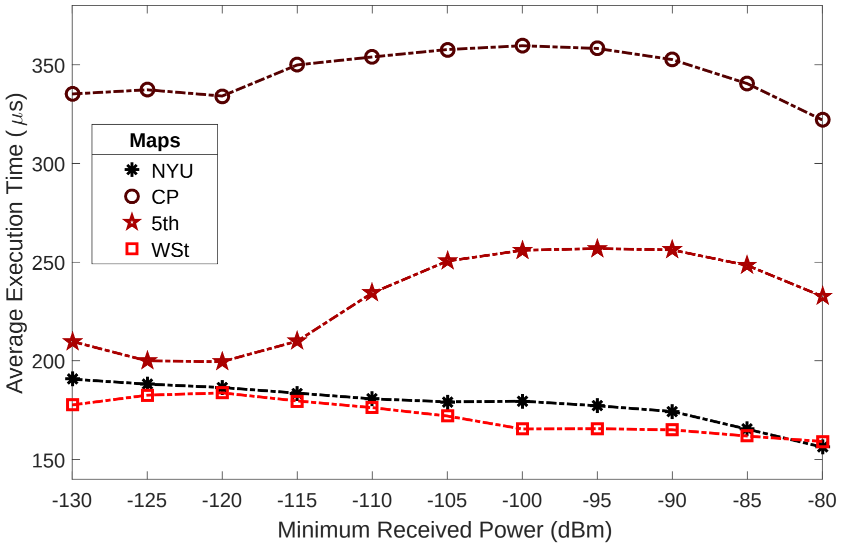
(b) Average Execution Time

Figure 3.5: Simulation results for 100 users attempting to connect, with a maximum interference of **-10 dB**. The maps in the figures' legend correspond to the areas shown in figs. 3.2 and 3.3.

the receiver would not only have a higher SNR, but also lower interference levels, since the AoA from the multiple interference sources would likely be different. Nevertheless, even for a low interference threshold, the multi-user spatial multiplexing algorithm had an average of at least 4.5 simultaneous users for the used parameters. In every situation, the average execution time for the 100 users attempting to connect did not exceed 600μs. While the required execution time is far higher than for the single user beamforming, it is still able to comply with the strict 5G requirements of a few milliseconds (given that the



(a) Average Simultaneous Users



(b) Average Execution Time

Figure 3.6: Simulation results for 100 users attempting to connect, with a maximum interference of **-15 dB**. The maps in the figures' legend correspond to the areas shown in figs. 3.2 and 3.3.

communications with the user take negligible time here).

To put the values above in perspective, consider the time required to suggest a single-user connection, 51 ns. This operation requires loading a single value from memory. If $|T|$ depicts the number of users already connected to the system, a multi-user spatial-multiplexing brute-force method with access to the power estimate tables would have to check its values $\mathbf{C}_{Tx} - |T|$ times to infer the possible signal power at the user's position, $|T|$ times to obtain the interference at its position, and $(\mathbf{C}_{Tx} - |T|) \times |T|$ to estimate the interference added by the new user to the connected users. In the same simulation environment, assuming an average of 5 simultaneously connected users (akin to fig. 3.6), the data loading operations alone would require more than 362 μs, or 81% more time than the results shown in fig. 3.6 for similar usage levels. Let's now consider the use of a larger codebook. Algorithm 3.2 shows that its execution does not depend on the codebook size. On the other hand, the brute-force approach would need to load and compare a much larger amount of data – for a 64-entry codebook, it would now require at least 1831 μs, or 816% more time than the proposed approach.

The achieved results show that the single user beamforming suggestions are formulated with negligible latency, while the multi-user spatial multiplexing method is able to run in under a millisecond. While these suggestions are prone to errors, the global result should yield a considerable gain, since they greatly reduce the required search space in order to obtain a suitable beamforming configuration. With the reduced search space, larger codebooks become possible, allowing for higher area spectral efficiency capabilities and paving the way for the next generation networks.

3.5 Summary

Chapter 2 ends with a question: although beamforming in the context of mmWave transmissions can deliver promising capabilities, how can we tame its latency and/or resource requirements? The work developed for this Chapter provides an answer to that question, based on the propagation fundamentals described before, as well as one external source of information: the device's position.

The validation of the proposed beam search method starts in section 3.1, where the system model and the problem formulation are described. To move forward with this formulation, one important assumption is made: the majority of new propagation paths arise from reflections on static obstacles (such as buildings), and thus a slow moving model of the signal power at each position can be built. In section 3.2, new methods that build upon that quasi-deterministic model are proposed, providing quick solutions when a BS employs a finite set of BF patterns. One of the methods concerns a simpler scenario where a user is alone in its communication channel, while the other enables beam selection such that multi-user spatial multiplexing becomes possible. Both methods share a critical trait: knowing the position of the users is essential to use the aforementioned model.

Having described the problem, the assumptions, and the solution, sections 3.3 and 3.4 confirm the validity of the solution with a simulation system. As the solution assumes a mmWave environment, a specialized ray-tracing software is employed to create the quasi-deterministic model. The results then show that the proposed methods can reliably determine suitable BF combinations for multiple users, well within the 1 ms latency goal established for 5G.

This concludes Part II of this Thesis, which follows a more traditional Telecommunications trajectory (*i.e.* related to transmitting more data using fewer resources). The development of the core concept behind this Part, solving beamforming from positioning, led to another interesting set of questions: is the reverse problem, obtaining the position from beamformed transmissions alone, possible to solve? If so, how efficient would a new solution be? The *de facto* approach to positioning is decades old, and technology has been evolving at a staggering pace. A better solution should be within grasp.

Part III

Positioning from Beamforming

4

Millimeter Wave Positioning Systems

Contents

Over the last decades, positioning systems have become increasingly pervasive, covering most of the planet's surface. Although they are accurate enough for a large number of uses, their precision, power consumption, or hardware requirements establish the limits for the adoption in new mobile devices.

Millimeter wave systems can achieve remarkable positioning accuracies due to the available bandwidth. However, as mentioned in the chapter 1, producing accurate estimates for NLOS positions represents a difficult task, limiting their practical applicability. This chapter provides an overview of the traditional positioning approaches, the state-of-the-art for mmWave positioning, and the system model for this Thesis' proposed positioning approach. The forthcoming Chapters then make use the devised system model to demonstrate its positioning capabilities.

4.1 Positioning Systems

Nowadays, an increasing number of tasks rely on GNSSs' precise localization capabilities for their operating success. The GPS, a GNSS, started in the 1970s, is based on a constellation of satellites broadcasting data frames at a very low rate of 50 bps to receivers on the Earth's surface. The signal is encoded using a pseudo-random sequence, unique to each satellite, transmitted at 1.023 million pulses (*chips*) per second, and each frame consists of 5 sub-frames of 300 bits each [44], as depicted below:

- Sub-frame 1: accurate timing information generated by the atomic clock embedded into the satellite itself;
- Sub-frames 2 and 3: precise orbital information used to compute the satellite's location, which remains valid for up to 4 hours (the *ephemeris*);
- Sub-frames 4 and 5: ionospheric conditions and the operating status of the whole system, typically updated every 24 hours (the *almanac*).

Depending on the existing information on the mobile device, the Time to First Fix (TTFF) of a stand-alone GNSS localization system can vary significantly. When the device lacks a valid almanac, then it must receive the full signal, consisting of 25 frames (12.5 minutes), also known as *cold start*. If the receiver was recently active, then it can perform a *warm start* by obtaining the ephemeris data, which takes up to 30 seconds. In optimal conditions, the receiver can acquire the GPS signal right away, returning a position estimate within a couple of seconds. This is known as *hot start*.

When the GPS was conceived, the system was designed for long periods of continuous navigation, having the TTFF relatively low importance. However, with the advent of heterogeneous mobile services, a low TTFF became vital for the user experience.

Since a mobile network BS can view the same satellites as a nearby mobile device, it has access to the desired satellites' time and orbital information, as well as to the device's coarse location. Thus, with A-GPS, the BS is capable of providing assistance to the device, minimizing its TTFF [44]. Depending on whether the position estimate is computed on the device or offloaded to the BS, A-GPS technologies can be classified as:

- Mobile Station Based (MSB) - the mobile device receives the ephemeris, almanac, time, and coarse location from the BS, enabling a hot start regardless of the starting conditions;
- Mobile Station Assisted (MSA) - the mobile device acquires raw satellites' signals and sends them to the BS, which computes and then returns the device location.

With MSB, the main energy cost comes from processing the GNSS signal after receiving the BS data. State-of-the-art low-power implementations claim requiring about 18 mJ per position fix when continuously tracking [45][46], with significant penalties when sporadic tracking is desired (e.g. [45] requires 504 mJ per fix when tracking the device once per minute).

MSA was proposed to avoid the sporadic tracking penalty, where the mobile device has to perform the costly signal synchronization with each visible satellite before an estimate [47]. With MSA, the mobile device just needs to capture and send a snapshot of the received GNSS signal. The duration of the captured signal must be a multiple of 1 ms, to ensure a coherent integration time. Considering the minimum sampling frequency of 2.046 MHz, each position fix requires transmitting at least 2046 bits. However, to obtain the desired GNSS positioning accuracy, most practical implementations capture the signal at more than 16 MHz, with durations exceeding 10 ms [48], requiring a transmission of hundreds of kilobits per position fix. This leads to significant energy costs, and thus MSB A-GPS approaches are often preferred over MSA.

Although A-GPS can solve the majority of problems associated with the start up latency, it still requires dedicated hardware whose average power consumption can easily exceed hundreds of mW [49]. To provide an alternative, multiple network-operated localization systems were considered in the past decade [11]. With Release 9 of Long Term Evolution (LTE) networks, the Observed-Time-Difference-of-Arrival (OTDoA) was introduced, possessing a theoretical achievable error similar to GNSS devices [50]. However, to achieve that error level, OTDoA has to operate under optimal conditions and has to employ expensive detection mechanisms, unfit for low-power Internet of Things (IoT) devices, as discussed in [51] (and further addressed in LTE Release 14). In practical scenarios, the average error fairly exceeds 20 m [52], and thus cannot be considered a high-accuracy outdoor positioning system. More recently, the works in [53] and [54] proposed enhancements to OTDoA through additional opportunistic measurements and Compressive Sensing (CS), respectively, obtaining near GNSS accuracy at the cost of expensive signal processing on the mobile device.

4.2 State-of-the-Art for Millimeter Wave Positioning Systems

The works developed in [55–59] aim to locate devices in both LOS and NLOS outdoor positions. The method in [55] applies CS on information gathered from static listeners, while in [56] multiple access points are used to create a location fingerprint database of received powers and AoA. In [57], the authors use multiple BF transmissions and an iterative algorithm to estimate the position and orientation of the device. The same parameters are obtained in [58], through the estimation of the AoA, Time-of-Arrival (ToA), and Angle-of-Departure (AoD), making simultaneous use of LOS and NLOS transmissions. However, the methods referred so far have difficulties complying with typical outdoor situations: [55] and [56] assume that each device is always in range of multiple static transceivers, while the other two methods struggle with NLOS locations, requiring multiple transmission paths reflecting in at least three different surfaces [57] or preferring not to disclose the performance results for those locations [58].

The method proposed in [59] overcomes the aforementioned restrictions by creating a fingerprint database of uplink pilots transmitted to a single massive MIMO BS that contains multiple antennas distributed over a limited area. Using a Gaussian process regression to resolve the position, this work achieves a RMSE of 34m. For the sake of comparison, let us consider the network-enabled OTDoA and the ubiquitous GNSS, both discussed in the previous sub-section. The former has a theoretical average error of about 10m [50], assuming optimal conditions and expensive detection mechanisms. On the other hand, state-of-the-art GNSS receivers are capable of obtaining higher accuracies, averaging 3m in continuous measurement scenarios [46], with significant penalties for sporadic measurements due to the extensive use of Kalman Filters [60]. Therefore, there is a significant performance gap between state-of-the-art mmWave systems and the existing outdoor positioning solutions, in the presence of NLOS. This and the following chapters present a new system that, making use of deep learning techniques, closes that gap.

For the 5G BSs, which are expected to be positioned in elevated positions of urban scenarios, the majority of the obstacles will be buildings, and thus static for a significant amount of time. Successive measurements of the received Power Delay Profile (PDP) at a given position are expected to remain comparable until a meaningful change in the surrounding space occurs. If a BS transmits a signal employing a sequence of directive BF patterns, so as to cover all possible transmission angles (and thus maximizing the covered space), then the receiver is able to gather multiple distinct PDPs. Due to the non-linear propagation phenomena in the presence of obstacles discussed in section 2.2, that set of PDPs is expected to have noticeable discontinuities throughout the target localization space, which provide significant spatial information. In [15], the use of the set of PDPs to produce the aforementioned BFF was proposed as a foundation for an accurate mmWave outdoor positioning method. The BFF positioning method has an additional attractive aspect: contrarily to most accurate positioning methods (including the method suggested in [59], GNSS, and OTDoA), it requires a single-anchor [11][61].

The information held in a BFF is a result of non-linear interactions and, therefore, requires a method that is able to sift through non-linear relationships. Given the requirements of the problem and the recent state-of-the-art results obtained when dealing with non-linear relationships, DL techniques become a powerful candidate to untangle the BFF. In [15], the use of Convolutional Neural Networks (CNNs) [62] was proposed to exploit the data structure within a BFF. The previous system was improved in [16] with a hierarchical structure, taking advantage of the BFFs' expected similarity along adjacent positions, at the cost of additional processing power. Finally, in [17], sequence-based DL techniques were added to the BFFs position estimates, enabling the system to track a mobile device. The remaining of this document concerns the proposed positioning techniques that were published in [15], [16], [17], and [19].

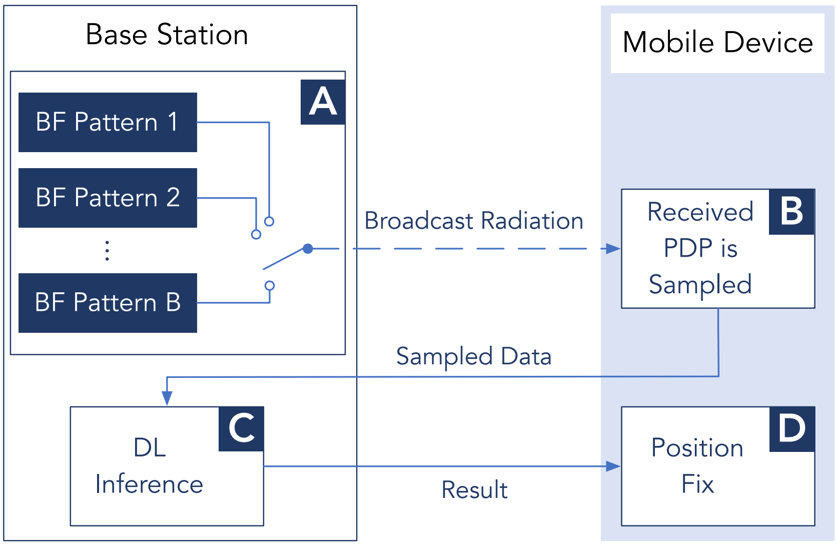


Figure 4.1: Overall scheme of the beamformed fingerprint positioning system. The mobile device samples the received PDPs from radiation transmitted through a fixed set of beamforming patterns, resulting in a unique beamformed fingerprint that can then be translated into its position.

4.3 Beamformed Fingerprints

The transmitted mmWave radiation, suffering from reflections, diffractions, and other phenomena, is shaped by the encountered obstacles. As result, a transmitted signal might have more than one propagation path between the BS and the receiver, each with a unique power attenuation and delay. From an information theory point of view, each new path carries additional information from the surrounding space, and thus strengthens the potential predictive power of the system. Based on this principle, the BFF can carry enough information to locate a listening mobile device.

A critical component of any learnable dataset is its consistency, as it then allows the system to extract helpful information from a trained mathematical model. In other words, the distribution of the data that is used at test time must be comparable to the distribution of the data seen at train time, to avoid the problem known as *dataset shift*. To ensure so, the input data must be gathered using an immutable methodology. As result, both the transmission and the receiving procedures must remain constant in order to obtain valid fingerprints. To comply with such requirements, the system depicted in fig. 4.1 was originally suggested in [15]. It operates in four distinct phases, as labeled in the diagram, whose details are further described below. In phase A, a BS will broadcast radiation using a constant set of beamforming patterns, while phase B focuses on measuring the resulting PDP at the target device. After all the required measurements are performed and transmitted back to the BS, phase C infers the device's position, which will be relayed back to it in phase D.

The transmitter BF directivity, one of the key aspects that will dictate the resolution of the information embedded in the BFF, is defined in phase A. The directivity determines how narrow the beam of transmitted radiation is. Therefore, increasing the directivity of a given transmission translates into a PDP containing information with higher specificity, focused in a particular sub-set of possible propagation paths. Furthermore, by focusing the radiation, the number of paths with enough energy to be detected by the receiver increases. Unfortunately, there is an associated trade-off: to fully cover all possible angles of transmission, higher BF directivities correspond to a higher number of PDP measurements required per position fix. Throughout this document, the exact mechanism to measure the timing of the non-zero samples within a PDP (*i.e.* a cluster of paths) is abstracted, knowing that it can be done through various real implementations[1].

Let us consider a fixed codebook $\mathbf{C}_{Tx}$ containing B_{Tx} BF patterns. Before a position estimate becomes possible, the BS must transmit the signal with the B_{Tx} BF patterns, which are expected to be transmitted in sequence. Assuming a BS with N_S antennas, the frequency-domain received signal for the i-th transmitter BF at a mobile device with N_R antennas, $r \in \mathbb{C}$, can be written as

$$r = \mathbf{w}^T \mathbf{H} \mathbf{f}_i s + \mathbf{w}^T \mathbf{z}, \tag{4.1}$$

where the superscript T denotes a matrix transpose, $\mathbf{w} \in \mathbb{C}^{N_R \times 1}$ corresponds to the (optional) beamforming at the receiver, $\mathbf{H} \in \mathbb{C}^{N_R \times N_S}$ is the channel matrix, $\mathbf{f}_i \in \mathbb{C}^{N_S \times 1}$ denotes the currently selected transmitter beamforming, $s \in \mathbb{C}$ is the signal to be detected, and $\mathbf{z} \in \mathbb{C}^{N_R \times 1}$ represents noise. Since the transmitter beamforming is codebook-based, it is important to state that $\mathbf{f}_i \in \mathbf{C}_{Tx}$ ($\mathbf{C}_{Tx} = \{\mathbf{f}_1, \ldots, \mathbf{f}_{B_{Tx}}\}$).

As the system transmits the sequence of beamformed patterns, it is important to avoid losing information due to destructive interference. As such, after the time allocated to the measurement of the desired fingerprint data for a given beamformed pattern, a minor time interval (T_{guard}) should be considered before the transmission of the following pattern, to account for longer paths with multiple reflections.

[1] Typical approaches rely on pseudo-random sequences [63], round-trip delays [64], and/or cross-correlations [65] (e.g. in [14], the PDPs were gathered through a correlation method).

In phase B, the process of obtaining the BFF from the transmitted signals must result in consistent data, regardless of the listening device. To ensure so, the second key information resolution dictating aspect, the PDP sampling rate, must be constant and enforced throughout the system. To understand how close the sampling rate is related to the resolution of the embedded information, consider a single propagation path between the BS and the receiver. As discussed in [66], the maximum theoretical spatial resolution for a single time-based measurement is given by

$$d_{th} = T \times c, \tag{4.2}$$

where d_{th} is the theoretical resolution of the distance in meters, T is the sampling period in seconds ($1/T$ is the sampling rate in Hz), and c is the speed of light in meters per second. Therefore, the maximum resolution of the hidden information provided by the measured delay of each path is inversely proportional to the selected sampling rate. However, similarly to the directivity in phase A, the sampling rate has associated trade-offs: using a higher sampling rate requires the allocation of additional radio spectrum resources, raises the minimum energy requirement for the detection of each path due to thermal noise, and also places tougher hardware requirements for the mobile devices.

If the system is expecting beamforming at the receiver, a fixed gain must also be established for all receivers. In that case, the receivers would have to define their own BF codebook, $\mathbf{C}_{Rx}$, containing B_{Rx} elements ($\mathbf{C}_{Rx} = \{\mathbf{w}_1, \ldots, \mathbf{w}_{B_{Rx}}\}$). The codebooks would have to be designed so as to search over all AoAs with similar gain, so as to avoid a scenario akin to the *orientation unaware* situation described in [61], where the device orientation becomes an extra variable. With BF at the receiver, the device would have to sample each transmitter BF B_{Rx} times, storing the maximum measured value for each sample within a PDP. The acquired data from the i-th transmitter BF, $\mathbf{x}_i$, can thus be written as

$$\mathbf{x}_i[n] = \max_{j=1,\ldots,B_{Rx}} r_j(nT), \quad n = 0, 1, \ldots, N-1, \tag{4.3}$$

where r_j is the time-domain sampled signal using the receiver beamforming $\mathbf{w}_j$, and N is the number of samples to be considered per PDP. It should be noted that the obtained fingerprint data ($\mathbf{X}$) has a negligible dependency on the mobile device orientation if the receiver BF codebook does cover all AoAs, since it considers the maximum value among all used receiver BFs.

After the required fingerprint data $\mathbf{X}$ is obtained, a previously trained DL method

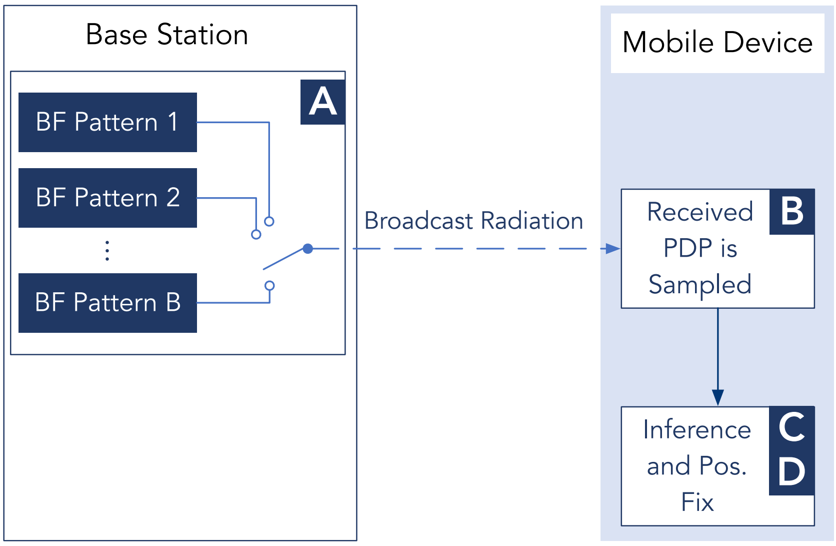


Figure 4.2: Alternative mode of operation of the system depicted in fig. 4.1, where the DL inference is performed in the mobile device. Although it does not require the upload of the sampled data, this mode of operation has additional storage and processing requirements.

can finally infer the device position in phase C (fig. 4.1). With a DL method, the system learns to cope with the non-linearities introduced by reflections and other propagation artifacts. Interestingly, the work in [14], released shortly after the original proposal of the BFFs [15], pointed out machine learning methods as a possible solution to cope with the non-linearities, which were the cause for their lack of positive NLOS experimental measurements.

It should be noted that each BS will have their own dataset and, therefore, their own model. The system performance is determined mainly by the data obtained in phase B and the DL architectures used in phase C, which are further analyzed in the following chapter.

During phase D, the device receives the position estimate from the BS. Phase C could be performed at the mobile device, as depicted in fig. 4.2, avoiding the data upload to the BS (and phase D altogether). However, the device would have to download millions of model weights from each BS, placing a significant memory constrain on the device, and thus herein the predictions are considered to be computed at the BS (as depicted in fig. 4.1). Moving the inference to the BS also allows the system to centralize the users' position information, enabling further applications (*e.g.* optimized traffic management and positioning-aided BF selection [10]). Nevertheless, it is an option to be considered, and it will be analysed throughout this Thesis.

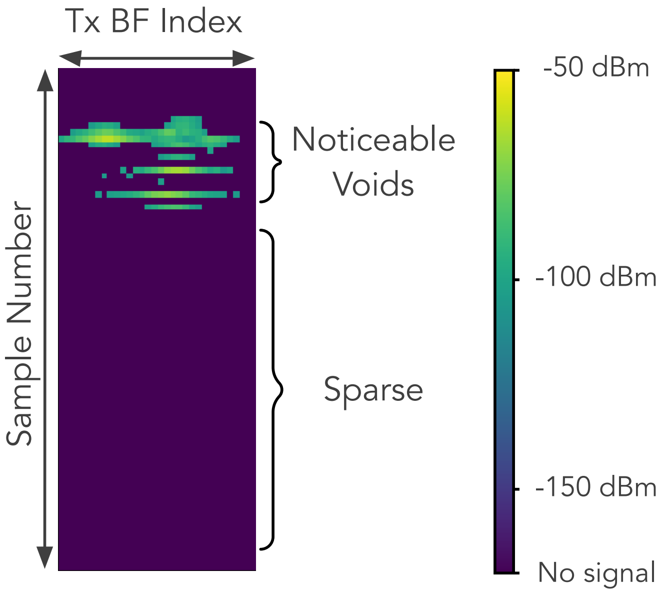


Figure 4.3: Standard example of a noiseless beamformed fingerprint from the experimental simulations, containing the PDP for each beamformed transmission on the vertical axis.

4.4 Beamformed Fingerprint Data Analysis

One of the aspects that dictate the potential spatial information embedded within a beamformed fingerprint is the selected sampling period (T). In fact, high quality data can be obtained with sampling frequencies exceeding 10 MHz (*i.e.*, $T < 100$ ns) – in such conditions, the radiation arriving from the multiple propagation paths is detected in clusters, containing voids that are large enough to be reliably detected [40]. The ability to distinguish these voids provides a meaningful shape to the resulting data, enhancing the learning capabilities of the system.

The multipath propagation inherent to these frequencies suggest us to gather a substantial number of samples per transmitter BF (N), so as to include even the longest paths and thus maximize the received information. However, by doing so, the resulting data will be sparse, as it is observable in the example plotted in fig. 4.3. In fact, due to this sparseness, the relative position of the acquired non-zero samples in the data contains the majority of the extractable information. Therefore, as it will be shown in chapter 5, a binary detection of the signal's existence when acquiring the data is preferable, instead of measuring the signals' power, further reducing the hardware requirements for the BFF positioning system. Furthermore, if the position inference is to be done at the BS, the use of binary PDPs also reduces the amount of data required to be transmitted back to it.

When examining the sampled data, it is interesting to notice a visual pattern that arises when the sequence of transmitted BF indices correspond to a continuous sweep over the azimuth (as in the simulations that resulted in fig. 4.3). Plotted as a 2D image, where the axes correspond to time and the BF index, and the color represents the detected power (or the signal existence), the formed image will likely have short lines along the BF direction. In other words, this means that physically adjacent BF patterns will likely end up having similar clusters when measured from the same location, and thus carrying partially redundant information. As a result, increasing the number of transmitted BF patterns without increasing their directivity has diminishing returns on the position inference accuracy. On the other hand, increasing the BF patterns directivity, which can be seen as increasing the spatial resolution of the captured information, should have a positive impact in the resulting accuracy.

Finally, the flexibility of the BFF positioning method regarding its radiation sources deserves to be highlighted. While most accurate positioning methods require three or more separate transmitters [11], the BFFs can be obtained from a single BS, enabling positioning estimates whenever there is mmWave coverage [61].

4.5 Beamformed Fingerprint Power Requirements

Any positioning system will be mostly used by mobile devices, and thus it is of paramount importance to assess their energy requirements. For the BFF positioning method, it is observable from fig. 4.1 that its energy requirements can be broken down into:

- *a)* Sampling the received radiation, so as to extract the BFF;
- *b)* Sending that BFF back to the BS and receiving the position fix; OR
- *c)* Performing inference on the device.

The energy required to sample the PDP consists mostly on the energy required by the whole mmWave RF front-end during the listening time (E_{Rx}). Considering the listening time needed by each of the $B_{Tx} \times B_{Rx}$ transmitted pulses required in order to obtain a BFF, we can estimate E_{Rx} as

$$E_{Rx} = P_{Rx}\big((T \times N) + T_{guard}\big)\big(B_{Tx} \times B_{Rx}\big), \tag{4.4}$$

where P_{Rx} is the average power required by the RF front-end, and $((T \times N) + T_{guard})$ is the time required per pulse. In [67], an assessment of the state-of-the-art for mmWave RF components concluded that a device's receiver front-end should require about 125 mW[2].

Let's assume now that inference is performed at the BS. In that case, the obtained data must now be transmitted to the BS. Considering that each of the $N \times B_{Tx}$ data samples contains k bits, if the system has an energy efficiency of E_{ef} Joules per transmitted bit, the required transmit energy (E_{Tx}) can be written as

$$E_{Tx} = (k \times N \times B_{Tx})\, E_{ef}. \tag{4.5}$$

Since the device will have mmWave antennas installed, it should be able to use a 5G-enabled mmWave connection. With the data from the study performed in [68], it is conservative to assume an uplink energy consumption of 0.2 μJ per transmitted bit[2] (or 5 Mbits per Joule).

As mentioned in the previous sub-section, when a small sampling period is used, the resulting data will be sparse, and thus it can be efficiently compressed. For instance, if the transmitted data contains exclusively pairs of non-zero values (the received power) and their respective positions in the data sample, the required transmit energy can be rewritten as

$$E_{Tx} = \Big(V \times \left(\lceil log_2\,(N \times B_{Tx})\rceil + k\right)\Big) E_{ef}, \tag{4.6}$$

where V is the number of valid entries (*i.e.*, non-zero entries), $\lceil log_2\,(N \times B_{Tx})\rceil$ is the number of bits required to encode all possible positions, and k represents the actual data for each valid entry. Furthermore, when the data is obtained through the simpler binary detection, the data in k (received power) becomes redundant, and thus

$$E_{Tx} = \left(V \times \lceil log_2\,(N \times B_{Tx})\rceil\right) E_{ef}. \tag{4.7}$$

Considering that the power required to receive the final result is negligible, the total required energy per position fix can be approximated by adding (4.4) to either (4.6) or (4.7) (for non-binary and binary data, respectively).

Let's now consider the other alternative for the position inference – the mobile device performs its own machine learning computations. Although DL architectures are often considered computationally demanding, in practice, one can build a complex model with

[2] Please note that mmWave systems are under exhaustive study, and thus these figures should improve over the next years.

a very limited set of computational operations. This spurred the development of energy-efficient computer architectures specifically tailored to them, often based on Graphics Processing Unit (GPU) or Field-Programmable Gate Array (FPGA) architectures [69][70]. In fact, for general purpose mobile devices such as the smartphone, a large number of them are having more and more dedicated hardware for DL computations. As such, to assess the energy consumption of the BFF position inference when it is computed in the mobile device, one must measure the energy consumption at the used dedicated hardware.

4.6 Summary

The third part of this Thesis proposes a new positioning approach, and aims to demonstrate its capabilities. Contrarily to beam search methods, there are proven positioning techniques with decades of existence, which section 4.1 aims to cover. Positioning is particularly challenging in the context of mmWaves, as discussed in section 4.2, with the new obstacle interactions preventing the use of geometric positioning methods in most scenarios. These two sections will be critical to provide a comparative assessment of the proposed approach.

The proposed positioning approach discussed in section 4.3 is built on the exact same assumption as the beam search method in Chapter 3: "the majority of new propagation paths arise from reflections on static obstacles (such as buildings), and thus a quasi-deterministic model of the signal power at each position can be built". However, this Chapter takes it a step further, with the analysis of the temporal behavior of that signal (PDP). Combining the unique mmWave propagation characteristics, which give rise to new transmission paths, with the directivity of beamformed transmissions, which limit the space scanned in a single transmission, we can build a powerful signal called beamformed fingerprint, which contains latent information about the surrounding space. This latent information can then be learned by a deep learning model, also leveraging the BFF data analysis made in section 4.4, which will be further assessed in Chapters 5 and 6.

To conclude this Chapter, section 4.5 builds the mathematical models to study the mobile device energy requirements for the BFF positioning system. Studying the energy requirements of a positioning approach is of upmost importance, as it will mostly be used in energy-constrained mobile environments. Consequently, the development of new approaches with lower requirements can lead to position-aware devices with smaller form factors.

The remaining of this part will build on the top of the concepts discussed on this Chapter, providing DL solutions to the positioning system and assessing the quality of the created models.

5
Beamformed Fingerprint Positioning

Contents

The beamformed fingerprint positioning problem can be seen as the supervised learning of the training set $\mathcal{T}$, whose samples are obtained from a fixed distribution $\mathcal{D}_{\mathcal{X}\times\mathcal{Y}}$. The input space $\mathcal{X} = \mathbb{R}^{(N\times B_{Tx})}$ corresponds to the set of possible BFFs, whereas the target space $\mathcal{Y} = \mathbb{R}^{d}$ is the set of all possible positions, where d depicts the dimension of the position space (2 or 3 for bidimensional or three-dimensional positions, respectively). The purpose of the BFF positioning system is then to train a mapping function $f : \mathcal{X} \mapsto \mathcal{Y}$ using $\mathcal{T}$, so as to be able to generalize to new, unseen samples.

The simplest DL architecture applicable to the BFF positioning problem is what is typically called a Deep Neural Network (DNN). The DNN is a circuit comprised of a number of basic elements called neurons that are stacked in multiple layers, denoted as fully connected layers. The vector containing the output of the i-th layer of neurons $\mathbf{n}_i$ can be written as

$$\mathbf{n}_i = a\left(\mathbf{U}_i\, \mathbf{n}_{i-1} + \mathbf{b}_i\right), \tag{5.1}$$

where $\mathbf{U}_i$ depicts the connection between neurons (also known as weight matrix), $\mathbf{b}_i$ is the firing thresholds vector (also known as bias), and a is an activation function, a nonlinear subdifferentiable function. The first layer ($\mathbf{n}_0$), also known as input layer, is fed in with the input data $\mathbf{X}$, which is a BFF in the context of this chapter.

Due to the nonlinear activation functions, a DNN is a good candidate to learn nonlinear phenomena [71], as commonly encountered in a mmWave transmission. To map the input fingerprint data to the target label, the network is trained using a gradient-based algorithm which updates the neurons' learnable parameters, $\mathbf{b}$ and $\mathbf{U}$ in equation (5.1). This supervised training is guided by a loss function, which can be seen as a measurement of the average similarity between the network predictions and the true labels. For the proposed system, the neural network is trained to perform a regression in the output layer, minimizing the Mean-Square-Error (MSE) to the data's labeled position $\mathbf{y}$, *i.e.*,

$$\hat{\mathbf{y}}^{*} = \underset{\hat{\mathbf{y}}}{\arg\min}\ \mathbf{E}\Big\{\left(\hat{\mathbf{y}}-\mathbf{y}\right)^{\mathbf{T}}\left(\hat{\mathbf{y}}-\mathbf{y}\right)\Big\}, \tag{5.2}$$

where $\hat{\mathbf{y}}^{*}$, which is the output of the neural network's last layer, denotes the estimated position given the input data $\mathbf{X}$. The usage of this loss function can be interpreted as a minimization of the euclidean distance between the labeled position and its estimate. After being trained with $\mathcal{T}$, the learnable parameters ($\mathbf{b}$ and $\mathbf{U}$) are locked, and the network is able to provide estimates for new, unseen data.

A DNN, as any DL architecture, can only have as much predictive power as the train-

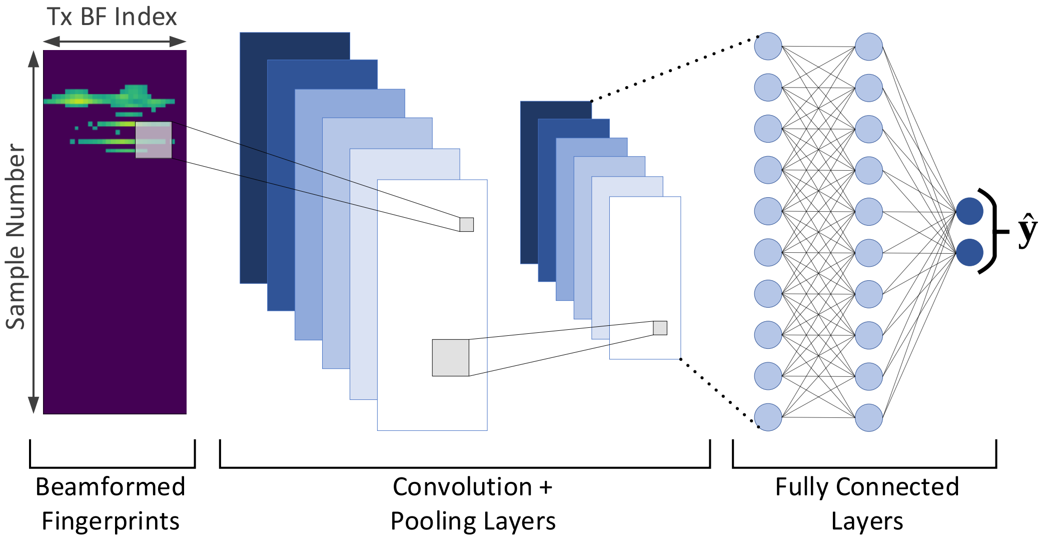


Figure 5.1: Even though the two dimensions within a BFF have disparate meanings, the data sequences along both dimensions carry significant information (as elucidated in section 4.4). Therefore, by using a CNN, the system can efficiently tap that source of information.

ing set $\mathcal{T}$ enables it to. In order to be effective when evaluating unseen data, the network should be able to generalize the information assessed while training, especially if the data is expected to be noisy. To do so, the network should be exposed to a sizable training set and possibly trained with regularization techniques (e.g. *dropout* [72]), forcing it to focus on the general attributes of the data, instead of memorizing the training set (also known as *overfitting*). A successful DL-based system must then be able to easily gather massive amounts of labeled data, which is not always possible. Fortunately, the BFF system, as well as any other outdoor positioning method, can use the GNSS as a last resort to accurately[1] label the captured input data. On the other hand, the same cannot be said for indoor positioning systems, which struggle to manually label the gathered data (as mentioned in [73]).

5.1 Enabling Convolutional Neural Networks

Consider now the two indexing dimensions of the BFF data samples, the time-domain sample number and the transmitter BF index. If the sequence of BF indices corresponds to a continuous sweep over the azimuth, as described in section 4.4, it is possible to extract information not only from the individual data points, but also from their sequence along those two dimensions. Therefore, even though the two dimensions have disparate meanings, the nature of the problem makes CNNs a good candidate for the problem at hand, as illustrated in fig. 5.1.

With CNNs, the convolutional layer is introduced, where the neural network can learn the most effective set of short filters to apply on the received data, and thus also extracting information from sequences within a sample. A convolutional layer can learn more than one feature from the previous layer's output, and thus subsequent layers are often seen as higher-order abstractions. For the i-th convolutional layer of neurons, $\mathbf{N}$, which is now a matrix, the output of the f-th feature can be written as

$$\mathbf{N}_i^f = a\left(\sum_{\tilde{f}=1}^{\tilde{F}} \left(\mathbf{U}_i^{f,\tilde{f}} \, \mathbf{N}_{i-1}^{\tilde{f}} \right) + \mathbf{1} \times b_i^f \right), \tag{5.3}$$

where $\tilde{F}$ is the number of features in the previous layer, $\mathbf{1}$ is a bi-dimensional matrix of ones, the bias b_l^f is now a single scalar, and each $\mathbf{U}_i^{f,\tilde{f}}$, now denoting a bi-dimensional filter, is a doubly block circulant matrix (which is a special case of a Toeplitz matrix). In this case, the input layer ($\mathbf{N}_0$) is fed in with the BFF data $\mathbf{X}$, which can be seen as a layer containing a single feature. Due to its new structure, if $\mathbf{U}_i^{f,\tilde{f}}$ is built from a $L1$ by $L2$ bidimensional filter, it will only contain $L1 \times L2$ learnable parameters. Although there is a different learnable filter for each pair of features on two subsequent convolutional layers, the number of learnable parameters in a convolutional layer is significantly lower than in a fully connected layer, for equally performing neural networks [62]. The enhanced performance per learnable parameter arises due to the filter bank structure of the convolutional layer, which enables the network to recognize the same patterns in different parts of the input data, effectively enforcing generalization.

Since each feature is a filtered copy of the previous layer's output, the total amount of data transported by each succeeding layer quickly becomes overwhelming. To cope with such data increase, and to improve the invariance against minor shifts, convolutional

[1] Even though typical civilian GNSS receivers have an average accuracy of 3m, the proliferation of systems similar to Japan's Quasi-Zenith Satellite System will enable sub-meter accuracies in particular areas.

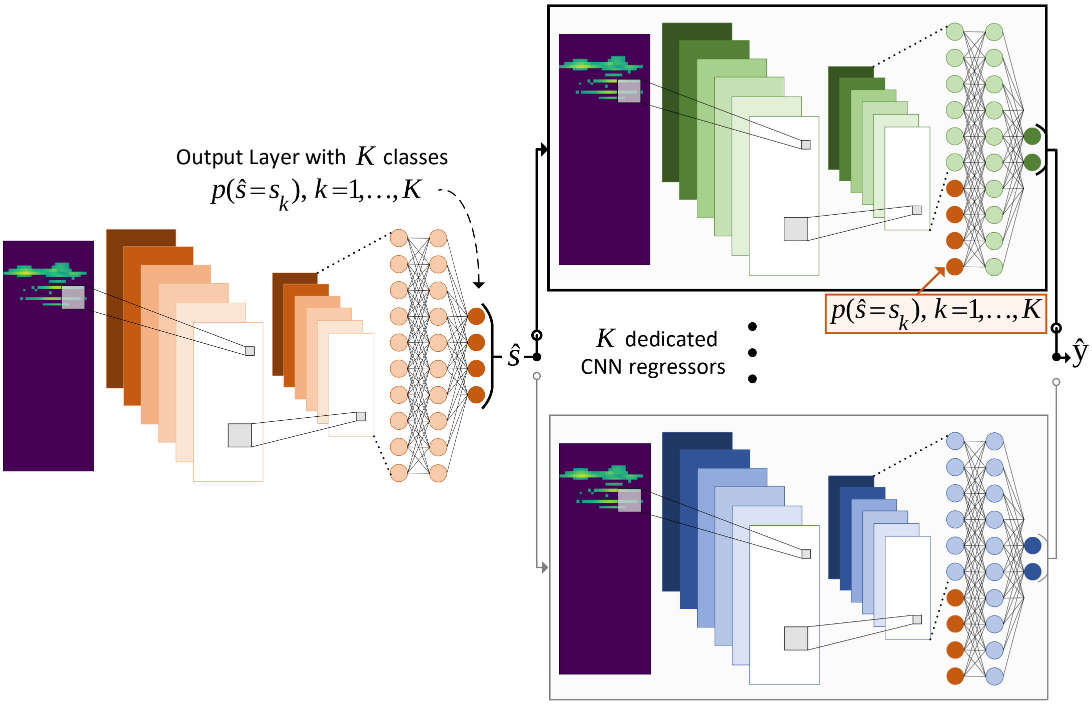

Output Layer with K classes
$p(\hat{s}=s_k),\ k=1,\ldots,K$
$\hat{s}$
K dedicated
CNN regressors
$p(\hat{s}=s_k),\ k=1,\ldots,K$
$\hat{\mathbf{y}}$

Figure 5.2: Overview of the proposed hierarchical architecture. Considering a solution space that can be divided into K highly correlated sub-regions, the hierarchical architecture first employs a CNN classifier that selects the most suitable sub-region $\hat{s}$. That sub-region's dedicated CNN regressor is then used to obtain the estimate, $\hat{\mathbf{y}}$. To enhance the regressor's precision, it is also fed with the output layer of the classifier, which can be seen as a coarse estimate. Note that each sub-model has its own set of learned weights, as indicated by the different colors.

layers are usually followed by pooling layers, where data is downsampled. In a typical CNN architecture, the network starts with the convolutional layers, whose output is then flattened for the subsequent fully connected layers.

5.2 Hierarchical Convolutional Neural Networks

The outdoor positioning problem maps a set of input data to a continuous space $\mathcal{Y}$, the position. Due to the physical laws that determine electromagnetic propagation, the same transmitted signal is expected to be highly correlated when measured in adjacent positions. In fact, if it was not for the non-linear phenomena introduced with mmWave frequencies, the received BFFs would have mostly smooth changes throughout the considered space. The non-linear phenomena introduces discontinuities in the BFF data, if assessed throughout a continuous route, segmenting the output space into multiple potential sub-regions, each with specific patterns in the input data. Given that clear segmentation, in [16] a hierarchy-based system was proposed to further refine the single BFF learning mechanism, as depicted in fig. 5.2. The concept of hierarchy has been widely used in the past to solve numerous problems. In this particular case, the approach was inspired by the work in [74], where the prediction outcome of a coarse model may trigger specialized fine-grain models, which help to handle complicated examples.

As explained above, each BS's covered space can be seen as a set of K sub-regions $\mathbf{S}$ ($\mathbf{S} = \{s_1, \ldots, s_K\}$, $\bigcup_{k=1}^{K} s_k = \mathcal{Y}$), which may overlap. If a dedicated CNN is assigned to each sub-region, containing a structure as defined in the previous section, those K CNNs can specialize on their own data partition. As adjacent positions are very likely to be highly correlated, and thus contain similar data patterns, each dedicated CNN will have fewer patterns to learn, thus facilitating the learning process. The sub-regions can be seen as coarse positions and, as result, identifying the sub-region s of a new data sample is easier than pinpointing its exact position. Therefore, a CNN classifier is used to predict the most likely $\hat{s}$, indicating which dedicated CNN should be used to estimate the device location. As mentioned, the predicted $\hat{s}$ can be seen as a coarse position estimate and, therefore, the selected regressor is also fed with the output layer of the CNN classifier, so as to enhance its precision.

Contrarily to image-based problems, where there are multiple lower level local features such as lines, curves, and colors to be learned and shared, the data in a BFF not only is sparse, but also changes dramatically throughout space. As such, in opposition to the architecture proposed in [74], which shares the first layers between all involved models, the architecture described in this section does not force learning the same basic patterns in the first layers. By not sharing those weights, not only each specialized regressor can completely focus its resources towards its sub-region, but also the global training procedure can be simplified from a three-pass learning algorithm [74], to a pair of steps: first the coarse classifier is trained, then all specialized regressors can be trained (the specialized regressors can be trained in parallel). To train the classifier, the cross-entropy between prediction and ground truth is minimized, such as

$$p(\hat{\mathbf{s}}) = \underset{p(\hat{s}_k),\ k=1,\ldots,K}{\arg\min} \ \mathbf{E}\Big\{ -\sum_k p(s_k|\mathbf{X}) \log(p(\hat{s}_k)) \Big\}, \tag{5.4}$$

where $p(\hat{\mathbf{s}})$ denotes the output vector of the classifier neural network, containing the predicted probabilities $p(\hat{s} = s_k)$ for a certain input data $\mathbf{X}$. It should be noted that the above formulation allows $\mathbf{S}$ to contain overlapped sub-regions, as $p(s_k|\mathbf{X})$, the true probability of being in s_k given the input data $\mathbf{X}$, can be 1 for multiple k[2]. After obtaining the classifier's output, the most suitable dedicated CNN $\hat{s}$ is selected by determining

$$\hat{s} = \underset{k=1,\ldots,K}{\arg\max} \ p(\hat{s} = s_k), \tag{5.5}$$

which in turn will provide the position estimate $\hat{\mathbf{y}}$.

As K grows, a trade-off is expected: the specialized CNNs have a smaller space to cover, and thus a smaller number of patterns to learn, while the CNN classifier has to select its answer from a wider range of solutions. Since the dedicated CNNs map their predictions to the complete space $\mathcal{Y}$, they might be able to recover from previous classification errors, as long as it is a recurrent (and thus learnable) mistake. On the other hand, non-recurrent misclassifications have a significant penalty on the system, especially when training, where a misclassified sample is tied to the training set for $\hat{s}$ (with $\hat{s} \neq s_k$). This can be seen as simultaneously adding noise to the training set for $\hat{s}$, while depriving s_k of meaningful samples, which can be particularly adverse when each sub-region has a small training set. The results in [74] also reflect the aforementioned trade-off, with hierarchical models outperforming traditional CNNs unless there are too many data partitions.

The application of the hierarchical model is nearly transparent to the BFF positioning system, as all changes occur in phase C of the method described in section 4.3. Contrarily to the work in [74], where the number of used fine-grain models has no upper bound, the discussed hierarchy model has a stable execution time (one coarse classification and one fine-grained regression), which is important for low-latency tasks such as positioning.

5.3 Simulation Apparatus

To evaluate the proposed system accuracy, a dataset using accurate mmWave ray-tracing simulations in the NYU area is used again, akin to the dataset described in section 3.3. Whereas the dataset described in section 3.3 held the average received power (for each used beamforming in a discrete position grid), the dataset used in this chapter contains the received PDP. In other words, it contains a ray-tracing generated BFF per considered position, totaling 160801 BFF samples drawn from a bidimensional position grid. Similarly to section 3.3, the propagation specifications in Table 5.1 were inherited from the experimental measurements in [40].

[2] For instance, a given data point can belong to sub-regions 1 and 2 if they partially overlap, resulting in a vector with $p(s_1|\mathbf{X}) = p(s_2|\mathbf{X}) = 1$.

Table 5.1: Ray-Tracing Simulation Parameters

Parameter Name	Value
Carrier Frequency	28 GHz
Transmit Power	45 dBm
Tx. Antenna Gain	24.5 dBi (horn antenna)
HPBW	10.9°
Transmitter Downtilt	10°
Codebook Size	32 (155° arc with 5° between entries)
Receiver Grid Size	160801 (400 × 400 m, 1 m between Rx, 1 m above the ground)
Samples per Tx. BF	82 (4.1 μs @ 20 MHz)
Assumed Rx. Gain	10 dBi (as in [30])
Detection Threshold	-100 dBm
Added Noise	$\sigma = [2, 10]$ dB (Log-Normal)

While the used ray-tracing software (Wireless InSite 3.0.0.1 [42]) was unable to control BF patterns, a physically rotating horn antenna was used, producing similar directive radiation patterns. For each of the 32 elements in $\mathbf{C}_{Tx}$, the received power data was sampled at 20 MHz over a spawn of 4.1 μs, which contained over 99% of the path data. Regarding BF at the receiver, a 10 dBi gain was considered (akin to [30], which contains a codebook with 8 entries). In the following simulations, noise is added to the obtained ray-tracing data following a log-normal distribution (also known as *slow fading*). The noise was introduced before applying a detection threshold of -100 dBm, which was selected due to the thermal noise for the considered bandwidth (-101 dBm). In all the shown simulations, the data is binarized after adding the noise and applying the detection threshold.

The resulting data was labeled with the corresponding bidimensional position, in a 400×400 m^2 area centered at the base station. When the area is split for the hierarchical model, only powers of 4 are considered for the number of partitions, where each physical dimension is subsequently bisected (e.g. when 64 partitions are considered, each dimension is bisected 3 times, resulting in partitions with 50×50 m^2).

The selected CNNs follow a typical architecture, whose hyperparameters (depicted in Table 5.2) were selected after the empirical testing of a random hyperparameter search [75]. The classification and the K regression CNNs share the same configuration and hyperparameters (Table 5.2), except for the input of the first fully connected layer and the output layer (see fig. 5.2). While sub-optimal, they are all trying to extract similar information given similar data, and thus using a single hyperparameter set yields satisfying results while alleviating the search complexity.

Table 5.2: CNN and Hierarchical CNN hyperparameters.

Parameter Name	**Value**
Convolutional Layers	1 layer (8 features with 3×3 filters)
Pooling Layers	2×1 max-pooling
Hidden Layers	12 (256 neurons each)
Regression Output	Linear with 2 Neurons (2D position)
Classification Output (Hierarchical CNN's 1st model)	Softmax with K classes
Epochs	Up to 1000 (early stopping [77] after 50 non-improving epochs)
Batch Size	64
Optimizer	ADAM[78]
Learning Rate	10^{-4}
Learning Rate Decay	0.995
Dropout	0.01

For each training epoch, a new noisy training set is generated, consisting of the original ray-tracing dataset entries with added random noise. Since the proposed positioning system is expected to be used to predict physical positions for which it already has training samples, the test set is also generated from noisy samples of the ray-tracing data. When evaluating the average results, a total of 10 test sets are used. The total training time required for the presented results was also taken in account — training any of the assessed architectures with the defined parameters takes less than 10 hours in a Nvidia GTX 780 Ti GPU, using Google's TensorFlow framework [76]. Note that, since the next chapter contains an augmented version of this chapter's system, the energy efficiency for all proposed approaches will be assessed there. Finally, since reproducibility is fundamental to validate and move forward on investigation, the simulation code and the used data are publicly available[3].

5.4 Simulation Results

In this section, the proposed system is simulated and evaluated by applying the data and the parameters discussed in the previous section. The results are obtained using a 32-element codebook dataset with binary detection, unless explicitly stated otherwise. When describing the following results, three levels of noise are usually considered: low ($\sigma = 2$ dB), moderate ($\sigma = 6$ dB), and high noise ($\sigma = 10$ dB).

[3] https://github.com/gante/mmWave-localization-learning

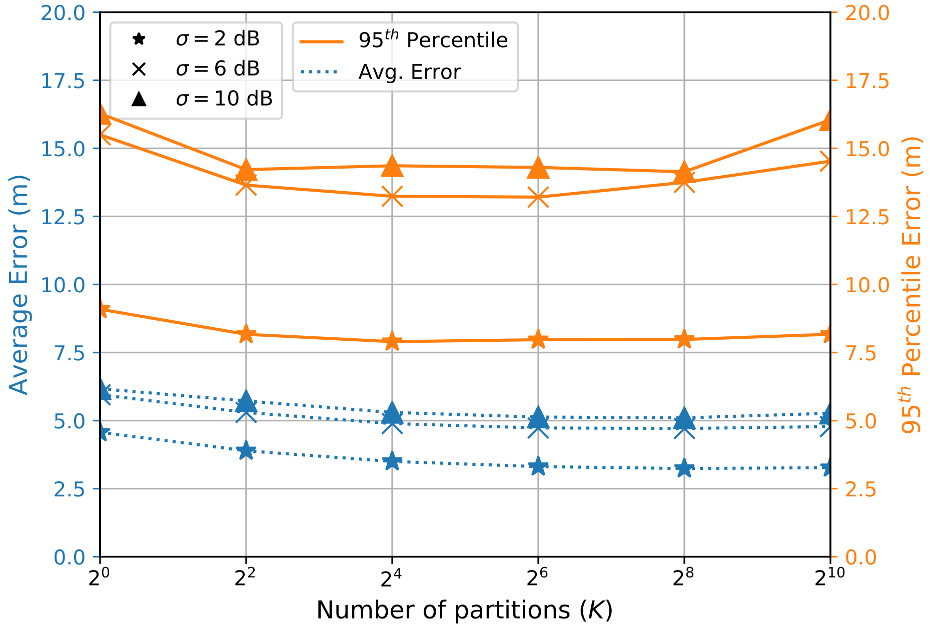


Figure 5.3: Average and 95th percentile prediction errors for multiple number of partitions and noise levels (σ). While it is a tool to extract additional accuracy, an excessive number of partitions has adverse consequences.

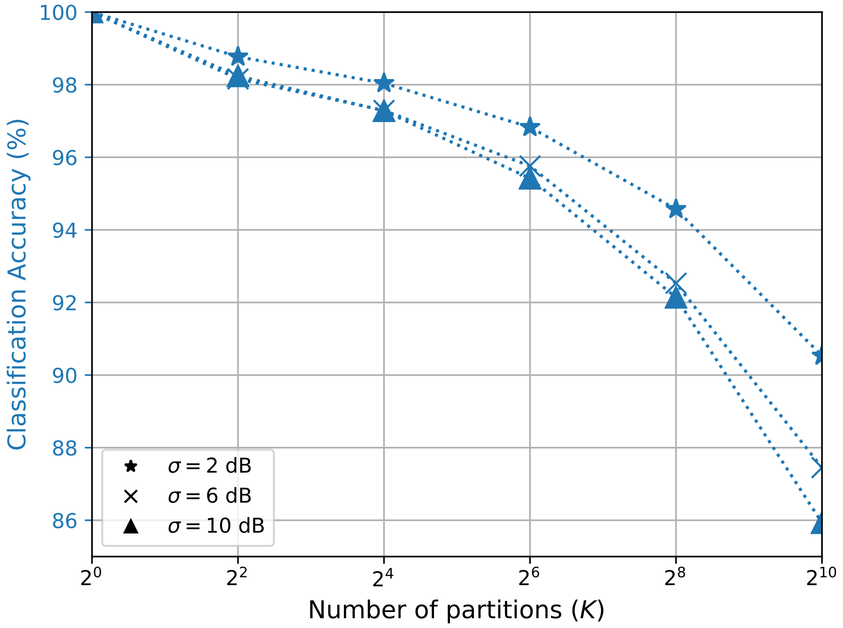


Figure 5.4: Classification accuracy for the classifier CNN, for multiple number of partitions and log-normal modeled noise levels (σ). As the number of partitions grow, so does the classification error, increasing the amount of incorrect data points fed to the specialized regressors.

The first assessed parameter is the number of data partitions (K), as shown in fig. 5.3. It is interesting to notice that the predictions for $K > 64$ keep roughly the same average

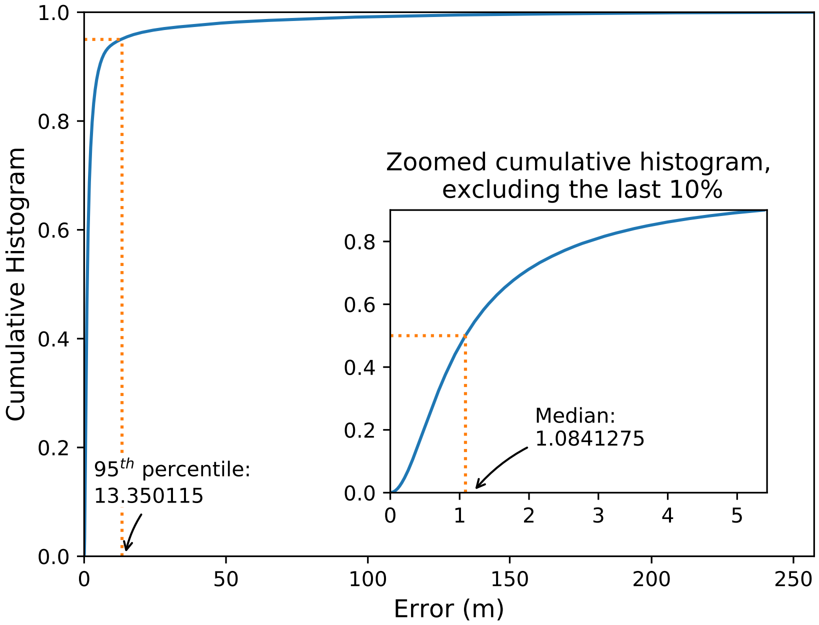


Figure 5.5: Cumulative histogram of the prediction errors, considering $K = 64$ and a noise σ of 6 dB. The least accurate predictions are very inaccurate ($>$ 100 m), and thus the average error is significantly larger than the median error.

error, with an increased 95th percentile error. This means that although more specialized regressors yield improved predictions on correctly classified samples, the higher number of misclassified samples (see fig. 5.3) reverts those gains, as mentioned in section 5.2. Considering a partition-less dataset (*i.e.*, $K = 1$), the average error ranges from 4.57 m to 6.17 m, for low and high noise values, respectively, with a 95th percentile error never exceeding 16.3 m. The best results were obtained when $K = 64$, with an average error ranging from 3.31 m to 5.13 m and a 95th percentile error never exceeding 14.3 m.

In fig 5.4, the classification accuracy for the classifier CNN is shown. In summary, the probability of selecting the incorrect specialized regressor grows with the number of partitions, which seems counterfactual with the results described before (fig. 5.3). However, the specialized regressors map their output to the complete output space, as described in section 5.2, and thus they are able to recover from the classifier's mistakes. It is important to clarify that the selected partitions (subsequent bisections of the considered area) are very likely to be sub-optimal. Nevertheless, they demonstrate the applicability of hierarchical data to the considered problem, achieving state-of-the-art performance with design minimal effort.

To analyze the gap between the average and the 95th percentile errors, fig. 5.5 plots the cumulative histogram for a moderate noise level with $K = 64$. The least accurate pre-

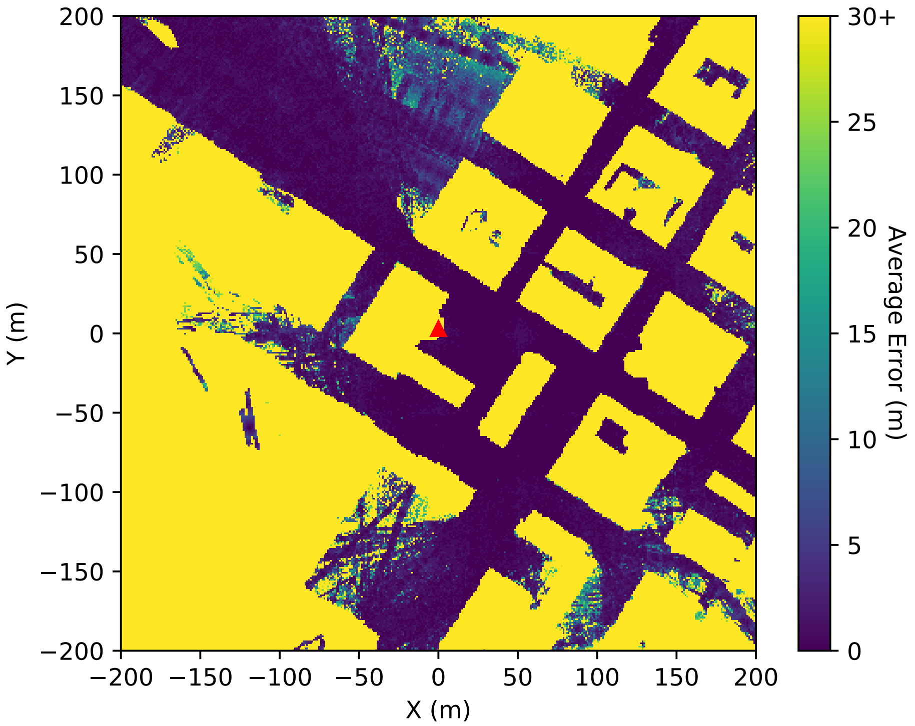


Figure 5.6: Average error per covered position, assuming $K = 64$ and a noise σ of 6 dB. Given that the transmitter is at the center of the image (red triangle), it is possible to confirm that being in NLOS positions is not critical for the proposed system.

dictions explain the gap between the two aforementioned errors, and also translate into a meaningful gap between median and average errors (1.08 m and 4.73 m, respectively). For this configuration, the predictions have a RMSE of 19.7 m, which denotes quite superior performance in all aspects when compared to [59], whose simulations obtained an RMSE of 35 m. Moreover, it is important to point out that [59] considers a lower noise level, with $\sigma = 5$ dB (6 dB was used in the shown experiments), and its numerical simulations do not contain NLOS positions, contrarily to this section's simulations, which means that simulations performed in the scope of this Thesis are more realistic and have considered a more conservative scenario.

The results plotted in fig 5.6 show that the simulated system is able to provide accurate predictions for most of the covered area. However, it is common knowledge that the performance of a CNN is very dependent on the quality of the training data. In fact, due to limitations of the used ray-tracing software with respect to the number of positions simulated, the considered dataset has a minimum physical distance of 1 m between any two samples. In fig. 5.7, the error for the original test set is shown for multiple spatially downsampled datasets, with additional minimum distance between training samples.

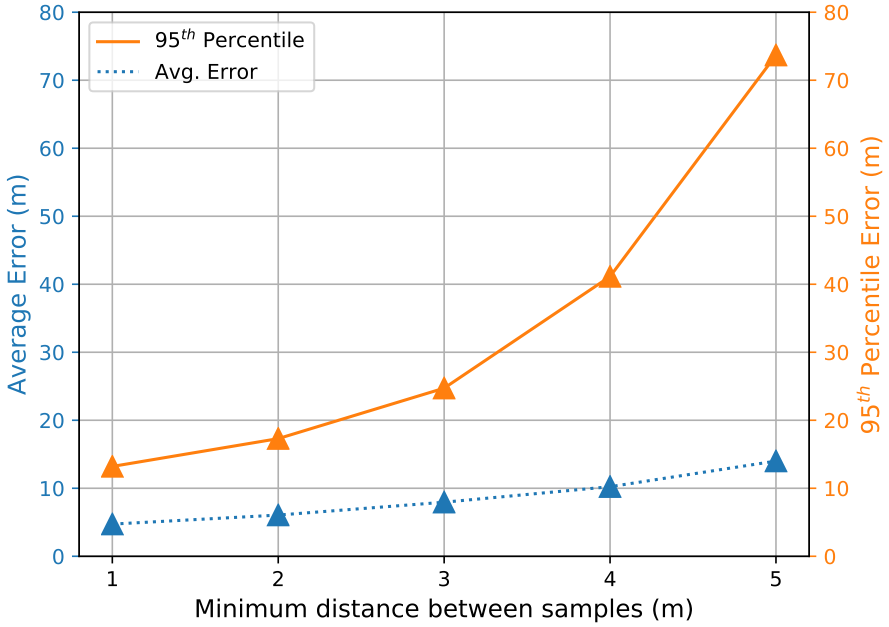


Figure 5.7: Average and 95^{th} percentile prediction errors for multiple downsampled datasets, assuming $K = 64$ and a noise σ of 6 dB. These datasets were downsampled considering a minimum physical distance between samples, and thus they possess inferior quality. The obtained performance is very dependent on the dataset quality, and thus the accuracy results presented in this section, obtained with a limited dataset due to software constraints, do not depict the maximum performance achievable with beamforming fingerprints.

The obtained performance is very dependent on the dataset quality, especially the 95^{th} percentile error. Therefore, it is important to realize that without the aforementioned dataset constraint, the accuracy results obtained in these simulations would be slightly superior.

The dataset quality is also greatly influenced by the sampling frequency, as it dictates the temporal resolution of the captured information. In fig. 5.8, the error results produced by datasets with different sampling rates are plotted. As discussed in Section 4.4, sampling periods larger than 100 ns are unable to properly capture the short voids that usually occur between the received radiation clusters, yielding data that lacks a critical shaping component. As it is observable, the error rises sharply in those circumstances. It should be noted that most experiences in this section were executed with a sampling frequency of 20 MHz, which, according to fig. 5.8, do not result in optimal error performance. The reason for that selection was twofold. First, the selected value achieves a good error performance *vs* required energy trade-off, as higher sampling frequencies yield a higher count of non-zero entries, and thus require additional device-side energy.

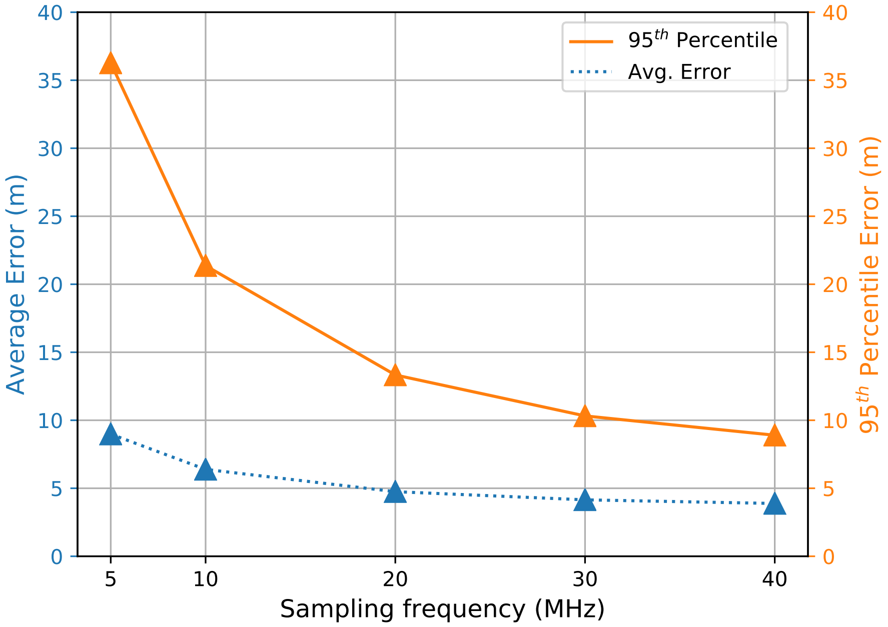


Figure 5.8: Average and 95th percentile prediction errors for multiple sampling frequencies, assuming $K = 64$ and a noise σ of 6 dB. As discussed in section 4.4, sampling frequencies below 10 MHz are unable to properly capture the voids between the received radiation clusters, resulting in greatly reduced performance.

Secondly, it demonstrates that the beamformed fingerprints are able to achieve good results with modest hardware requirements, as current LTE mobile devices are able to sample at this frequency. If more aggressive sampling rates are selected, such as the 800 MHz used during the practical measurements in [14], the predictions could potentially become more accurate. However, using higher sampling rates would have its drawbacks: the mobile devices would spend considerably more energy throughout the positioning process, and additional expensive radio spectrum bandwidth would be required.

The impact of different dataset types is assessed in fig. 5.9. To compare against the baseline, created through the binary measurement of the beamformed radiation generated by a 32-element codebook, two additional datasets were considered: one that uses a smaller codebook with 16 entries and the same binary measurement scheme, and other that uses the same codebook, but adopts higher-resolution measurements (floating point data). The 16-element codebook, which was created using the odd-numbered entries from the original codebook, while having nearly the same mmWave coverage, resulted in a higher prediction error. Nevertheless, using a smaller codebook provides significant energy savings, as it will be analyzed in the following chapter, resulting in a trade-off be-

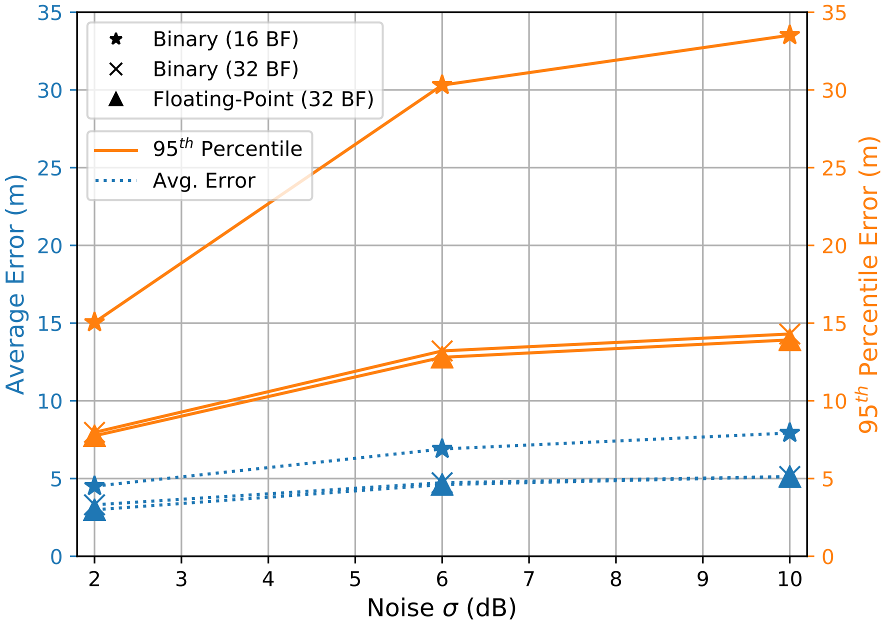


Figure 5.9: Average and 95th percentile prediction errors for multiple dataset types and noise levels, assuming $K = 64$. The 32-element codebook dataset, while providing approximately the same coverage as the 16-element codebook, yields a considerably lower prediction error. On the other hand, the usage of floating-point data shows very marginal benefits, and thus, given the additional hardware requirements, it is not recommended.

tween accuracy and energy consumption for the proposed system[4]. On a different note, the utilization of floating point data, instead of a single bit representation, resulted in marginal positioning benefits, particularly in a low-noise environment, where the average error saw a reduction of 9% (3.02 m). Given that multi-level analog-to-digital conversion translates into significantly higher energy requirements, and a high number of signal discretization levels might not be available to all mobile devices, the cost of using non-binary data clearly outweighs the benefits.

5.5 Summary

Having defined the problem in Chapter 4, this Chapter studies possible DL architectures that are able to provide a position estimate given a single BFF sample. In section 5.1, it is demonstrated that the input data has similarities to images and, as such, CNNs can be used to obtain better results than conventional DNNs. Moreover, the output space can

[4] In fact, assuming the availability of enough resources, multiple instances of the proposed system can be run in parallel, targeted at specific energy consumption or accuracy requirements.

be divided into multiple coarse regions, giving rise to the hierarchical structure discussed in section 5.2.

As with the technique proposed in Chapter 3, simulating the complete system requires a complex apparatus, including ray-tracing mmWave propagation simulation, realistic environments, and DL models. Section 5.3 describes the complete simulation environment, and provides a link to working simulation code, essential to reproduce the results.

The results are then laid out in section 5.4. This section studies how the system performs as we change multiple parameters, including signal noise, minimum distance between training samples, and BFF sampling frequency. It is possible to conclude that, even using a modest set of parameters, NLOS scenarios are not a problem for the proposed system. In fact, it is able to beat the state-of-the-art accuracy for NLOS mmWave positioning, and that accuracy is on par with GNSS systems. This is particularly remarkable if we take into account that the BFF system only requires a single anchor (*i.e.*, a single BS) to work and, if the mobile device already possesses mmWave communication capabilities, no dedicated hardware is required. Nevertheless, the following Chapter proposes a further refinement to the system, and its results section will assess a critical component: energy consumption.

6

Beamformed Fingerprint Tracking

Contents

The previous Chapter described DL architectures that are able to convert a single BFF into a position estimate. They can be seen as versatile architectures, enabling position estimates whenever there is mmWave coverage. However, many practical systems request localization services during a significant amount of time, and their movement can be seen as an additional source of information. Through the inspection of the sequence of positions, it is fairly easy to categorize the movement type: pedestrians have a very limited speed, cars' steering angle is reduced, and so forth. Moreover, the system should be able to learn how to segment $\mathcal{Y}$ (*e.g.* cars shouldn't go over sidewalks, boats are limited to water), and thus push the estimates into positions coherent with their movement types. Therefore, by having information regarding past positions and the expected trajectory, the system can infer the range of physically plausible positions, and thus can enhance its position estimates.

In this section, the use of sequence-based DL architectures for the BFF positioning system is proposed. This new set of architectures aim to learn the mapping function $f : \mathcal{X}^M \mapsto \mathcal{Y}$, where M is the input sequence length (or the system's memory size). Consequently, the training set $\mathcal{T}$ is now obtained from the fixed distribution $\mathcal{D}_{\mathcal{X}^M \times \mathcal{Y}}$, where $\mathcal{X}^M$ is now the set of possible BFF sequences.

6.1 Long Short-Term Memory Networks

The default DL architecture to deal with sequences is the Recurrent Neural Network (RNN). In recent decades, multiple variants of RNNs were proposed, namely the Long Short-Term Memory (LSTM) networks [79], which were developed to help with the vanishing and exploding gradient problems that often plagued vanilla RNNs' training. LSTMs are known for their good (and often state-of-the-art) results in multiple sequence-based tasks, including indoor tracking using WiFi fingerprints [73]. Therefore, being a suitable candidate, this sub-section discusses the application of LSTMs to learn from sequences of BFFs.

Unlike DNNs, RNN-based architectures have an internal state that allows them to retain information as a sequence is being processed. This mechanism allows a model to process sequences of arbitrary length, while keeping an understanding of the chain of events. It also effectively shares the model's trained weights as it traverses the sequence, which, as mentioned in section 5.1, aids the generalization process.

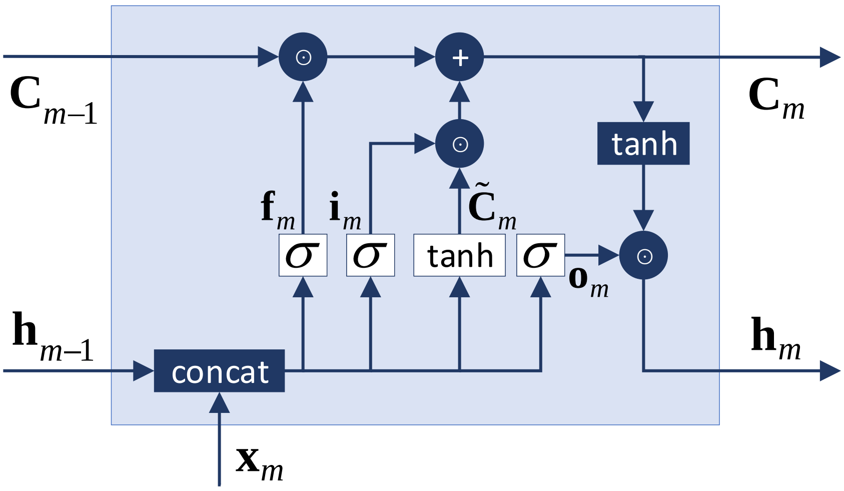


Figure 6.1: Block diagram of the m-th LSTM module, as described on equations (6.1)-(6.6). The activation functions depicted on a white background contain the learnable weights.

Each step of the sequential model can be abstracted within a LSTM module (containing multiple LSTM units), as depicted in figure 6.1. This module abstraction is in fact the consequence of unrolling the LSTM, as the weights are shared between modules. The output of the m-th LSTM module can be written as

$$\mathbf{h}_m = \mathbf{o}_m \odot \tanh\left(\mathbf{C}_m\right), \tag{6.1}$$

where $\mathbf{C}_m$ is the *cell state*, $\mathbf{o}_m$ is the *output gate*, $\odot$ denotes the Hadamard product, and $\tanh(\cdot)$ represents the hyperbolic tangent function. The output gate, containing a mixture of the current input sample being assessed and the previous module's output, selects which parts of the cell state's information are to be passed to the module output. More specifically, the output gate is written as

$$\mathbf{o}_m = \sigma\left(\mathbf{U}_o\left[\mathbf{h}_{m-1}; \mathbf{x}_m\right] + \mathbf{b}_o\right), \tag{6.2}$$

where $\sigma\left(\cdot\right)$ denotes the sigmoid function. Consistently with the previous sections' notation, $\mathbf{U}$, $\mathbf{b}$, and $\mathbf{x}$ represent weights, bias, and BFF data (as a vector), respectively.

The cell state is defined as

$$\mathbf{C}_m = \mathbf{f}_m \odot \mathbf{C}_{m-1} + \mathbf{i}_m \odot \tilde{\mathbf{C}}_m, \tag{6.3}$$

in which $\mathbf{f}_m$ represent the *forget gate*, and $\mathbf{i}_m$ the *input gate*. The forget gate controls which information should the cell state discard, relative to its own past state, while the input

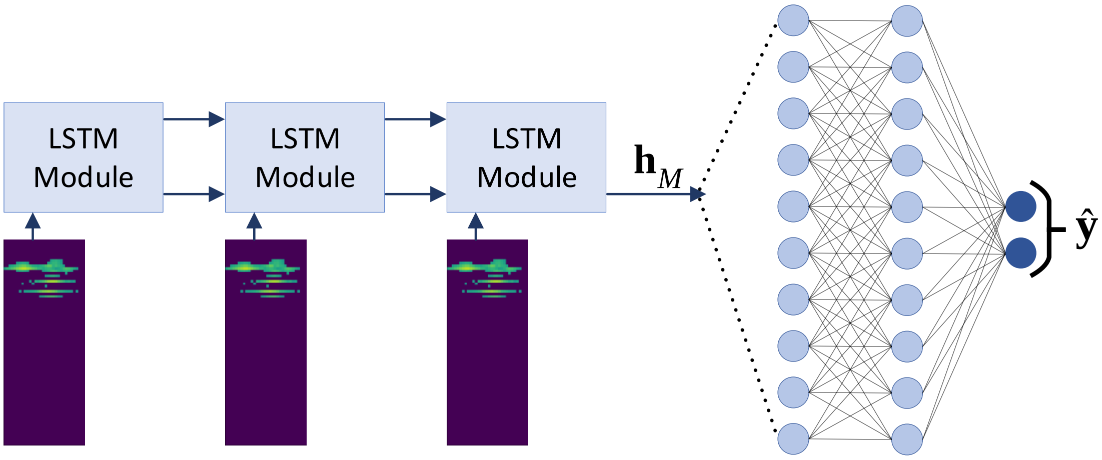


Figure 6.2: Representation of an LSTM model applied to the BFF data, with $M = 3$. Each LSTM module is as depicted in fig. 6.1, where the first module's historical inputs ($\mathbf{h}_0$ and $\mathbf{C}_0$) are randomly initialized.

gate filters the information contained in $\tilde{\mathbf{C}}_m$, which will then be added to the cell state. These two gates' expressions are given as

$$\mathbf{f}_m = \sigma\big(\mathbf{U}_f\,[\mathbf{h}_{m-1};\mathbf{x}_m] + \mathbf{b}_f\big) \tag{6.4}$$

and

$$\mathbf{i}_m = \sigma\big(\mathbf{U}_i\,[\mathbf{h}_{m-1};\mathbf{x}_m] + \mathbf{b}_i\big), \tag{6.5}$$

while the candidate values to be added to the cell state, $\tilde{\mathbf{C}}_m$, are given by

$$\tilde{\mathbf{C}}_m = \text{tahn}\big(\mathbf{U}_c\,[\mathbf{h}_{m-1};\mathbf{x}_m] + \mathbf{b}_c\big). \tag{6.6}$$

Throughout equations (6.1)-(6.6), there are two different activation functions: the sigmoid and the hyperbolic tangent. The former, whose output ranges from 0 to 1, is used as an information filter (*gates*), while the later, ranging from -1 to 1, adds the critical non-linearities, while limiting the output range of the data that is passed between LSTM modules. As shown in fig 6.2, fully connected layers are usually placed after the last LSTM module, mapping its last output vector $\mathbf{h}_M$ to the desired output information ($\hat{\mathbf{y}}$).

Compared to a traditional RNN, an LSTM adds the cell state which, as it can be seen above, adds three sets of learnable weights. However, the addition of the cell state allows the system to latch on to particular information parts, and thus improve the quality of the system's memory. In the particular case of outdoor positioning, it should help the system to retain details such as the movement category and direction, even if the user temporarily stops moving (which often happens in practice).

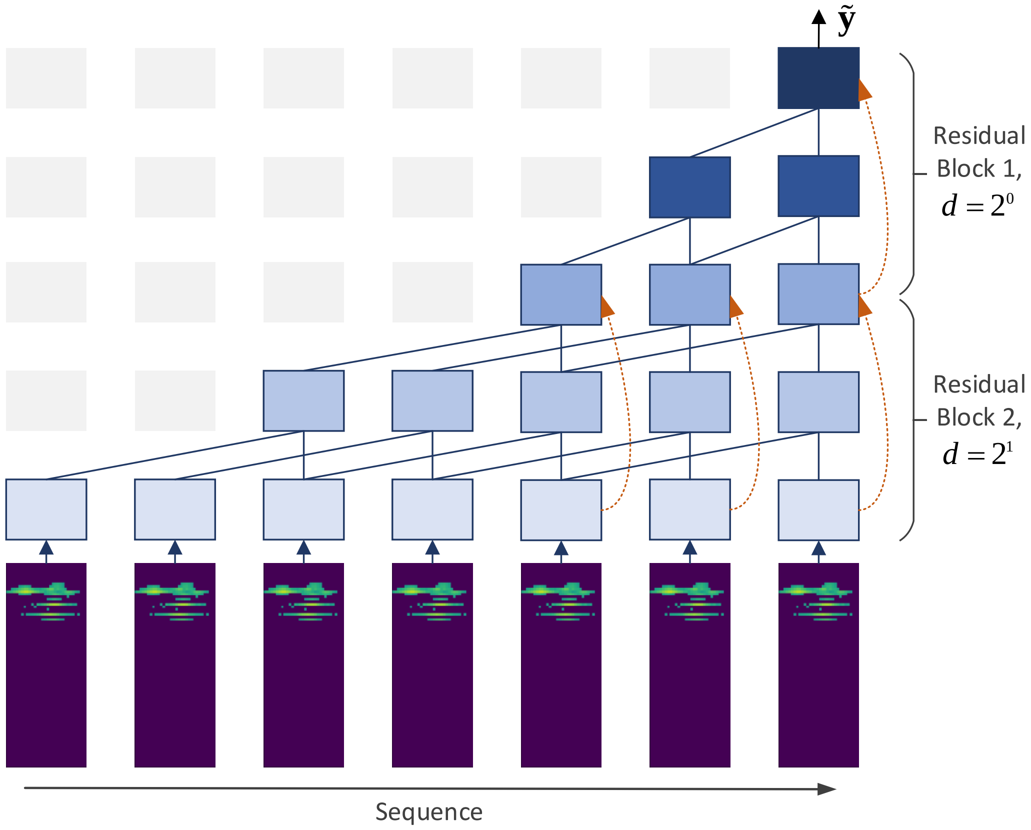


Figure 6.3: Core of a TCN model with $M = 7$, excluding the output layer after the last residual block's output ($\tilde{\mathbf{y}}$). With each subsequent residual block, the receptive field increases exponentially, due to the dilation factor d. The dashed lines depict the residual connections.

6.2 Temporal Convolutional Networks

Although LSTMs are an effective tool to learn from sequences, they are often notoriously difficult to train [80]. Moreover, as discussed in [18], there are multiple sequence-based problems for which CNN provide the best solution (e.g. audio synthesis in [81], where the convolution is applied over the time-domain). To harness the potential of the convolution operation, which is naturally suited to handle sequences, while being able to process sequences of arbitrary length, the TCN were proposed in [18]. In its original paper, TCNs surpassed LSTMs in multiple tasks where LSTMs were the state-of-the-art [18]. The work made for this Thesis is the first to apply TCNs in the context of positioning.

TCNs, when compared to a typical CNN, have three key differences. First and foremost, any non-sequence-dimension (*feature*) size mismatch between two subsequent layers is dealt through a 1D convolution [82]. This ensures that for each step in the input sequence, there is a single corresponding step in each hidden layer (as observable in fig. 6.3).

If the convolution is to be applied directly over the sequence dimension, its size can quickly become unbearable. As such, the second feature of a TCN is the introduction of dilated convolutions, which enable an exponentially large receptive field. The dilated convolution operation F on element m of the sequence $\mathbf{x}$, using a filter f, is defined as

$$F[m] = \big(\mathbf{x} *_d f\big)[m] = \sum_{l=1}^{L} f[l] \cdot \mathbf{x}[m - d \cdot l], \tag{6.7}$$

where L is the length of the dilated convolution, and d is the dilation factor. Since d is set to grow exponentially with the depth of the network, each subsequent layer can be interpreted as a *zoom out* in the sequence data, enabling the network to perceive larger sequences with few learnable parameters. If the TCN's receptive field is larger than the input sequence, the input sequence can be zero-padded.

Finally, the last key element of a TCN is the use of the residual block [83]. With the TCN's residual block, the network has access to the original input data every two dilated convolution layers, which is critical to stabilize large networks. More formally, if $\mathbf{x}$ is the input of a given residual block, its output $\tilde{\mathbf{y}}$ defined as

$$\tilde{\mathbf{y}} = a\big(\mathcal{F}(\mathbf{x}) + \mathbf{x}\big), \tag{6.8}$$

where a is an activation function, and $\mathcal{F}$ represents a series of transformations corresponding to the two dilated convolutions within the residual block (with 1D convolutions being used to match $\mathbf{x}$ to $\mathcal{F}(\mathbf{x})$, if needed). By stacking these residual blocks, a TCN is built. The output of the last residual block, $\tilde{\mathbf{y}}$, must then go through the output layer, so as to extract the desired prediction ($\hat{\mathbf{y}}$).

6.3 Simulation Apparatus

The tracking system described in this Chapter is an extension of the single-point estimate system studied in Chapter 5. As such, most of the simulation apparatus for this section is identical to that Chapter's, with one addition: the sequences of BFFs.

To generate the sequences of BFFs for LSTMs and TCNs, three types of synthetic user paths were carefully designed: static, pedestrian-like, and vehicle-like. While users in

Table 6.1: Synthetic Path Generation Parameters

Parameter Name	Pedestrian-like	Vehicle-like
Default Speed (m/s)	1.4	8.3
Maximum Speed (m/s)	2.0	13.9
Maximum Acceleration (m/s^2)	0.3	3
Maximum Direction Change ($^\circ/s$)	10.0	5.0
p(No Movement Change)	0.8	0.8
p(Full Stop)	0.1	0.02
p(Speed Change)	0.05	0.05
p(Direction Change)	0.05	0.13

static paths remain in the same position for its complete duration, users following the other two path types move according to the specifications depicted in table 6.1. The pedestrian-like paths were generated with the typical human preferred walking speed (5 km/h), but can quickly stop or change their direction. On the other hand, vehicle-like paths were generated with higher default speed (30 km/h) and maximum allowed acceleration, but with a restricted steering angle. The probabilities depicted in the bottom half of table 6.1 are applied once per second, where a *full stop* stops a user for a second, before restarting its movement in a random direction (uniformly sampled) with the default speed, and the *speed* and *direction changes* modify the existing speed or direction by a value uniformly sampled between the specified maximum and its negation (e.g. a vehicle-like path can decelerate or accelerate by an amount between -3 and 3 m/s^2).

To mimic typical civilian GNSS receivers, the sequences of BFFs are then created by drawing a noisy BFF sample once per second (*i.e.* sampled at 1 Hz), resulting in sequences as depicted in fig. 6.4. To be representative of a real scenario, where most users are moving, there is a ratio of $8:1$ moving to static paths (the moving paths are evenly distributed between pedestrian- and vehicle-like paths). The test sequences, corresponding to 20% of the generated paths, are hidden while training, to avoid a simple memorization of paths. As in the previous chapter, the total training time required for the presented results was also taken in account – training any of the assessed architectures with the defined parameters takes less than 10 hours on a Nvidia GTX 780 Ti GPU, using Google's TensorFlow framework [76]. Similarly, since reproducibility is fundamental to validate and move forward with investigation, the simulation code and the used data are also publicly available[1].

The experiments in the next section will also assess the energy efficiency of the posi-

[1] https://github.com/gante/mmWave-localization-learning

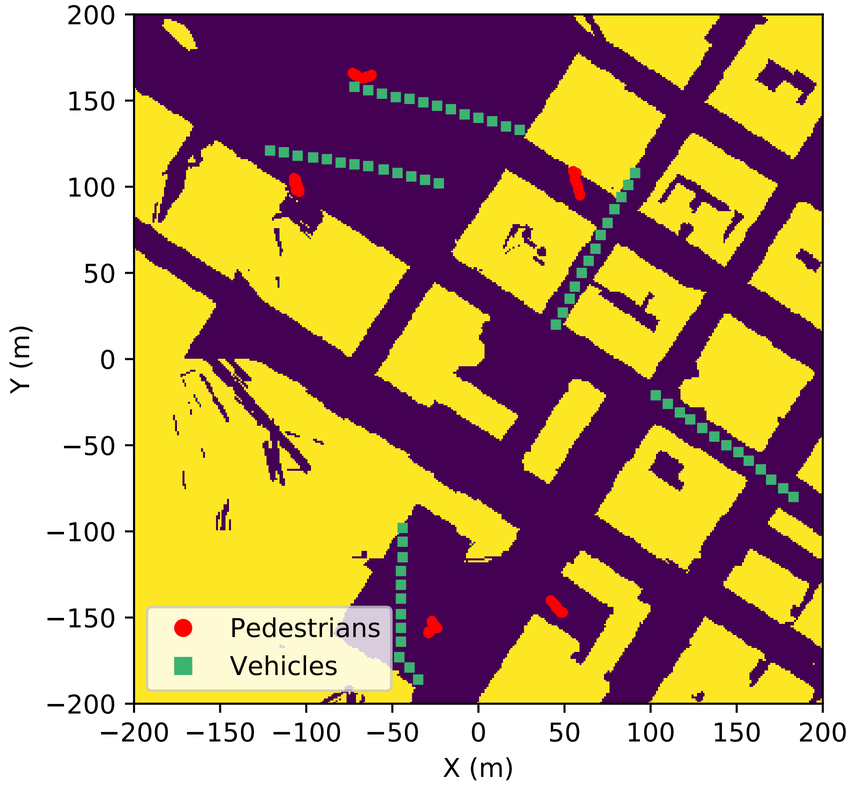


Figure 6.4: Examples of the generated sequences, sampled at 1 position per second during 13 seconds. The pedestrian-like sequences have a low average speed and can frequently change their direction, while the vehicle-like sequences display the opposite behavior. The dark area corresponds to the positions present in the BFF dataset.

tioning method, including the DL architectures from both chapters. When the position inference is done in the BS, the equations from section 4.5 will be used. However, to assess the energy consumption when the computations are to be performed in the mobile device, a low-power, edge GPU is used (Nvidia Jetson TX2 [69]). The repository with the simulation code linked above also includes all the instructions required to reproduce the energy consumption measurements for this family of devices.

6.4 Simulation Results

Throughout this Chapter, two DL architectures suited to deal with the tracking problem were presented: LSTMs and TCNs. The accuracy obtained with both architectures for multiple sequence lengths (M) is plotted in fig. 6.5, where it is clear that TCNs outperform LSTMs in the context of BFF tracking. For LSTMs, the achievable accuracy gets

Table 6.2: LSTM and TCN Hyperparameters

Parameter Name	Value for LSTMs	Value for TCNs
LSTM Units	512	—
TCN Blocks	—	$[2,3]$ (depending on M)
TCN Filter Length	—	3
TCN Features	—	512
MLP Layers	2 (512 neurons each)	0
Regression Output	Linear with 2 Neurons (2D position)	
Total # of Sequences	720918	
Sequence Length (M)	$[4, 13]$	
Epochs	Up to 100 (early stopping [77] after 5 non-improving epochs)	
Batch Size	64	
Optimizer	ADAM	
Learning Rate	5×10^{-5}	5×10^{-4}
Learning Rate Decay	0.995	

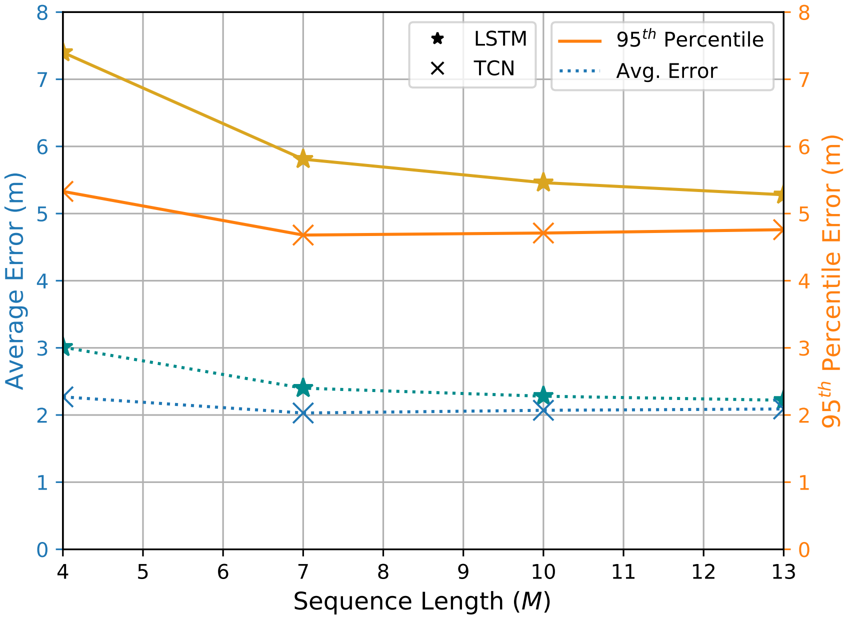


Figure 6.5: Prediction error comparison of LSTMs and TCNs for multiple values of M, considering an average noise value ($\sigma = 6$ dB). On the considered BFF tracking problem, TCNs outperform LSTMs, especially for shorter sequences.

better as M increases, with visible diminishing returns. TCNs, on the other hand, saturate their performance with short sequences ($M = 7$), and their performance slightly decays with longer sequences.

In fig. 6.6, the performance of the TCN architecture for multiple noise values is presented, considering the best performing sequence length ($M = 7$). As expected, higher

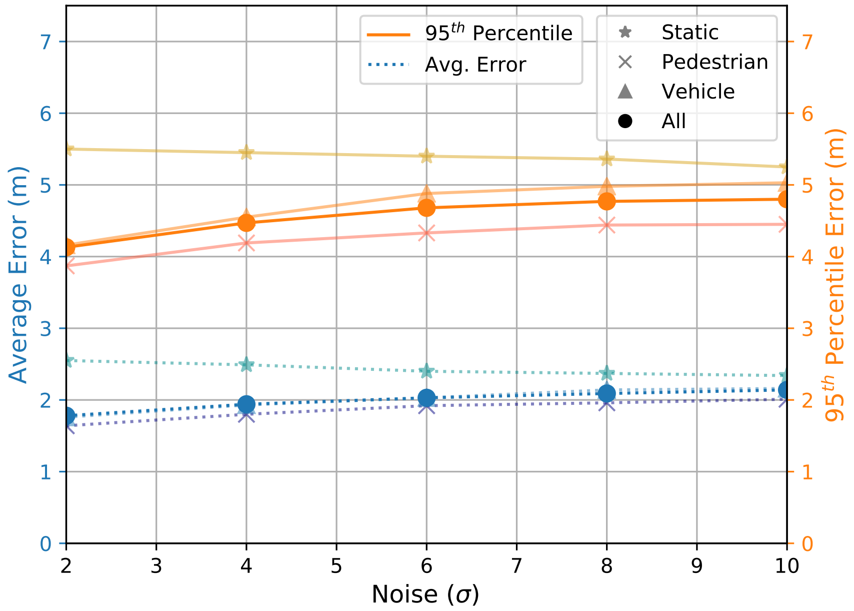


Figure 6.6: Performance of the TCN architecture by sequence type for multiple noise values, with $M = 7$. Due to the higher number of moving paths seen during training, the system is better suited to track moving targets.

noise values correspond to higher estimation errors. At a low noise level ($\sigma = 2$ dB), the TCN can achieve average and 95^{th} percentile errors of 1.78 and 4.13 m, respectively, which corresponds to an average error reduction of 46% (48% for the 95^{th} percentile) when compared to the best results for the single BFF positioning system, as shown in Chapter 5. Also in fig. 6.6, the error for the three types of generated sequences is shown. Static sequences have slightly worse accuracy, likely due to the unbalanced sequence type distribution, and to the fact that pedestrian-like sequences are quite similar to them. That performance difference also depends on the noise level: with $\sigma = 2$ dB the static sequences' error is $\sim 35\%$ larger, compared to $\sim 15\%$ with $\sigma = 10$ dB. Therefore, a high noise value acts as a strong regularizer, forcing the model to generalize and resulting in smaller error discrepancies. It is important to mention that these results were obtained with randomly generated synthetic paths and, as such, no movement type segmentation nor traffic rules were included in the data. A real-world dataset would very likely observe these phenomena, which would be an additional source of information to aid the learning system.

The positioning task often sees error spikes, which are undesirable. In the results described for the single BFF positioning method, it is clear that there are significant spikes,

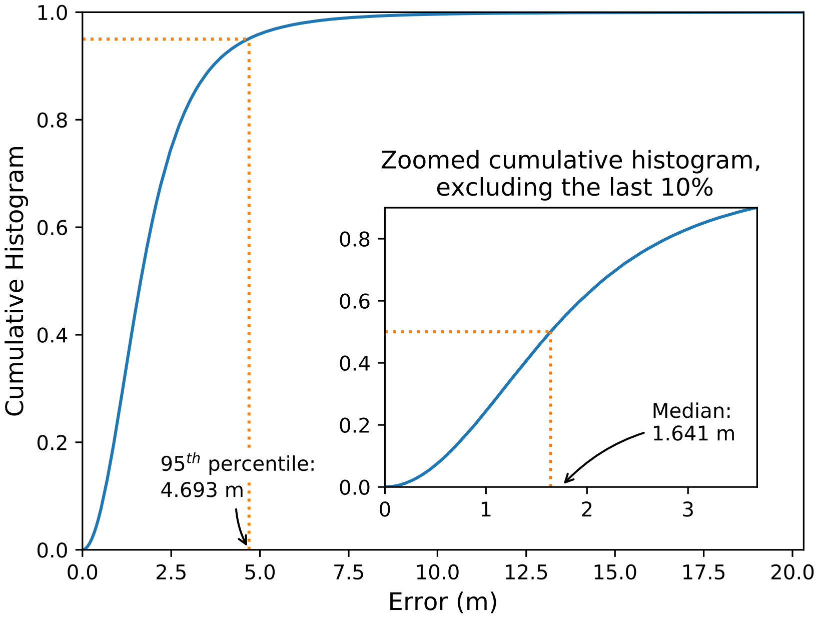


Figure 6.7: Cumulative histogram of the error obtained for the TCN architecture, assuming sequences of 7 BFFs and a noise σ of 6 dB. Due to the use of multiple BFFs per position estimate, the model is better suited to deal with occasional noise spikes in the samples, resulting in moderate error for the top percentiles.

as its RMSE is significantly higher than its average error (19.7 m *vs.* 4.73 m, for $K = 64$ and $\sigma = 6$ dB). From a statistical point of view, the use of sequences should attenuate that issue, as it is very unlikely that the multiple BFFs gathered throughout several seconds all suffer from a noise spike. By assessing fig. 6.7, it is visible the positive impact of the tracking methods, with the error peaking at 20 m. In fact, the RMSE for a sequence of 7 BFFs and a noise of 6 dB is 3.64 m — 5.41× smaller than the previous results in Chapter 5, and 9.62× (almost an order of magnitude) smaller than the results in [59].

Throughout section 4.3, two features were pointed out as major influences of the BFF positioning accuracy: the number of received paths, and the selected sampling frequency. Considering the used frequency of 20 MHz, as well as the maximum theoretical spatial resolution per path, given in eq. (4.2), it is interesting to notice that the proposed system does leverage the information from multiple paths, as its error is far below the single-path limit of 15 m. In fig. 6.8, the distribution of sequences and errors by the total number of detected path clusters (*i.e.*, the number non-zero entries for all BFFs in a sequence) is shown. Although there are visible diminishing returns, the number of received paths has a positive impact on the prediction error, as expected.

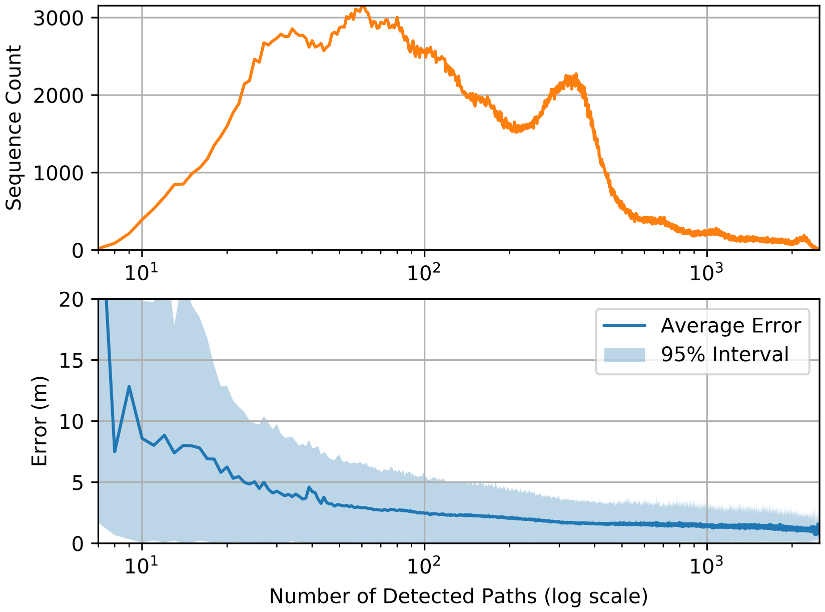


Figure 6.8: Distribution of sequences and errors by the total number of detected paths (*i.e.*, sum of non-zero entries in the BFFs), assuming sequences of 7 BFFs and a noise σ of 6 dB. When few paths are detected, the prediction error soars.

Table 6.3: Number of learnable parameters, training time, and inference throughput in a desktop GPU for the tested DL architectures

DL Architecture	L. Parameters	Training Time	Inference Throughput
CNN	3.37×10^6	328 mins	19.15×10^3 predictions/s
Hierarchical CNN	220×10^6	651 mins	9.439×10^3 predictions/s
LSTM	7.15×10^6	453 mins	4.249×10^3 predictions/s
TCN	7.67×10^6	432 mins	3.849×10^3 predictions/s

The last set of accuracy results is shown in fig 6.9, where the average prediction error was computed for each position. Comparing with fig 3.2a, it is possible to verify that the system was able to return an accurate estimate whenever there is mmWave coverage. Moreover, the obtained error has no visible dependence on whether the position was in LOS or NLOS. As such, the proposed BFF tracking system achieves state-of-the-art accuracy for NLOS positioning with mmWave.

Having thoroughly discussed the accuracy achievable by the considered DL architectures, it is also important to compare the corresponding computational cost. To that end, Table 6.3 compares three important attributes: the models' size in terms of learnable parameters, the time required to train them, and their inference throughput rate. The reported results were averaged over five runs with an Nvidia GTX 780 Ti GPU, using

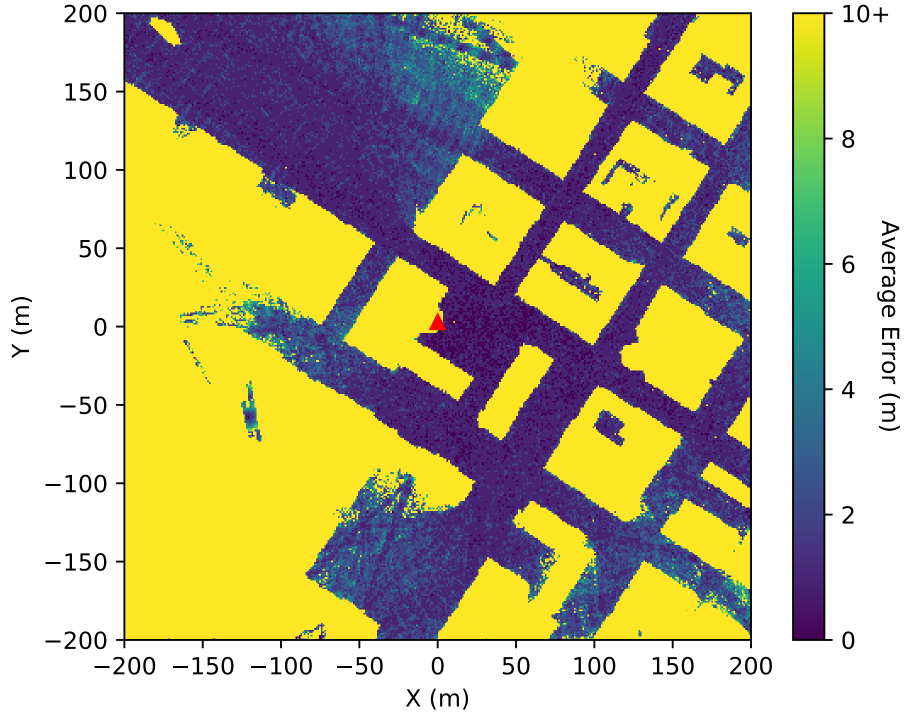


Figure 6.9: Average error per covered position for the TCN architecture, assuming sequences of 7 BFFs and a noise σ of 6 dB. Given that the transmitter is at the center of the image (red triangle), it is possible to confirm that being in a NLOS position is not a constraint for the proposed system.

Google's TensorFlow framework [76] and the hyperparameters considered throughout most of this subsection ($\sigma = 6$ dB, $K = 64$, $M = 7$, and the hyperparameters reported in Tables 5.2 and 6.2).

Observing the top half of Table 6.3, it is noticeable that the hierarchical CNN uses far more learnable parameters than its non-hierarchical counterpart, while also requiring roughly double computational time for training and inference. The cause is straightforward, as the hierarchical CNN contains $K + 1$ models, and thus has roughly as many times more learnable parameters when compared to the standard CNN. The hierarchical CNN is therefore an expensive source of accuracy gain, which should only be used if spare resources are available.

In the presence of tracking, the choice between LSTMs and TCNs is less clear-cut: while TCNs have better accuracy and training time, they also have more learnable parameters and a lower inference throughput. It is also important to notice that TCNs only have a shorter training time because early stopping is employed, and the performance

Table 6.4: Energy consumption of inference for the tested DL architectures on a mobile GPU (Nvidia Jetson TX2). The power values are in mW.

	Samples/s	GPU Power	Memory Power	Total Power	mJ/sample
CNN	3194.4	780	1141	3820	1.196
HCNN	1577.3	773	1210	4010	2.542
TCN	1367.1	4284	2271	9259	6.773
LSTM	1833.4	3747	2273	8682	4.735

on the validation set converges in fewer epochs. The main cause for this is the aforementioned difficulty in training LSTMs. Another interesting observation is that the tracking systems assessed in this Thesis contain roughly two times more learnable parameters than the standard CNN, and have less than half the accuracy error. This demonstrates that LSTMs and TCNs can build efficient representations of the system model.

6.4.1 Comparison with the State-of-the-Art

To finalize this Thesis' experimental results, the device-side energy efficiency of the BFF position estimates is assessed. This quantity, the energy-efficiency of a position estimate at the device-side, often dictates whether a positioning method can be deployed or not. The current positioning methods already place a significant energy strain to powerful mobile devices such as smartphones [84], let alone to smaller devices with more restricted energy sources. Therefore, to replace the *de facto* positioning method (A-GPS MSB), a new contender must provide a higher energy-efficiency with at least similar accuracy levels (as the existing accuracy is good enough for most uses).

As mentioned in the previous sections, the proposed system has two modes of operation: either the mobile device sends the BFFs to the BS, delegating the inference process, or the mobile device computes the position estimate itself. The first mode of operation, where the mobile device can be unaware of the methods used to estimate its position, has its energy requirements independent of the DL architecture used. The equations shown in section 4.5 are evaluated with the parameters from table 5.1. Considering $P_{Rx} = 125$ mW [67] and $T_{guard} = 2.9$ μs, so that a PDP is collected every 7 μs, the system would require 0.224 mJ to obtain the data regarding the received radiation. The previous value considers $B_{Tx} = 32$ and $B_{Rx} = 8$, as discussed in the previous apparatus sections. From the simulations performed, the average number of non-zero entries per sample (V) ranged from 63.38 to 68.62, for $\sigma = 10$ dB and $\sigma = 0$ dB, respectively. Therefore, assuming

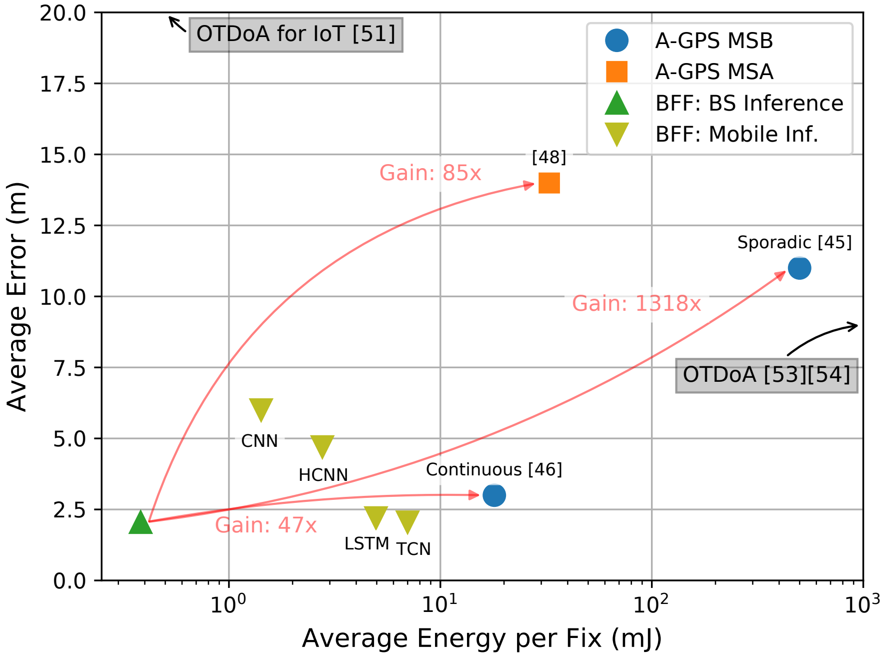


Figure 6.10: Average error *vs* average energy required per position fix for the positioning technologies discussed in this paper. The proposed system is plotted for its two operation modes, depending on where the DL inference is computed. As it is observable, the proposed system has an accuracy comparable to A-GPS, while achieving energy efficiency gains exceeding 47× per position fix.

a network energy efficiency of 0.2 μJ per transmitted bit [68], the system would need between 0.152 mJ ($\sigma = 10$ dB) and 0.165 mJ ($\sigma = 0$ dB) to upload the gathered information to the BS, on average. Combining both, the mobile device would need to spend between 0.376 and 0.389 mJ per position fix.

During the second mode of operation, the computations happen at the mobile device, and thus the energy consumption greatly depends on the DL architecture used. The DL architectures mentioned throughout chapters 5 and 6 were implemented in a mobile GPU (Nvidia Jetson TX2 [69]), with the results shown in table 6.4. Note that the energy required to obtain the BFF (0.224 mJ) must be added to the values shown in the table, representing the cost of inference for each position estimate. Although this mode of operation seems to be less energy-efficient, the used GPU is still over-dimensioned for the problem (only a few samples per second are required), and there are known techniques to reduce the energy consumption at the inference stage, as elaborated in section 7.2.

In order to comparatively assess the energy consumption of the proposed methods for positioning and tracking, the energy consumption values for other positioning methods are revisited[2]. For MSB A-GPS systems, the used data was taken directly from [45] and [46], which correspond to two state-of-the-art low-power A-GPS chips. Since the periodicity of the position fixes has a great impact on the energy consumption and accuracy of the MSB A-GPS method, two data points were considered: one for continuous measurements of one fix per second, resulting in full A-GPS accuracy and an average energy consumption of 18 mJ [46], and another for sporadic measurements (once per minute), with decreased accuracy and an average energy consumption of 504 mJ [45]. When evaluating MSA A-GPS systems, the majority of energy consumption goes to the uplink transmission, which depends on the network used. Therefore, to enable a fair comparison, it is assumed that the MSA A-GPS system also has access to energy efficient mmWave networks. In order to obtain a position fix, the system in [48] takes binary samples of the GNSS signal at 16.368 MHz during 10 ms. Considering a network energy efficiency of 0.2 μJ per uploaded bit [68], that system would require 32.736 mJ to obtain a position fix with an average error of 14 m.

Putting all together, fig. 6.10 plots the energy consumption versus the accuracy for the aforementioned methods. For better visualization, all data points for BFF positioning use the same medium noise value, $\sigma = 6$ dB, and for the case where the inference is done at the BS, the most accurate DL architecture is used. For all sequence-based methods, a sequence length of $M = 7$ is considered. When compared to the assessed A-GPS implementations, the BFF positioning system with inference made at the BS shows energy efficiency gains of 47× for continuous measurements (*vs* A-GPS MSB), and 85× for sporadic position fixes (*vs* A-GPS MSA using a mmWave network), while keeping slightly better accuracy levels. Furthermore, the proposed method is available whenever there is mmWave coverage, while requiring no additional hardware at devices with mmWave capabilities. As such, in this Thesis we can conclude that BFF-based positioning methods can dethrone GNSS-based methods as the default commercial positioning systems, with their much higher energy efficiency.

6.5 Summary

Many positioning systems, such as all GNSS-based approaches, leverage the fact that movement is bound by physical laws, particularly with respect to velocity and accelera-

[2] To the best of our knowledge, the energy consumption for other mmWave positioning methods has yet to be studied.

tion. Those techniques are often referred as *tracking*, and can increase the performance of a positioning system significantly. This chapter brings the same techniques to augment the solutions adopted in Chapter 6, through the use of models that can digest sequences of BFFs.

Two DL architectures are considered in this Chapter. Section 6.1 presents LSTMs, the standard modern approach to learn sequences. However, LSTMs are notoriously difficult to train and very sensitive to the hyperparameters selected. Consequently, section 6.2 discusses a potential alternative, called TCN.

The apparatus to simulate a model that is able to track a user is quite similar to the apparatus in the previous Chapter, except for one distinction: the movement of the users must also be simulated (as covered in section 6.3). As expected, in section 6.4 we can see that the tracking approaches are far superior than their non-tracking counterparts. The new accuracy results place the BFF positioning system even farther away from other mmWave positioning systems that can handle NLOS positions (especially in terms of RMSE), and can even beat GNSS systems.

After the accuracy of the BFF system was validated, one question remained: how does its energy consumption fare against known positioning methods'? The last part of section 6.4 concludes that a BFF positioning system can beat all known alternatives by a significant margin, especially if DL inference is delegated to the BS. To conclude this Part, subsection 6.4.1 places all obtained results in perspective: the positioning approach proposed in this Thesis has better accuracy than the incumbent positioning method (A-GPS MSB), while delivering massive energy efficiency gains.

Part IV

Conclusions

7 Conclusions and Future Work

Contents

The introduction of millimeter wave communications in the context of 5G will open up a significant amount of bandwidth, resulting in massive theoretical improvements. However, bringing those improvements to practice is not a trivial task, as the physics underlying the radiation propagation change dramatically. This Thesis tackled two of the major millimeter wave communication problems under active research: beamforming selection and positioning.

7.1 Conclusions

In the second Part of this Thesis, two new beam search algorithms were proposed. These methods create a set of likely viable beamforming configurations, given the users' positions and a pre-constructed model, which can be tested first as part of the standard beam search procedure (as opposed to a random search). Although they do not guarantee a feasible solution (let alone the optimal solution), they are the first algorithms to be able to generate highly plausible multi-user spacial-multiplexing beamforming solutions with a high number of served users under a very limited time frame. In particular, the simulations shown in this Thesis demonstrate that non-optimized code can generate beamforming combinations for more than 10 simultaneous users under 600 μs, far below the 1 ms objective for 5G, in very realistic simulation environments. While the suggestions are prone to error, the global result should yield a considerable gain, since they greatly reduce the required search space. With a reduced search space, larger codebooks become possible, allowing for higher area spectral efficiency capabilities and paving the way for next generation communication networks.

In the context of outdoor positioning, using millimeter waves means that the typical geometrical approaches are no longer reliable for NLOS positions, which are a common occurrence. The third Part of this Thesis focuses on the concept of beamformed fingerprints, which was coined in a recent work, proposing a new positioning approach that is able to outmatch many existing positioning techniques in several aspects.

The beamformed fingerprint, which is able to gather detailed latent information with respect to the surrounding environment, can be used as the data source to train a deep learning model, which is able to convert subsequent beamformed fingerprints into a device's position. The resulting models, which can either do *ad hoc* positioning or tracking (depending on the model architecture), are the first systems to achieve NLOS mmWave outdoor positioning accuracies on par with GNSS systems, outclassing all other mmWave positioning methods in this regime. In fact, the simulated tracking systems are able to deliver average and 95^{th} percentile errors of 1.78 and 4.13 m, respectively, which are considerably below the 3 and 7 m (again, average and 95^{th} percentile errors) for A-GPS MSB commercial systems.

Demonstrating the accuracy levels of a positioning method is a critical milestone, but no positioning method is actually deployed if it is not practical. Consequently, a significant part of this Thesis is devoted to practical aspects of the beamformed fingerprint positioning method, with special focus on the mobile device. The signal acquisition at the mobile device is assumed to be done with bandwidth levels already used by LTE networks, which are far below the expected for mmWave communication channels, and assumes simple binary samples of the signal existence. This means that any mobile device with mmWave capabilities should be able to use the proposed method straight away (as opposed to GNSS approaches, which require dedicated legacy hardware). The simplicity of the beamformed fingerprint signal has another critical aspect: if the positioning inference is delegated to the BS, the mobile device only needs to spend less than 0.4 mJ per position fix, which is approximately 47 times more energy efficient than low-power A-GPS MSB commercial systems. These gains are even bigger for sporadic position fixes (at least 85 times more efficient), and the proposed system would not suffer from accuracy degradation under that regime, as opposed to GNSS systems. Finally, as the cherry on top of the cake, the beamformed positioning system only requires a single BS to operate, as opposed to at least three (as in most methods), which means that it can deliver position estimates whenever there is mmWave coverage. Weighing in all benefits, the proposed beamformed positioning method proposed in this Thesis has all the arguments to dethrone GPS as the standard commercial positioning method.

7.2 Future Work

In Section 4.3, the fingerprint data was established as the critical aspect of a fingerprint positioning method, and thus it is one of the key subjects of this Thesis. However, as seen in the sections 5.4 and 6.4, the used Machine Learning (ML) method also plays an important role in the outcome, both in terms of achievable accuracy and energy efficiency. As such, this section lists recent techniques that can potentially be used to improve the obtained results, as future work.

The idea of data segmentation was developed throughout section 5.2 and validated in section 5.4. The first stage of the hierarchical CNN is a classifier where the probabilities of each sub-region matter, and thus the calibration of its output would likely yield better downstream results [85]. However, while an explicit hierarchical representation of the data is helpful, that representation also requires one additional step when training the system, and the optimal representation of the hierarchy might change over time. To handle this problem, the concept of manifold regularization [86–91] can be used, where better representations are learned from the data while training the model, through rank minimization of the observed results in the hidden layers. More specifically, if the output matrix for a set of inputs at a given layer is denoted by **N**, the rank minimization problem can be written as

$$\min_{\mathbf{L},\mathbf{E}} \text{rank}(\mathbf{L}) + \lambda ||\mathbf{E}||_l, \ s.\ t.\ \mathbf{N} = \mathbf{L} + \mathbf{E}, \tag{7.1}$$

where **L** is **N**'s low rank approximation, **E** is the approximation error, $\lambda > 0$ is an hyperparameter that controls the tolerance to approximation errors, and $|| \cdot ||_l$ indicates a certain regularization strategy (e.g. Frobenius Norm). The rank minimization problem is known to be NP-hard, but fortunately the nuclear norm ($|| \cdot ||_*$) can be used as a relaxation of the problem, as minimizing it corresponds to the minimization of the rank's convex envelope [92]. Therefore, eq. (7.1) can be rewritten as

$$\min_{\mathbf{L},\mathbf{E}} ||\mathbf{L}||_* + \lambda ||\mathbf{E}||_l, \ s.\ t.\ \mathbf{N} = \mathbf{L} + \mathbf{E}, \tag{7.2}$$

where

$$||\mathbf{L}||_* = \text{trace}\left(\sqrt{\mathbf{L}^*\mathbf{L}}\right). \tag{7.3}$$

The exact implementation details to obtain the low-rank approximation **L** can vary, with methods based on Laplacian matrices and Augmented Lagrangian Multipliers (ALMs) being used in the aforementioned references. Solving the manifold regularization problem through ALM is mandatory for DNNs trained with large datasets on GPUs, as it is compatible with batch updates [93] (as opposed to solving the problem through Laplacian matrices, which would result in matrices too large to fit in GPU memory). With manifold regularization, the hierarchical representation of the data can be implicitly incorporated in the model, usually resulting in improved results.

Manifold regularization is also tied to implementations of multimodal and multi-task learning [88, 94, 95]. In practical scenarios, there will be multiple positions that are covered by more than one BS — multimodal learning can help training a model from multiple data sources, while multi-task learning would enable a single model for an area covered by several BSs. Therefore, by being able to train a model with an unified loss function $\mathcal{L}$ that includes manifold regularization, the system can potentially capture more information about the target area, and thus yielding better predictions.

A different concept that might also result in improved models is attention [96–98], which is typically applied to sequences. With the attention mechanism, a model can learn to focus on subtle details of the data sequence, and more easily digest long data sequences from heterogeneous patterns. To apply attention over a packed sequence of vectors $\mathbf{X} \in \mathbb{R}^{M \times d_x}$ (where d_x is the length of each vector), three sets of learnable weights are needed: $\mathbf{U^Q}$, $\mathbf{U^K}$, and $\mathbf{U^V}$, all d_x by d_k matrices (where d_k is an hyperparameter). By multiplying $\mathbf{X}$ by those weights, we obtain $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$, which stand for *query*, *key*, and *value*, respectively. The output attention matrix, which will be used as an input to further NN layers, is then given by

$$\text{Attention}\left(\mathbf{Q}, \mathbf{K}, \mathbf{V}\right) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{7.4}$$

Each row of the attention output is a weighted sum of the of the rows in the value matrix, where the weights are given by the softmax of a score (in this case, a scaled dot product) between the keys and the considered query (row). Intuitively, we obtain how relevant is each element in the sequence to predict a target at the sequence member under evaluation. The attention mechanism can be expanded into multi-head attention [97], where each *head* can be seen as a traditional attention element. The multi-head attention enables the model to focus on multiple details over diverse sequence elements, and can be written as

$$\text{MultiHead} = \text{concat}\left(\text{head}_1, \ldots, \text{head}_h\right)\mathbf{U^O}, \tag{7.5}$$

where

$$\text{head}_i = \text{Attention}\left(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i\right), \tag{7.6}$$

h is the number of heads, and $\mathbf{U^O}$ is a learnable projection matrix, whose objective is to project the output of all attention heads into the size of a single head's output, so that the multi-head attention output size is invariant to the number of heads. Please note that each attention head has its own set of query, key, and value weights, and thus they can learn to focus on different details. In the context of this problem, the most straightforward application of the attention mechanism is over tracking, i.e., over the sequence of

BFFs. From a high-level perspective, it should enable the model to distinguish subtle trajectory changes – e.g. if a ground vehicle has moved to the right-most lane just before an intersection, it is likely that it will turn right in that intersection.

The attention technique described above increases significantly the complexity of the model, which also translates to lower energy efficiency. It is also possible to alter the model in the opposite direction, sacrificing some accuracy in order to obtain higher model energy efficiency. In [99], the authors show that it is possible to train a smaller model from a larger model, retaining most of the larger model's precision, in a technique called distillation. The drawback of using distillation is that not only some precision is lost, but also training becomes more expensive – given that a model per BS is needed, the cost can be non-negligible. Finally, if the hardware used for the DL inference allows it, it is possible to trade some accuracy for energy efficiency through operations with fewer bit resolution. In fact, it was shown in [100] that representing the weights of a DNN with a single bit can result in very competitive results. This particular solution is particularly beneficial if the BFF position inference is to be computed at the mobile device, as the cost for transmitting and storing the networks' weights is drastically reduced.

Bibliography

[1] J. Wells, "Faster than fiber: The future of multi-g/s wireless," IEEE Microwave Magazine, vol. 10, no. 3, pp. 104–112, May 2009.

[2] S. Ishii, M. Kinugawa, S. Wakiyama, S. Sayama, and T. Kamei, "Rain attenuation in the microwave-to-terahertz waveband," Wireless Engineering and Technology, vol. 07, pp. 59–66, 01 2016.

[3] P. Pirinen, "A brief overview of 5G research activities," in 5G for Ubiquitous Connectivity (5GU), 2014 1st International Conference on, Nov 2014, pp. 17–22.

[4] 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.

[5] T. S. Rappaport, R. W. Heath, R. C. Daniels, and J. N. Murdock, Millimeter Wave Wireless Communications. Prentice Hall, 2014.

[6] J. C. Liberti and T. S. Rappaport, Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications. Prentice Hall, 1999.

[7] C. D. Durgin, Space-Time Wireless Channels. Prentice Hall, 2003.

[8] S. Sun, T. S. Rappaport, R. W. Heath, A. Nix, and S. Rangan, "Mimo for millimeter-wave wireless communications: beamforming, spatial multiplexing, or both?" IEEE Communications Magazine, vol. 52, no. 12, pp. 110–121, December 2014.

[9] 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, no. 2, February 2014.

[10] J. Gante, G. Falcao, and L. Sousa, "Data-aided fast beamforming selection for 5G," in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), April 2018, pp. 1183–1187.

[11] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: From 1G to 5G," IEEE Communications Surveys Tutorials, vol. 20, no. 2, pp. 1124–1148, Secondquarter 2018.

[12] K. Witrisal, P. Meissner, E. Leitinger, Y. Shen, C. Gustafson, F. Tufvesson, K. Haneda, D. Dardari, A. F. Molisch, A. Conti, and M. Z. Win, "High-accuracy localization for assisted living: 5G systems will turn multipath channels from foe to friend," IEEE Signal Processing Magazine, vol. 33, no. 2, pp. 59–70, March 2016.

[13] M. Koivisto, A. Hakkarainen, M. Costa, P. Kela, K. Leppanen, and M. Valkama, "High-efficiency device positioning and location-aware communications in dense 5G networks," IEEE Communications Magazine, vol. 55, no. 8, 2017.

[14] O. Kanhere and T. S. Rappaport, "Position Locationing for Millimeter Wave Systems," 2018 IEEE Global Communications Conference, Dec. 2018. [Online]. Available: https://arxiv.org/abs/1808.07094

[15] J. Gante, G. Falcao, and L. Sousa, "Beamformed fingerprint learning for accurate millimeter wave positioning," in 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), Aug 2018, pp. 1–5.

[16] J. Gante, G. Falcao, and L. Sousa, "Enhancing beamformed fingerprint outdoor positioning with hierarchical convolutional neural networks," in 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2019, pp. 1473–1477.

[17] J. Gante, G. Falcão, and L. Sousa, "Deep Learning Architectures for Accurate Millimeter Wave Positioning in 5G," Neural Processing Letters, Aug 2019. [Online]. Available: https://doi.org/10.1007/s11063-019-10073-1

[18] S. Bai, J. Zico Kolter, and V. Koltun, "An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling," arXiv e-prints, p. arXiv:1803.01271, Mar. 2018.

[19] J. Gante, L. Sousa, and G. Falcão, "Dethroning GPS: Low-Power Accurate 5G Positioning Systems using Machine Learning," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 2020.

[20] T. S. Rappaport, Wireless Communications, Principles and Practice. Prentice Hall, 2002.

[21] 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, no. 8, pp. 1390–1436, Aug 2011.

[22] H. T. Friis, "A note on a simple transmission formula," Proceedings of the IRE, vol. 34, no. 5, pp. 254–256, May 1946.

[23] R. Crane, "Prediction of attenuation by rain," IEEE Transactions on Communications, vol. 28, no. 9, pp. 1717–1733, Sep. 1980.

[24] C. Balanis, Antenna Theory: Analysis and Design. Wiley, 2015. [Online]. Available: https://books.google.pt/books?id=PTFcCwAAQBAJ

[25] H. Zhao, R. Mayzus, S. Sun, M. Samimi, J. K. Schulz, Y. Azar, K. Wang, G. N. Wong, F. Gutierrez, and T. S. Rappaport, "28 GHz millimeter wave cellular communication measurements for reflection and penetration loss in and around buildings in New York city," in 2013 IEEE International Conference on Communications (ICC), June 2013, pp. 5163–5167.

[26] ITU-R, "Effects of building materials and structures on radiowave propagation above about 100 mhz," https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.2040-1-201507-I!!PDF-E.pdf, accessed: 26th of October, 2019.

[27] ITU-R, "Propagation by diffraction," https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.526-14-201801-I!!PDF-E.pdf, accessed: 26th of October, 2019.

[28] S. Kutty and D. Sen, "Beamforming for millimeter wave communications: An inclusive survey," IEEE Communications Surveys Tutorials, 2016.

[29] Y. Kim, H. Y. Lee, P. Hwang, R. K. Patro, J. Lee, W. Roh, and K. Cheun, "Feasibility of mobile cellular communications at millimeter wave frequency," IEEE Journal of Selected Topics in Signal Processing, vol. 10, no. 3, April 2016.

[30] T. Obara, Y. Inoue, Y. Aoki, S. Suyama, J. Lee, and Y. Okumurav, "Experiment of 28 GHz band 5G super wideband transmission using beamforming and beam tracking in high mobility environment," in 2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Sep. 2016, pp. 1–5.

[31] T. E. Bogale, L. B. Le, and X. Wang, "Hybrid analog-digital channel estimation and beamforming: Training-throughput tradeoff," IEEE Transactions on Communications, vol. 63, no. 12, Dec 2015.

[32] E. Bjornson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: ten myths and one critical question," IEEE Communications Magazine, vol. 54, no. 2, pp. 114–123, February 2016.

[33] J.-H. Lee and J.-Y. Lee, "Optimal beamforming-selection spatial precoding using population-based stochastic optimization for massive wireless MIMO communication systems," Journal of the Franklin Institute, vol. 354, no. 10, pp. 4247–4272, 2017.

[34] "Qualcomm Snapdragon X50 modem information document," https://www.qualcomm.com/media/documents/files/qualcomm-snapdragon-x50-5g-modem-infographic.pdf, accessed: 31st of December, 2019.

[35] R. Deng, S. Chen, S. Zhou, Z. Niu, and W. Zhang, "Channel fingerprint based beam tracking for millimeter wave communications," 2019.

[36] M. Dong et al., "Simulation study on millimeter wave 3D beamforming systems in urban outdoor multi-cell scenarios using 3D ray tracing," in 2015 IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Aug 2015.

[37] S. Baek et al., "Comparison analysis of outdoor channel characteristics at 28 GHz and 2 GHz using 3D ray-tracing technique," in 2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall), Sept 2014.

[38] C. Wang et al., "Ray-tracing based performance evaluation of 5G mmwave massive MIMO in hotspots," in 2016 International Symposium on Antennas and Propagation (ISAP), Oct 2016.

[39] J. Brady and A. Sayeed, "Beamspace mu-mimo for high-density gigabit small cell access at millimeter-wave frequencies," in 2014 IEEE 15th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), June 2014, pp. 80–84.

[40] Y. Azar, G. N. Wong, K. Wang, R. Mayzus, J. K. Schulz, H. Zhao, F. Gutierrez, D. Hwang, and T. S. Rappaport, "28 GHz propagation measurements for outdoor cellular communications using steerable beam antennas in New York city," in 2013 IEEE International Conference on Communications (ICC), June 2013, pp. 5143–5147.

[41] "Open source new york 3D building model," http://www1.nyc.gov/site/doitt/initiatives/3d-building.page, accessed: 19th of February, 2019.

[42] "Wireless InSite web-page," https://www.remcom.com/wireless-insite-em-propagation-software/, accessed: 19th of February, 2019.

[43] Z. Zhang, J. Ryu, S. Subramanian, and A. Sampath, "Coverage and channel characteristics of millimeter wave band using ray tracing," in 2015 IEEE International Conference on Communications (ICC), June 2015, pp. 1380–1385.

[44] F. van Diggelen, A-GPS: Assisted GPS, GNSS, and SBAS. Artech House, 2009.

[45] "Super-E: low power and good performance (white paper)," https://www.u-blox.com/en/white-papers, accessed: 19th of February, 2019; Requires registration.

[46] "MediaTek MT 3339 datasheet," https://labs.mediatek.com/en/chipset/MT3339, accessed: 19th of February, 2018.

[47] S. C. Wu, W. I. Bertiger, D. Kuang, S. Nandi, L. J. Romans, and J. M. Srinivasan, "MicroGPS for low-cost orbit determination," TDA Progress Report, vol. 42-131, Nov. 1997.

[48] T. Nguyen Dinh and V. La The, "A novel design of low power consumption GPS positioning solution based on snapshot technique," in 2017 International Conference on Advanced Technologies for Communications (ATC), Oct 2017, pp. 285–290.

[49] N. Vallina-Rodriguez, J. Crowcroft, A. Finamore, Y. Grunenberger, and K. Papagiannaki, “When assistance becomes dependence: Characterizing the costs and inefficiencies of A-GPS,” SIGMOBILE Mob. Comput. Commun. Rev., vol. 17, no. 4, pp. 3–14, Dec. 2013.

[50] S. Fischer, “Observed time difference of arrival (OTDOA) positioning in 3GPP LTE„” in Qualcomm Technologies Inc., White Paper, Jun 2014.

[51] S. Hu, A. Berg, X. Li, and F. Rusek, “Improving the performance of OTDOA based positioning in NB-IoT systems,” in 2017 IEEE Global Communications Conference, Dec 2017, pp. 1–7.

[52] G. T. 37.857, “Study on indoor positioning enhancements for UTRA and LTE,” in Rel. 13, V13.1.0, Jan 2016.

[53] Z. Z. M. Kassas, J. Khalife, K. Shamaei, and J. Morales, “I hear, therefore I know where I am: Compensating for GNSS limitations with cellular signals,” IEEE Signal Processing Magazine, vol. 34, no. 5, pp. 111–124, Sep. 2017.

[54] C. Chen and W. Wu, “Three-dimensional positioning for LTE systems,” IEEE Transactions on Vehicular Technology, vol. 66, no. 4, pp. 3220–3234, April 2017.

[55] X. Han, J. Wang, W. Shi, Q. Niu, and L. Xu, “Indoor precise positioning algorithm using 60GHz pulse based on compressive sensing,” Journal of Mathematics and Computer Science, 2016.

[56] Z. Wei, Y. Zhao, X. Liu, and Z. Feng, “DoA-LF: A location fingerprint positioning algorithm with millimeter-wave,” IEEE Access, vol. 5, 2017.

[57] A. Shahmansoori, G. E. Garcia, G. Destino, G. Seco-Granados, and H. Wymeersch, “Position and orientation estimation through millimeter-wave MIMO in 5G systems,” IEEE Transactions on Wireless Communications, vol. 17, no. 3, pp. 1822–1835, March 2018.

[58] Z. Abu-Shaban, X. Zhou, T. Abhayapala, G. Seco-Granados, and H. Wymeersch, “Error bounds for uplink and downlink 3D localization in 5G mmwave systems,” IEEE Transactions on Wireless Communications, pp. 1–1, 2018.

[59] V. Savic and E. G. Larsson, "Fingerprinting-based positioning in distributed massive MIMO systems," in 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), Sept 2015.

[60] L. R. W. Mohinder S. Grewal and A. P. Andrews, Global Positioning Systems, Inertial Navigation, and Integration, 2nd ed. Wiley, 2007.

[61] A. Guerra, F. Guidi, and D. Dardari, "Single-anchor localization and orientation performance limits using massive arrays: MIMO vs.beamforming," IEEE Transactions on Wireless Communications, vol. 17, no. 8, pp. 5241–5255, Aug 2018.

[62] Y. Bengio, Y. LeCun, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436–444, 2015.

[63] 3GPP, "Evolved universal terrestrial radio access (E-UTRA); LTE positioning protocol (LPP)," in 3rd Generation Partnership Project (3GPP), TS 36.355 V14.5.1, April 2018.

[64] G. Mao, B. Fidan, and B. D. Anderson, "Wireless sensor network localization techniques," Computer Networks, vol. 51, no. 10, pp. 2529 – 2553, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128606003227

[65] T. S. Rappaport, J. H. Reed, and B. D. Woerner, "Position location using wireless communications on highways of the future," IEEE Communications Magazine, vol. 34, no. 10, pp. 33–41, Oct 1996.

[66] F. Lemic, J. Martin, C. Yarp, D. Chan, V. Handziski, R. Brodersen, G. Fettweis, A. Wolisz, and J. Wawrzynek, "Localization as a feature of mmWave communication," in 2016 International Wireless Communications and Mobile Computing Conference (IWCMC), Sep. 2016, pp. 1033–1038.

[67] W. B. Abbas, F. Gomez-Cuba, and M. Zorzi, "Millimeter wave receiver efficiency: A comprehensive comparison of beamforming schemes with low resolution ADCs," IEEE Transactions on Wireless Communications, vol. 16, no. 12, pp. 8131–8146, Dec 2017.

[68] S. Buzzi and C. D'Andrea, "Energy efficiency and asymptotic performance evaluation of beamforming structures in doubly massive MIMO mmwave systems,"

IEEE Transactions on Green Communications and Networking, vol. 2, no. 2, pp. 385–396, June 2018.

[69] D. Franklin, "Nvidia developer blog: Nvidia jetson tx2 delivers twice the intelligence to the edge," https://devblogs.nvidia.com/jetson-tx2-delivers-twice-intelligence-edge/, accessed: 11th of January, 2020.

[70] "Intel(r) movidius(tm), intel movidius neural compute stick," https://software.intel.com/en-us/neural-compute-stick, accessed: 11th of January, 2020.

[71] H. N. Mhaskar and C. A. Micchelli, "How to choose an activation function," in Advances in Neural Information Processing Systems 6, J. D. Cowan, G. Tesauro, and J. Alspector, Eds. Morgan-Kaufmann, 1994, pp. 319–326. [Online]. Available: http://papers.nips.cc/paper/874-how-to-choose-an-activation-function.pdf

[72] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," Journal of Machine Learning Research, vol. 15, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[73] S. Bai, M. Yan, Y. Luo, and Q. Wan, "RFedRNN: An end-to-end recurrent neural network for radio frequency path fingerprinting," in Recent Trends and Future Technology in Applied Intelligence, M. Mouhoub, S. Sadaoui, O. Ait Mohamed, and M. Ali, Eds. Cham: Springer International Publishing, 2018, pp. 560–571.

[74] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di, and Y. Yu, "HD-CNN: Hierarchical deep convolutional neural networks for large scale visual recognition," in 2015 IEEE International Conference on Computer Vision (ICCV), Dec 2015, pp. 2740–2748.

[75] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," J. Mach. Learn. Res., vol. 13, pp. 281–305, Feb. 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2188385.2188395

[76] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale

machine learning," in 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16), 2016, pp. 265–283. [Online]. Available: https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf

[77] R. Caruana, S. Lawrence, and L. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," in Proceedings of the 13th International Conference on Neural Information Processing Systems, ser. NIPS'00. Cambridge, MA, USA: MIT Press, 2000, pp. 381–387. [Online]. Available: http://dl.acm.org/citation.cfm?id=3008751.3008807

[78] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," CoRR, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[79] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[80] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28, ser. ICML'13, 2013, pp. III–1310–III–1318. [Online]. Available: http://dl.acm.org/citation.cfm?id=3042817.3043083

[81] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," arXiv e-prints, p. arXiv:1609.03499, Sep. 2016.

[82] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015, pp. 3431–3440.

[83] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 770–778.

[84] L. A. Tawalbeh, A. Basalamah, R. Mehmood, and H. Tawalbeh, "Greener and smarter phones for future cities: Characterizing the impact of GPS signal strength on power consumption," IEEE Access, vol. 4, pp. 858–868, 2016.

[85] N. Culakova, D. Murphy, J. Gante, C. Ledezma, V. Hovhannisyan, and A. Mosca, "How to calibrate your neural network classifier: Getting true probabilities from a classification model," pp. 3499–3500, 2020. [Online]. Available: https://doi.org/10.1145/3394486.3406700

[86] J. Yu, C. Zhu, J. Zhang, Q. Huang, and D. Tao, "Spatial pyramid-enhanced netvlad with weighted triplet loss for place recognition," IEEE Transactions on Neural Networks and Learning Systems, pp. 1–14, 2019.

[87] C. Hong, J. Yu, J. You, X. Chen, and D. Tao, "Multi-view ensemble manifold regularization for 3D object recognition," Information Sciences, vol. 320, pp. 395 – 405, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025515001978

[88] C. Hong, J. Yu, J. Zhang, X. Jin, and K. Lee, "Multi-modal face pose estimation with multi-task manifold deep learning," IEEE Transactions on Industrial Informatics, pp. 1–1, 2018.

[89] J. Yu, Y. Rui, and D. Tao, "Click prediction for web image reranking using multimodal sparse coding," IEEE Transactions on Image Processing, vol. 23, no. 5, pp. 2019–2032, May 2014.

[90] C. Hong, J. Yu, D. Tao, and M. Wang, "Image-based three-dimensional human pose recovery by multiview locality-sensitive sparse retrieval," IEEE Transactions on Industrial Electronics, vol. 62, no. 6, pp. 3742–3751, June 2015.

[91] F. Zhuang, L. Huang, J. He, J. Ma, and Q. He, "Transfer learning with manifold regularized convolutional neural network," in Knowledge Science, Engineering and Management, G. Li, Y. Ge, Z. Zhang, Z. Jin, and M. Blumenstein, Eds. Cham: Springer International Publishing, 2017, pp. 483–494.

[92] M. Fazel, H. Hindi, and S. Boyd, "Rank minimization and applications in system theory," in Proceedings of the 2004 American Control Conference, vol. 4, June 2004, pp. 3273–3278 vol.4.

[93] H. Ouyang, N. He, L. Tran, and A. Gray, "Stochastic alternating direction method of multipliers," in International Conference on Machine Learning, 2013, pp. 80–88.

[94] C. Hong, J. Yu, J. Wan, D. Tao, and M. Wang, “Multimodal deep autoencoder for human pose recovery,” IEEE Transactions on Image Processing, vol. 24, no. 12, pp. 5659–5670, Dec 2015.

[95] J. Yu, X. Yang, F. Gao, and D. Tao, “Deep multimodal distance metric learning using click constraints for image ranking,” IEEE Transactions on Cybernetics, vol. 47, no. 12, pp. 4014–4024, Dec 2017.

[96] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, “A decomposable attention model for natural language inference,” CoRR, vol. abs/1606.01933, 2016. [Online]. Available: http://arxiv.org/abs/1606.01933

[97] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” CoRR, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[98] J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teh, “Set transformer,” CoRR, vol. abs/1810.00825, 2018. [Online]. Available: http://arxiv.org/abs/1810.00825

[99] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in NIPS Deep Learning and Representation Learning Workshop, 2015. [Online]. Available: http://arxiv.org/abs/1503.02531

[100] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, “Binarized neural networks,” in Proceedings of the 30th International Conference on Neural Information Processing Systems, ser. NIPS’16. Red Hook, NY, USA: Curran Associates Inc., 2016, pp. 4114–4122.