UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Amélioration de la précision des cartes de couverture réseau par des techniques de lancer de rayons

Bernard Tamba SANDOUNO

Centre Inria d'Université Côte d'Azur, équipe Diana

Présentée en vue de l'obtention du grade de docteur en Informatique **d'**Université Côte d'Azur

Dirigée par : Chadi BARAKAT, Directeur de recherche, Centre Inria d'Université Côte d'Azur, Sophia Antipolis, France

Co-dirigée par : Thierry TURLETTI, Directeur de Recherche, Centre Inria d'Université Côte d'Azur, Sophia Antipolis, France

Devant le jury, composé de :

Guillaume URVOY-KELLER, Professeur des universités, I3S, Sophia Antipolis, France

André-Luc BEYLOT, Professeur des universités, INP-ENSEEIHT, Toulouse, France

Marcelo DIAS DE AMORIM, Directeur de recherche, Sorbonne Université, CNRS, LIP6, Paris, France

Hossam AFIFI, Professeur des universités, Télécom SudParis, Paris, France

Leonardo LIZZI, Associate Professor, Université de Trento, Trente, Italie

AMÉLIORATION DE LA PRÉCISION DES CARTES DE COUVERTURE RÉSEAU PAR DES TECHNIQUES DE LANCER DE RAYONS

Enhancing outdoor radio frequency mapping with Ray Tracing techniques

Bernard Tamba SANDOUNO

Jury :

Président du jury

Guillaume URVOY-KELLER, Professeur des universités, I3S, Sophia Antipolis, France

Rapporteurs

André-Luc BEYLOT, Professeur des universités, INP-ENSEEIHT, Toulouse, France
Marcelo DIAS DE AMORIM, Directeur de recherche, Sorbonne Université, CNRS, LIP6, Paris, France

Examinateurs

Hossam AFIFI, Professeur des universités, Télécom SudParis, Paris, France
Leonardo LIZZI, Associate Professor, Université de Trento, Trente, Italie

Directeur de thèse

Chadi BARAKAT, Directeur de recherche, Centre Inria d'Université Côte d'Azur, Sophia Antipolis, France

Co-directeur de thèse

Thierry TURLETTI, Directeur de Recherche, Centre Inria d'Université Côte d'Azur, Sophia Antipolis, France

---

Université Côte d'Azur

Résumé

Le lancer de rayons est un modèle de propagation déterministe qui estime la puissance du signal reçu en des emplacements géographiques spécifiques, en prenant en compte de manière exhaustive des facteurs environnementaux en 3D qui influencent la propagation du signal. Cette grande précision du lancer de rayons se fait toutefois au prix d'une charge de calcul élevée due aux complexités de ses processus. Cette charge de calcul se caractérise par une consommation élevée de la mémoire vive et un temps d'exécution élevé, rendant difficile le calcul efficace de la puissance du signal reçu.

Conscients de la complexité inhérente au lancer de rayons, diverses techniques d'accélération ont émergé au fil du temps pour améliorer son efficacité computationnelle. Ces techniques optimisent les processus au sein du lancer de rayons, permettant ainsi des réductions significatives du temps d'exécution pour l'estimation de la puissance du signal à des emplacements individuels. Cependant, lorsqu'elles sont appliquées à la génération de cartes de fréquences radio sur des régions étendues telles que des villes ou des pays, ces techniques d'accélération perdent leur efficacité. Cette limitation découle de la conception initiale du lancer de rayons, conçue pour l'évaluation de la réception du signal en point à point plutôt que pour la cartographie radio fréquence à grande échelle.

En réponse à cette problématique, cette thèse propose des optimisations adaptées spécifiquement à la génération de cartes de fréquences radio précises à grande échelle sans entraîner de surcharge computationnelle excessive. En affinant tous les aspects du pipeline du lancer de rayons avec un accent particulier sur la cartographie ra-

dio fréquence, nos optimisations visent à résoudre les inefficacités prédominantes dans les solutions actuelles de cartographie radio fréquence à grande échelle. Tout au long de ce travail, nous démontrons les lacunes des approches existantes et validons l'efficacité de nos optimisations en réduisant considérablement les temps d'exécutions tout en maintenant la précision de l'estimation dans des limites acceptables.

Nous proposons d'abord une approche innovante du processus traditionnellement complexe de génération de rayons, en minimisant efficacement le nombre de rayons lancés à partir d'une antenne tout en préservant la précision. Adaptée aux conditions spécifiques du site, cette méthode prend en compte l'environnement de propagation lors du lancement des rayons, contrairement à l'approche traditionnelle qui lance les rayons dans tous les sens possibles. En mettant en œuvre cette technique de génération de rayons spécifique au site et en réimaginant le processus de test de réception, nous réalisons une réduction remarquable de près de 1200 fois du temps d'exécution du lancer de rayons pour la génération de cartes radio fréquence dans des terrains avec de légères variations d'altitude. En étendant cette solution à des terrains divers, nous introduisons des optimisations complètes à travers tous les processus de lancer de rayons, méticuleusement adaptées aux objectifs de la cartographie radio fréquence. Ces améliorations aboutissent à une augmentation de 50 fois de la vitesse de génération de cartes radio fréquence à grande échelle sur des terrains variés, démontrant la polyvalence et l'efficacité de notre approche.

Mots-clés: Modèles de propagation, modèle déterministe, lancer de rayons, cartographie radio fréquence, réseaux cellulaires, estimation de puissance.

Abstract

Ray Tracing is a deterministic propagation modeling approach that estimates the signal power received in specific geographical locations based on a thorough consideration of the 3D environmental factors that influence signal propagation. This high accuracy of Ray Tracing is however at the cost of a high computational load due to the complexities of the processes within its workflow. This computational demand is characterized by a high memory consumption and a high execution time, making it challenging to efficiently compute the received signal power.

Acknowledging the complexity inherent in Ray Tracing, various acceleration techniques have emerged over time to enhance its computational efficiency. These techniques optimize processes within Ray Tracing, facilitating significant reductions in execution time for signal power estimation at individual locations. However, when applied to the generation of Radio Frequency (RF) maps across expansive regions like cities or countries, these acceleration techniques lose their efficiency. This limitation arises from Ray Tracing's original design, tailored for point-to-point signal reception assessment rather than large-scale RF mapping.

In response, this thesis proposes optimizations designed specifically for generating accurate and precise RF maps at large scale without incurring excessive computational overhead. By refining all aspects of the Ray Tracing pipeline with a focus on RF mapping, our optimizations aim to address the inefficiencies prevalent in current large-scale RF mapping solutions. Throughout this work, we demonstrate the shortcomings of existing approaches and validate the effectiveness of our optimizations in significantly reducing execution times while maintaining estimation

accuracy within acceptable bounds.

We first propose an innovative approach to the traditionally complex ray generation process, by effectively minimizing the number of rays launched from an antenna while preserving the accuracy. Tailored to specific site conditions, this method considers the propagation environment when launching rays, in contrast to the traditional approach that launches rays in a brute-force fashion. Further, by implementing this site-specific ray generation technique and re-imagining the reception test process, we achieve a remarkable reduction of nearly 1200 times in Ray Tracing execution time for RF map generation in terrains with slight altitude variations. Extending this solution to diverse terrains, we then introduce comprehensive optimizations across all Ray Tracing processes, meticulously tailored for RF mapping objectives. These enhancements culminate in a 50-fold increase in the speed of large-scale RF map generation across varied terrains, showcasing the versatility and efficiency of our approach.

Keywords: Propagation models, deterministic model, Ray Tracing, RF mapping, cellular networks, power estimation

To my parents: Facely Sandouno and Sia Fanta Mamadouno

"Entrust your works to the LORD, and your plans will succeed."

— Bible, Proverbs 16:3

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Chadi Barakat, Thierry Turletti and Walid Dabbous, for their invaluable guidance, support, and encouragement throughout this journey. Their expertise, patience, and unwavering belief in my abilities have been instrumental in shaping this thesis.

Additionally, I am immensely grateful to Yamen Alsaba for his exceptional support during the initial stages of my thesis. At a time when I felt overwhelmed and uncertain, Yamen patiently answered all my questions and provided invaluable insights that laid the foundation for my research.

I am also immensely grateful to my colleagues both at Inria and YDATA. Their camaraderie, insightful discussions, and collaborative spirit have enriched my research experience and contributed significantly to the development of this work.

To my friends and family, I owe a debt of gratitude that words cannot adequately express. Your unwavering support, encouragement, and understanding have sustained me through the highs and lows of this academic pursuit. Thank you for being a constant source of love, motivation, and inspiration.

Lastly, I extend my heartfelt thanks to all those who offered their prayers and positive thoughts. Your spiritual support has been a source of strength and comfort, guiding me through challenges and imbuing me with resilience.

This thesis is a culmination of the collective efforts, encouragement, and blessings of many individuals, and for that, I am profoundly grateful.

Table of Contents

List of Figures

List of Tables

List of Acronyms

2D	2 Dimensions
3D	3 Dimensions
3GPP	3rd Generation Partnership Project
4G	4th Generation
5G	5th Generation
AI	Artificial Intelligence
API	Application Programming Interface
ARCEP	Autorité de régulation des communications électroniques, des postes et de la distribution de la presse
AZB	Angular Z-Buffer
BP	Breaking Point
BS	Base Station
BVH	Bounding Volume Hierarchy
CDF	Cumulative Distribution Function
cm	centimeters
CQI	Channel Quality Indicator
dB	Decibel
dbm	Decibel milli-watts
dBW	Decibel Watts

FDTD	Finite Difference Time Domain
FSPL	Free Space Path Loss
G_{rx}	Gain of the receiving antenna
G_{tx}	Gain of the transmitting antenna
GB	Giga Byte
GHz	Giga Hertz
GIS	Geographical Information System
Hz	Hertz
IGN	Institut national de l'information géographique et forestière
INSEE	Institut national de la statistique et des études économiques
IP	Internet Protocol
itu	International Telecommunication Union
km	kilometers
LIDAR	Light Detection and Ranging
LOS	Line of Sight
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
m	meters
Mbps	Mega bit per second
MHz	Mega Hertz
mmWave	Millimeter wave
MNO	Mobile Network Operator
NLOS	Non Line of Sight
NR	New Radio
osm	open street map
P_{rx}	Power of the receiving antenna

P_{tx}	Power of the transmitting antenna
PDF	Probability Distribution Function
PL	Path Loss
QoS	Quality of Service
RF	Radio Frequency
RMa	Rural Macro cell
RMSE	Root Mean Squared Error
ROI	Ray Object Intersection
RT	Ray Tracing
Rx	Receiving antenna
s	seconds
SBR	Shooting and Bouncing Ray Tracing
SIMD	Single Instruction Multiple Data
SINR	Signal to Noise plus Interference Ratio
SNR	Signal to Noise Ratio
TCP	Transmission Control Protocol
THz	Tera Hertz
Tx	Transmitting antenna
UE	User Equipment
UHF	Ultra High Frequency
UMa	Urban Macro cell
UMi	Urban Micro cell
VHF	Very High Frequency

Chapter 1

Introduction

Over the years, mobile traffic has undergone significant evolution, and by 2023, it accounted for almost 60% of internet traffic according to Statista [11]. This increasing traffic is due to the evolving capabilities introduced by successive generations of mobile communication. Specifically, the latest generations, namely 4G and 5G, offer to users high speed mobile broadband internet access, enabling activities such as web browsing, video streaming, and online gaming. To fully leverage these services, it becomes crucial for mobile users to have high data rates in their preferred locations: home, school, work, grocery store,...

However, variations in received signal power among mobile users exist, due to factors like mobile service plans, distance from base stations, obstacles between users and base stations, the type of the zone (rural, suburban, or urban) and employed technologies. Due to these numerous factors that affect mobile users' internet usage, knowing in advance the expected received signal power in their areas of interest becomes crucial. Such information enables them to make informed decisions regarding their mobile services such as selecting service plans that align with the signal conditions in their specific locations.

Moreover, understanding the variations in received signal power allows mobile users to optimize their activities, such as streaming high-quality videos, engaging in online

gaming, and conducting business transactions, without interruptions or degraded performance. This awareness becomes particularly crucial in environments where obstacles or interference may impact signal strength, emphasizing the need for users to stay informed and adapt their mobile usage accordingly. In essence, having a prior understanding of the expected received signal power not only enhances the overall mobile user experience but also facilitates more efficient utilization of available services in diverse and dynamic settings.

It is in this context that generating Radio Frequency maps, aiming to provide comprehensive signal strength levels and bitrate capacity estimations throughout a whole country [1, 12], is gaining significant interest. The primary objective of these maps is to assist end-users in gaining a clearer understanding of the quality of the service delivered by their Mobile Network Operators based on their geographical locations. Additionally, these maps serve as valuable tools for MNOs themselves, enabling them to troubleshoot and enhance their network performance as needed. This improvement may involve strategic adjustments in antenna placement and configuration, ultimately contributing to an optimized QoS [12]. Furthermore, these RF maps play a crucial role for telecommunication regulatory authorities, allowing them to verify whether the specifications set by MNOs are being met [1]. Additionally, by offering insights into signal strength and bitrate capacity at country level, these maps facilitate informed decision-making from telecommunication regulators.

Nevertheless, generating such maps is quite challenging due to the inherent complexity of the environment which is composed of factors like topography, buildings, and obstacles. To fulfill the aforementioned objectives effectively, RF maps should cover a large area and reach a high level of precision to accurately estimate the signal strength across all individual locations within a country. Due to this complexity, telecommunication regulators and the majority of MNOs furnish end-users instead with the so-called coverage maps that illustrate the areas serviced by MNOs antennas. Figure 1.1 is an example of such map provided by the French Telecommunication Regulator.

Figure 1.1: An example of 4G coverage map of France. This Figure is taken from [1].

Major MNOs such as AT&T [13], Telecom Italia [14], and Orange [15], provide their customers with such maps, illustrating both covered and uncovered regions by their respective antennas. These maps serve as valuable tools for users in selecting an MNO that sufficiently covers their areas of interest, ensuring uninterrupted access to internet services. Conversely, in some countries, regulators employ their models to generate coverage maps, offering users two key advantages: the ability to compare MNOs' estimations with those of the regulator and the opportunity to make informed choices based on network coverage preferences. Additionally, private companies, aiming to recommend optimal mobile service plans to mobile end-users, leverage proprietary in-house coverage maps that provide more detailed information than those offered by MNOs and regulators [16–18]. As users increasingly seek such information when subscribing to new service plans, companies like ZoneADSL & Fibre [19] report a monthly visits ranging from 500K to 750K users.

While these maps serve as invaluable tools for identifying areas lacking coverage,

it's essential to note that they do not furnish information about the bitrate received at specific locations. This information becomes crucial, especially in areas covered by multiple MNOs, aiding users in making informed decisions regarding received signal power.

To mitigate this issue, various techniques are employed by stakeholders in the mobile broadband sector. In France, the regulatory body conducts annual measurements using mobile devices at different locations [20]. The outcomes of these measurements are then depicted on an RF map, illustrating the signal strength levels across these locations. Alternatively, tools like Ookla [12] and nperf [21] utilize speed test datasets to generate similar maps. However, both measurement and speed test approaches suffer from a lack of scalability, as they can only estimate signal power in the specific locations where the tests or measurements were conducted. Given the limited scope of these locations, these approaches fail to provide a comprehensive overview of signal levels throughout an entire country.

To address this challenge, researchers in this field utilize propagation model-based RF mapping, as discussed in the literature [22–28]. Propagation models consist of sets of equations and/or algorithms designed to estimate the path loss between end-users' locations and base stations. These models differ from each other in the level of accuracy they offer and the time it takes them to perform the estimation.

The empirical propagation modeling approach, for example, allows the rapid generation of RF maps using offline calibrated models on real data, resulting in a short execution time. These models are fast because they do not incorporate detailed information about the environment, such as buildings and topography. They estimate the received signal power relying on parameters like frequency, distance, zone type, average street width, mean buildings height,... [29]. While these models are quick for generating RF maps, they suffer from low accuracy due to their weak capacity to account for the details of the propagation environment.

Similarly, stochastic models are even faster than empirical models when generating

RF maps. They represent the environment as a set of random variables, using a probability density function to estimate the received signal power. These models are particularly useful for generating RF maps in extensive areas, as they require minimal information about the propagation environment, leading to shorter execution times [29]. Nevertheless, due to their inherent randomness and their lack of consideration for environmental details, stochastic models are the least accurate propagation models.

While the empirical and stochastic models are useful for their faster execution times, they are not accurate enough to assist users in choosing the most suitable MNO. This limitation arises from their neglect of user-specific environmental factors when estimating received signal strength.

It is in this context that Ray Tracing emerges as a highly regarded propagation model renowned for its precision in determining the paths taken by radio waves and evaluating the received signal power at diverse locations within the specified area. RT achieves its accuracy by integrating a comprehensive 3D map and environmental characteristics when estimating received signal power. Moreover, RT considers various multipath propagation phenomena, including reflection, refraction, and diffraction [30] in a deterministic fashion across the obstacles present in the environment. The consideration of these fine-grained details contributes significantly to enhancing the quality of generated RF maps. Consequently, RT is proven to produce high-quality RF maps that accurately depict the received signal power levels across an entire country. This capability empowers mobile end-users to make informed decisions about their mobile service plans, enabling them to choose the MNO that best suits their needs in terms of downlink bitrate. Simultaneously, MNOs can leverage these detailed maps to identify areas with bad signal quality and take targeted actions to enhance the performance of their network coverage in those specific areas.

Nevertheless, despite the high accuracy offered by RT, it comes at the expense of significant memory consumption and computational load. This complexity leads to long execution time making RT impractical for generating RF maps in complex envi-

ronments [29–31]. This complexity is due to the intensive computations incurred by its processes. For instance, the technique employed to generate rays in RT has been identified as computationally heavy [32]. Additionally, the ray/object intersection test process determining whether a ray intersects with an object, may lead to long execution times in complex scenarios if used naively [33–35]. Further, the reception test process, which involves checking whether a ray is received by a set of reception points, suffers from resource-intensive matrix computations and greedy memory consumption [36].

It is also crucial to note that RT was not initially designed for RF mapping purposes [36]. Instead, its primary purpose is to compute the path loss between a given transmitter and a given receiver. Various techniques in the literature attempt to adapt RT for RF mapping purposes [22–28]. However, these adaptations often result in increased complexity, making it challenging, if not impossible, to generate accurate and precise RF maps in complex environments.

1.1 Objectives and Contributions

Therefore, the objective of this thesis is to optimize RT processes for large-scale RF mapping purposes. We aim to preserve RT's accuracy while reducing its execution time and ensuring its applicability to diverse scenarios. The ultimate goal is to generate precise and accurate RF maps depicting countrywide signal power levels, within an acceptable execution time.

To optimize RT processes, we started by proposing a novel ray generation method. Indeed, in RT waves are modeled by rays, i.e, they are the ones sent and received by antennas. The conventional ray generation process involves launching rays in all directions without a prior knowledge of the receivers' positions in order to fully cover antennas 3D radiation pattern. However, this process follows a complex and time-consuming algorithm. In contrast, our new ray generation technique quickly and iteratively finds the optimal number of required rays to fully cover an area of

interest. Our technique is site specific, i.e., for each scenario it gives the optimal number of rays that need to be launched in order to fully cover the area without any blank zone. By minimizing the number of rays that need to be launched, our method reduces the overhead incurred by rays that will never reach the reception point. Moreover, our technique overcomes the complexity and the computational slowness of the conventional technique. It was able to launch up to 1 million rays within 5 seconds, hence reducing the overhead due to the ray generation process. We performed simulations with our new technique in different scenarios and a validation with respect to a state-of-the art model implementing the conventional ray generation technique was carried out. Through these simulations, we could see that our solution for ray generation is flexible, robust, and computationally fast at almost no cost.

After validating our model, we made an extensive simulation study on the accuracy/complexity trade-off of RT depending on whether we are in an urban, suburban, or rural area. This extensive work is meant to help one choose the suitable number of rays to launch given a specific scenario while having sufficient information about the complexity required to reach the sought level of accuracy. We performed different simulations involving many scenarios to assess the complexity and the accuracy of RT in those cases. The latter helped us to draw recommendations about the number of rays to launch in each scenario and the expected accuracy depending on the available resources.

Furthermore, we extended our work to subsequent RT processes by optimizing them for RF mapping purposes. The conventional approach for generating RF maps using RT involves discretizing the area of interest with a limited number of reception points, as outlined in previous studies [26–28]. These reception points are distributed throughout the area according to some statistical distribution to ensure comprehensive coverage. The traditional RT algorithm is then performed for each pair of transmitting antenna and reception points. However, the drawback of this approach becomes apparent in subsequent processes, particularly the reception test, which is employed naively and lacks optimization for RF mapping purposes. Indeed,

it involves checking, for each pair of ray and reception point, whether the ray is received by the given reception point. This can result in excessive computational load and potential memory limitations, particularly in complex scenarios including large number of rays and reception points, posing challenges to the effective generation of RF maps.

To address this limitation, we introduce a novel reception test method tailored for regions characterized by minimal variations in terrain altitude. Instead of going through complex matrix computations, we simply capture the footprints of rays on a given horizontal plane and subsequently utilize this information to generate the RF map on a 2D plane. This innovative approach eliminates the need of discretizing the area of interest, hence removing the traditional reception test. While existing models struggle with 16-hours execution time and frequently encounter memory limitations, our solution delivers the RF map in only 50 seconds. By combining our aforementioned site-specific ray generation and our new reception test processes, we could divide by almost 1200 the execution time of RT, with a memory usage accounting for less than 2% compared to baseline solutions.

Finally, we propose a fast RT solution fully designed to generate RF maps at large-scale taking into account the 3D details of the environment. In the literature, none of the existing RT techniques are designed for RF mapping purposes and/or are not sufficiently optimized to meet the requirements of large-scale RF map generation with account for the topography of the surface. Our approach introduces innovative schemes based on 3D triangulation to model an environment of interest and we leverage efficient hardware acceleration techniques to generate RF maps at large-scale. The acceleration technique leveraged in this solution as well as the cartography solution that we propose are both tuned to meet the requirements of generating RF maps at large-scale while considering the 3D topography of the terrains and the information on the buildings. For this, we speed up the ray/object intersection test by adapting a fully optimized RT engine developed by Intel called *Embree* [37]. Since Embree is based on hardware acceleration techniques, we were able to efficiently accelerate the

ray/object intersection test process. Moreover, we leverage the geometrical properties of the triangular mesh used during the ray/object intersection test to generalize our reception test to 3D RF mapping. It consists in directly performing a spatial intersection between the 3D triangles and the circular cross-sections of the rays. Thanks to all these underlying RT processes being introduced, we were able to move directly from the triangular mesh of the terrain to the RF map production after proposing signal power estimators for triangles and coloring them accordingly. This contribution forms an optimized pipeline fully designed for RF mapping purposes and capable to reach the goal of RF map generation at large-scale in any type of terrain including areas with high terrain altitude's variations.

Simulations of our RF mapping solution in 3D large-scale scenarios show that our approach outperforms existing ones both in terms of execution time and memory consumption. Indeed, even without leveraging the efficiency of our site-specific ray generation technique, our approach was 50 times more efficient than traditional RT approaches to RF mapping. Due to memory limitations, existing solutions were even unable to produce RF maps in urban areas involving multiple antennas and requiring precise maps. In contrast, our solution generates a precise and accurate RF map in all types of terrains with a difference as small as 1Mbps on average compared to existing solutions. We also show that our approach preserves the accuracy of the RF map when changing the resolution step (or the size of triangles), i.e., with a 50 meters resolution step, we achieve almost the same accuracy as when the resolution step gets near zero. Furthermore, we demonstrate that in extremely complex scenarios, the reception test as used in traditional approaches may account for more than 98% of the overall RT complexity. We then highlight how our 3D reception test, tailored for complex terrain types, is able to tackle this issue with only a slight increase of the execution time while changing the resolution step. We equally show how the ray/object intersection test in our solution which is based on Embree behaves independently from the number of triangles, i.e., one can choose a fine-grained size for triangles to improve the RF map precision without degrading the performance of the intersection test.

1.2 Thesis outline

The rest of this thesis is organized as follows.

- In Chapter 2, we present the fundamentals of radio propagation models. Here, we explain how propagation modeling approaches stand out as the optimal choice for estimating the received signal power across an entire country. Additionally, we provide a comprehensive overview of the three propagation models, highlighting the distinctions among them in terms of both accuracy and complexity.

  Our emphasis extends to showcasing how the use of RT aligns seamlessly with our objectives, aiming to furnish mobile broadband stakeholders with a practical tool for making informed decisions. We also focus in this Chapter on RT processes, from the generation of rays to the computation of the received signal power. We particularly focus on how these processes account for RT complexity in large-scale complex scenarios. Finally, we present an overview of the acceleration techniques that attempt to optimize RT and make it usable in diverse scenarios.

- In Chapter 3, we present the conventional approach of generating rays in RT, with a dig in depth in its technical details. Additionally, we explore alternative state-of-the art methods designed to address the complexities inherent to the conventional approach. Within this Chapter, we demonstrate how our site-specific approach not only outperforms these methods but also proves highly advantageous in large-scale scenarios featuring thousands of antennas. Furthermore, we provide a thorough examination of the technical details involved in deriving this approach. This includes a presentation of simulations conducted across various terrains, showcasing the efficiency, flexibility, and robustness of our approach.

- In Chapter 4, we introduce the concept of RF map generation. Within this Chapter, we explore the current state-of-the art techniques employed for generating such maps, with a particular emphasis on highlighting the inefficiencies of

RT processes in these scenarios. We provide an overview of the techniques that attempt to optimize RT for this purpose. In contrast, we show the performance of our new reception test algorithm solely designed to accelerate the generation of RF maps in areas with minimal terrain variations. We detail the optimization brought by this approach and illustrate its efficiency when combined with our aforementioned ray generation technique. The description of its technical details also shed the light on our algorithm's limitation to areas with only slight variations in the terrain.

- In order to account for the adaptability of all RT processes to RF mapping, we present in Chapter 5 our novel RF map rendering technique. This Chapter presents a generalization of our reception test technique designed to accommodate various types of terrain. We present the hardware accelerations leveraged as well as the optimizations brought to all RT processes tailoring them for the specific demands of RF mapping. We present how this solution outperforms existing ones, and also demonstrate its capacity to generate precise RF maps at large-scale where current methods failed. Furthermore, we showcase our cartography solution that employs a 3D triangular mesh-based RF map rendering.

- Chapter 6 serves as a comprehensive summary of all the contributions made throughout this work. In this Chapter, we engage in discussions regarding the impact of our contributions and present future perspectives for enhancing our work, alongside addressing any limitations inherent in our approaches.

- Throughout this thesis, we meticulously construct an in-house RT tool that incorporates all our contributions. In the appendices, we provide detailed technical information about our implementation to facilitate the reproducibility of our work.

List of Publications

Conferences

- Bernard Tamba Sandouno, Chadi Barakat, Thierry Turletti, Walid Dabbous, "**Optimizing Ray Tracing Techniques for Generating Large-Scale 3D Radio Frequency Maps**", in proceedings of the 25th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Perth, Australia, June 2024.
- Bernard Tamba Sandouno, Yamen Alsaba, Chadi Barakat, Walid Dabbous, Thierry Turletti, "**A novel approach to mobile outdoor QoS map generation**", in proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Glasgow, Scotland, UK, March 2023.
- Bernard Tamba Sandouno, Yamen Alsaba, Chadi Barakat, Walid Dabbous, Thierry Turletti, "**Site-specific ray generation for accurate estimation of signal power**", in proceedings of the 25th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), Montreal, Canada, October 2022.

Journals

- Bernard Tamba Sandouno, Yamen Alsaba, Chadi Barakat, Walid Dabbous, Thierry Turletti, "**A novel approach for ray tracing optimization in wireless communication**", in Elsevier Computer Communications Journal, vol. 209, pp. 309-319, September 2023.

Demo

- Bernard Tamba Sandouno, Timothée Poincheval, Benjamin Gervais, Edouard Delgatte, "**Leveraging semi-deterministic propagation models for Large scale RF Mapping: A case study of France**", in proceedings of the 25th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Perth, Australia, June 2024.

Chapter 2

Background and state-of-the-art

2.1 Electromagnetic waves

Electromagnetic waves represent a form of energy that travels through vacuum or a medium via oscillations of electric and magnetic fields. These waves were mathematically conceptualized by James Clerk Maxwell's work in 1865 [38], where he established the relationship between the magnetic and the electric fields. Maxwell stated that a changing electric field induces a changing magnetic field and vice versa. This mechanism creates an electromagnetic wave that helps the energy to be propagated through any given medium [38]. During this propagation, the electric and the magnetic fields maintain a perpendicular orientation to each other and to the direction of propagation of the wave, as illustrated in Figure 2.1.

This relationship established by Maxwell led to the formulation of a set of equations known as Maxwell's equations, which are the most elegant way of describing the relationship between the electric and the magnetic fields. In a homogeneous and isotropic medium, i.e. where the electric and magnetic properties are uniform in all directions and in all points in space, the local form of Maxwell's equations also known as Maxwell's curl equations give the expression of the two fields in a given space and time as described below.

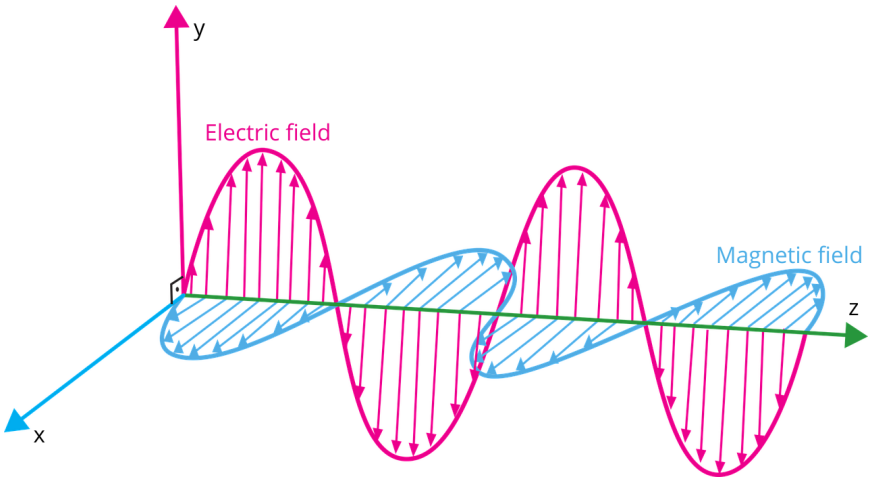


Figure 2.1: Electromagnetic wave

Maxwell's formulation of Gauss electric and magnetic laws are given respectively by (2.1) and (2.2).

$$\nabla \cdot E = \frac{\rho}{\epsilon} \tag{2.1}$$

$$\nabla \cdot B = 0 \tag{2.2}$$

The Maxwell-Ampere and Maxwell-Faraday laws are respectively given by (2.3) and (2.4).

$$\nabla \times B = \mu \left(J + \epsilon \frac{\partial E}{\partial t} \right) \tag{2.3}$$

$$\nabla \times E = -\frac{\partial B}{\partial t} \tag{2.4}$$

In this equation:

- E is a vector representing the electric field.
- B is a vector representing the magnetic field.
- ρ is the electric charge density.
- $J = \sigma E$ represents the current density.
- ϵ is the electric permittivity of the medium.

- μ is the magnetic permeability of the medium.
- σ is the conductivity of the medium.

The characteristics of the medium namely, ϵ, μ and σ, determine how electromagnetic waves propagate within the medium. Simply said, they determine the medium's influence on the behavior of the electromagnetic waves. The velocity c of an electromagnetic wave is therefore dependent on the characteristics of the medium as given by Equation (2.5).

$$c = \frac{1}{\sqrt{\epsilon\mu}} \tag{2.5}$$

Usually, the permittivity and the permeability are used in their relative forms, i.e. with regards to the ones of the free space. Their values are given by (2.6) and (2.7).

$$\epsilon_r = \frac{\epsilon}{\epsilon_0} \tag{2.6}$$

$$\mu_r = \frac{\mu}{\mu_0} \tag{2.7}$$

In this equation:

- ϵ_r is the relative permittivity of the medium.
- μ_r is the relative permeability of the medium.
- $\epsilon_0 = 8.854 \times 10^{-12} F/m$ (Farads per meter) is the free space permittivity.
- $\mu_0 = 4\pi \times 10^{-7} H/m$ (Henrys per meter) is the free space permeability.

Including these two equations in (2.5) shows that the velocity of an electromagnetic wave in free space is the same as the speed of light, which is $3 \times 10^8 m/s$.

Maxwell' equations can be also written as functions of the magnetic excitation H and the electric induction D, which are linked to the electric and the magnetic fields by (2.8) and (2.9).

$$H = \frac{B}{\mu} \tag{2.8}$$

$$D = \epsilon E \tag{2.9}$$

At time t, having knowledge of the electric field E (or D) and the magnetic field B (or H) allows the computation of the power flow carried by the electromagnetic wave. This computation is facilitated by the Poynting vector, which characterizes both the direction and magnitude per unit of surface area of the power flow carried by the wave. The Poynting vector, denoted as S aligns with the direction of propagation of the wave. Its instantaneous magnitude is the power density in Watts per meter squared of the wave at a given time and space. Its value is given by Equation (2.10) [32], where H^* is the complex conjugate of the magnetic field.

$$S = E \times H^* \tag{2.10}$$

2.2 Radio Propagation

In wireless communication, the information is sent from a Base Station in the form of electrical signals. These signals undergo a transformation into RF waves by an antenna for its transmission through the wireless medium. Those waves are transmitted towards the receiver according to the radiation pattern of the transmitting antenna. Waves are sent continuously on a propagation channel in space in order to fully cover the direction on which the waves are being transmitted. As these RF waves traverse the wireless medium, they interact with the surrounding environment, encountering influences from the medium itself, the channel, and various objects in their path. This RF wave behavior is known as Radio Propagation. Throughout the process of radio propagation, the waves interact with the environment of propagation, causing the attenuation of the signal power. The level of this attenuation is tightly linked to the type of environment and more specifically, to the objects present in the environment and to the coverage of the antennas.

Three types of environments are commonly considered in radio propagation, each of them having specific behavior when waves propagate in them [29].

- **Urban area:** Also referred to as Dense area, this type of environment is characterized by a high population density and contains numerous tall buildings with multiple floors. In such areas, the extent of the signal coverage is limited, necessitating the use of micro cell antennas. This is due to the high concentration of people that results in a large number of connections to the BSs during each time slot. Hence, having small coverage areas help to deliver high signal power to end mobile users present in the micro cells. Additionally, this is motivated by the attenuation caused by the surrounding structures. Indeed, these high-rise structures block the signal, thus preventing them to cover larger distances. Therefore, to effectively cover urban regions, a significant deployment of micro cell antennas is required, often resulting in the installation of hundreds of antennas within a relatively small radius.

- **Suburban area**: This type of environment corresponds to small cities characterized by a landscape of houses and trees, as opposed to tall buildings, with a moderate population density per square kilometer. Suburban areas employ a combination of micro cell and macro cell antennas. Micro cells are strategically placed in city centers to provide high signal power to mobile users in densely populated areas. On the other hand, macro cells cover the remaining regions, offering larger coverage distances but with lower signal power compared to micro cells. Consequently, the number of antennas in suburban areas is generally lower than in densely populated urban areas.

- **Rural area**: This type of area is typically composed of small number of houses, large number of trees and/or mountains, hills, valleys, plains. In these areas most of the obstructions come from the topography of the terrain and the trees. They are typically composed of macro cell antennas able to cover more than 10km in some cases. Those antennas are usually mounted on tall pylons located on mountains that are then able to cover large distances. Due to the large coverage distance of those antennas, these areas are usually composed of small number of antennas. Hence, the signal power in those areas is weak and often contains

blank zones, i.e. areas not covered by any antenna.

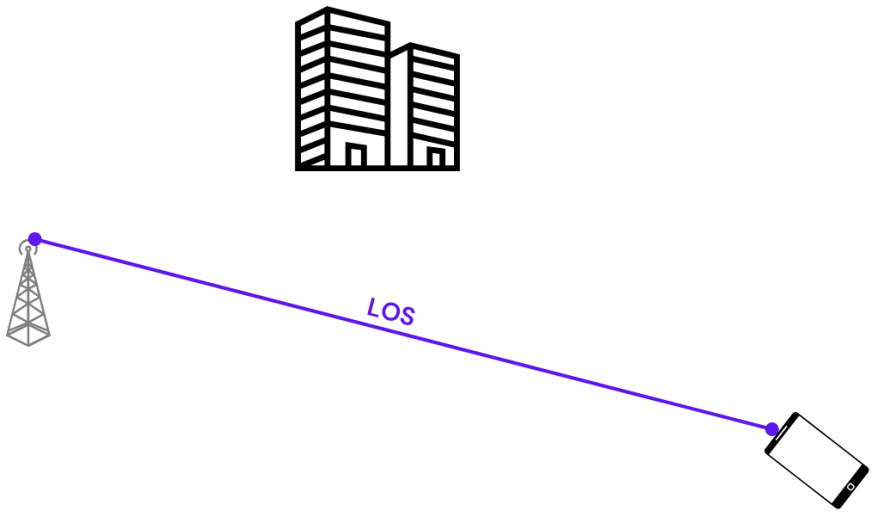


Figure 2.2: LOS scenario between a transmitter and a receiver

During radio propagation in these aforementioned areas, the signal may encounter various obstacles in their journey from the transmitter to the receiver. The type of these obstacles depends on the specific characteristics of the area. Urban areas feature buildings as primary obstacles, while rural areas contain trees and the high variation of the terrain topography as obstacles.

When the signal interacts with these obstacles, it undergoes changes due to the electric and magnetic properties of the materials constituting the obstacles. The properties of these materials determine how much the signal is attenuated before continuing its journey. For example, lossy materials tend to absorb a significant portion of the signal power, while lossless materials do not attenuate it. Additionally, these obstacles modify the trajectory of the signal, potentially changing its path and/or dividing it into different other signals, leading the signal to be received by following multiple paths. This phenomenon is referred to as Multipath propagation. During multipath propagation, two distinct scenarios are observed. In the Line-Of-Sight scenario, the signal reaches the receiver without passing through any obstacles. This occurs when there are no obstacles in the direct path between the transmitter and the receiver as shown in Figure 2.2. In this case, the signal power is not affected by the character-

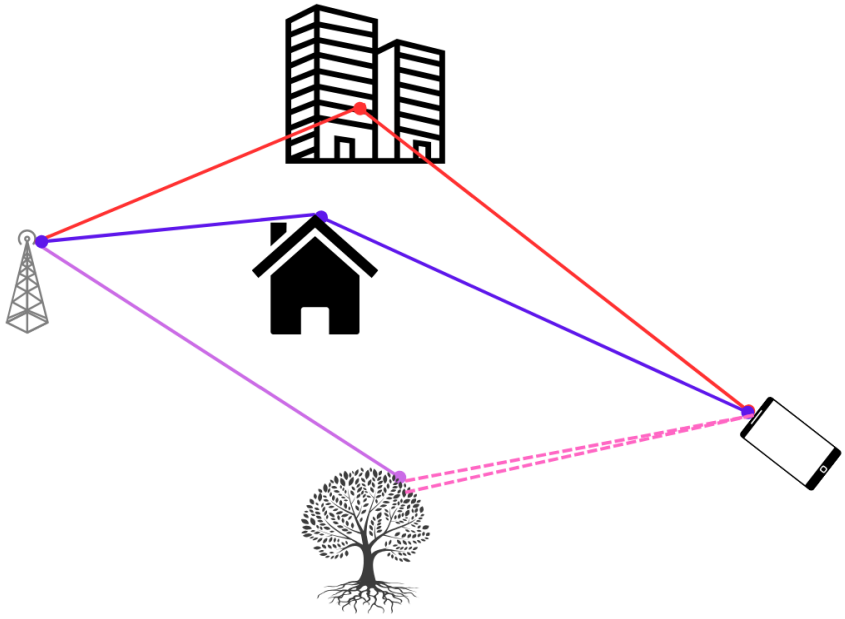


Figure 2.3: Propagation mechanisms

istics of the objects in the environment. Conversely, the Non-Line-Of-Sight occurs when there exist obstructions in the direct path between the transmitter and the receiver. When the signal encounters these obstacles, different propagation mechanisms can occur, as illustrated in Figure 2.3. Usually, four propagation mechanisms occur during radio propagation: Reflection, Refraction, Diffraction and Scattering.

- **Reflection**: This mechanism occurs when the size of the obstacle exceeds the dimensions of the signal's wavelength. This typically corresponds to the signal hitting the ground or buildings' walls. Across various types of environments, this stands out as the most prevalent mechanism, primarily due to the prevalence of buildings, houses, and varied surface topography. In the case of reflection, the signal deviates from its initial trajectory, following what is known as the reflected path.

- **Refraction**: It has the same condition of occurrence as the reflection. Indeed, according to Snell's law [39], a signal hitting a surface goes through two mechanisms: reflection and refraction. The latter involves the signal being absorbed

by the material and passing through it. This is the mechanism responsible for signal reception in indoor environments. Figure 2.4 shows a signal hitting a large surface with the occurrence of reflection and refraction.

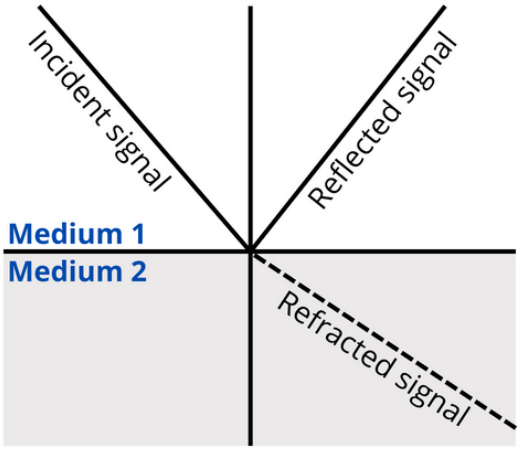


Figure 2.4: Reflection and Refraction on a large surface

- **Diffraction**: This mechanism takes place when the size of the surface is not significantly larger than the wavelength of the signal. Such surfaces include corners of buildings, edges of any objects including the relief: mountains or hills. When diffraction occurs, the signal bends around the surface's edge [40], continuing its journey along a deviated path. This property is particularly valuable in rural areas, since it enables the signal to propagate beyond obstacles such as mountains or hills.

- **Scattering**: Scattering occurs when the size of the surface is smaller than the wavelength of the signal. Common examples of such surfaces include the leaves of trees. When the signal encounters these leaves, they scatter into multiple directions, with each scattered component continuing its journey towards its deviated path.

The prevalence of each of these mechanisms is hence dependent on the wavelengths of both the channel and the obstacles. The flow chart provided in Figure 2.5 shows which propagation mechanism occurs depending on the signal and the obstruction's

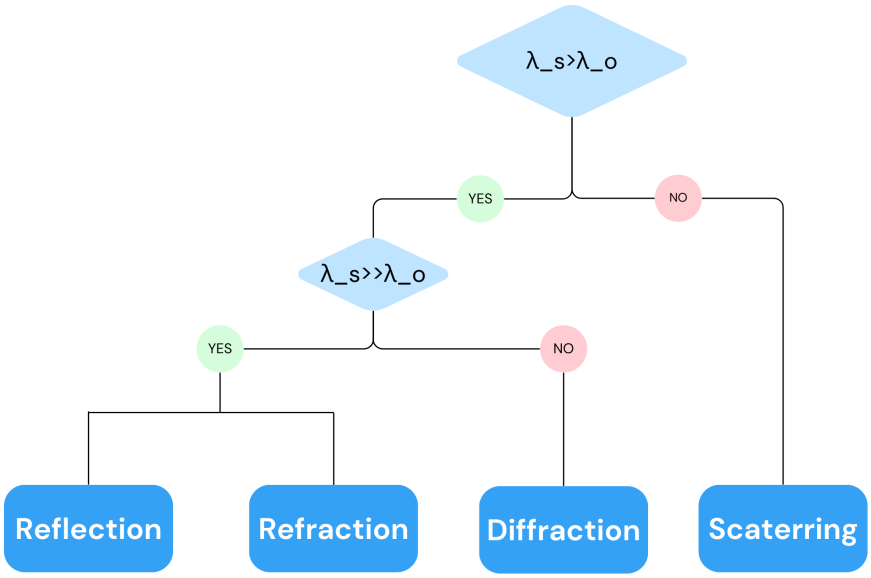


Figure 2.5: Flowchart of propagation mechanisms

wavelengths λ_s and λ_o respectively.

Apart from these propagation mechanisms, the signal is also attenuated by other surrounding mechanisms during radio propagation as shown below.

- **Channel**: The wireless channel through which waves propagate is tightly linked to the wavelength of the transmitting antenna. This wavelength's width plays an important role in determining the extent of the signal coverage. In cases where the wavelength is large, waves exhibit lower sensitivity to the surrounding environment and are less susceptible to interference, enabling them to travel over long distances. Conversely, signals characterized by a smaller wavelength showcase higher level of sensitivity to environmental factors. In this scenario, the signal becomes more vulnerable to attenuation caused by objects in its path and is significantly affected by climatic conditions such as rain or snow. Consequently, signals with smaller wavelengths exhibit smaller coverage areas.

- **Small-scale Fading**: This phenomenon results from multipath propagation, where a transmitted signal interacts with the surrounding environment, leading to the reception of the signal through multiple paths. These signals upon

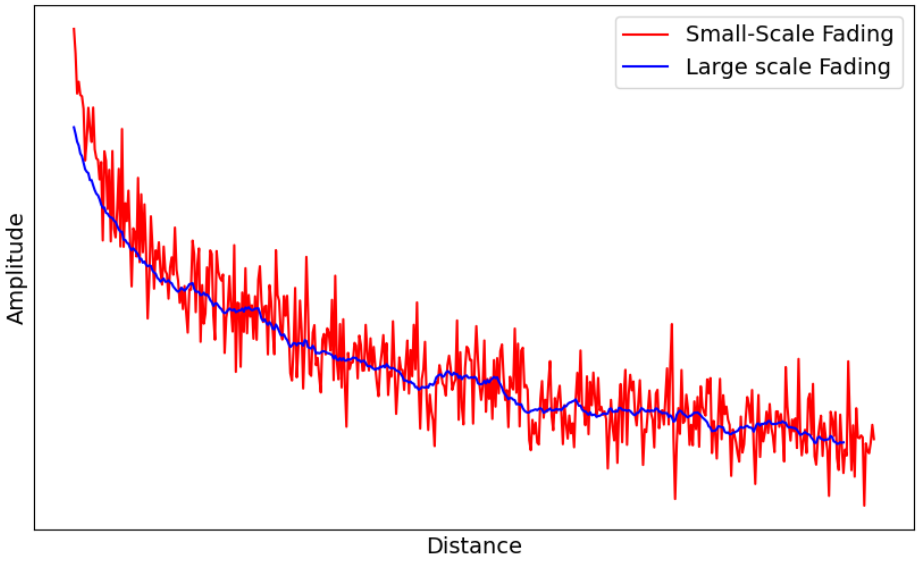


Figure 2.6: Small-scale vs Large-scale fading

reception arrive at different time intervals causing phase shifts in the received signal. The phase shifts, arising from the superposition of signals with different arrival times, give rise to interference during reception. This interference can manifest as constructive interference, amplifying the signal power, or destructive interference, causing a complete attenuation of the signal power.

Small-scale fading is characterized by a rapid and unpredictable fluctuations in both signal amplitude and phase over short distances or brief periods of time. These fluctuations result in signal variations that impact the quality of wireless communication links. Understanding and managing small-scale fading is crucial for designing effective communication systems. Different techniques are introduced in some mobile communication technologies to mitigate the effect of small-scale fading.

- **Large-scale Fading**: Refers to the gradual attenuation of signal strength as it propagates over large distances. This fading phenomenon is a consequence of factors like free space loss, absorption, and scattering, contributing to the progressive weakening of the transmitted signal as it moves away from the source. Figure 2.6 illustrates the distinction between large-scale and small-scale fading,

with the former exhibiting smooth fluctuations over long distances, while the latter undergoes rapid changes over shorter distances. The Figure shows that the large-scale fading is the mean of the fluctuations of small-scale fading [2].

Large-scale fading contains free space path loss which is the attenuation due to the free space. Additionally, it involves shadowing, which represents signal attenuation caused by obstacles present in the propagation path. The large-scale fading process is often mathematically modeled as a log-normal distribution [29], incorporating a mean and standard deviation dependent on the specific environmental conditions.

- **Doppler spread**: This phenomenon is a type of small-scale fading that occurs when the mobile receiver is in motion. The movement of the mobile receiver relative to the signal source creates a frequency shift known as Doppler Shift. The extent of this shift is linked to factors like the relative velocity between the mobile receiver and the transmitter, as well as the frequency of the transmitted signal. The observed frequency f at a given distance from the transmitter is given by (2.11) [32].

$$f = \left(1 + \frac{\Delta v}{c}\right) f_0 \tag{2.11}$$

In this equation:

  - $\Delta v(ms^{-1})$ is the relative velocity of the moving object regarding the fixed transmitting antenna.
  - $c = 3 \times 10^8 ms^{-1}$ is the velocity of the RF wave in free space.
  - $f_0(Hz)$ is the frequency of the channel.

Figure 2.7 is an example of Doppler shift on a moving vehicle. The Figure shows the shifts undergone by the signal's wave fronts as they vehicle moves towards the antenna.
In scenarios involving multipath propagation, the Doppler shift calculated for each path differs, causing these signals to reach the receiver with varying phases

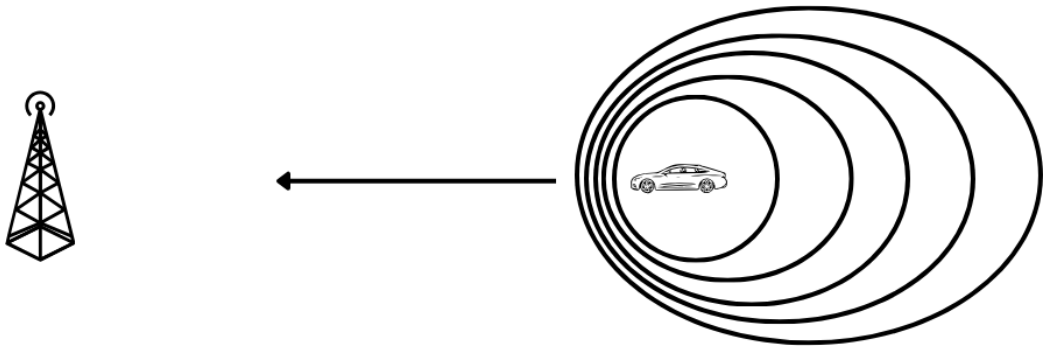


Figure 2.7: Doppler effect on a moving vehicle

and amplitudes. The frequency domain variability resulting from this phenomenon is quantified as Doppler spread, serving as a metric for the rate of change of the Doppler shifts.

- **Noise**: The received signal undergoes influence from different sources of noise at the receiver level. Primarily originating from the electronic components of the mobile receiver, this noise leads to a reduction in the received signal power. The relationship between noise and the received signal strength gives the Signal to Noise Ratio, which is a crucial metric for determining the bitrate upon reception. When noise surpasses the received signal strength, the bitrate approaches zero. Conversely, lower levels of noise correspond to higher bitrates, highlighting the significance of managing and minimizing noise for optimal signal reception.

- **Interference**: In addition to the internal interference expressed by multipath fading, signals are also affected by interference coming from surrounding antennas. Indeed, a mobile device receives traffic on a given channel frequency. Therefore, the signal emitted by all surrounding sources on that given frequency will be received by the mobile device. Hence, when the useful signal is received at the same time as other signals from surrounding antennas, it causes interference. This interference is considered as noise that is added to the background noise to form the so-called Signal to Interference plus Noise Ratio. In the case where this interference is caused by a neighboring antenna, this interference can be as high as the signal, hence completely attenuating the signal. However, strategic

network planning is usually set to mitigate such interference. This is achieved by avoiding the use of identical frequencies on neighboring antennas, hence minimizing the likelihood of signal overlap and interference, thereby optimizing overall signal reception quality.

This overview of the radio propagation mechanism showing the interactions undergone by the signal from its origin at the BS to its reception by the mobile device shows the inherent complexity of the wireless channel. This complexity shows the challenging nature of estimating the received signal power at a given location. Notably, the channel's unpredictability adds an additional layer of complexity, as it is not known in advance. This unpredictability poses challenges in accurately estimating the signal strength, further emphasizing the complex nature of wireless communication channels.

2.3 Propagation models

In the early years of the 20th century, facing the challenges posed by the unpredictable wireless channel, propagation models emerged in the 1900s. These models are essentially sets of equations and algorithms designed to tackle the task of estimating the received signal strength at a mobile receiver over a wireless channel. Their primary function is to compute the path loss, which represents the attenuation experienced by the signal during radio propagation. This path loss is meant to incorporate all the mechanisms in the wireless channel that attenuate the signal. Since it is the attenuation during radio propagation, the PL is directly linked to the transmitted signal power P_{tx} in Watts, and received signal power P_{rx} in Watts by the formula in (2.12).

$$PL(dB) = 10 \log \left(\frac{P_{tx}}{P_{rx}} \right) \tag{2.12}$$

Propagation models serve as a cost-effective solution for estimating received signal power across various geographical locations [29]. Relying on parameters such as the

distance to the transmitting antenna, propagation channel frequency, and antenna gains, some of these models provide estimates of the received signal strength at any given location.

Despite the uncontrollable nature of the wireless channel, these models offer a practical means of addressing the complexities of signal propagation. However, as highlighted earlier, the wireless channel's dynamic and changing nature, as one moves from one point to another makes its accurate modeling very challenging. Given the diversity of environments, finding a universal model applicable to all types of propagation environments is difficult. Consequently, propagation models adopt three distinct approaches to address this challenge, corresponding to three types of propagation models. Empirical models compute the PL using equations obtained from experimentation and measurements performed in a city or a country. Stochastic models estimate the PL by modeling the environment using a set of random variables. Finally deterministic models take detailed information about the environment when estimating the path loss.

2.3.1 Empirical propagation models

Empirical propagation models are developed through experimentation and/or measurements conducted in specific cities or countries, considering the unique characteristics of those environments. These models aim to capture the path loss as a function of various terrain features, such as streets, buildings, and transmitting/receiving antennas' information. While the equations generated from these experiments are designed for general application across different terrains, it is acknowledged that their performance is optimal when applied in environments having the characteristics of their origin [41].

Recognizing this limitation, ongoing research in this field has led to the evolution of these models. Various optimizations have been introduced to incorporate detailed characteristics of the specific environments in which they are employed. The journey

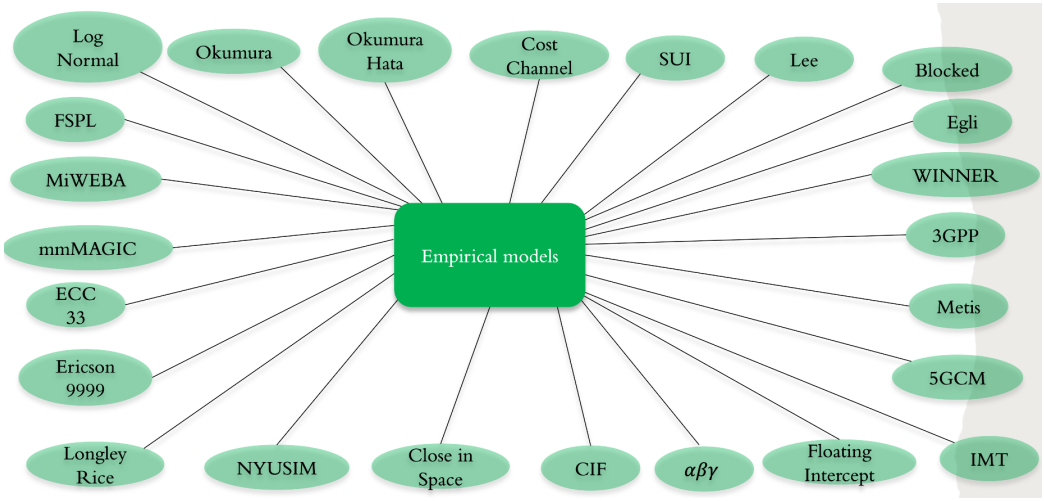


Figure 2.8: List of the most known empirical propagation models

from the original Friis equation in 1946 [42], which did not consider the environment, to the contemporary 3GPP models [43, 44] reflects this evolution, with the latter incorporating calibration parameters such as average street width and building height.

Over the years, dozens of empirical models were developed, each representing an enhancement of their predecessors. These models, often designed for specific frequency bands, complement each other in providing comprehensive coverage. Furthermore, these models are known for their ease of implementation and straightforward computation of path loss once calibrated with specific parameters. Figure 2.8 shows the most known empirical propagation models [29].

Nevertheless, despite their efficiency in estimating the path loss, these empirical models face challenges related to their accuracy. Calibration parameters, while useful, remain rough approximations of the considered area, leading to imprecise estimations. Notably, the models lack fine-grained information about the environment, as demonstrated in Figure 2.9, where two mobile devices located at the same distance from the transmitting antenna experience identical path losses despite one being in LOS and the other one in NLOS. This issue arises from the models' limited consideration of the local environment as it neglects the impact of individual buildings and the terrain's topography, which play crucial roles in real-world signal propagation mechanisms.

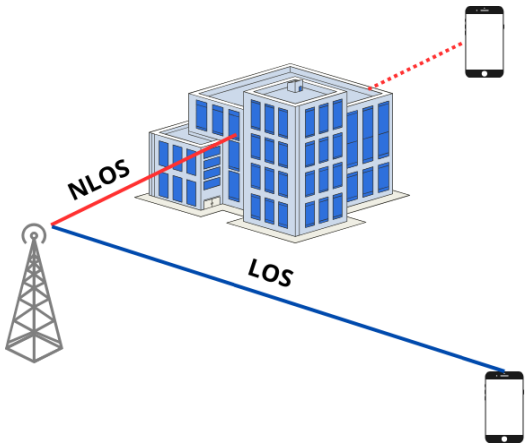


Figure 2.9: Empirical model's issue in a real-life scenario

To facilitate a better understanding of these models, we present here three empirical models, emphasizing on the derivation of their path loss formulas.

2.3.1.1 Free Space Path Loss

The FSPL model also known as Friis equation is the simplest model employed to estimate the path loss [45]. Designed in 1946 by the radio engineer Harald Friis [42], this model operates under the assumption of an unobstructed path between the transmitter and the receiver. It provides a fundamental understanding of how signal strength diminishes with the distance in unobstructed transmission scenarios. In this idealized scenario, the signal undergoes attenuation simply due to the effect of the distance, resulting in a path loss that is inversely proportional to the square of the distance traveled by the signal. Therefore, as the signal covers a larger distance, the path loss increases, leading to a decrease in the received signal power.

Given that the Friis equation is formulated for free space conditions, representing LOS scenarios where signals reach the receiver without encountering obstacles, it is the one that is commonly employed to compute the LOS path loss. The FSPL formula's is obtained as follows.

Consider two isotropic antennas, i.e., antennas that radiate uniformly in all directions: one at the transmission and the second at the reception. The transmitting

isotropic antenna sends signal in all directions of a sphere. At distance d, the power density P_s which is the power per unit of surface is given by (2.13).

$$P_s = \frac{P_{tx}}{4\pi d^2} \tag{2.13}$$

The receiving isotropic antenna receives the signal in all directions. The aperture A_e of such antenna is derived through thermodynamic considerations [46] and is given by (2.14).

$$A_e = \frac{\lambda^2}{4\pi} \tag{2.14}$$

Hence, the power received by the isotropic antenna is how much signal it collects given its aperture, as shown in (2.15).

$$P_{rx} = P_s A_e = P_{tx} \left(\frac{\lambda}{4\pi d} \right)^2 \tag{2.15}$$

This formula is generalized to all types of antennas by the introduction of the notion of antenna gains. Indeed, in real life, antennas do not radiate uniformly in all directions as it is the case with the idealized isotropic antenna. Rather, antennas radiate in privileged directions determined by the gain of the antenna that defines how well it radiates in a given direction. Hence, Equations (2.13) and (2.14) respectively include the gain G_{tx} of the transmitting antenna and the gain G_{rx} of the receiving antenna, to account for the amplification of the signal over a certain direction. Therefore, the Friis equation for any type of antenna is given by the formula in (2.16).

$$P_{rx} = P_{tx} G_{tx} G_{rx} \left(\frac{\lambda}{4\pi d} \right)^2 \tag{2.16}$$

Finally, the propagation path loss according to (2.12) is given by (2.17).

$$PL(dB) = 20 \log \left(\frac{4\pi d}{\lambda} \right) - G_{tx}(dB) - G_{rx}(dB) \tag{2.17}$$

In this equation:

- λ is the wavelength of the wave.
- d is the distance traveled by the ray from the transmitter Tx to the receiver Rx.

2.3.1.2 Okumura-Hata Model

The Okumura-Hata model is an extension of the Okumura Model [47]. This extension was brought by Hata et al., since the original Okumura model was designed to work only in the city of Tokyo in Japan, where it was formulated [48]. It served for long time as the standard for planning terrestrial mobile systems in Japan [49, 50]. To overcome this geographical limitation, Hata leveraged the fitting curves from Okumura's report [47] to introduce a novel propagation loss formula that is applicable across various terrains, and not only in Japan [29].

The resulting model called Okumura-Hata model was also formulated from experiments conducted in the city of Tokyo in Japan. It is specifically tailored for urban areas and incorporates correction factors to accommodate suburban and rural environments. Additionally, it includes correction factors for mobile devices based on the terrain type. The formula is suitable for UHF and VHF bands, with a minimal formulation error, under the following conditions:

- The frequency range f_c must be between 100-1500 MHz.
- The distance d ranges from 1 to 20 km.
- The antenna height h_{tx} must be between 30 and 200 meters.
- The receiving antenna height h_{rx} must be between 1 to 10 meters.

The standard formula for propagation loss as obtained from Hata's derivation is given for urban areas over the aforementioned conditions as:

$$PL_{urban}(dB) = 69.55 + 26.16 \log f_c - 13.82 \log h_{tx} - a(h_{rx}) + (44.9 - 6.55 \log h_{rx}) \log d \tag{2.18}$$

In (2.18), $a(h_{rx})$ is the correction factor in dB of the receiving antenna height. This correction factor depends on the type of the coverage area. For small and medium cities, i.e. cities with average building height not more than 15 meters, this correction factor is given by (2.19).

$$a(h_{rx}) = (1.1 \log f_c - 0.7)h_{rx} - (1.56 \log f_c - 0.8) \tag{2.19}$$

For large cities, i.e. the ones featuring average building heights of more than 15 meters, the correction factor is expressed as follows:

$$a(h_{rx}) = \begin{cases} 8.29(\log 1.54h_{rx})^2 - 1.10 & , f_c \leq 200MHz \\ 3.2(\log 11.75h_{rx})^2 - 4.97 & , f_c \geq 400MHz \end{cases} \tag{2.20}$$

The Hata model as shown above is designed for urban areas. To make it works in suburban and rural areas, correction factors are introduced. For suburban areas, the correction factor $K_{suburban}$ is given in equation (2.21).

$$K_{suburban}(dB) = 2\left[\log\left(\frac{f_c}{28}\right)\right]^2 + 5.4 \tag{2.21}$$

Hence, the path loss in suburban is the path loss in (2.18) minus the correction factor in (2.21) as shown below.

$$\begin{aligned} PL_{suburban}(dB) &= PL_{urban}(dB) - K_{suburban}(dB) \\ &= 64.15 + 26.16 \log f_c - 13.82 \log h_{tx} - a(h_{rx}) + \\ &\quad (44.9 - 6.55 \log h_{rx}) \log d - 2\left[\log(f_c/28)\right]^2 \end{aligned} \tag{2.22}$$

Finally in rural areas, the correction factor K_{rural} is given below as:

$$K_{rural}(dB) = 4.78(\log f_c)^2 - 18.33 \log f_c + 40.94 \tag{2.23}$$

Consequently, the path loss for rural areas is given by Equation (2.24).

$$\begin{aligned} PL_{rural}(dB) &= PL_{urban}(dB) - K_{rural}(dB) \\ &= 28.57 + 44.49 \log f_c - 13.82 \log h_{tx} - a(h_{rx}) + \\ &(44.9 - 6.55 \log h_{rx}) \log d - 4.78 (\log f_c)^2 \end{aligned} \tag{2.24}$$

Finally, since suburban and rural areas meet the condition of a small and medium cities, equation (2.19) is used to compute the correction factor of the mobile receiver's height $a(h_{rx})$ for these areas.

This model although intuitive and applicable to any type of terrain is not applicable on frequencies above 1.5GHz, i.e. this model cannot be applied to estimate the received signal power for some 4G LTE/LTE-A and 5G NR frequencies. Moreover, the model is not applicable on some urban micro cell antennas, due to the distance considered that starts only from 1km. Rather, the model is suitable for large cells where the height of the BS is higher than all the surrounding obstacles [29].

2.3.1.3 3GPP Model

This model was developed by 3GPP which is the consortium responsible for standardizing the 4G LTE/LTE-A and 5G NR mobile communication generations. This model is designed to represent the dynamic environments of 4G and 5G technologies in various terrains: urban, suburban, and rural settings, both indoor and outdoor. It allows users to employ a path loss model tailored to their specific environment and technological requirements. Developed through extensive experimentation, the equations incorporated in these models include numerous calibration parameters, ensuring a faithful reproduction of real-world conditions. These parameters are drawn on global information about the environment such as streets, buildings, and other elements critical to mobile communication infrastructure.

For 4G LTE/LTE-A, 3GPP considers three outdoor scenarios corresponding to different 4G use cases [43] as presented below.

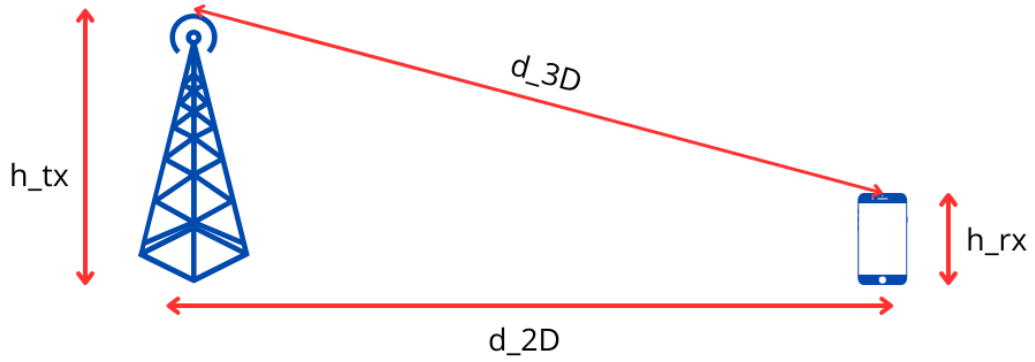


Figure 2.10: 3GPP distance and height definition

- 3D-UMi: This scenario depicts urban environments equipped with micro cell antennas. It characterizes densely populated areas with a high concentration of UEs and BSs strategically positioned below surrounding structures.

- 3D-UMa: Similar to the previous scenario, this setting involves urban areas, but with macrocell antennas. Unlike micro cells, macro cells cover larger distances and have BSs located above surrounding buildings. This configuration is commonly found in suburban settings.

- 3D-RMa: This scenario is designed for rural regions featuring macro cell antennas. In contrast to urban scenarios, rural areas often employ macro cells to cover large distances with few buildings, reflecting the lower population density characteristic of rural environments.

For these scenarios, the 3GPP models give the path loss for both LOS and NLOS use cases. A LOS probability function is also given to determine whether there is LOS or not and hence to decide which path loss formula to apply. Additionally, the path loss models outlined for 4G LTE/LTE-A are designed for applications within the frequency range of 2-6 GHz, which aligns with the typical frequency band used in 4G communication systems. Figure 2.10 is an illustration depicting the definitions of the distances within the 3GPP path loss models.

For 3D-UMi in LOS, the following conditions must hold.

- The default BS height is 10 meters.
- $1.5m \leq h_{rx} \leq 22.5m$

The path loss is defined as:

$$PL = PL_{3D-UMi-LOS} = \begin{cases} 22\log(d_{3D}) + 28 + 20\log(f_c) & , 10m < d_{2D} < d'_{BP} \\ 40\log(d_{3D}) + 28 + 20\log(f_c) - & \\ 9\log\left[(d'_{BP})^2 + (h_{tx} - h_{rx})^2\right] & , d'_{BP} < d_{2D} < 5000m \end{cases} \tag{2.25}$$

The BP's distance, d'_{BP}, is a function of the channel frequency f_c in Hz, the RF wave velocity in free space c and the effective antenna heights at BS and UE: h'_{tx} and h'_{rx} respectively, that are functions of the real heights of the antennas [51].

$$d'_{BP} = 4h'_{tx}h'_{rx}\frac{f_c}{c} \tag{2.26a}$$

$$h'_{tx} = h_{tx} - 1 \tag{2.26b}$$

$$h'_{rx} = h_{rx} - 1 \tag{2.26c}$$

For hexagonal cell layout in 3D-UMi-NLOS scenario, the condition for h_{tx} and h_{rx} are exactly the same. The 2D distance d_{2D} must satisfy: $10m < d_{2D} < 2000m$. The path loss is finally given as follows.

$$PL_{3D-UMi-NLOS} = max(PL'_{3D-UMi-NLOS}, PL_{3D-UMi-LOS}) \tag{2.27a}$$

$$PL'_{3D-UMi-NLOS} = 36.7\log(d_{3D}) + 22.7 + 26\log(f_c) - 0.3(h_{rx} - 1.5) \tag{2.27b}$$

In order to determine whether there is LOS or not, a probabilistic function is given by (2.28).

$$Pr_{LOS} = \begin{cases} 1 & , d_{2D} \leq 18m \\ \frac{18}{d_{2D}} + \exp\left(-\frac{d_{2D}}{36}\right)\left(1 - \frac{18}{d_{2D}}\right) & , d_{2D} > 18m \end{cases} \tag{2.28}$$

The 3D-UMa LOS path loss is exactly the same as 3D-UMi in LOS as given in (2.25) with only a difference in the parameter value of h_{tx}, whose default value is now 25

meters instead of 10m as in 3D-UMi and satisfies $10m < h_{tx} < 150m$.

In the case of 3D-UMa-NLOS, new parameters and new conditions are included.

- $10m < d_{2D} < 5000m$.
- $10m < h_{tx} < 150m$ and its default value is 25 meters.
- The height of the UE h_{rx} satisfies $1m < h_{rx} < 10m$ and its default value is 1.5 meters.
- The street width is introduced and must satisfy: $5m < W < 50m$, with a default value of 20 meters.
- The average building height must satisfy: $5m < h < 50m$ and its default value is 20 meters.

The path loss model is given as follows.

$$PL_{3D-UMa-NLOS} = max(PL'_{3D-UMa-NLOS}, PL_{3D-UMa-LOS}) \tag{2.29}$$

$$\begin{aligned} PL'_{3D-UMa-NLOS} = {} & 161.04 - 7.1\log(W) + 7.5\log(h) - (24.37 - 3.7(\frac{h}{h_{tx}})^2)\log(h_{tx}) + \\ & (43.42 - 3.1\log(h_{tx}))(\log(d_{3D}) - 3) + 20\log(f_c) - \\ & (3.2(\log(17.625))^2 - 4.97) - 0.6(h_{rx} - 1.5) \end{aligned} \tag{2.30}$$

The condition for determining which of LOS or NLOS formula to use is given by (2.31), showing the LOS probability.

$$Pr_{LOS} = \begin{cases} 1 & , d_{2D} \leq 18m \\ \left[\frac{18}{d_{2D}} + \exp\left(-\frac{d_{2D}}{63}\right)\left(1 - \frac{18}{d_{2D}}\right)\right]\left[1 + C'(h_{rx})\frac{5}{4}\left(\frac{d_{2D}}{100}\right)^3 \exp\left(-\frac{d_{2D}}{150}\right)\right] & , d_{2D} > 18m \end{cases} \tag{2.31}$$

In this equation:

$$C'(h_{rx}) = \begin{cases} 0 & , h_{rx} \leq 13m \\ \left(\frac{h_{rx}-13}{10}\right)^{1.5} & , 13m < h_{rx} \leq 23m \end{cases} \tag{2.32}$$

Finally for the 3D-RMa, the following conditions are applied in the case of LOS.

- The h_{tx} default value is 35 meters and satisfies $10m < h_{tx} < 150m$.
- The height of the UE h_{rx} satisfies $1m < h_{rx} < 10m$ and its default value is 1.5 meters.
- The street width satisfies $5m < W < 50m$, with a default value of 20 meters.
- The average building height must satisfy: $5m < h < 50m$ and its default value is 5 meters.

The path loss model is then given by (2.33).

$$PL_{3D-RMa-LOS} = \begin{cases} PL_1 & , 10m < d_{2D} < d_{BP} \\ PL_2 & , d_{BP} < d_{2D} < 10000m \end{cases} \tag{2.33}$$

In this equation:

$$\begin{cases} PL_1 = 20\log(\frac{40\pi d_{3D} f_c}{3}) + min(0.03h^{1.72}, 10)\log(d_{3D}) - min(0.044h^{1.72}, 14.77) + 0.002\log(h)d_{3D} \\ PL_2 = PL_1 + 40\log(\frac{d_{3D}}{d_{BP}}) \\ d_{BP} = \frac{2\pi h_{tx} h_{rx} f_c}{c} \end{cases} \tag{2.34}$$

In the case of NLOS, all the conditions remain the same as the LOS case, except for the distance that must satisfy: $10m < d_{2D} < 5000m$. The path loss for 3D-RMa in NLOS is derived as follows.

$$PL_{3D-RMa-NLOS} = max(PL'_{3D-RMa-NLOS}, PL_{3D-RMa-LOS}) \tag{2.35}$$

$$PL'_{3D-RMa-NLOS} = 161.04 - 7.1\log(W) + 7.5\log(h) - (24.37 - 3.7(\frac{h}{h_{tx}})^2)\log(h_{tx}) + (43.42 - 3.1\log(h_{tx}))(\log(d_{3D}) - 3) + 20\log(f_c) - (3.2(\log(11.75h_{rx}))^2 - 4.97) \tag{2.36}$$

The LOS probability in 3D-RMa is given by (2.37).

$$Pr_{LOS} = \begin{cases} 1 & , d_{2D} \leq 10m \\ \exp\left(-\frac{d_{2D}-10}{1000}\right) & , d_{2D} > 10m \end{cases} \tag{2.37}$$

After deriving all these formulas that give the path loss for these scenarios, 3GPP also derives on the same basis the path loss models for a frequency range from 0.5GHz to 100GHz. This is meant to generalize the aforementioned path loss models and make them work on the 5G NR technology. More information on these models can be found in [44].

2.3.2 Stochastic propagation models

As demonstrated in Section 2.2, the propagation channel is inherently unpredictable as it is characterized by various random phenomena. It is in response to this inherent randomness of the propagation channel that stochastic models have been developed.

Also known as statistical models, stochastic models constitute a set of propagation models that describe the wireless environment through the utilization of random variables following specific statistical distributions. Specifically, these models capture the fading experienced by signals as they travel between the transmitter and the receiver. This trip is influenced by the presence of obstacles in the propagation environment, leading to the occurrence of multipath propagation and hence to both small-scale and large-scale fading.

Small-scale fading is inherently a random process, as it involves unpredictable rapid fluctuations in the signal amplitude and phase over short periods of time and/or

distance. Various stochastic propagation models have been developed to characterize this phenomenon and derive the Probability Density Function of the received signal power based on the fluctuations in the small-scale fading [52, 53]. These models include Rayleigh fading, Rice fading, and Nakagami-m fading. Furthermore, the mean of the fluctuations in small-scale fading, known as large-scale fading, is itself a random process. This randomness is due to the shadowing effect caused by obstacles in the propagation path and is modeled by the log-normal shadowing model.

Nevertheless, when estimating the amplitude of the received signal, these models do not account for details in the propagation environment. Rather, they provide a global PDF illustrating the signal distribution. Not considering even little information about the current area where these stochastic models are applied, results in a coarse estimation of the received signal power. Indeed, the small and large-scale fading phenomena arise from obstacles present in the environment, and the positions and extents of these obstacles vary across different terrains. Therefore, failure to consider these individual obstructions leads to a low accurate model.

To provide a more practical understanding of how these models operate, we introduce three of them below: the Rayleigh and Rice fading models, which characterize small-scale fading, and the log-normal fading model, which models the impact of shadowing.

2.3.2.1 Rayleigh fading

Rayleigh fading is employed to model small-scale fading in situations where there is no direct LOS between a transmitter and a receiver. This scenario occurs when the transmitted signal undergoes multiple reflections and scatterings off various obstacles, which is particularly common in urban and suburban areas with numerous objects. Therefore, in NLOS scenarios, the amplitude α of the signal follows the Rayleigh

distribution, as described by Equation (2.38).

$$P(\alpha) = \frac{2\alpha}{\Omega} \exp\left(-\frac{\alpha^2}{\Omega}\right), \quad \alpha \geq 0 \tag{2.38}$$

In this equation, $\Omega = \overline{\alpha^2}$ represents the average power of the small-scale fading.

2.3.2.2 Rice fading

Rice fading is the generalization of Rayleigh fading. It is applicable in scenarios where there is a dominant ray, namely a LOS component whose amplitude is higher than that of the others. In the case where such condition is fulfilled, the received signal amplitude α follows the Rice distribution as given in (2.39).

$$P(\alpha) = \frac{2\alpha(1+n^2)\exp\left(-n^2\right)}{\Omega} \exp\left(-\frac{(1+n^2)\alpha^2}{\Omega}\right) I_0\left(2n\alpha\sqrt{\frac{1+n^2}{\Omega}}\right), \alpha \geq 0 \tag{2.39}$$

In this equation:

- I_0 is the modified Bessel function of the first kind [54, 55].
- Ω remains the average power of the fading.
- n is the diffuse parameter and is related to the Rice factor by $K = n^2$.

The Rice factor K is the ratio between the power of the LOS component and all the NLOS ones [2]. When there is no LOS component, i.e $K \to 0$ ($n \to 0$), the Rice fading in (2.39) simplifies to the Rayleigh fading in (2.38).

2.3.2.3 Log-normal Fading

Since shadowing or large-scale fading is the mean of the fluctuations of the small-scale fading, the random variable of concern is the mean of the received signal power. Studies show that the shadowing follows a log-normal distribution [29, 43, 44, 56]. It

is defined as $\psi = 10 \log \mathbb{E}[\alpha^2]$ and follows the log-normal distribution given in (2.40).

$$P(\psi) = \frac{1}{\sqrt{2\pi}\sigma_\psi} \exp\left(-\frac{(\psi - \mu_\psi)^2}{2\sigma_\psi^2}\right) \tag{2.40}$$

In this equation σ_ψ and μ_ψ are respectively the standard deviation and the mean of the random variable ψ. The values of σ_ψ and μ_ψ can be determined empirically depending on the type of the terrain.

2.3.3 Deterministic propagation models

Deterministic models also referred to as site-specific propagation models are based on the fundamental laws governing electromagnetic signal propagation, namely the Maxwell equations [3, 30]. These models simulate the interactions between the electric and magnetic fields of the signal with their environment throughout their journey from the transmitter to the receiver. The simulation aids in computing the received signal strength at the mobile receiver by incorporating detailed information about the propagation environment.

These models leverage fine-grained data, including a 3D map of the environment containing obstructions to estimate the signal attenuation caused by these objects. Deterministic models meticulously account for the effects of these objects to estimate the attenuation of the signal during its propagation. The effect of buildings, streets, relief, and trees surrounding the mobile receiver is considered by these models when computing the path loss. Furthermore, as objects in the propagation environment can cause both small-scale and large-scale fading, deterministic models can accurately represent these phenomena, enabling precise estimation of the attenuation caused by objects in the propagation path.

Unlike empirical models that provide generic path loss formulas applicable to all environments, deterministic models tailor their calculations based on local information surrounding the mobile receiver to compute the path loss. Now, two mobile users

located at the same distance of an antenna are less likely to have the same signal power estimations if they are not surrounded by the same exact obstructions.

Due to their consideration of the effects of all the obstacles in the propagation environment, deterministic models are the most accurate propagation models that exist [29, 30]. The path loss and consequently the received signal power estimated by deterministic models are extremely accurate as they reflect the reality of the mobile receiver. Moreover, deterministic models are applicable to all types of terrains and to all mobile communication technologies.

Nevertheless, this accuracy is at the cost of high computational complexity due to the fine-grained details of the environment considered. These details involve a lot of computation to accurately determine the interaction of the signal with the objects surrounding them. This complexity of deterministic models leads to a high execution time and a high memory consumption. Furthermore, due to their complexity, it is often challenging if not impossible to use these models in some complex scenarios and use cases.

In the upcoming sections, we present the most used deterministic models nowadays, namely the Finite Difference Time Domain and Ray Tracing. The former is an analytical solution to Maxwell's equations and the second is an optical ray-based method.

2.3.3.1 Finite Discrete Time Domain

As highlighted in Section 2.1, Maxwell's equations establish the relationship between the electric and magnetic fields. Determining the values of these fields at a specific time and location facilitates the computation of the power density of waves using the magnitude of the Poynting vector as defined in (2.10). However, solving Maxwell's equations is a challenging task due to their complex nature, represented as spatial and temporal partial differential equations in their expanded forms, with no exact solution existing in the literature [57–60].

The FDTD model introduced by Kane Yee in 1966 [61, 62], provides a deterministic approach to numerically solve Maxwell's equations in both spatial and temporal domains. It leverages the precision of these equations by approximating the set of differential equations using finite difference methods. The fundamental concept behind FDTD involves discretizing both the time and the space domains, enabling the computation of electric and magnetic fields at specific time-space pairs. Finite differences are applied to approximate Maxwell's equations in these discretized points. Once the fields are determined, the power density at these points is computed by evaluating the magnitude of the Poynting vector. This approach enhances the accuracy of FDTD, as it considers the local properties of the environment under study. Notably, for each discretized point, the characteristics of the materials, including the permittivity ϵ, the permeability μ, and the conductivity σ, are taken into account during the field computation.

FDTD employs the final difference or central difference technique to approximate derivatives of fields, enabling the numerical resolution of the differential Maxwell's equations. This approach is derived based on the Taylor series and is defined as follows.

Consider a function $f(x \pm \frac{\Delta x}{2})$ that is C^∞ at a given point x, i.e. f has continuous derivatives of all orders at x. The Taylor series of f is expressed as:

$$f\left(x \pm \frac{\Delta x}{2}\right) = f(x) \pm \frac{f'(x)}{1!}\left(\frac{\Delta x}{2}\right)^1 + \frac{f^{(2)}(x)}{2!}\left(\frac{\Delta x}{2}\right)^2 \pm \frac{f^{(3)}(x)}{3!}\left(\frac{\Delta x}{2}\right)^3 + \cdots$$
$$\Leftrightarrow \frac{f\left(x + \frac{\Delta x}{2}\right) - f\left(x - \frac{\Delta x}{2}\right)}{\Delta x} = f'(x) + \frac{f^{(3)}(x)}{3!}\frac{(\Delta x)^2}{2^3} + \cdots$$

Based on this, the second-order central difference approximation of the derivative of f is expressed in (2.41) as:

$$f'(x) = \frac{\partial f}{\partial x} = \frac{f\left(x + \frac{\Delta x}{2}\right) - f\left(x - \frac{\Delta x}{2}\right)}{\Delta x} + \mathcal{O}(\Delta x)^2 \approx \frac{f\left(x + \frac{\Delta x}{2}\right) - f\left(x - \frac{\Delta x}{2}\right)}{\Delta x} \tag{2.41}$$

To estimate the received signal power at any given location within the area of in-

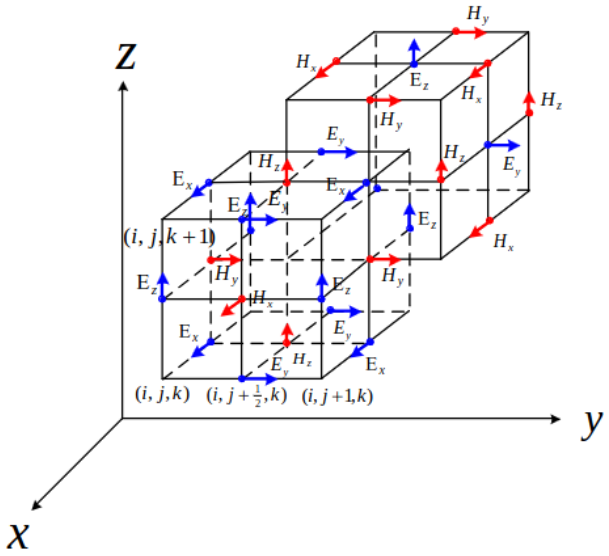


Figure 2.11: 3D Yee Grid. This Figure is taken from [2].

terest, FDTD discretizes the propagation environment using 3D grids or cubes. This discretization involves placing consecutive virtual cubes in the environment, collectively forming the Yee grid. Figure 2.11 illustrates an example of this grid, and its construction is outlined as follows.

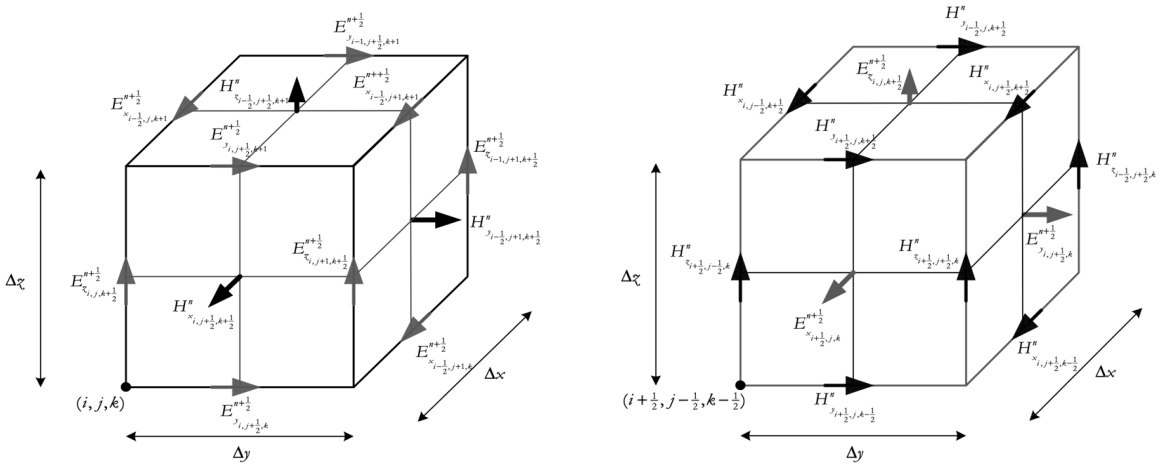
For the initial cube:

- Electric fields are positioned at all the edges of the cube, resulting in 12 electric fields, corresponding to the 12 edges of the cube.
- Magnetic fields are positioned at the midpoints of the cube's face, resulting in 6 magnetic fields, corresponding to the 6 faces of the cube.

Subsequent cubes are connected to the preceding one in a way that at common surfaces:

- The edges of the previous grid become the midpoints of the current grid's faces.
- The midpoints of the previous grid become the edges of the current grid.

This technique is applied so on and so forth until the entire space is covered with



(a) Magnetic field computation (b) Electric field computation

Figure 2.12: Zoom on Yee grids to compute the electric and magnetic fields. This Figure is taken from [3].

cubes of the same size, determined by Δx, Δy and Δz. To ensure stability, the cube size must be a fraction of the wavelength. Consequently, as the frequency increases, the number of cubes used also increases. Additionally, since FDTD staggers the fields in both space and time, the time step Δt must adhere to the condition: $\Delta t < \frac{\sqrt{(\Delta x)^2+(\Delta y)^2+(\Delta z)^2}}{C_{\max}}$, where $C_{\max}$ represents the maximum velocity of the different mediums in the propagation environment.

After obtaining the Yee-grid for a given propagation environment, the subsequent step involves approximating the different fields on each cube using the central difference method in (2.41). To perform these approximations, each field is identified by its coordinates in both 3D space and time. The following notation is used for field identification, where X is either x, y, or z.

$$H_X(x,y,z,t) = H_X(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = H^n_{X_{i,j,k}} = H^n_X[i,j,k]$$
$$E_X(x,y,z,t) = E_X(i\Delta x, j\Delta y, k\Delta z, n\Delta t) = E^n_{X_{i,j,k}} = E^n_X[i,j,k]$$

Figure 2.12 provides a zoom on one of these cubes to aid in approximating the field values. Figure 2.12a assists in computing the magnetic fields. For example,

the x-component of the Maxwell-Faraday equation given in (2.4) is approximated as follows.

$$\mu\left(\frac{H_x^{n+1}[i,j+\frac{1}{2},k+\frac{1}{2}]-H_x^{n}[i,j+\frac{1}{2},k+\frac{1}{2}]}{\Delta t}\right)=\left(\frac{E_y^{n+\frac{1}{2}}[i,j+\frac{1}{2},k+1]-E_y^{n+\frac{1}{2}}[i,j+\frac{1}{2},k]}{\Delta z}\right)-\left(\frac{E_z^{n+\frac{1}{2}}[i,j+1,k+\frac{1}{2}]-E_y^{n+\frac{1}{2}}[i,j,k+\frac{1}{2}]}{\Delta y}\right)$$

This equation is further used to derive the expression of $H_x^{n+1}[i,j+\frac{1}{2},k+\frac{1}{2}]$. The y and z components are derived in the same manner. This operation is repeated for all the magnetic fields on the cubes.

Similarly, the electric fields' values are approximated using Maxwell-Ampere's equation in (2.3). Figure 2.12b helps in approximating the x-component of the electric field, which, for the provided example, is given below.

$$\epsilon\left(\frac{E_x^{n+\frac{1}{2}}[i+\frac{1}{2},j,k]-E_x^{n-\frac{1}{2}}[i+\frac{1}{2},j,k]}{\Delta t}\right)=\left(\frac{H_z^{n}[i+\frac{1}{2},j+\frac{1}{2},k]-H_z^{n}[i+\frac{1}{2},j-\frac{1}{2},k]}{\Delta y}\right)-\left(\frac{H_y^{n}[i+\frac{1}{2},j,k+\frac{1}{2}]-H_y^{n}[i+\frac{1}{2},j,k-\frac{1}{2}]}{\Delta z}\right)-J_x^n[1+\frac{1}{2},j,k]$$

This equation is also used to compute the value of $E_x^{n+\frac{1}{2}}[i+\frac{1}{2},j,k]$. All the other electric fields and their components are computed in the same fashion.

It is important to note that, at a given time step, the value of the magnetic field at a given position depends on its value at the previous time step and its neighboring electric fields. Similarly, at a given time step, the value of the electric field at a given position depends on its value at the previous time step and its magnetic fields.

Once the approximations for all the fields are completed, the FDTD algorithm is run. The values of the electric fields and magnetic fields are computed until a steady state is reached for all spatial components of the fields. After this convergence, the power density corresponding to each position is computed using the magnitude of

Algorithm 1 3D FDTD algorithm

INPUT:

- **MaxTimeStep:** The maximum number of time steps to reach the steady state
- **N:** Number of discretized points
- **H:** A table of size $N \times 3$ containing the magnetic fields at each position
- **E:** A table of size $N \times 3$ containing the electric fields at each position
- H_0**:** Value of the magnetic field at position $(0, 0, 0)$
- E_0**:** Value of the electric field at position $(0, 0, 0)$

INITIALIZATION:

- H[0] $\leftarrow H_0$
- E[0] $\leftarrow E_0$
- For $i = 1$ to N:
  - $H[i] \leftarrow [0, 0, 0]$
  - $E[i] \leftarrow [0, 0, 0]$

BEGIN:

```
while i < MaxTimeStep do
    for j = 0 to N do
        H[j] ← Update equation of H[j]
        E[j] ← Update equation of E[j]
    end for
    i ← i + 1
end while
for j = 0 to N do
    Power[j] ← E[j] · H[j]
end for
```

END

the Poynting vector. The FDTD algorithm to compute the power density for all the discretized points is given in Algorithm 1.

As a second-order approximation of Maxwell's equations, FDTD stands out as a highly accurate method for estimating the received signal power at any given location within a propagation environment. It comprehensively considers the material characteristics within the propagation area, naturally incorporating the effects of all obstacles.

However, its iterative nature makes parallelization challenging, resulting in a high

computational load. Additionally, the discretization scheme employed by FDTD demands a high amount of memory, particularly in large environments. This demand is accentuated in high-frequency scenarios, where smaller cube sizes lead to an increased number of discretization points, resulting in high resource consumption and higher execution times.

2.3.3.2 Ray Tracing

At high frequencies, Maxwell's equations, which govern electromagnetic waves, approximate the Eikonal equation [30, 63] given in Equation (2.42), where S represents the Eikonal function and $n = \sqrt{\epsilon_r \mu_r}$ denotes the refractive index of the medium.

$$|\nabla S|^2 = n^2 \tag{2.42}$$

The Eikonal equation shown above serves as the foundational principle describing the trajectory of light rays as they traverse a medium [39]. This elucidates the duality that exists between electromagnetism and geometrical optics. This duality states that, at high frequencies, waves exhibit the same behaviors as light rays and vice-versa, i.e. all the physical properties applied to light rays hold true for electromagnetic waves.

It is from this intrinsic duality that the Ray Tracing propagation model has emerged. RT is a deterministic propagation modeling approach that simulates the behavior of radio waves from the transmitter to the receiver using light rays. As the fundamental units of RT, rays share the same physical properties as light rays, as summarized below.

- They adhere to Fermat's principle of least time: In homogeneous mediums, rays propagate in straight lines.
- They follow the Snell's laws of reflection and refraction. First, when an incident ray meets a boundary between two mediums, the incident ray, the reflected ray, and the normal to the surface lie within the same plane, known as the incident

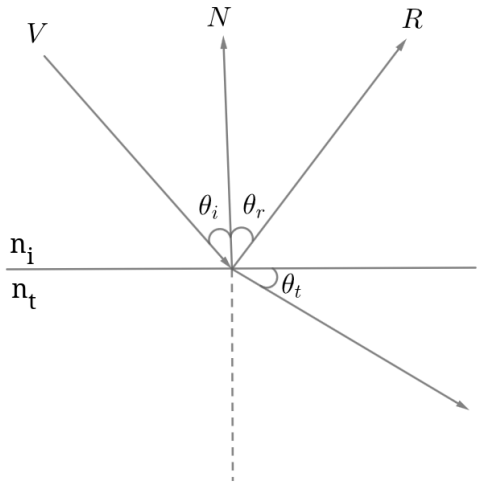


Figure 2.13: Illustration of the laws of reflection and refraction

plane.

Additionally, the angle of incidence θ_i is equal to the angle of reflection θ_r as shown in Equation (2.43) and depicted in Figure 2.13.

$$\theta_i = \theta_r \tag{2.43}$$

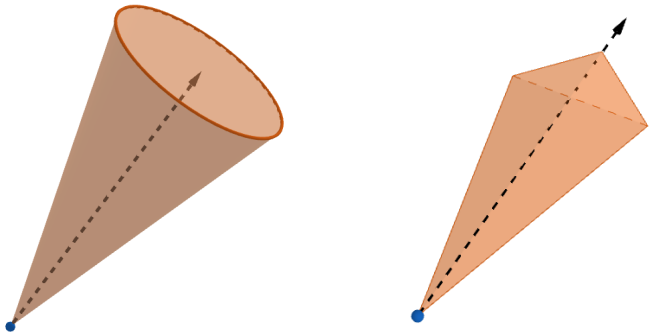
This law also determines the direction of propagation of the reflected ray R whenever it hits the boundary separating two mediums as a function of the incident ray's direction V and the normal to the surface N as shown in Equation (2.44), where $\cdot$ is the scalar product.

$$\mathbf{R} = \mathbf{V} - 2(\mathbf{V} \cdot \mathbf{N}) \cdot \mathbf{N} \tag{2.44}$$

The second Snell's law links the angle of incidence to the angle of refraction θ_t as shown in Equation 2.45, where n_i and n_t are the refractive indices of the two mediums respectively.

$$n_i \sin\theta_i = n_t \sin\theta_t \tag{2.45}$$

- Rays also obey to the law of diffraction when encountering a boundary between two mediums.



(a) A ray cone (b) A ray cube

Figure 2.14: Illustration of a ray cone and a ray tube

- A ray carries energy contained within a cone or tube surrounding the central ray as illustrated in Figure 2.14.

In order to estimate the received signal power at a given location, RT considers all the 3D information about the topography and the buildings. Since rays propagate in the same fashion as light rays, the interaction of each individual ray with the environment is tracked. This includes the buildings hit by the rays, the impact of the topography and the attenuation brought by them. These rays are tracked until their reception by the receiving antenna. The final power received by the receiving antenna accounts for all the mechanisms that rays undergo during radio propagation, as it is the sum of the energies carried by all its received rays.

Taking into account these fine-grained details of the propagation environment and their effect on waves, RT stands out as a very accurate propagation model. They are able to accurately estimate the received signal power in any kind of propagation environment settings. Therefore, RT is the most used deterministic model over FDTD and other methods. Since it provides the most accurate environment modeling and offers the possibility to compute the received signal power at a given position without the need to iterate over all the environment of propagation. Furthermore, RT is less complex than FDTD, since the latter is iterative and challenging to be parallelized and its convergence time may be exponential in complex scenarios such as when dealing with mmWaves.

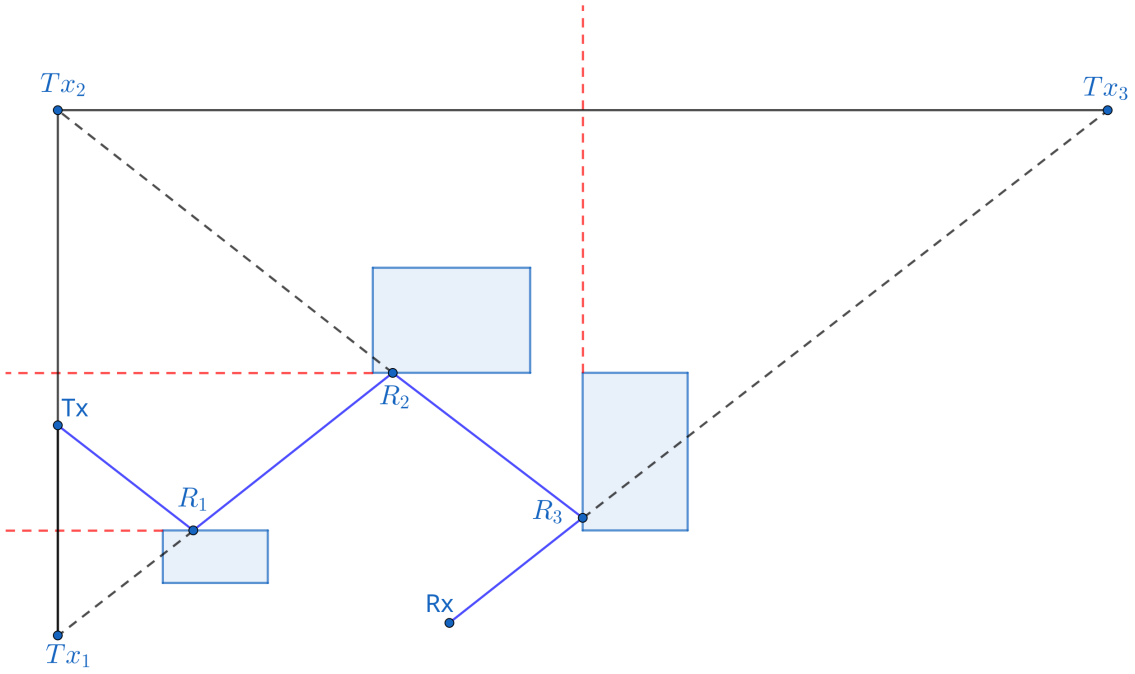


Figure 2.15: Ray path finding using the image method

There exists two types of RT models depending on how rays are launched and how the ray/object interaction is performed: the *Image method* and the *Shooting and Bouncing Ray Tracing* method.

The image method aims at constructing the exact path followed by rays between the transmitter Tx and the receiver Rx. It is based on a recursive construction of the transmitter's image while considering all the obstacles present in the environment as potential reflectors. Starting from the receiver, the images of all the objects are traced regarding Tx and rays with correct path between Rx and Tx regarding the objects are considered as valid rays. The other rays not leading to Tx are discarded. The process used by the image method to construct the exact path is outlined below and illustrated in Figure 2.15 that shows only the correct path that leads to the exact path towards the receiver.

1. Locate the image or the symmetry Tx_1 of the transmitter Tx regarding all the obstacles' facets.

2. Locate the image or the symmetry Tx_2 of Tx_1 regarding all the other obstacles' facets.

3. Recursively continue this process for all these images until a correct path towards the receiver Rx is found.

4. Identify all the obstacles and images in the correct path.

5. Starting from Rx, trace a straight line between Rx and the last image found (Tx_3 in the Figure) and locate the intersection point with the obstacle (R_3 in the Figure).

6. From the intersection point, trace a straight line towards the next image.

7. Recursively continue this process until the first reflection point (R_1 in the image).

8. Finally, trace a line, between the first reflection point R_1 and the transmitter Tx. From Figure 2.15, the obtained reflection sequence is: $Tx - R_1 - R_2 - R_3 - Rx$.

9. Redo the steps 4-8 for all the correct paths found.

As shown Figure 2.15, the image method is very precise, since the exact path between Tx and Rx is constructed, hence being able to estimate with high precision the signal power received by Rx.

Nevertheless, this method has a complexity of $O(N^K)$ [64] with N and K being respectively the number of buildings' facets and the maximum number of reflections allowed. In dense areas, containing a lot of complex buildings and where a large number of reflections is required, the computation time of this method explodes, making it not suitable for those scenarios.

In contrast, the SBR approach [64] is a brute force method that launches rays in all possible directions from Tx in order to reach the receiver. All these rays are then tracked throughout the wireless medium until their reception by the receiving antenna. The SBR method is composed of three main processes: (i) the ray generation, (ii) the Ray Object Intersection test and (iii) the reception test.

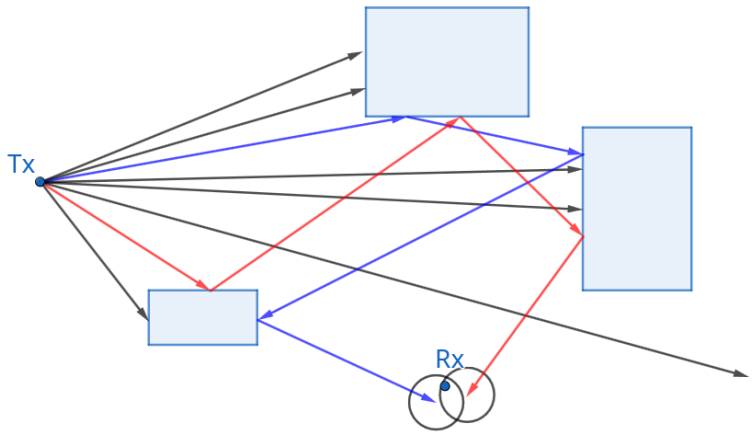


Figure 2.16: Illustration of SBR

Ray generation consists of generating a finite set of rays around Tx in order to fully cover its 3D radiation pattern. Once rays are generated, they are launched in the area, geometrically tracked, and their intersections with buildings are tested. When a ray hits a building's wall, reflection and refraction occur. Diffraction occurs when a building's edge is hit. Using the laws of geometrical optics and uniform theory of diffraction, the reflected, refracted, and diffracted rays are traced, and the process continues until the maximum number of reflections and diffraction is reached. On the receiver side, a reception test is carried out in order to determine rays that are being received by the receiving antenna and that must be accounted for in the calculation of the received power. Before performing this reception test, rays that are below a certain signal strength threshold, the ones that reach the maximum number of reflections and/or diffraction, and the ones that go out of the area of interest are discarded. The reception test is performed only with the rest.

Figure 2.16 depicts an example of SBR. From this Figure, rays are launched in all directions, they are tracked until they are near Rx. We can see from it that the SBR method is relatively less accurate than the image method since the exact path between Tx and Rx is not taken, but rather an acceptable approximation of it as opposed to Figure 2.15 where the exact path is taken in the image method.

On the positive side, it has far less complexity than the image method and offers

the best accuracy/complexity tradeoff. It is also applicable to any type of terrain without an exponential growth of its complexity. Indeed, its complexity is in the order of $O(NK)$ [64], with N being the number of rays launched from Tx and K is the number of reflections, hence being independent of the number of buildings, making it a good candidate to apply in complex scenarios. This is why the focus of this thesis is on SBR, due to its less complexity as opposed to the image method.

2.4 State-of-the-art on SBR

SBR is the most cost-effective RT model as it offers the best accuracy/complexity tradeoff and is applicable to complex scenarios with only a linear complexity as opposed to the image method. Still, due to the inherent complexity of RT itself, the SBR method has high computational load. This load is due to the overhead incurred by the processes involved in SBR. All these processes include complex computations that lead to a high execution time and a high memory consumption, making them challenging if not impossible to be used in certain scenarios.

Due to this complexity, different researches were carried out throughout the years to accelerate SBR and make it usable with less computational load. Since SBR's complexity comes from its processes, each of these acceleration techniques tackles a specific issue in a given SBR process. In the upcoming sections, we present the complexities of the ray generation, and the Ray Object Intersection test processes as well as the state-of-the-art works to optimize them. These two processes are the most complex ones when performing the point-to-point power estimation with SBR. Furthermore, we present the state-of-the-art work on the use of SBR for generating RF maps showing the signal power level anywhere in a given propagation environment.

2.4.1 Ray Generation

To generate rays from an antenna and ensure that rays fully cover its 3D radiation pattern, the icosahedron technique is the most commonly used technique in the lit-

erature. An icosahedron is a polyhedron with 20 faces and 12 vertices as shown in Figure 2.17. The icosahedron technique consists in subdividing all the 20 faces of an icosahedron into multiple uniform equilateral triangles. The vertices of the obtained triangles are then projected on a unit sphere centered at the antenna's position. These vertices are then used afterwards to launch rays uniformly on the sphere in order to fully cover the isotropic pattern of the antenna. This method is widely used since it helps to fully cover an area of interest without any blank zone, i.e. without any area being missed on the sphere when rays are generated.

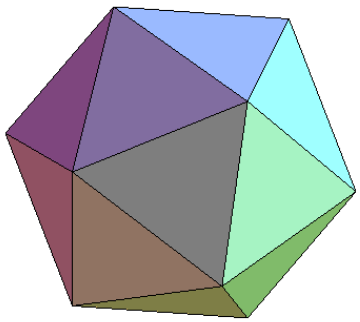


Figure 2.17: Example of a regular icosahedron. This Figure is taken from [4].

Nevertheless, it has the disadvantage of being a brute-force method since it launches rays in all possible directions without any prior knowledge of the environment of propagation. Indeed, launching rays in all possible directions may lead to a large number of rays that will a priori never reach the targeted receiver. Furthermore, since this technique has no prior knowledge about the environment, a receiver located 10 meters away from the antenna and the other one located 10000 meters from it, will have the same number of rays launched when estimating their received signals' power. Yet in reality, to estimate the received signal power for a given receiver, less rays are necessary for a receiver located near the antenna and more are necessary for a far located one.

When this happens, the unnecessary rays launched for the receiver located near the antenna will go through all subsequent SBR processes. For instance, they are tested

with all the buildings in the propagation environment to determine whether they hit each of them. This leads to a high unnecessary computational load specifically in dense areas composed of a large number of buildings. Additionally, at each reflection, a reception test is performed to test whether these rays have been received or not by the targeted receiver. Here, a large matrix computation is performed leading to a high memory consumption and the inability in some cases for SBR to estimate the received signal power due to memory limitations.

The effects of these unnecessary rays launched using the icosahedron technique are even worse in large-scale scenarios involving multiple antennas. In these scenarios, the number of unnecessary rays is multiplied by the number of antennas, hence leading to serious overhead in the SBR execution time and potential memory limitations that prevent the estimation of the received signal power by a given receiver.

Furthermore, the icosahedron technique has a high computational load. Indeed, the algorithm beneath ray generation is quite complex leading to high execution time, specifically in scenarios with large number of rays.

To overcome this complexity, different approaches are used in the literature. First, Matlab in their SBR implementation computes offline the direction and the maximum angular separation of each of the vertices of the geodesic structure obtained from the icosahedron. Three fixed triangle subdivision numbers, called tessellation frequencies, are chosen. These three tessellation frequencies are used to compute the vertices' coordinates and their maximum angular separation offline, and these values are then simply loaded in the memory [65] whenever SBR is used in Matlab. Nevertheless, this approach lacks flexibility because the number of rays launched is limited to those three choices making it impossible for Matlab users to adapt the number of rays to be launched depending on the scenario they have. Moreover, as it is using the icosahedron technique, many rays can be useless since rays are launched in all possible directions and go through bunch of unnecessary computations.

Further, the authors in [32] propose the use of a technique based on the Fibonacci

golden spiral [66]. This technique was proposed to overcome the lack of flexibility of the icosahedron technique. Indeed, with the icosahedron technique, it is not possible to choose a given number of rays to generate in a given scenario. Rather, the tessellation frequency decides of the number of rays to be generated. In the golden spiral approach, the authors were able to evenly generate points on a sphere in order to launch rays passing by them. This technique has the advantage of being flexible, i.e., one can generate any desired number of rays. However, the technique gives no clue about the suitable number of rays to launch in a given environment. Hence one must try different number of rays depending on each scenario before finding the best setting. On the other hand, this golden spiral approach is as brute force as the icosahedron technique with no prior knowledge of the environment of interest when generating rays. As shown earlier, this often leads to SBR inability to estimate the received signal power in some scenarios.

It is to account for the complexity, the lack of flexibility and the brute-force nature of the icosahedron technique, that we propose in this thesis a site-specific ray generation technique presented in Chapter 3. Our technique outperforms existing solutions to the icosahedron technique issue, because on the first hand it solves the complexity of the icosahedron technique by its ability to generate a large number of rays in a reasonable time using an iterative and adaptive process. On the other hand, our method is site specific, i.e., depending on the coordinates of the antenna and the radius of the area of interest, it only generates necessary rays that can be potentially received by the end mobile receiver. For instance, our method will generate less rays for computing the signal power for a receiver located just near the antenna and will generate more for another one that is far away. In the same manner, it generates more rays for a propagation environment of 5000 meters radius and less for a propagation environment of 500 meters.

Furthermore, for a given scenario, our method produces a range of the number of rays that can be launched. We proved that within this range, one can choose any number of rays at almost no loss in terms of accuracy. Hence, one can choose to

launch the minimum possible number of rays in a given propagation scenario while maintaining the accuracy of the signal power estimation. The latter has the advantage that, with smaller number of rays, less intersection tests with the buildings are needed as well as a smaller number of reception tests, hence reducing the high computational load of SBR.

2.4.2 Ray Object Intersection Test

The other process of particular interest in SBR is the ROI test. Indeed, when rays are generated from transmitting antennas, they are launched in the propagation environment and a simulation of their interactions with the objects present in the environment is performed. This simulation typically involves testing for each individual ray whether they hit the objects or not. Whenever a ray hits an object, the corresponding propagation mechanism is applied according to Figure 2.5 and the deviated ray is tracked until its reception by the receiver or until the energy it carries is below a certain defined threshold.

A naive ROI test algorithm checks for each pair of ray and building whether there is an intersection or not. This intersection test is systematically performed regardless of the direction of the ray and the position of the object. If N objects' facets are present in the environment and M rays are launched, $(N \cdot M)^K$ intersection tests are performed [33], with K being the the maximum number of reflections. In complex environments including a large number of buildings and large number of generated rays and where many reflections and/or diffraction are required, this process becomes out of hand and may account for more than 90% of the SBR overall complexity [33, 34].

Fortunately, different researches introduced optimizations to this naive ROI algorithm. Current SBR implementations include those optimizations that help to reduce the computational complexity of the algorithm. These techniques are mostly space division techniques that intelligently divide the space in order to perform targeted intersections. This helps to reduce the number of tests performed as they prevent the

systematic test between all the rays and all the buildings. In a nutshell, they come up with techniques that help to reduce the $(N \cdot M)^K$ complexity of the ROI test.

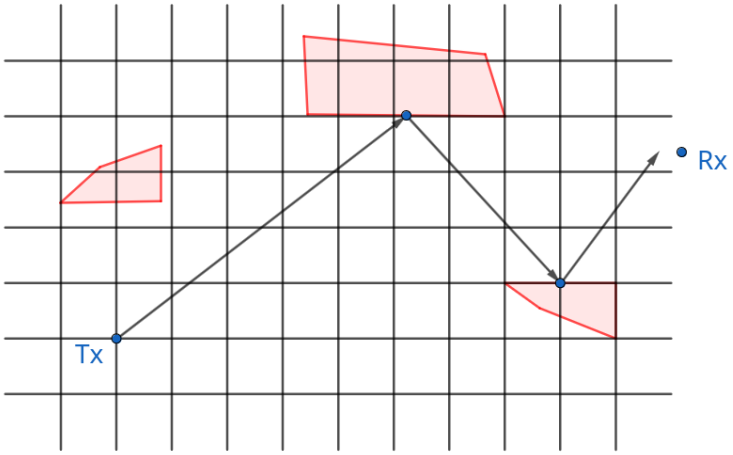


Figure 2.18: Uniform space division

One of these techniques is called the uniform division technique [67]. It is a pre-processing technique that divides the space into small cells. The space is divided into cells of the same size and each ray's direction is tracked to determine in which cell they lie in. This technique assumes that the edges of the cells are obstacles. Hence, whenever a ray is in a cell, the intersection test is performed with all the four edges that form the cell in order to determine the next cell the ray will enter in. The directions of all the rays are iteratively tracked in this fashion until the maximum number of reflections and/or diffraction is reached or until rays are received as shown in Figure 2.18.

This technique helps to reduce the number of candidate tests and therefore the computational complexity of the ROI test. It is particularly efficient when the obstacles coincide with the edges of the cells [30, 67]. Nevertheless, this criterion is hard to meet in real life scenarios, since the objects in a given environment may have completely different shapes, i.e. we can have as many shapes as objects in the environment. Furthermore, it is quite challenging if not impossible to find a cell's dimension that can enclose all the obstacles in such a way to meet this criterion. This is why the

performance of the uniform division degrades when this criterion is not met, making it not suitable for accurate SBR.

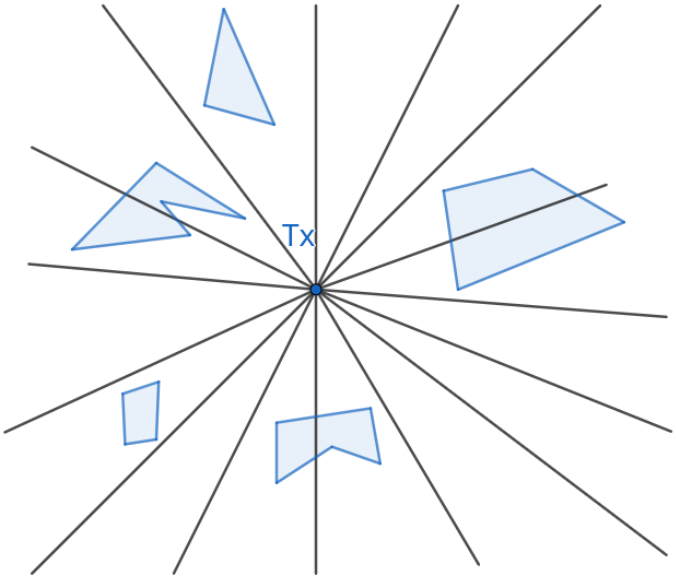
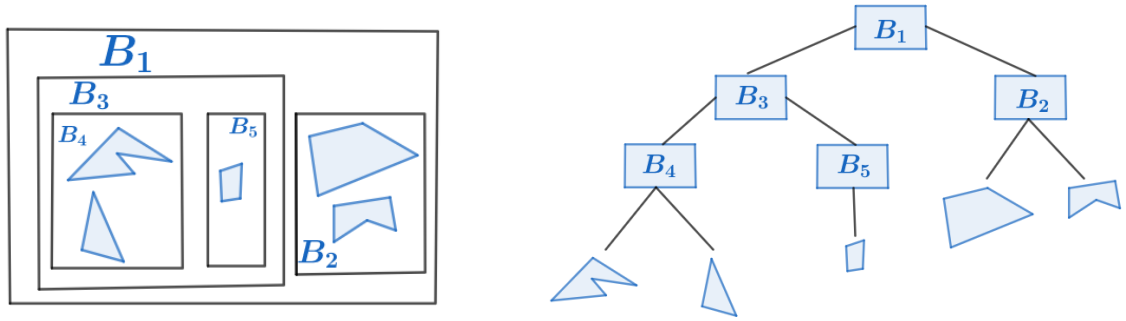


Figure 2.19: AZB space division

To overcome this lack of accuracy of the uniform division when not applied on the right scenario, the Angular Z-Buffer [33] approach was introduced. As the uniform division approach, AZB employs a smart division of the space in order to reduce the number of tests performed. As shown in Figure 2.19, the space is divided into angular regions according to the transmitting antenna if in LOS or the images of the antenna if in NLOS and the facets belonging to each region are stored. The idea beneath AZB is that whenever a ray is launched in a region, the intersection tests are performed only with the facets present in that region. To continue tracking the ray's behavior, the algorithm continues to execute in a recursive fashion until the maximum number of reflections and/or diffraction is reached.

This approach helps to drastically reduce the number of intersection tests performed without any loss in terms of accuracy. Due to this, it has become a widely used technique and has been improved throughout the years for better division of the



(a) Bounding volumes

(b) Tree structure of the volumes

Figure 2.20: BVH tree construction

space [68, 69]. Furthermore, this technique is source dependent, i.e. for all the transmitting antennas present in the environment, a regional subdivision is performed. In complex scenarios, where multiple antennas exist, this approach can easily become cumbersome and may incur overhead in the SBR process.

To overcome this antenna-dependency of the AZB algorithm, another acceleration technique based on a method used in computer graphics called Bounding Volume Hierarchy was introduced. BVH introduces another space division scheme based on bounding volumes. Objects present in the environment are wrapped in bounding volumes in a recursive fashion as in Figure 2.20a showing an example of bounding volumes of the environment shown in Figure 2.19. After the bounding volumes are constructed, a tree of these volumes is derived with individual objects being the leaves of the tree as in Figure 2.20b: with this setting, the intersection test is performed only between rays and the bounding volumes. Whenever a ray intersects a bounding volume, a recursive intersection test is performed with the children of the bounding volume until the corresponding intersected object is found. The intersection tests are not performed with the other bounding volumes, hence drastically reducing the intersection tests.

It has been proven that with BVH, the complexity of the intersection test is in the order of the logarithm of the objects present in the environment [70]. Furthermore, BVH has the advantage of being source-independent, i.e. the tree structure is built

only once and is the same for all the antennas. Hence avoiding the occurrence of extra overhead due to the large number of antennas present in the environment of propagation.

Additionally, since the same operation is performed on all the rays, BVH also benefits from the Single Instruction Multiple Data paradigm. The latter allows the intersection tests to be parallelized on the different processing units present in the computer used. SIMD is present on most recent CPUs and its combination with the efficiency of BVH helps to perform faster intersection tests. For this reason, most of the current implementations of RT use BVH to perform their intersection tests.

2.4.3 Reception Test and Power computation

As said earlier, in SBR multiple rays are launched from the transmitting antenna towards the mobile receiver, and these rays go through different mechanisms before their reception. Since the exact path is not traced between the mobile receiver and the transmitting antenna, a reception test needs to be performed between all the rays and the receiver in order to determine which ones are being received.

Due to the fact that in reality rays launched from an antenna are cone-shaped with a circular cross-Section as shown in Figure 2.14a, this reception test simply consists in checking whether the mobile receiver or the reception point is inside the ray's cone or not. Since the ray is propagating, this consists in checking if the radius of the ray is less than the minimum distance between the reception point and the ray as illustrated in Figure 2.21.

This condition is mathematically modeled by considering the radius of the ray's cross-Section when it is at its minimum distance to the reception point. Afterwards, the reception point is centered by this cross-Section called the reception sphere as illustrated in Figure 2.22. In this mathematical modeling, the ray is said to be received if it passes through this reception sphere.

Let P be the reception point on which the reception sphere is centered as shown

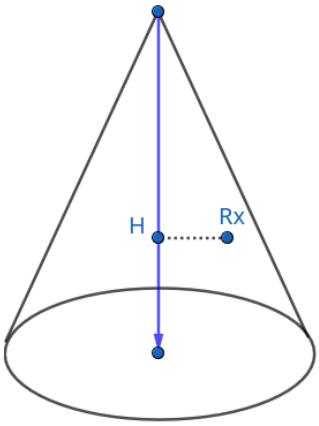


Figure 2.21: A reception point inside a ray cone

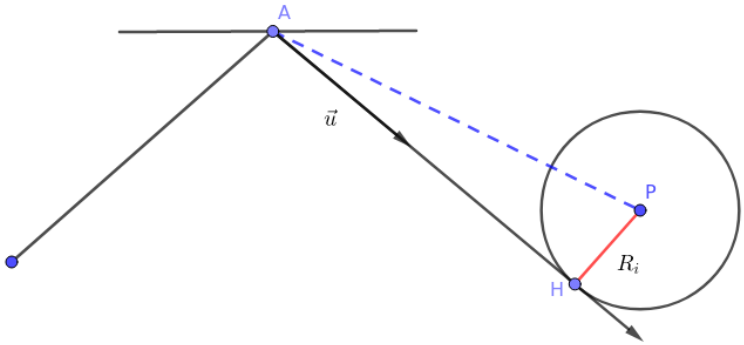


Figure 2.22: Reception test process

in Figure 2.22. The i^{th} ray is received by point P if the distance between P and its orthogonal projection on the ray (H) is less than or equal to the radius R_i of the ray at point H, i.e. if $\|\mathbf{HP}\| \leq R_i$. Point H, the orthogonal projection of point P is found using (2.46), where $\mathbf{u}$ is the unit vector of the direction of the reflected ray given by Equation (2.44).

$$\begin{aligned} AH &= \mathbf{AP} \cdot \mathbf{u} \\ H &= AH \cdot \mathbf{u} + A \end{aligned} \tag{2.46}$$

This reception test helps to determine the rays that are being received by the reception point. The sum of the energies carried by these rays determine the total

power received by the reception point and is given by Equation 2.47.

$$P_{rx} = P_{tx} G_{rx} G_{tx} \left| \frac{\lambda}{4\pi} \sum_{m=1}^{M} \prod_{n=1}^{N_m} \frac{R_{mn}}{d_m} e^{j\frac{2\pi}{\lambda}(d_m - d_{los})} \right|^2 \tag{2.47}$$

In this equation:

- d_{los} is the LOS distance between the transmitter and the receiver.
- M is the number of rays received.
- N_m is the number of reflections of the m^{th} ray.
- d_m is the unfolded distance traveled by the m^{th} ray.
- R_{mn} corresponds to the coefficient of reflection of the m^{th} ray on its n^{th} reflection.

The coefficient of reflection of a ray when it hits a surface is given by the Fresnel coefficients and it depends on the polarization of the transmitting antenna, whether vertical or horizontal. The vertical coefficient R_s which value is given in Equation (2.48), is used when the incident field E is perpendicular to the plane of incidence. On the other hand, the horizontal polarization R_p, which value is given in Equation (2.49), is used when the electric field is parallel to the plane of incidence.

$$R_s = \frac{\epsilon_r \cos\theta_i - \sqrt{\epsilon_r - \sin^2\theta_i}}{\epsilon_r \cos\theta_i + \sqrt{\epsilon_r - \sin^2\theta_i}} \tag{2.48}$$

$$R_p = \frac{\cos\theta_i - \sqrt{\epsilon_r - \sin^2\theta_i}}{\cos\theta_i + \sqrt{\epsilon_r - \sin^2\theta_i}} \tag{2.49}$$

In this equation:

- ϵ_r is the relative permittivity of the material hit by the ray.
- θ_i is the angle of incidence of the ray on the surface. Its value is determined by the dot product between the unit vectors of the incident ray $\mathbf{u}$, and of the

normal to the surface $\mathbf{n}$ as shown in Equation (2.50).

$$\theta_i = \arccos\left(-\mathbf{u} \cdot \mathbf{n}\right) \tag{2.50}$$

This expression of the received signal power in Equation (2.47) helps to derive the SINR, which gives the ratio between the useful signal and the noise. The latter is composed of the background noise N and of the interference coming from other antennas emitting on the same frequency as the principal antenna. To compute the SINR, one must compute the signal power coming from all antennas. The signal power S_i of the i^{th} antenna is computed using (2.47) based on the rays received from this antenna. The maximum signal $S = \max_i\{S_i\}, \forall i$ is considered as the signal source and all the others acting on the same frequency are considered as sources of interference. Thus, the interference I is determined by $I = S - \sum_{f_i=f} S_i$, where f_i and f correspond respectively to the frequency of the i^{th} antenna and the frequency of the signal source. Based on these, the formula of the SINR is given in Equation (2.51).

$$SINR = \frac{S}{I + N} \tag{2.51}$$

This value of the SINR is the one that is further used to compute the bitrate received by the mobile receiver. It allows for instance to determine the downlink CQI that is used to determine the corresponding bitrate depending on the technology of the BS, whether 3G, 4G or 5G. This SINR value also serves to determine the maximum capacity of a wireless link using the Shannon capacity formula in (2.52) as a function of the channel bandwidth B in Hz.

$$C = B \log_2(1 + SINR) \tag{2.52}$$

It is important to note that in the power computation formula in Equation (2.47), we only considered the reflection as the only propagation mechanism attenuating the signal. Indeed, reflection aligns with our objective of generating outdoor RF maps at large-scale. Considering diffraction in this specific use case simply adds more

complexity on a model that is already complex with only minor gains in terms of accuracy [30, 65].

2.4.4 SBR for RF Map Generation

SBR is designed to compute the path loss between a transmitting antenna and a receiving antenna. However, when it comes to generating RF maps, the issue stems from the fact that SBR was not primarily designed for this purpose. All the processes involved in the SBR algorithm as well as the acceleration techniques were not designed for RF mapping purposes. Still, to make SBR usable for this need, many researchers passed by the discretization of the space. This consists in discretizing the propagation environment using a limited number of reception points, as outlined in previous studies [26–28]. These reception points are distributed throughout the area according to some statistical distribution to ensure comprehensive coverage. All the aforementioned SBR processes are run over these reception points and the signal power associated with each receiver is calculated. The powers of these points serve as basis to generate the resulting map either discretely or by performing spatial interpolation to create a continuous map. This approach is used for example in [26] and [27] to generate a multi-cell coverage map in the case of mmWave frequency band and small-cell networks. Following the same idea, Matlab's RT toolkit performs a uniform sampling of the area with a certain resolution step between the reception points [65].

Although intuitive, this approach has several drawbacks. First, the map produced is not always precise as the precision is related to the number of reception points considered. To the best of our knowledge, no work exists about the appropriate distribution of reception points that gives the best precision over the entire map. Second, depending on the number of reception points, some rays may simply not be considered. Indeed, only rays that are received by the chosen reception points are accounted for. This can lead to a serious accuracy problem when areas are considered not to receive rays while in reality the issue comes from the fact that certain reception

points were not taken into account in those areas.

In this approach, one must be careful about the number of reception points. Indeed, insufficient density can result in imprecise maps, as it fails to capture the interference between received rays. Conversely, sampling at high rates comes with an inherent complexity caused by the reception test process, which is needed to determine if rays have been received or not by the reception points. As opposed to a point-to-point scenario, where this reception test consists of a low complex matrix computation; for RF mapping purposes, this process is subject to high computational load. Indeed, for each pair of ray and reception point, one must check whether the ray is received or not. In scenarios requiring high precision as in [23] and [24], large matrix computations need to be performed in order to determine which rays have been received by which reception points. Moreover, in large-scale scenarios, namely at city-level, millions of rays are launched and thousands or even millions of reception points are considered depending on the level of precision sought. Along with the large number of reflections that may be needed in some scenarios, the reception test can quickly get out of control. Due to this complexity, this current SBR implementation is unable to produce precise RF map in complex scenarios.

To the best of our knowledge, none of the existing RT techniques are designed for RF mapping purposes and/or are not sufficiently optimized to meet the requirements of RF map generation. To mitigate this, we propose in this thesis two different RF mapping schemes.

On one hand, we propose a new reception test algorithm solely designed to accelerate the generation of RF maps in areas with minimal terrain variations. Instead of discretizing the propagation environment to produce the RF map, our idea was to merely capture the footprint of rays on a given plane rather than performing the reception tests on a set of reception points. By doing so, we removed the discretization and the reception test processes which account for most of the overhead in SBR when generating RF maps. Therefore, we were able to divide by almost 1200 the execution time of SBR with less than 2% of memory usage as compared to baseline solutions.

On the other hand, we propose a new RF map rendering technique which in fact generalizes our reception test approach by accounting for the variations in the topography of the propagation environment. Our fast RF mapping solution is fully designed to generate RF maps at large-scale. The acceleration technique leveraged in our solution as well as the cartography solution that we propose are both tuned to meet the requirements of generating RF maps at large-scale while considering the 3D topography of the terrains and the information on the buildings. For this, we speed up the ROI test by adapting a fully optimized RT engine called *Embree* [37] developed by Intel. As shown later, Embree's complexity is only ray-dependent, i.e., the cost of the ROI test remains almost the same while changing the number of triangles (aka the resolution of the map). This property helps us to generate high quality RF maps with a large number of triangles without increasing the complexity of the ROI process. Moreover, we leverage the geometrical properties of the triangular mesh used during the ROI test to propose a new reception test scheme that is suited to RF map generation and does not require the consideration of reception points. Our scheme consists in directly performing a spatial intersection between the triangles and the circular cross-sections of the rays, instead of going through the complex matrix computations in the current reception test approach. Thanks to these underlying RT processes being introduced, we are able to move directly from the triangular mesh of the terrain to the RF map production after proposing signal power estimators for triangles and coloring them accordingly. This forms an optimized pipeline fully designed for RF mapping purposes and capable to reach the goal of RF map generation at large-scale, regardless of the type of the terrain. Our simulation results show that this optimized pipeline can divide by a factor of 50 the RF map rendering time while limiting the difference in bitrate estimation to less than 1Mbps on average as compared to existing approaches.

2.5 Summary

In their journey from the transmitting antenna to the mobile receiver, RF signals go through different propagation mechanisms that attenuate the energies they carry. This attenuation is mostly due to the obstacles present in the wireless medium. These obstacles usually modify the trajectory of the signals and divide them into multiple other signals. This phenomenon known as multipath propagation creates different fluctuations in the signal amplitude and phase known as small-scale and large-scale fading, depending on whether this fluctuation is rapid or smooth over time.

At the reception by the mobile device, different other interference coming from surrounding antennas and the components of the device itself add noise to the signal, giving rise to the so-called SINR, that gauges the amount of useful signal over the noise and therefore determines the download bitrate received by the mobile device at a certain point in time.

All these phenomena make it hard to model the wireless medium and to estimate the attenuation undergone by the signal due to the objects present in the propagation environment. This issue led to the emergence of propagation models, which are set of equations and algorithms designed to estimate this attenuation referred to as path loss. Depending on the factors considered during this path loss estimation, three different propagation models exist. The empirical model is based on measurements performed in a city, the stochastic one models the random nature of the medium and the deterministic model considers thorough details about the objects to estimate this path loss. All these three modeling approaches have their pros and cons and each of them is useful depending on the use case at hand. Table 2.1 is the summary of the differences between these modeling approaches as well as their complexity and the accuracy they offer.

Since the goal of this thesis is to accurately estimate the received signal power in any given geographical location, we focus on SBR which is a type of deterministic model offering the best accuracy/complexity trade-off. Despite its efficiency over

Which model to choose?	Empirical model	Deterministic model	Stochastic model
Based on	Measurements	Maxwell's equations	PDF
Complexity	Very easy to implement	Algorithms are very complex	Easy to implement
Accuracy	Accurate, but depends on how well it is calibrated	Very accurate	Less accurate
Computation	Less computational load	Lot of computational load	No computational load
Sensitivity to change	Less sensitive to change in the propagation area	Very sensitive to change in the propagation area	Less sensitive to change in the propagation area
Carrier frequency	Some models do not work for all carrier frequency	Works for all frequency	Works for all frequency
Need for data	Data needed only for sake of calibration	The 3D map of the environment is required	Few information on the environment is enough
Type of terrain	Some models do not work on all terrain profile	Works on all type of terrain	Works on all terrain profile

Table 2.1: Comparison between propagation models

other deterministic models, SBR is still computationally complex and was not primarily designed for RF mapping purposes. This issue in the SBR design makes it quite complex when applied to do such task, leading to its inability to generate RF maps in some complex environments due to hardware limitations. To overcome this, we propose a site-specific ray generation method that accounts for the environment on the information before generating rays. This helps to reduce the number of rays launched and hence to reduce the overhead generated by the subsequent SBR processes. Furthermore, we propose RF mapping schemes fully designed and optimized to produce RF maps at large-scale without significant loss in terms of accuracy. Additionally, since our focus in the upcoming chapters will be on SBR, we will be referring to it simply as RT.

Chapter 3

Site-Specific Ray Generation

3.1 Introduction

Ray Tracing as a propagation model that is based on the light/wave duality [30] gives very good accuracy when estimating the received signal power. Since it accounts for all the interactions between the rays and the objects present in the environment, hence considering all the attenuation caused by these objects. Nevertheless, due to the consideration of these fine-grained information when estimating the path loss, RT suffers from a high execution time. This slowness mainly comes from the high computational load and high memory consumption of the processes involved in the RT workflow. A process of particular interest in its workflow and of particular need for estimating the path loss between a transmitting antenna and a receiving antenna is the Ray Generation one, also known as Ray Launching.

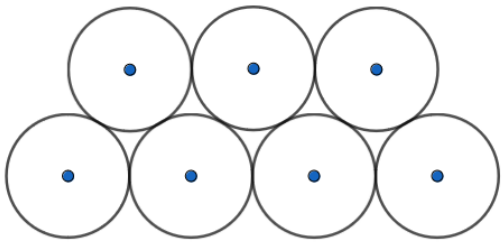


Figure 3.1: Cross-sections of rays touching each other

Ray Generation is a process that consists of generating rays in all directions in order to fully cover a transmitting antenna's 3D radiation pattern. Ideally, to achieve this full coverage, an infinite number of rays must be generated, which is not practically feasible. Rather, to approximate reality with a finite number of rays, generated rays are modeled as tubes or cones centered by a line (the ray) as in Figure 2.14, with the most used ones being ray cones. These ray cones are generated afterwards in such a way to fully cover the radiation pattern of the transmitting antenna.

To avoid having zones in the antenna's pattern not covered by the ray cones, the radius of the cross-sections of these cones must be well chosen. This radius is a function of the distance traveled by the rays and the angular separation of the ray with its neighbors.

First, if these ray cones only touch each other as in Figure 3.1, this will lead to small areas between them that are not covered by the rays. These gaps between the rays if not handled continue to increase as rays propagate, leading to big areas not covered by them. Hence, with these gaps the targeted receiver may appear as not receiving rays or may miss some rays that it receives in reality, therefore leading to inaccurate estimation of the received signal power.

The radius of rays' cones in this case where their cross-sections touch each other is derived from Figure 3.2. For a relatively small angular separation α between a ray and its neighbors, such that $\sin(\alpha) \approx \alpha$, this radius R is given by Equation (3.1).

$$R = \frac{\alpha d}{2} \tag{3.1}$$

To avoid having these gaps in the propagation environment and obtain an accurate estimation of the received power, the ray cones must slightly overlap, which passes by increasing their radius R by a constant multiplicative factor. The minimum multiplicative factor that helps to fully cover the gaps is obtained as follows.

In order to have an overlap that covers the gap in Figure 3.1, the radius R of the rays must be stretched up to the median of the equilateral triangle formed by the centers

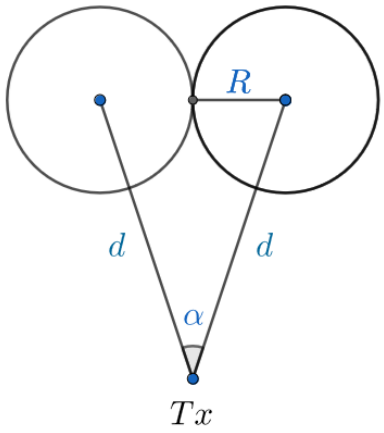


Figure 3.2: Calculation of the cone's radius when rays touch each other

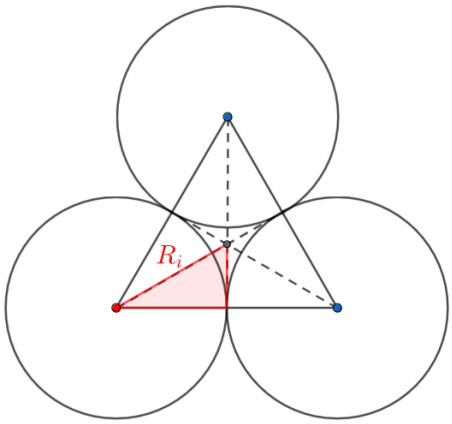


Figure 3.3: Construction of rays overlap to avoid blank zones

of the rays as shown in Figure 3.3. Therefore, the new radius of the rays that avoids any gap is the distance between the center of the ray and the median of the equilateral triangle. From the triangle highlighted in Figure 3.3, this new radius value R_i helps to find the value of the constant multiplicative factor as given in Equation (3.2).

$$R_i = \frac{2}{\sqrt{3}} R \Leftrightarrow \frac{R_i}{R} = \frac{2}{\sqrt{3}} \tag{3.2}$$

This constant multiplicative factor of $2/\sqrt{3}$ helps to obtain the radius rays should have in order to fully cover the area of interest without any gap. It is obtained by replacing the radius R in Equation (3.1) by its value in the new Equation (3.2).

Therefore, the radius R_i of the i^{th} ray's cross-section after traveling the distance d_i is given by (3.3) [64], with α_i being the angular separation of the i^{th} ray with its neighbors.

$$R_i = \frac{\alpha_i d_i}{\sqrt{3}} \tag{3.3}$$

Using this new radius enables a slight overlap between neighboring rays that allows to obtain a full coverage of the area as shown in Figure 3.4. This full coverage has the advantage that all the area of interest is fully covered by the rays and that the receiver receives all the rays it is supposed to receive from the transmitting antenna. This ensures the accuracy of the model, hence its compliance with reality.

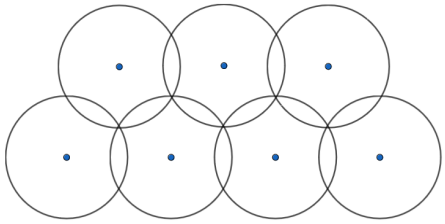


Figure 3.4: Avoidance of gaps due to the choice of the right radius

Nevertheless, there is an issue that arises from this overlapping criterion in order to ensure full coverage of the area. This is when at the reception, the receiver falls within the overlapping area as depicted in Figure 3.5. When this happens, the receiver is said to receive two rays while in reality only one ray is received, leading to an overestimation of the received signal power as proved in the literature [71, 72]. This issue is known as double counted error and is solved in the literature by considering the energy carried only by one of the neighboring rays and not both. Therefore, when computing the signal power received by a reception point using Equation (2.47), one has to make sure that all double counted rays are removed in order to avoid overestimating the received power. This simply consists in removing all the rays that follow exactly the same paths before arriving to the reception point [71, 72].

Another issue with the ray generation process is the choice of the angular separation α_i between a ray and its neighbors. On a 2D plane, one can choose the same constant

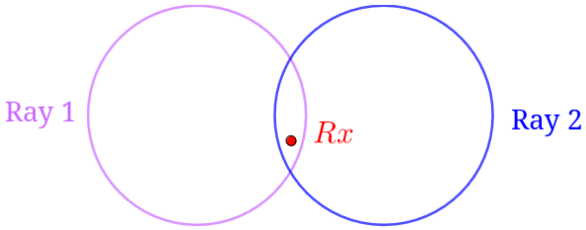


Figure 3.5: Illustration of double counted ray error

value $\alpha = \alpha_i, \forall i$ to separate rays and still fully cover the area with no gaps [64]. Nevertheless, in 3D rays are generated in such a way to fully cover a unit sphere around the antenna, supposing the latter is isotropic. At a given distance d from the antenna, a sphere has two degrees of freedom namely the elevation θ and the azimuth φ angles. These two angles must be appropriately chosen in order to avoid gaps. Indeed, applying a constant angle for both the elevation and the azimuth in 3D leads to inaccuracies and gaps in the propagation area [64] as shown in Figure 3.6. From this Figure, the gaps do not stem from the rays having the same azimuths, but rather from the fact that the same angular steps are chosen for both the elevation and the azimuth. These gaps as shown in the Figure are more severe than in 2D scenarios, since they lead to whole areas not covered at all by rays.

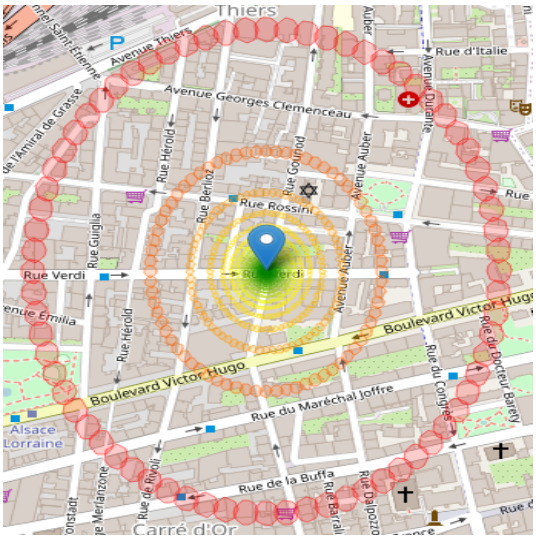


Figure 3.6: Gaps when using a constant angular separation for both azimuth and elevation angles

This problem is solved in the literature by using the icosahedron technique, which consists in subdividing the faces of a regular icosahedron into multiple uniform equilateral triangles. This subdivision approximates the sphere by equilateral triangles which vertices serve to launch rays. The icosahedron technique solves all the aforementioned issues and helps to fully cover the area of interest without any gaps. However, the main drawback with this technique is its computational slowness due to the complexity of the algorithm used to evenly generate rays. Furthermore, it has the disadvantage of generating lot of overhead in the subsequent RT processes. Indeed, it launches rays in all possible directions regardless of the position of the receiving antenna [73]. Most of these rays will a priori never reach the receiver but will still go through all the subsequent computations to determine whether or not they hit buildings and whether they have been received by the mobile device. The number of these wasted rays is related to the size of the area of interest, the larger it is, the higher will be the number of wasted rays. Hence, although the icosahedron technique solves the gap removal issue in RT, its high computational load makes it not the right choice for generating rays mainly in complex scenarios where many antennas are involved and where a lot of interactions with the obstacles exist.

To solve the gap removal issue without introducing further overhead in RT, we propose a new ray generation technique that quickly and iteratively finds the optimal number of required rays to cover the receiving area of interest. Our technique is site-specific, i.e., for each scenario it gives the optimal number of rays that need to be launched in order to fully cover the area without any blank zone. By minimizing the number of rays that need to be launched, our method reduces the overhead incurred by those rays that never reach the receiver. Moreover, by generating rays in a flexible and iterative way, we were able to overcome the complexity and the computational slowness of the icosahedron technique. By utilizing our technique, we achieved the capability to launch up to 1 million rays within 5 seconds on a laptop equipped with 16GB of memory and 7 processors running at 1.8GHz. As a result, we effectively minimized the overhead associated with the ray generation

process. Simulations were then performed with this new technique in different real-life scenarios and a validation with respect to the state-of-the-art model implementing the original icosahedron technique was carried out. After validating our model, we made an extensive simulation study on the accuracy/complexity trade-off of RT depending on whether we are in an urban, suburban, or rural area. This is meant to help one choose the suitable number of rays to launch given a specific scenario while having sufficient information about the complexity required to reach a sought level of accuracy. Through all these simulations, we were able to observe that our solution for ray generation was flexible, robust, and computationally fast at almost no cost compared to standard RT. Therefore, through our work we provide a full pipeline that, depending on the location of the antenna, the coverage area, the type of the terrain and the computational resources available, gives the minimum possible number of rays required to accurately estimate end-users' signal power received and their download bitrate.

The rest of this Chapter is organized as follows. In section 3.2, we deeply explain the algorithm behind the icosahedron technique, from the construction of the geodesic sphere to the ray launching process. In the same section, we explain the techniques used in literature to overcome its complexity. section 3.3 contains the technical details of our site-specific ray generation technique as well as how it can be implemented. The validation of the technique as well as its performance evaluation are presented in section 3.4. Our sensitivity analysis on how the accuracy of our model evolves with the number of rays is presented in section 3.5. The accuracy/complexity tradeoff study that helps one choose the number of rays to launch in a given scenario is detailed in section 3.6. Finally, the summary of this Chapter is presented in section 3.7.

3.2 State-of-the-art on the Icosahedron technique

Performing SBR on an isotropic antenna requires launching an infinite number of rays from the antenna in order to fully cover its spherical radiation pattern. Nevertheless,

this is practically unfeasible, rather, a discretization of the antenna's spherical pattern is performed in order to launch a finite set of rays that adequately approximate the sphere's surface.

The approximation of sphere's curvature has been extensively studied over the years across various fields like architecture, engineering, and molecular biology. Originating from the Platonic Solids, named after the Greek philosopher Plato, this concept involves regular and convex polyhedrons in 3D Euclidean space, where all faces are identical, and the same number of faces meet at each vertex. The polyhedrons satisfying these conditions are in the number of 5:

- Tetrahedron: 4 faces and 4 vertices;
- Cube: 6 faces and 8 vertices;
- Octahedron: 8 faces and 6 vertices;
- Dodecahedron: 12 faces and 20 vertices;
- Icosahedron: 20 faces and 12 vertices.

All these polyhedrons can be circumscribed by spheres containing all their vertices as shown in Figure 3.7. Through this circumscription process, the sphere is approximated by the polyhedron it contains. The icosahedron and dodecahedron best approximate the sphere due to their higher number of vertices and faces. Notably, they are duals of each other, implying that the faces and vertices of one are derived from the vertices and faces of the other.

Despite this duality, the icosahedron is primarily chosen for approximating spherical patterns due to its triangular facets. Triangles, being the smallest polygons, are commonly used for geometric approximations. This preference makes the icosahedron ideal for approximating the spherical pattern of an isotropic antenna.

Nevertheless, the 20 faces of the icosahedron remain a rough estimation of the curvature of the sphere, since, with the latter, one cannot capture all part'] of a sphere.

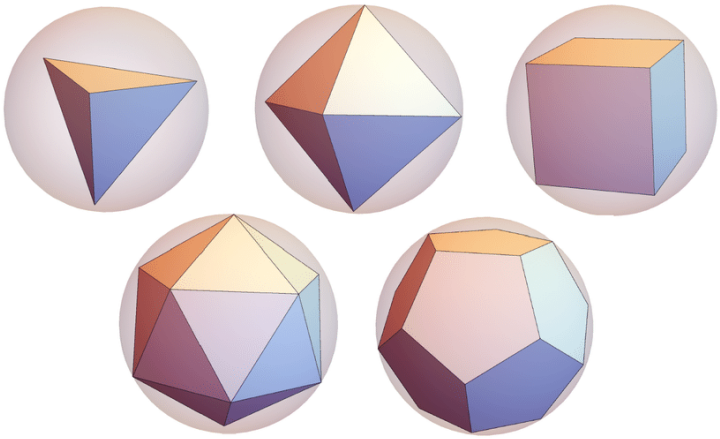


Figure 3.7: Platonic solids and their circumscribed spheres. From left to right: tetrahedron, octahedron, cube, icosahedron and dodecahedron. This Figure is taken from [5].

To mitigate this, a fine-grained approximation of the sphere is performed. It consists in subdividing the triangular facets of the icosahedron into multiple other equilateral triangles. The rate of this subdivision is known as the tessellation frequency, and it determines the quality of the approximation. The higher this frequency is, the better the sphere is approximated. To obtain the geodesic structure that approximates the sphere based on the icosahedron, the process below is followed.

1. Construct the icosahedron in Euclidean coordinates. The 12 vertices of the icosahedron are located in the coordinates given in Equation (3.4), where $\phi = (1+\sqrt{5})/2$ is the golden ratio. The resulting icosahedron is as the one presented earlier in Figure 2.17.

$$(\pm\phi, \pm 1, 0), (\pm 1, 0, \pm\phi), (0, \pm\phi, \pm 1) \tag{3.4}$$

2. Construct the circumscribed sphere to the icosahedron. This sphere must pass through all the vertices of the icosahedron as the sphere in Figure 3.7.

3. Choose n, the tessellation frequency that determines how much the edges of the icosahedron are subdivided.

4. For each face ABC of the icosahedron, choose one edge. Let AB be the chosen

edge and C the vertex facing it.

5. Subdivide the segment AB into n segments of equal length. Let $P_0, P_1, ..., P_n$ be the chosen points after this subdivision, where $P_0 = A$ and $P_n = B$.

6. Trace the segment $CP_n = CB$ and trace all the segments parallel to CP_n and bounded by $P_1, ..., P_{n-1}$ and the edge AC.

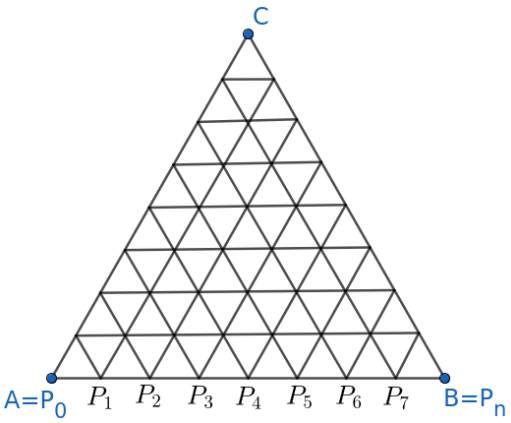


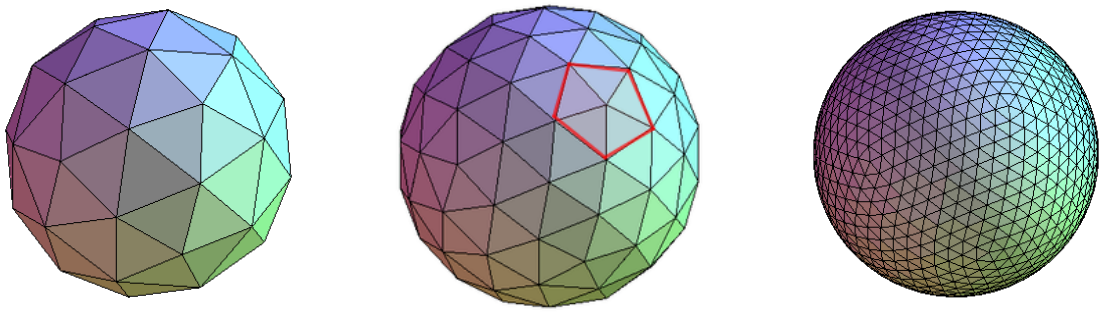
Figure 3.8: A tessellated triangle with $n = 8$

7. Repeat this operation for the edges: AC and AB and obtain a triangle as in Figure 3.8 showing an example of a tessellated equilateral triangle.

8. Repeat steps 3-7 for all the faces of the icosahedron. This will result in small equilateral triangles on each face of the original icosahedron.

9. Once the tessellated geodesic structure is obtained, project all its vertices on the circumscribed sphere of the original icosahedron. The resulting geodesic sphere is composed of triangles that better capture the curvature of the sphere as shown in Figure 3.9. The Figure shows geodesic spheres for different values of the tessellation frequency.

The vertices of the geodesic sphere obtained from this process are used to directly launch the rays from the transmitting antenna. By launching rays from the center of the geodesic structure toward the vertices, rays are launched with a certain angular separation with their neighbors. By applying Equation (3.3) on each ray, while



(a) $n = 2$ (b) $n = 3$ (c) $n = 10$

Figure 3.9: Geodesic spheres for different values of the tessellation frequency. This Figure is taken from [4].

considering the maximum angular separation, these rays' cones will slightly overlap with their neighbors in such a way to ensure comprehensive coverage of the 3D space without any gap.

Note that, from the obtained geodesic sphere, most of the vertices have six neighboring vertices since they are centers of the hexagon surrounding them. Only 12 vertices corresponding to the vertices of the original icosahedron have each, five neighboring vertices as they are surrounded by pentagons. Indeed, according to the Euler-Descartes law on transformable polyhedrons to spheres [74], it is impossible to fill a sphere with only hexagons (or vertices with six neighbors). For the case of the geodesic sphere obtained from the icosahedron, 12 pentagons (or vertices with five neighbors) must exist in order to obtain the sphere. One of this pentagon is highlighted in Figure 3.9b. Thus, the total number of rays produced using the icosahedron method can be determined as follows.

When tessellating a single triangle n times, the number of resulting small triangles is n^2. Given that an icosahedron comprises 20 faces, the total number of triangles or faces in the resulting geodesic structure is $20n^2$. Each triangle possessing 3 edges leads to $60n^2$ edges in total. These edges determine the number of vertices from which rays will be emitted. According to Euler's law, each of the 12 vertices is connected to 5 edges, while the rest are linked to 6. Thus, the total number of vertices is calculated by summing up those with five edges and those with six.

Considering the 12 vertices with 5 edges, they contribute to a total of 60 edges. Consequently, the remaining edges amount to $60n^2 - 60 = 60(n^2 - 1)$, representing the edges shared by the vertices with 6 edges. As each vertex is connected to 6 edges, the total number of vertices with 6 edges is derived as $60(n^2 - 1)/6 = 10(n^2 - 1)$. Hence, the total number of vertices forming the resulting geodesic structure is given by Equation (3.5), where V, V_5, and V_6 denote respectively the overall number of vertices, the number of vertices with 5 edges, and the number of vertices with 6.

$$V = V_5 + V_6 = 12 + 10(n^2 - 1) = 10n^2 + 2 \tag{3.5}$$

In order to ensure that the obtained geodesic sphere is fully covered by the rays without any gap, the radius of rays' cones must be as in Equation (3.3). Since each ray has five or six neighbors, the maximum angular separation between a ray and its neighbors is considered to determine its radius in Equation (3.3) and avoid gaps. The maximum angular separation ensures that there is a slight overlap between rays' cones and all their surrounding neighbors. Choosing another angle would lead to gaps between rays and their neighbors, hence not meeting the requirements of SBR ray generation.

The tessellation frequency and consequently the number of rays launched using the icosahedron technique determines the accuracy of the RT algorithm. Large tessellation frequency leads to a large number of rays launched which leads to a more accurate RT model. Indeed, with a large tessellation frequency, the angular separation between rays and their neighbors is small and according to Equation (3.3), smaller ray cones will be launched. The latter increases the accuracy of SBR because, as explained earlier, SBR as opposed to the image method does not find the exact path between the transmitter and the receiver. Rather, an approximation of the path is done based on the radius of the ray cone.

Since the exact path is not defined in SBR, the receiver is said to receive a ray if it lies in the cross-section of that ray, whether it is at the edge of the cross-section or at

its center. The closer the receiver is to the center of the ray's cross-section, the more accurate will be the ray path estimation performed by SBR. Hence, when small ray cones propagate, with the effect of the small angular separation, their radius does not increase faster with the distance, hence the radius remains small even at the reception. Since the radius is small, the shift between the estimated path and the exact one is negligible, which leads to the exact estimation of the received signal power.

Moreover, since with small radius ray cones a large number of rays is launched, the receiver is more likely to be reached by rays. Rays will follow different paths throughout space and even if some rays are blocked by obstacles, others may reach the receiver, due to the fact that no propagation path is missed. This property is very important, since waves propagate in a continuous fashion and in order to reflect the reality, the number of propagation paths missed should be close to zero.

On the opposite side, when small number of rays are launched, the angular separation values are large and with the distance, the radius of rays becomes large. At the reception, the shift between the exact and the estimated paths is more likely to be high due to the large distance that may be between the center of the ray and the position of the receiver. Hence ending up in these scenarios having a rough estimation of the received signal power.

Additionally, with this setting some ray paths may be missed as opposed to the small angular separation case. This is due to the large radius of rays and that only few number of rays are launched. This leads to either the receiver to be completely missed or not to receive all the rays it had to, thus having a coarse estimation of the received signal power with these types of rays.

Yet, despite their accuracy, choosing a large number of rays has a big impact on the complexity of SBR. Indeed, when rays are launched, each of them is tracked and they go through all SBR subsequent processes. Hence, large number of rays launched leads to high computational complexity, hence slowing down the execution time of SBR. Moreover, as the icosahedron technique consists of launching rays from all the

vertices of the geodesic sphere, the more rays are launched, the more wasted rays there will be. All these rays that will a priori never reach the receiver will go through all these cumbersome computations, hence slowing down SBR and making it unusable in some scenarios as emphasized in section 2.4.1.

On the other side, choosing a small number of rays although not as accurate as large number of rays, helps to maintain low complexity with SBR. Indeed, since less rays are launched, only small computations are needed for the subsequent SBR processing. Furthermore, the number of rays wasted due to the inherent functioning of the icosahedron technique is less, hence less unnecessary computation is performed. All these mechanisms help to reduce the computational complexity of SBR and thus to make faster estimation of the received signal power using SBR.

Having in one hand the accuracy obtained when large number of rays is launched and on the other hand the low complexity when only few rays are launched, makes it difficult to choose, for a given use case what is the suitable number of rays that gives the best accuracy/complexity tradeoff. This kind of information is necessary since some scenarios may require a large number of rays to reach a high accuracy, while some others may stick with an acceptable level of accuracy to avoid complexity. Furthermore, given the available resources at hand in a given scenario, this kind of information may help to determine the number of rays necessary to ensure a sought level of accuracy given the resources available.

Unfortunately, to the best of our knowledge, no work in the literature giving such information exists. This leads to trying random number of rays before finding the settings that suit the scenario at hand. In complex scenarios, this operation can easily become cumbersome due to the high execution time incurred when trying a given setting.

In addition to this lack of information about the number of rays to choose with the icosahedron technique in a given scenario, the icosahedron technique is itself a very complex algorithm. Indeed, the algorithm explained above is computationally

heavy, and according to Equation (3.5), the number of rays launched is quadratically proportional to the tessellation frequency. This leads to having high ray generation time mainly in scenarios where large number of rays are generated [32].

This is why Matlab [65] in their SBR implementation defined three fixed tessellation frequencies $n = \{2^8, 2^9, 2^{10}\}$ corresponding to three predefined levels of accuracy. Matlab defined these levels of accuracy as a function of the angular separations between rays and their neighbors. The *high* angular separation corresponding to $n = 2^8$ is the least accurate level, since only 40,962 rays are launched. The *medium* angular separation corresponding to $n = 2^9$ is the medium level of accuracy since 163,842 rays are launched. Finally, the *low* angular separation corresponding to $n = 2^{10}$ is the highest accurate level, since 655,362 rays are launched.

These three levels of accuracy, namely high, medium and high, were defined by Matlab to avoid running the cumbersome icosahedron technique whenever SBR is used. Instead, Matlab developers launch the algorithm for these predefined levels of accuracy offline and store the coordinates of the vertices and their maximum angular separations in the device's disk. Whenever Matlab's users choose a given level of accuracy, this information is simply loaded from the disk to the memory. Following this approach, they could avoid the computational complexity of the icosahedron technique but generated a lack of flexibility in their SBR method. Indeed, one cannot choose another tessellation frequency apart from the ones defined by Matlab.

To avoid this lack of flexibility, Matlab followed a different approach in its 2022b release in August 2022. Instead of simply loading the values to the memory, Matlab allows it users to choose the average angular separation they want between rays. Once the latter is chosen, the nearest tessellation frequency that gives a range of angular separations in which the user-defined angular separation lies in, is chosen and the corresponding rays are generated and launched. Whenever a new tessellation frequency is chosen, the icosahedron technique is executed and cached during the user running session. This is meant to avoid recomputing the coordinates of rays because whenever a new tessellation frequency is set, the SBR execution time is increased due

to the overhead incurred by the icosahedron technique.

Another issue that stems from the icosahedron technique apart from its computational complexity is its lack of flexibility in predetermining the number of rays as pointed out by authors in [32]. When using the icosahedron technique, one does not choose a number of rays to launch, rather the tessellation frequency determines the number of rays to launch. This leads to a discrete jump of $10(2n+1)$ rays launched, between n and $n+1$ tessellation frequencies, with no possibility to choose a value in between. In scenarios where a tradeoff is sought between accuracy and complexity, one becomes very limited with the degree of freedom. Consider a use case where n gives a poor accuracy and $n+1$ gives good accuracy but with high complexity, it becomes impossible to find in this specific use case a tradeoff and one must stick either with a complex model or a less accurate one. It is to avoid this that these authors used instead the Fibonacci spiral [66] that evenly discretizes a given number of points on a sphere, and is hence able to launch any given number of rays. Despite its flexibility, this approach may lead to gaps in the radiation pattern of the antenna mainly when a few number of rays are launched. Indeed, this even distribution does not always ensure the radius of ray cones to be as in Equation (3.3), hence leading to small areas not covered by ray cones. To avoid this, a large number of rays should be launched from the antenna, leading to subsequent complexities in the SBR processes.

Finally, all these solutions or alternatives to the icosahedron technique do not solve its brute force nature. Rather the Matlab solution simply applies the icosahedron technique, and the Fibonacci spiral approach is as brute force as the icosahedron technique. This shows the inability of these approaches to solve the computational complexity of the icosahedron technique. Furthermore, none of the existing solutions give the suitable number of rays to be launched in a given scenario that gives the best accuracy/complexity tradeoff. In the upcoming sections, we present our site-specific ray generation technique that accounts for the environment when launching rays. We furthermore present our accuracy/complexity study that gives the suitable number of rays to launch in a given scenario.

3.3 Site-specific Ray Generation

To account for the limitations of state-of-the-art ray generation techniques, we propose a new technique that generates rays based on local information about the propagation environment. As opposed to current techniques that launch rays in a brute-force fashion, our technique generates the minimum number of rays necessary to fully cover the potential area where the receiver is located in. We reach this goal by following an iterative and adaptive approach that helps to fully cover the area of interest without any gap while minimizing the overlap between adjacent ray cones.

To ensure that there is no gap when generating rays, all the rays generated in our approach comply with Equation (3.3). In 3D there are 2 degrees of freedom, namely the azimuth ($0 \leq \theta \leq 2\pi$) and the elevation ($0 \leq \varphi \leq \pi$), and as shown in Figure 3.6, choosing the same angular separation for these two angles leads to gaps in the area of interest. To avoid this, we set all the elevations in our approach to be iterative and adaptive, i.e the elevation on which the current ray is launched to be dependent on the previous ray's elevation. This dependency is meant to ensure that there is an overlap between rays that have neighboring elevations. Technically speaking, the goal is to find an elevation step $\Delta\varphi_i$ so that the rays on the next elevation φ_{i+1} overlap with their neighbors having φ_i as elevation angle in such a way to avoid any gap. Once $\varphi_{i+1} = \varphi_i + \Delta\varphi_i$ is found, the process continues to find φ_{i+2}, so on and so forth until $\varphi \geq \pi$.

To illustrate further our idea, we consider the illustration in Figure 3.10. Given the elevation angle φ_i, we look for the $\Delta\varphi_i$ so that the cross-sections of Ray i and Ray $i+1$ located on the same azimuth overlap with each other in such a way to avoid any gap. The value of φ_{i+1} found will help to find φ_{i+2} and so on. Proceeding this way has the advantage of being adaptive, i.e., the full area is covered with less possible number of rays launched.

From Figure 3.10, the cross-sections of Ray i and Ray $i+1$ overlap with each other, if condition (3.6) is met, with d_i being the distance traveled by the i^{th}, d_{i+1}

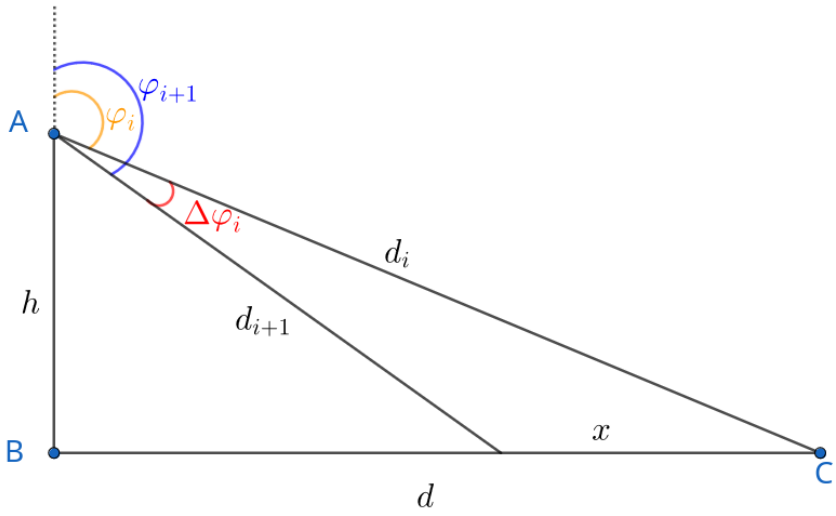


Figure 3.10: Illustration of our ray generation procedure

the distance Ray $i+1$ should travel in order to meet (3.6), and α_i is the angular separations of both rays with their neighbors. The term k is the overlapping factor that determines how much the rays must overlap. Indeed, it is not trivial to find how much the rays must overlap in order to remove any gap. Therefore, the parameter k helps to control the overlap and ensure the full coverage of the area of interest.

$$x = Radius_0 + Radius_1 \Leftrightarrow x = \frac{\alpha_i(d_i + d_{i+1})}{k} \tag{3.6}$$

The value of $\Delta\varphi$ at each iteration that avoids any gap is derived as follows. According to the law of cosines, the value of x in Figure 3.10 is given in Equation (3.7).

$$x = d_i^2 + d_{i+1}^2 - 2d_i d_{i+1}\cos(\Delta\varphi_i) \tag{3.7}$$

Let's consider the angular separation $\Delta\varphi_i$ to be small, i.e. $\Delta\varphi_i$ is less than 15°. Therefore, the small angle approximations $cos(\Delta\varphi_i) \approx 1$ and $sin(\Delta\varphi_i) \approx \Delta\varphi_i$ can be applied to it. By taking this approximation, Equation (3.7) resolves to Equation (3.8).

$$x = d_i - d_{i+1} \tag{3.8}$$

Since Equations (3.7) and (3.8) are equal, by replacing the value of x, we obtain the following equation linking the two neighboring distances.

$$(k - \alpha_i)d_i = (\alpha_i + k)d_{i+1} \tag{3.9}$$

From Figure 3.10, the values of d_i and d_{i+1} are respectively given by Equations (3.10) and (3.11), where h is the height of the antenna.

$$d_i = -\frac{h}{\cos(\varphi_i)} \tag{3.10}$$

$$\begin{aligned} d_{i+1} &= -\frac{h}{\cos(\varphi_{i+1})} \\ &= -\frac{h}{\cos(\varphi_i + \Delta\varphi_i)} \\ &= -\frac{h}{\cos(\varphi_i) - \Delta\varphi_i \sin(\varphi_i)} \end{aligned} \tag{3.11}$$

By replacing the values of d_i and d_{i+1} in Equation (3.9), and by deriving the value of $\Delta\varphi_i$, we obtain Equation (3.12).

$$\Delta\varphi_i = \frac{2\alpha_i}{(\alpha_i - k)\tan\varphi_i}, \qquad \pi/2 < \varphi_i \leq \pi \tag{3.12}$$

Furthermore, the value of k that helps to fully cover the area with no gap was found through binary search. This was done by applying Equation (3.12) to launch rays and by changing the value of k. The optimal value that removes any gaps is therefore $k = 2\sqrt{3}$.

Note that, the value of α_i, which is the maximum angular separation of the rays, depends on the angular distances in both elevation and azimuth, $\alpha_i = \max(\Delta\varphi_{i-1}, \Delta\theta_i)$. The value of $\Delta\theta_i$ is found at each iteration using the angular distance formula [75] in Equation (3.13) with the constant value β being the angle between rays in azimuth. This constant value β has a big impact on the number of rays launched since it is the one that decides of how many rays are launched for each given elevation $\Delta\varphi_i$.

$$\Delta\theta_i = \cos^{-1}\left[(cos\beta - 1)\sin(\varphi_i)^2 + 1\right]. \tag{3.13}$$

On the other hand, in Equation (3.12), we have the condition $\pi/2 < \varphi_i \leq \pi$; this is because when $\varphi_i \leq \pi/2$, the signal is not received by any receiving antenna due to the fact that most of the building's facets are vertical and that the transmitting antennas are usually higher than the receiving ones. However, following the same approach, interested readers can derive the formula for the case $\varphi_i \leq \pi/2$.

Our method is said to be site-specific, because we consider local information about the environment of interest in order to generate the optimal number of rays in that specific scenario. Practically, our method adapts the angle from where the first ray must be launched in order to fully cover the potential area where the receiver is. To do this, an initial value of the elevation φ_0, is chosen as a starting point. This value is chosen with regards to the radius of the area or the maximum coverage distance and the height of the antenna used. From Figure 3.10, this coverage distance is d and the value of φ_0 is derived in Equation (3.14) with h being the height of the antenna.

$$\varphi_0 = \pi - \arctan(\frac{d}{h}) \tag{3.14}$$

Our algorithm to iteratively identify the elevation angles of rays is summarized in Algorithm 2 shown below. Given the maximum coverage distance, the constant azimuthal angular separation β, the height of the antenna as well as its cartesian coordinates, the algorithm computes the directions on which the rays are launched as well as their maximum angular separations. These two information will be used to launch the rays and further track them to determine whether they are received or not by the targeted receiver. Figure 3.11 is a 2D projection of rays launched using our site-specific and iterative ray launching technique. The Figure shows that ray cones fully cover the area of interest with no gaps.

Nevertheless, our new ray generation technique has one limitation. It fails when the

Algorithm 2 Site-specific ray generation algorithm

INPUT:

- **d:** Maximum coverage distance or Radius of the area
- **h:** Height of the antenna
- β**:** Azimuthal angular separation
- **Tx:** (x,y,z) coordinates of the antenna

INITIALIZATION:

- k $\leftarrow 2\sqrt{3}$
- $\Delta\varphi \leftarrow 0$
- $\varphi \leftarrow \pi - \arctan(d/h)$
- $X \leftarrow [\,]$: $N \times 3$ vector containing the (x,y,z) coordinates of the rays, where N is the number of rays
- $Angle \leftarrow [\,]$: Angular separations of rays

BEGIN:

while $\varphi \leq \pi$ **do**
  for γ **in** $[0, 2\pi]$ **with step** β **do**
    X.append($[\sin(\varphi)\cos(\gamma), \sin(\varphi)\sin(\gamma), \cos(\varphi)]$)
  end for
  $\triangle\theta \leftarrow \cos^{-1}\left[(cos\beta - 1)\sin(\varphi)^2 + 1\right]$
  $\alpha \leftarrow max(\triangle\theta, \triangle\varphi)$
  $Angle$.append(α)
  $\triangle\varphi \leftarrow \frac{2\alpha}{(\alpha - k)\tan\varphi}$
  $\varphi \leftarrow \varphi + \triangle\varphi$
end while
Direction $\leftarrow \frac{X - Tx}{\|X - Tx\|}$
Return Direction, Angle

END

condition $\triangle\theta \geq \triangle\phi$ is not met. This happens on the direct bottom of the antenna, hence leading to the area at the direct bottom of the antenna not being covered. Nevertheless, this issue arises only with isotropic antennas, which correspond to a theoretical antenna not used in practice. Still, even in these cases, one can easily solve this issue by merely launching an additional ray with the unit direction vector $[0, 0, -1]$. However, when other antennas such as omnidirectional ones are used, this can simply be ignored. Indeed, these types of antennas have a donut-shaped radiation pattern and do not by design cover the direct bottom and the top of the antenna.

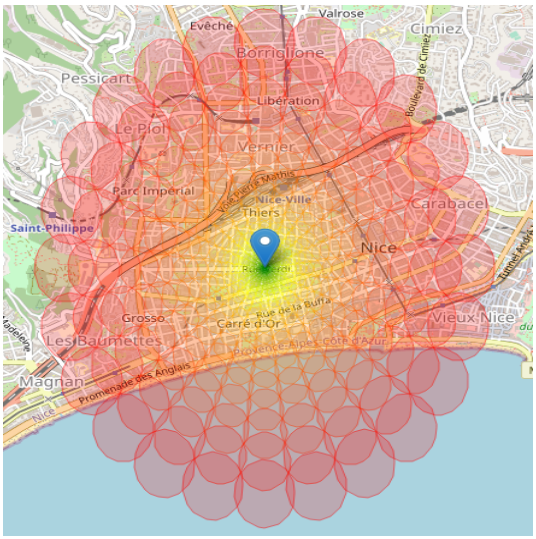


Figure 3.11: Our ray generation technique without any gaps and with minimum overlap between rays

3.4 Numerical simulations

To validate our site-specific approach, we implemented a Ray Tracer from scratch. The latter uses our site-specific approach to generate rays. The details of this Ray Tracer's implementation are provided in the appendix. Using our in-house Ray Tracer, we validated our site-specific ray generation against the approach in the literature that uses the icosahedron technique for ray generation. As explained in section 3.2, Matlab in their SBR implementation uses the icosahedron technique. Hence, we checked the correctness of our technique by comparing it with the Matlab implementation of SBR.

The main challenge with the validation of our technique against the icosahedron technique is the choice of the number of rays to launch. For a first validation of the accuracy of the received signal power, and for the purpose of fairness, we set the number of rays to be comparable in both cases. Indeed, in the icosahedron technique, rays are launched in all directions, while in ours, only the optimal number of rays useful in each scenario is launched. Since our technique launches rays only below the horizon (starting with a $\varphi_0 \geq \pi/2$), we adapt for each of the three level of angular separations in Matlab, the number of rays that need to be launched in our case. As

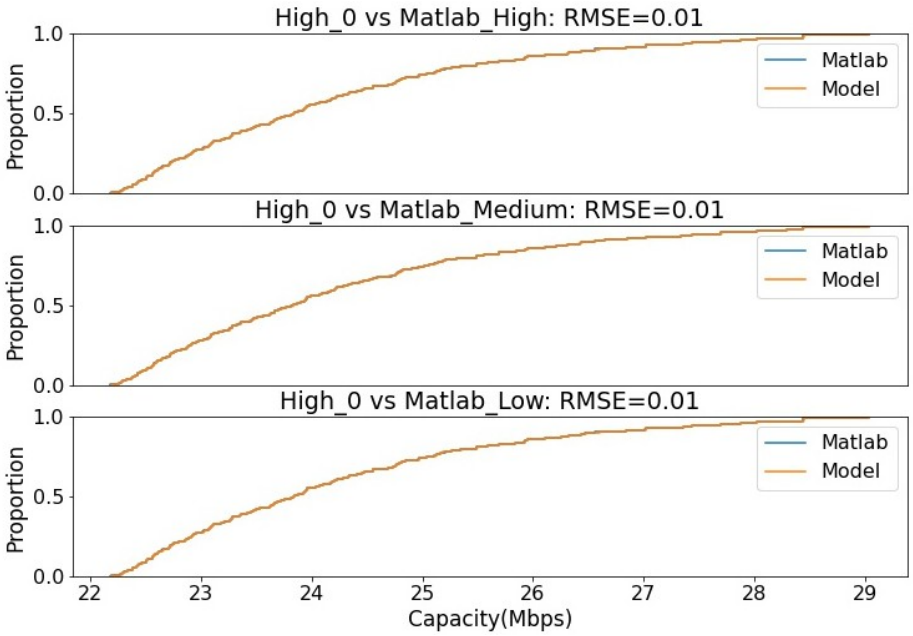


Figure 3.12: *High* angular separation in LOS

shown earlier, in these three levels, 40,962, 163,842 and 655,362 rays are launched respectively for the high, medium, and low angular separations [65]. For each of the latter case, we set the values of β (the constant azimuthal distance that determines the number of rays launched in azimuth at each elevation) to 2.35°, 1.19° and 0.75°, which corresponds respectively to 19,252, 77,005 and 308,020 rays launched in our model. Simply said, where Matlab launches 40,962 rays, 19,252 rays are the equivalent for our model and so on. For the simulation setup, we set the maximum coverage distance to 5000 meters.

We start by assessing the sensitivity of our technique regarding the number of rays launched by Matlab. We compare each of our cases to Matlab's three resolutions. We repeat this process for different maximum numbers of reflections allowed: 0, 2 and 4. Moreover, we test the accuracy of our technique on three different urban environments in the city center of Nice in France. For the bitrate computation, we used the Shannon capacity formula in Equation (2.52), which computes the maximum ideal capacity of a link, with a bandwidth of 1MHz and a receiver noise power of -107dBm.

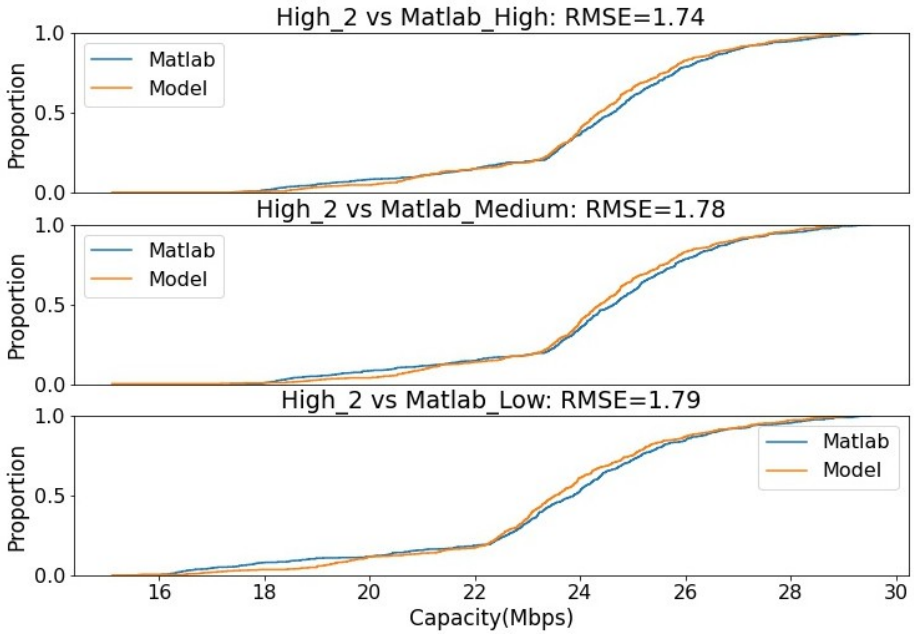


Figure 3.13: *High* angular separation with 2 maximum number of reflections

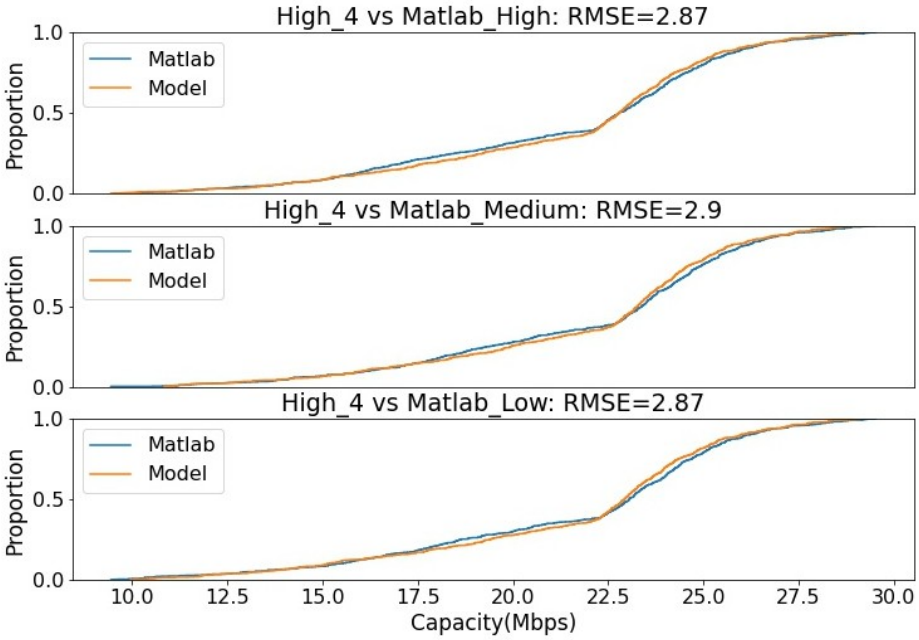


Figure 3.14: *High* angular separation with 4 maximum number of reflections

Figures 3.12, 3.13 and 3.14 show a comparison between our solution in the case of a *high* angular separation and all the three other scenarios available in Matlab for different maximum number of reflections. The figures show a direct comparison between our solution and Matlab's using the CDF of the bitrate estimation. We can observe from these figures that our bitrate or capacity estimation has the same distribution as Matlab for the three resolution scenarios of Matlab (High, Medium, and Low) as compared to the *High* resolution scenario in our model. This gives an idea on the fact that by using our approach that is adaptive and gives the less possible number of rays in a reasonable time, we can accurately estimate the signal power and consequently the download bitrate of end users. Furthermore, from the CDF we can also see that our method is not that sensitive to the number of rays launched in Matlab. We will explain this sensitivity more deeply in the next section. We can further observe that the average error made by our technique regarding the one of Matlab increases slightly and is less than 3 Mbps in all cases. This remains a good trade-off regarding the advantages offered by our technique as explained earlier.

By zooming on the case of high angular separation with 4 reflections, Figure 3.15 shows the absolute error distribution for bitrate estimation between our solution and Matlab's. We can see from this Figure that the error follows a Gaussian distribution centered around 0, i.e., most of the errors made by our model are around 0. This highlights the ability of our model to accurately estimate the signal power and the bitrate as compared to the widely used icosahedron technique.

Our simulations were performed on three urban terrains in the city of Nice in France. Since the plots for the three terrains were bringing similar results, only the plots for one terrain were shown in Figures 3.12, 3.13 and 3.14. We summarize here the results obtained from the simulations performed on the three terrains in Tables 3.1, 3.2 and 3.3.

For each terrain, the simulation was done with different angular separations and different number of reflections. Each column is a comparison between the estimation performed by our technique and the one of Matlab. For example, the column **Med_4**

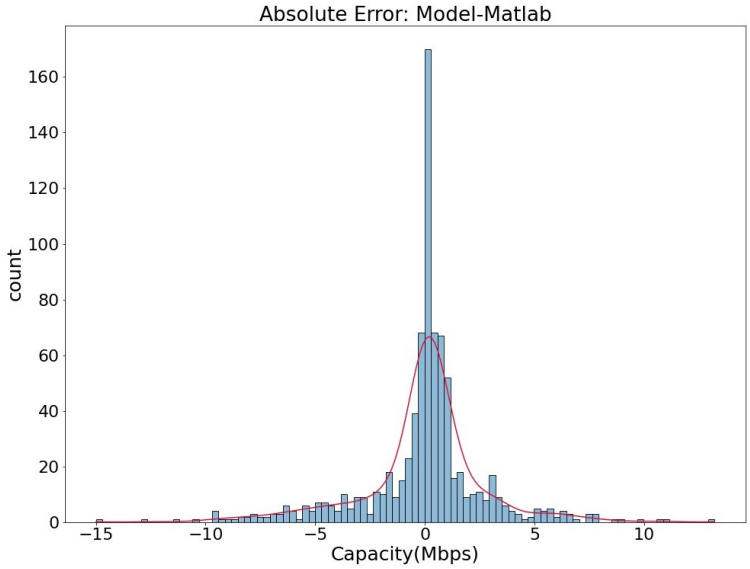


Figure 3.15: Mean Absolute Error distribution

	High_0	**High_2**	**High_4**	**Med_2**	**Med_4**
Matlab_High	0	1.51	2.48	1.45	2.41
Matlab_Medium	0	1.58	2.45	1.47	2.41
Matlab_Low	0	1.55	2.54	1.4	2.49

Table 3.1: RMSE (in Mbps) for the 1st terrain

means that we are comparing our *medium* angular separation with 4 reflections to the 3 angular separations available in Matlab. Each cell represents the RMSE in Mbps between our results and the ones of Matlab. The tables show slight variations of the RMSE from one terrain to another due to the differences in the terrains themselves. These small variations highlight the robustness and scalability of our technique, and its ability to be accurate regardless of the terrain used. Through all these simulations, one can see that our method is correct, robust to terrain change and capable of maintaining the accuracy of RT while launching less rays.

	High_0	**High_2**	**High_4**	**Med_2**	**Med_4**
Matlab_High	0	1.39	2.14	1.29	1.94
Matlab_Medium	0	1.47	2.26	1.39	2.12
Matlab_Low	0	1.38	2.28	1.27	2.18

Table 3.2: RMSE (in Mbps) for the 2nd terrain

	High_0	**High_2**	**High_4**	**Med_2**	**Med_4**
Matlab_High	0.01	1.74	2.87	1.69	2.73
Matlab_Medium	0.01	1.78	2.9	1.74	2.69
Matlab_Low	0.01	1.79	2.87	1.76	2.59

Table 3.3: RMSE (in Mbps) for the 3rd terrain

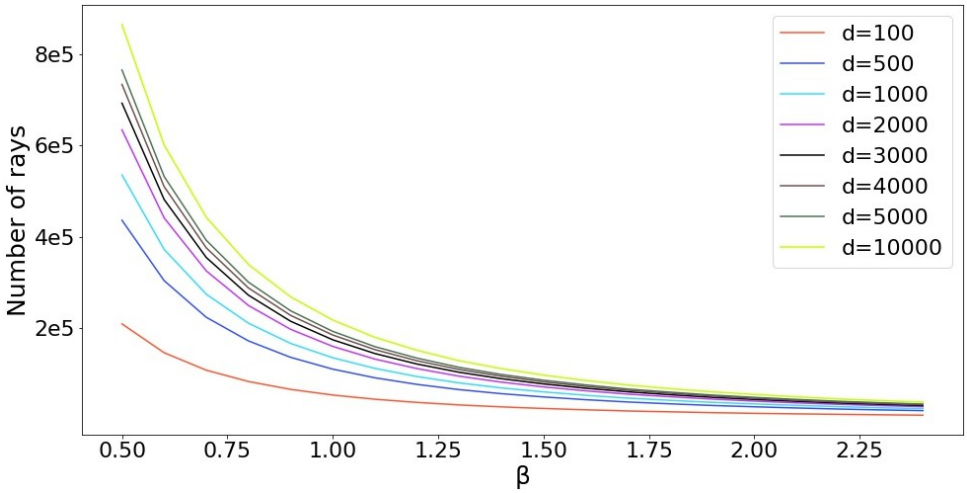


Figure 3.16: Number of rays launched in different scenarios

3.5 Sensitivity Analysis

After explaining the correctness, robustness, and the cost of our site-specific ray generation technique, we dig deeper in this section into the gain it offers through a sensitivity analysis of the results versus the number of rays. As a site-specific method, we aim at optimizing the number of rays launched by reducing the number of rays wasted which allows us to reduce the computational load of RT. In traditional RT, for covering an area of 100 meters radius, one need to launch as much rays as in the case of 5000 meters radius (i.e., with no consideration of the receiving area).

However as shown in Figure 3.16, our technique optimizes the number of rays launched by taking into account the maximum coverage distance of the area of interest and the height of the antenna. The x-axis of this Figure represents β, the constant azimuthal angular separation and the y-axis determines the number of rays launched. The Figure gives for different coverage area distance values d, the number of rays that

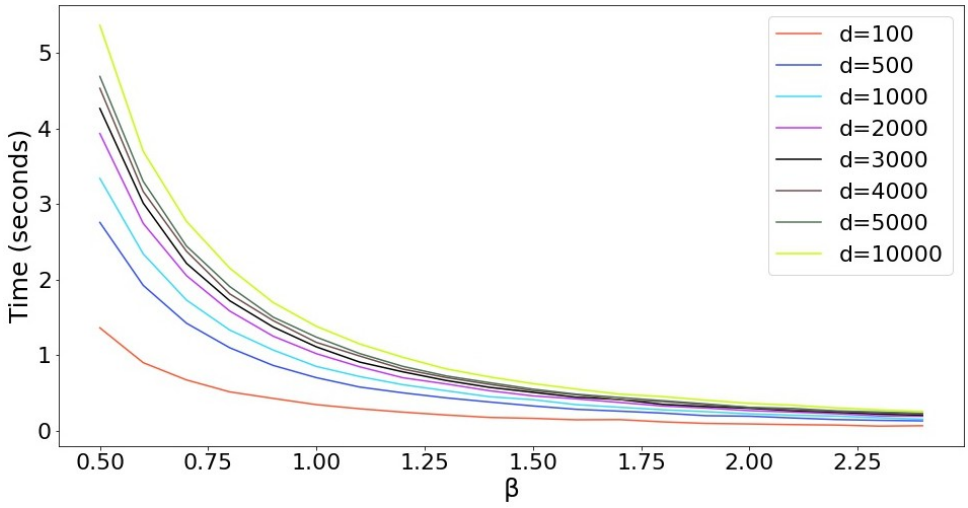


Figure 3.17: Time to generate rays in different scenarios

are launched as a function of β.

The first intuitive observation is that the larger the azimuthal angular separation is, the less rays are launched, which is intuitive as the angular step between adjacent rays increases. Second, the Figure provides the minimum number of rays necessary to fully cover the propagation area without any gap for each value of β and d. The curves in the Figure thus help to have a sense of how many rays are effectively launched by our method, and consequently how many are saved compared to the icosahedron technique. We can observe that the gain obtained depends on the coverage distance d: smaller coverage areas need less rays than larger ones. We can noticeably see that at $\beta = 0.5°$, almost 1 million rays are necessary for 10000 meters distance, while only 200,000 rays are enough for the 100 meters case. Our method can then automatically save almost 800,000 rays to be launched when switching between these two environments, which later helps to reduce the computational load and the high memory consumption of RT.

As our method reduces the complexity of the icosahedron technique by generating less rays in an adaptive and flexible way, we make different simulations to assess the time taken by our model to generate rays. Rays' generation time includes the time to find the azimuth and elevation of each ray at departure and the time to launch the rays

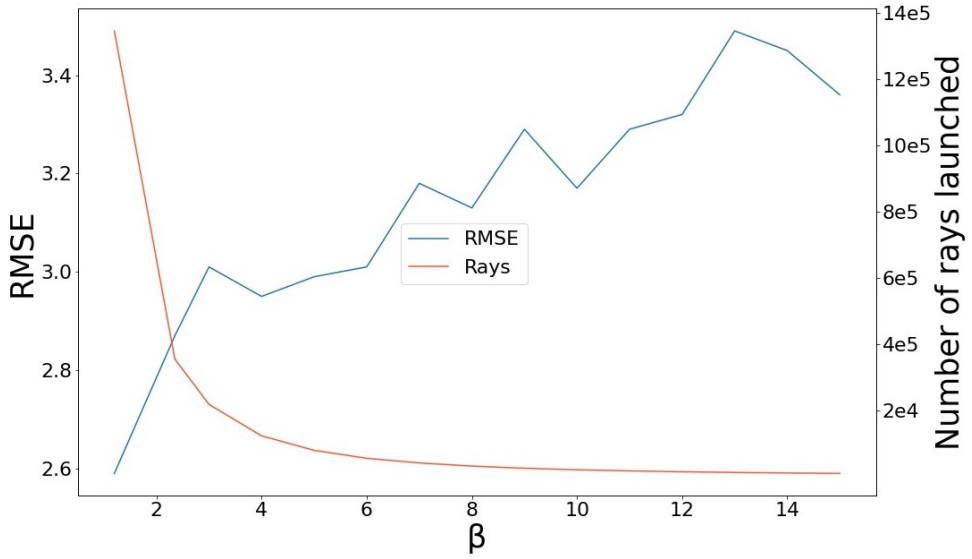


Figure 3.18: Sensitivity study on Terrain 3

given those angles as given in Algorithm 2. The average time taken by our approach to generate rays is shown in Figure 3.17. The Figure plots the generation time as a function of the azimuthal angular separation β for different coverage distances d. This helps to get an idea of what is the time required to launch a certain number of rays. Naturally we see that the smaller the coverage distance is, the less time is required. For instance, at $\beta = 0.5°$, it takes 1.5 seconds to generate rays at $d = 100$ meters, while it takes 5.5 seconds for $d = 10000$ meters. This comes from the ability of our method to minimize the number of rays necessary in each specific scenario. Moreover, we see that almost 1 million rays can be launched by our technique in almost 5 seconds. This result shows the ability of our technique to launch large number of rays in a negligible time, hence highlighting its lower complexity.

We move our performance evaluation further by assessing the sensitivity of our approach regarding the number of rays launched. Said differently, we seek to evaluate the change of accuracy of our signal power estimation compared to the icosahedron technique when rays are launched. We changed the number of rays by varying the value of the azimuthal angular separation β. We performed new simulations for different values of β by setting the values of the distance d and the height of the

$\beta(°)$	**1.19**	**2.35**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Terrain 1	2.49	2.54	2.57	2.74	2.78	2.88	3	3.04	2.98	3.1	2.99	3.19	3.21	3.18	3.18
Terrain 2	2.18	2.28	2.39	2.51	2.6	2.73	2.7	2.86	2.88	2.84	2.82	2.95	2.97	2.93	2.99
Terrain 3	2.59	2.87	3.01	2.95	2.99	3.01	3.18	3.13	3.29	3.17	3.29	3.32	3.49	3.45	3.36

Table 3.4: RMSE (Mbps) vs β for all the 3 terrains

antenna to 5000 meters and 30 meters respectively. Each of our simulation results was compared to Matlab. We compared our results to Matlab's *low* angular separation, because of its high accuracy compared to its two others defined resolutions. A range of β values were taken with 15° being the maximum, which is needed by the small angle approximation in our technique. For each value of β in this range, we compute the RMSE of the bitrate estimate with respect to the *low* resolution of Matlab and show it in Figure 3.18. The value of the number of rays launched is also given as a function of β to help having a better sense of the efficiency of our technique.

Based on this Figure, it is evident that our technique demonstrates low sensitivity to changes in the value of β, specifically regarding accuracy loss. For example, even when rays are launched at 2° and others at 15°, the RMSE only varies slightly, ranging from 2.6 to 3.4 Mbps. This property is very important since one can safely choose an azimuthal angular separation of 15°, hence launching less rays and ending up reaching almost the same level of accuracy as for other rays using a smaller value of β. Therefore, the minimum possible number of rays can be launched at a lower cost. Since less rays are launched, the computational load and the memory consumption of RT can be reduced.

Figure 3.18 shows the studies for only one terrain. We performed the simulations on two different other terrains to check the robustness of our results. Table 3.4 shows the summary of the RMSE values of the bitrate estimation obtained in each case. Despite the slight variations from one terrain to another, we can observe from this table that the difference in terms of accuracy is still very small. This confirms the robustness of our technique and its ability to launch the minimum possible number of rays while keeping the overall accuracy within acceptable range. Furthermore, our method can be easily applied in a realistic scenario where there exist multiple antennas. Since

antennas are independent, they are treated independently from each other.

3.6 Accuracy/Complexity trade-off

In the previous section we show the effect of a chosen value of β on the accuracy in order to help users choose the right number of rays to launch depending on their scenario. In this section, we dig in depth in this study by considering different types of terrains in large-scale scenarios involving multiple antennas. This is to fully help users get the most out of our site-specific technique. Our extensive study shows the pros and cons of a given number of rays in terms of both accuracy and complexity.

As the number of rays to be launched depends mostly on the value of β chosen, our study highlights on one hand the effect of β on the bitrate estimate and on the other hand, the computational complexity as a function of β and the type of the terrain. In a nutshell, we aim to emphasize on the accuracy/complexity trade-off of RT based on the number of rays launched and the type of terrain. The latter is meant to help users have full control of the technique and to tune it depending on the level of accuracy sought and the available resources.

To stress the effect of the number of rays launched on the computational complexity, we simulated our model on a more powerful server featuring 32 processors and 94GB memory. This allows us to perform our complexity/accuracy trade-off based on real-world datasets in order to show the effectiveness of our site-specific ray generation model in real life scenarios involving multiple antennas. These real-world datasets include information about the buildings and the antennas in the chosen environments. More details on these datasets are provided in Chapter 5.

For our study, we considered three different types of terrains: Urban, Suburban and Rural. For the sake of highlighting the robustness of our technique, we took two different cities for each terrain type. Table 3.5 gives the description of these six terrains and we see that the number of buildings and antennas decreases when moving from urban to rural areas.

Terrain	Terrain 1	Terrain 2	Terrain 3	Terrain 4	Terrain 5	Terrain 6
Zone	Urban	Urban	Sub-urban	Sub-urban	Rural	Rural
Buildings	5141	5406	2861	1031	755	431
Antennas	8	6	2	5	1	1

Table 3.5: Terrains description

For all these terrains, the studied area is 1km radius around a given location in the city. Within this radius, 5000 random reception points are taken, whose heights were set to 1.5 meters. The latter corresponds to the average height of a mobile phone held by a person. To this height, we added the elevation regarding sea level of the reception point. Furthermore, to ensure the stability of the model, we perform our experiments 10 times on different 5000 random reception points chosen by changing the seed used to generate them. Simply said, for each value of β we have 10 different experiments run on different sets of reception points. Also, for each experiment, the value of β is the same for all the multiple antennas in the areas considered and the interference coming from surrounding antennas is accounted for when computing the receiver's download bitrate.

For each terrain shown in Table 3.5, we performed the accuracy assessment as a function of β and also the computational complexity assessment in terms of execution time and memory consumption.

3.6.1 Accuracy

In our site-specific ray generation model, the larger the value of β is, the larger will be the angular separation of a ray cone with its neighbors and according to Equation (3.3), the larger its radius will be. In an obstacle-free environment, the space is covered by big overlapping circles when large values of β are used. Small values of β instead cover the space by small overlapping circles. However, in complex environments, the big radius of rays tends to overlap, or even fully cover some obstacles, therefore showing places as covered by rays while in reality they are not. This typically happens in areas behind obstacles. We will be referring to this phenomenon

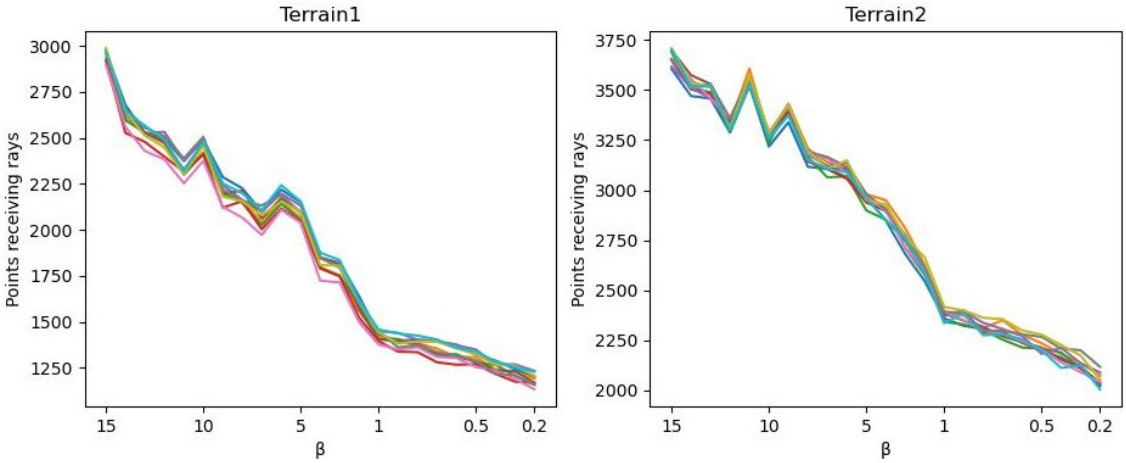


Figure 3.19: Urban areas: Colors correspond to different runs of the experiment

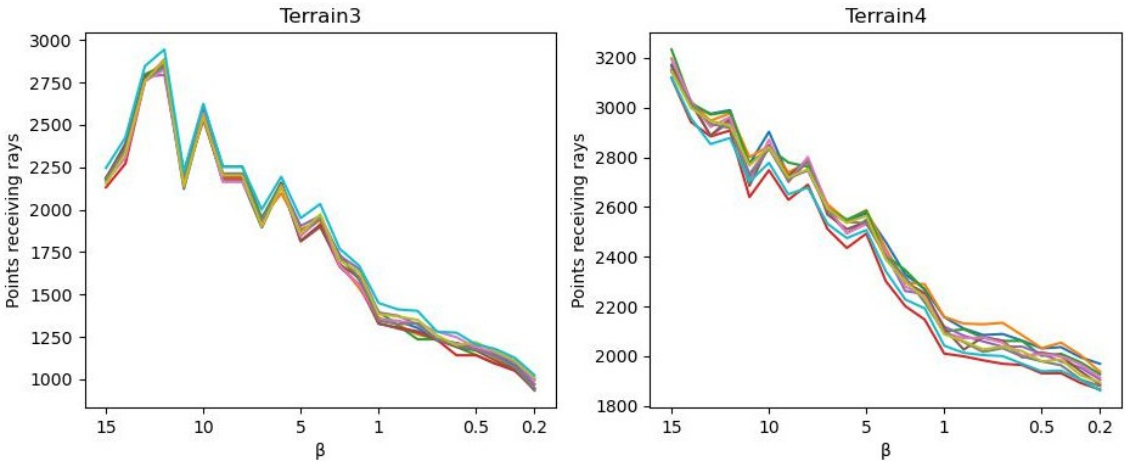


Figure 3.20: Suburban areas: Colors correspond to different runs of the experiment

as *ray overestimation*. In the case where small radius rays are launched, rays will be reflected instead by the obstacles and no overestimation should occur. In this section we assess how often this overestimation occurs and link it to the number of obstacles and the value of β used to generate rays. We further study how this phenomenon affects the accuracy of our model and give insights about the suitable values of β to use depending on the use case in order to ensure the best accuracy of the model.

First, to quantify the ray overestimation, we evaluate how many of the 5000 reception points receive rays for each value of β. Knowing that small values of β are more accurate than the large ones in terms of ray overestimation, we assess how many

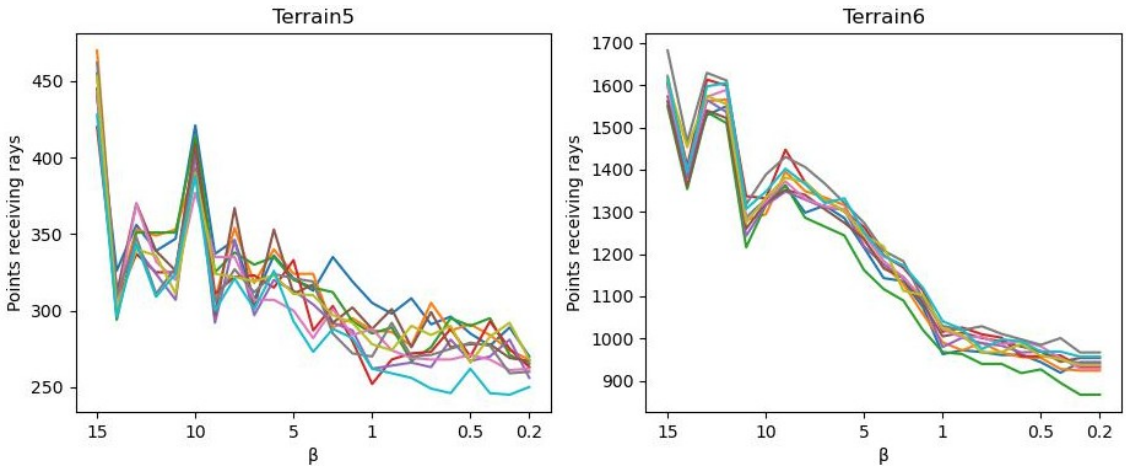


Figure 3.21: Rural areas: Colors correspond to different runs of the experiment

points receive rays for each value of β and how far are these values between large and small values of β depending on the type of the terrain.

Figures 3.19, 3.20 and 3.21 show how the number of reception points varies as a function of β. The 10 line plots in each Figure correspond to the same experiment repeated over 10 different sets of 5000 randomly selected points: Each colored line plot corresponds to a given set's result. From these figures and as expected, we see that the number of points receiving rays decreases as the value of β decreases. However, as the value of β gets close to 0, the level of overestimation starts to be negligible, and the curves tend to be less and less inclined. Nevertheless, the extent of this overestimation is more or less severe from one terrain to another. In rural areas for instance as shown in Figure 3.21, the difference of points receiving rays between large and small values of β is relatively small. The decrease is in the order of 43%. However, in suburban and urban areas, the relative decrease is more than 50% for both cases. This high overestimation in urban and suburban areas typically comes from the complexity of the environment, i.e, the high number of obstacles that are present in those areas. This is to say that in complex environments, large values of β are subject to ray overestimation phenomenon. Nevertheless, in less complex environments, the overestimation is relatively small, i.e., one can launch a small or a large number of rays with only small loss in terms of precision.

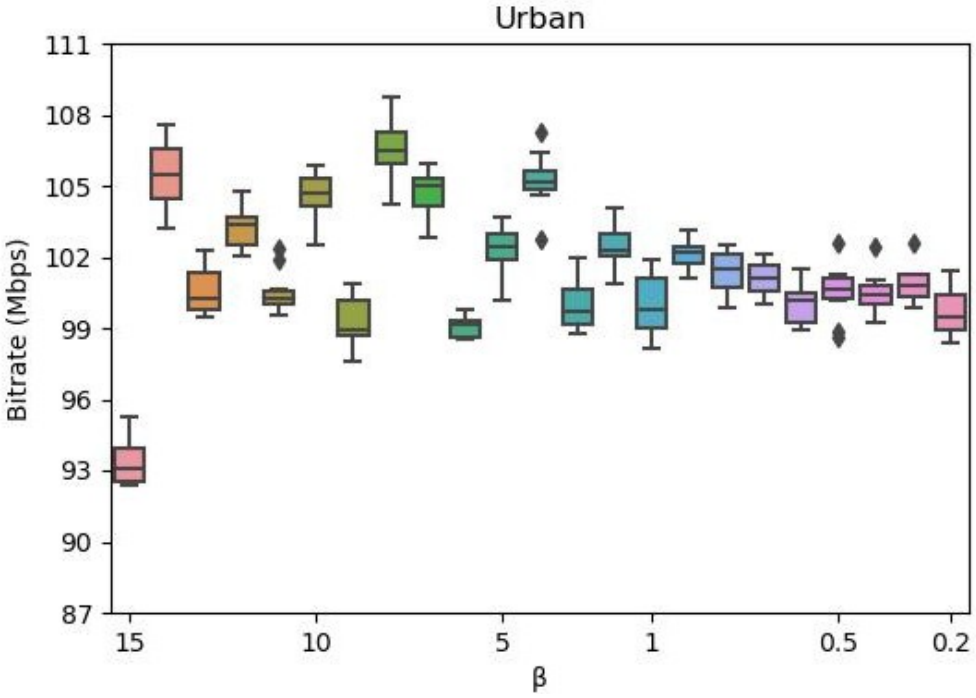


Figure 3.22: Average bitrate values in an urban area: Terrain 1

We move further our study to assess the impact of the azimuthal angular separation β on the bitrate estimation accuracy. This is meant to give a clear idea about the loss and gain in terms of bitrate depending on the type of terrain. To do this, we run the same experiment over the same sets of 5000 reception points and compute the downlink bitrate for each of these points that receive the signal power. This downlink bitrate considers interference coming from surrounding antennas. For each of the 10 experiments and for each value of β, we take the average bitrate over all the points receiving rays. As result, we obtain a matrix containing the average bitrate for each value of β and for all the 10 experiments.

Figures 3.22, 3.23 and 3.24 are box plots showing these results for Terrain 1, 4 and 6. We can observe from these figures how the average bitrate varies as a function of β. In urban areas as shown in Figure 3.22, we see high fluctuations for a wide range of β values and the bitrate starts to converge around $\beta = 1°$. The latter means that if one wants to maintain a high accurate model in urban areas, it is better to choose small values of β, mainly where the model starts to converge. The complexity related

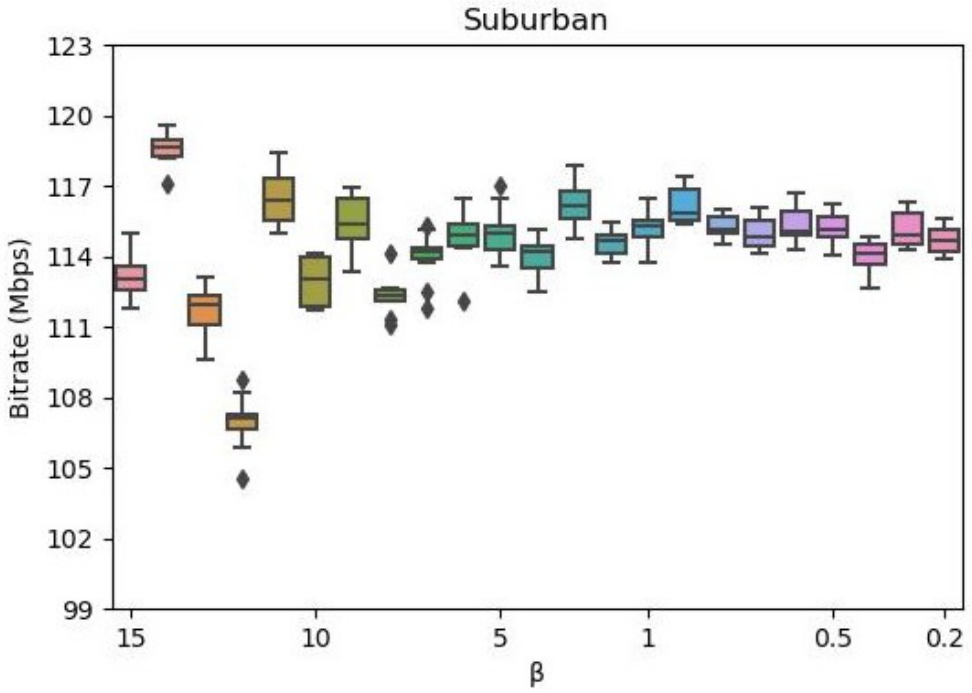


Figure 3.23: Average bitrate values in a suburban area: Terrain 4

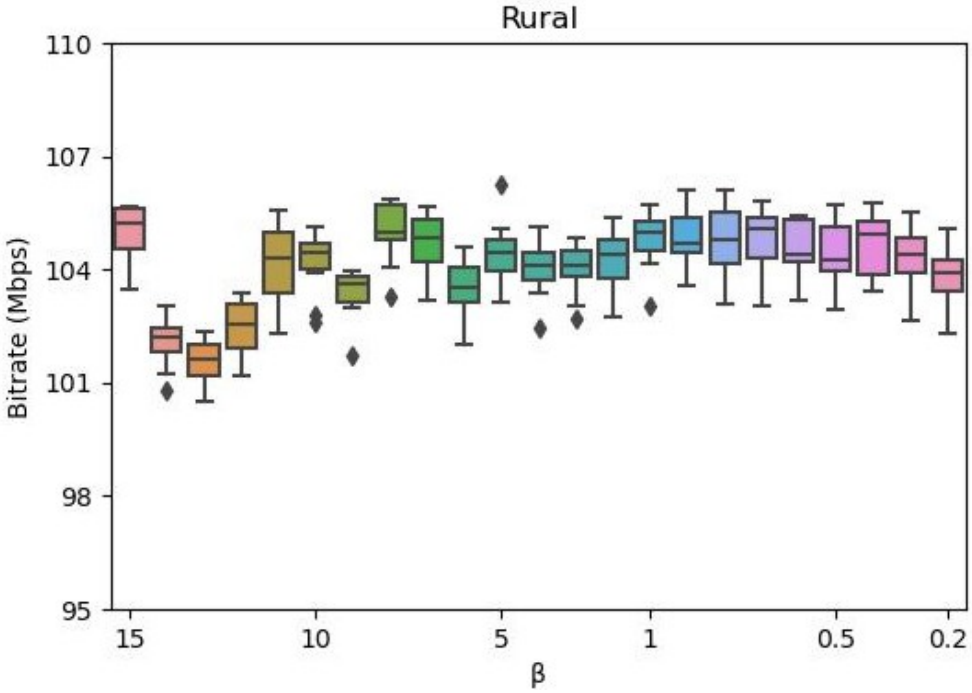


Figure 3.24: Average bitrate values in a rural area: Terrain 6

to this choice will be highlighted in section 3.6.2.

According to Figure 3.23, in suburban areas, we see that the convergence is faster than in urban areas. Indeed, the model starts to converge around $\beta = 6°$. This means that medium values of β can be used to launch rays in suburban areas while still maintaining the model's high level of accuracy.

Finally, from Figure 3.24, we see that the variation of the average value of the bitrate in rural areas is quite negligible, since the model converges around $\beta = 11°$. Moreover, the difference of bitrate between the largest value of β and where it starts to converge is around 2 Mbps. This is to say that the model remains accurate despite the value of β chosen in the range we consider. We can thus conclude that in environments with small number of obstacles, β has almost no effect on the model's estimation accuracy.

In summary, urban areas are more sensitive to the overestimation incurred by large values of β due to the large number of obstacles present in these environments. This leads the model's accuracy to also suffer from this phenomenon. Therefore, to have an accurate RT downlink bitrate estimation in urban areas, it is better to choose smaller values of β. However, suburban, and mostly rural areas are less sensitive to this phenomenon. Thus, larger values of β can be chosen, i.e., a smaller number of rays can be launched in those environments with just a little effect on the model's overall accuracy.

3.6.2 Complexity

We study in this section how the memory consumption and the execution time varies depending on the value of β and the type of terrain. We performed these simulations on a Linux server running on an Intel(R) Xeon(R) CPU @ 2.6 GHz with 32 processors and 94GB memory.

Figures 3.25, 3.26 and 3.27 show the memory consumption and the execution time needed to compute the bitrate for all the reception points as a function of β in urban, suburban, and rural areas respectively. From these figures we can see that the

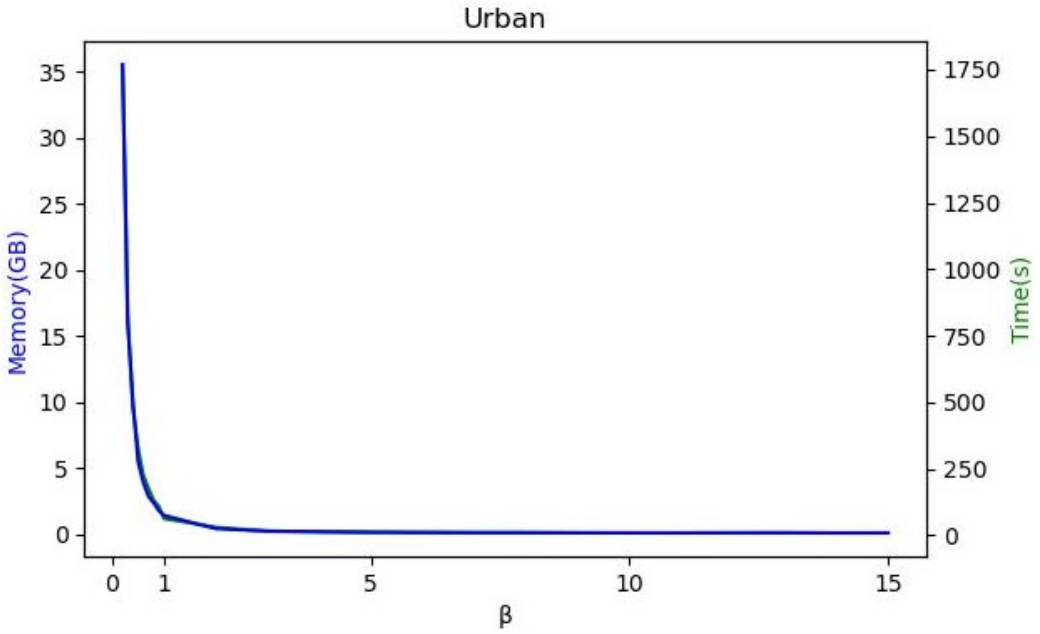


Figure 3.25: Complexity in an urban area: Terrain 1

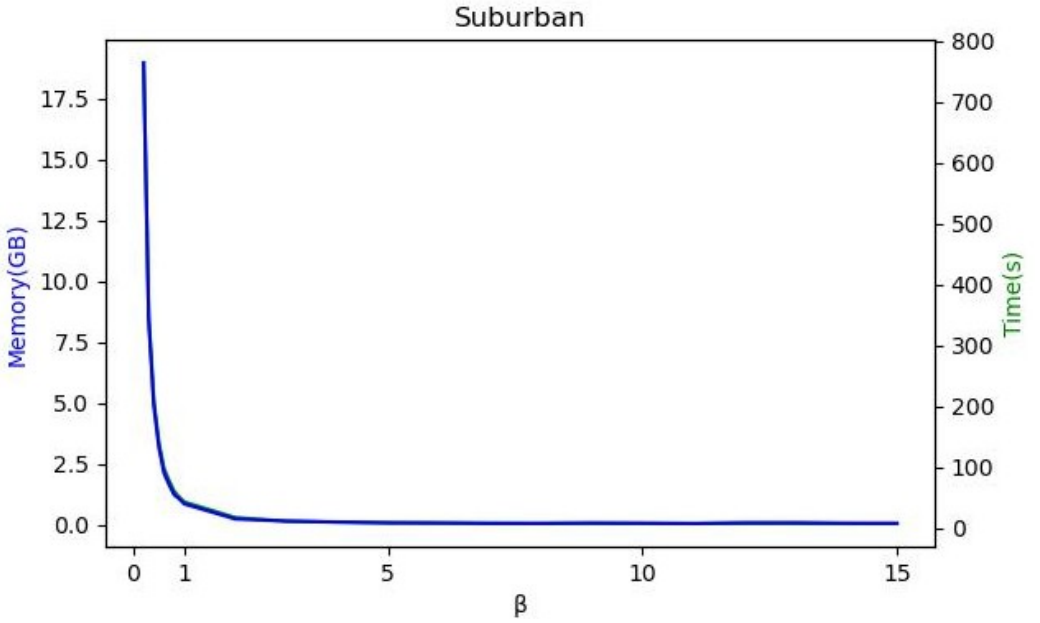


Figure 3.26: Complexity in a suburban area: Terrain 4

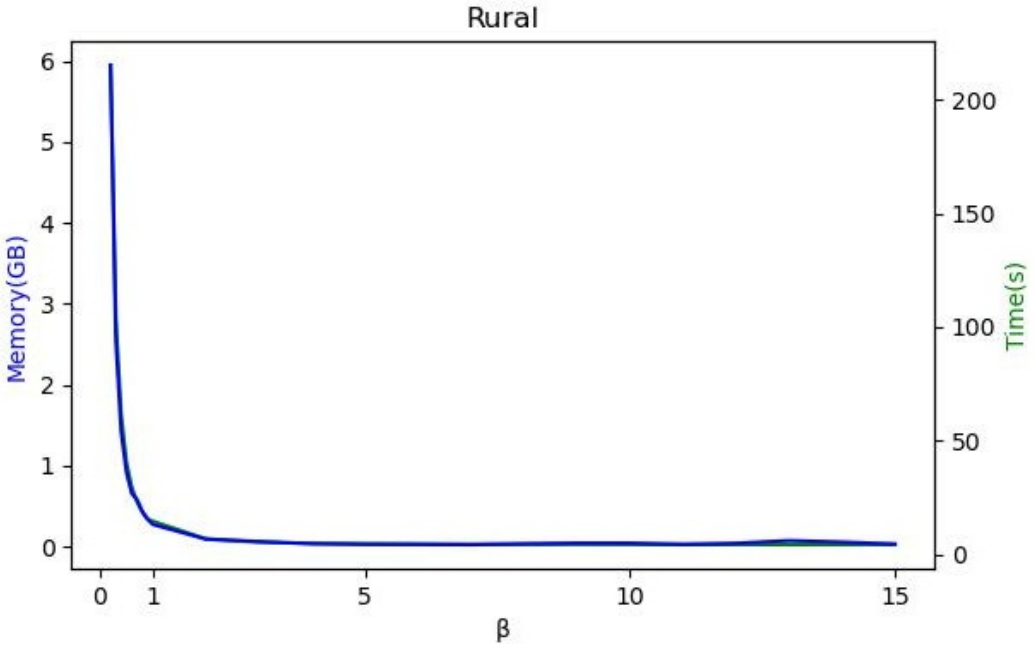


Figure 3.27: Complexity in a rural area: Terrain 6

memory consumption and the execution time have almost the same distribution and they are inversely proportional to the value of β. This behavior is normal since with large values of β, the number of launched rays is small, leading to less computational complexity, which is the opposite of what happens with smaller values of β where a large number of rays is generated. Furthermore, we can see that for $\beta > 1$, the memory consumption and the execution time for all the three terrains drop drastically as compared to the case of $\beta < 1$, due to the large number of rays launched in the latter case.

Urban areas contain a lot of antennas, this means that the number of rays generated will be large, as all the antennas are launching rays at the same time. This large number of rays leads to a high computational complexity as opposed to suburban and rural areas and as can be seen in Figure 3.25. From this Figure, we see that more than 37% of memory is consumed with an execution time around 30 minutes for the smallest value of β considered.

While in suburban areas, the smaller number of antennas and buildings leads to a

lower complexity. From Figure 3.26, the smallest value of β consumes more than 19% of memory with an execution time around 13 minutes.

Nevertheless, in rural areas, resource consumption is way less as opposed to the previous ones due to the small number of antennas present in the considered areas. According to Figure 3.27, the memory consumption with the lowest value of β is less than 6% with an execution time around 5 minutes.

3.6.3 Accuracy vs. Complexity Trade-off

Our study on both the complexity and the accuracy of our site-specific solution helps to choose the right number of rays to generate depending on the level of accuracy sought and the number of resources available. This allows us to tackle the accuracy/complexity trade-off that is crucial in RT. In practice, a user with unlimited resources willing to run RT with high accuracy can go with small values of β regardless of the terrain. Another one with limited resources willing to have an acceptable accuracy can go with any values of β in rural areas, and medium values of β in suburban and urban areas. For suburban areas this has no impact on the accuracy, however for urban areas, one must accept to lose a bit of accuracy to adapt to the constraint on computing resources.

3.7 Summary

In this Chapter we presented the ray generation process of SBR. The icosahedron technique, which is the most used one in the literature has several drawbacks as it launches rays in a brute fashion without any acquaintance of the propagation environment. Its other alternatives and/or optimizations also launch rays in all possible directions leading to a big overhead in SBR. To solve this issue, we introduced in this Chapter, a site-specific ray generation technique that launches rays regarding the information about the environment. Due to this, our technique generates the

minimum number of rays necessary to fully cover the area of interest without any gaps. Although less rays are launched compared to the state-of-the-art approach, we prove that our technique still maintains the accuracy of RT. Since we launch less rays, we reduce the high computational load and the high memory consumption of RT. Moreover, we solve the computational slowness of the state-of-the-art technique by the ability of our new technique to generate thousands of rays in few seconds. Furthermore, we first introduce a practical trade-off between accuracy and complexity, allowing users to select the appropriate number of rays based on the terrain type and available resources.

Chapter 4

2D RF Map Generation

4.1 Introduction

Ray Tracing as a deterministic propagation modeling approach accurately estimates the signal power received in a given location due to its ability to consider fine-grained information about the environment of interest. Due to this high level of accuracy obtained with RT, it is considered to be an appropriate tool for accurate RF maps generation in mobile wireless networks. Indeed, generating such maps is gaining a lot of momentum nowadays [1, 12], since they help end users to better understand the QoS offered by their MNOs depending on their living place. They are even useful to MNOs themselves, since with these maps they can troubleshoot their own network and improve their QoS when needed. Telecommunication regulatory authorities can also use these RF maps to check if MNOs' specifications are met.

Nevertheless, generating such accurate maps is quite challenging. Several factors like hardware limitations, the size of the area of interest and the huge number of antennas and buildings present in the area make it often impossible to generate those maps using RT. This impossibility comes from the fact that RT was not primarily designed for RF mapping purposes. The processes within RT are solely designed to compute the pathloss between a given transmitter Tx and a given receiver Rx.

Although RT was not primarily designed for this purpose, it is still used to generate these maps. The intuitive way that is used in the literature to generate RF maps using RT is the discretization of the area of interest using a limited number of reception points as outlined in previous studies [26–28]. These reception points are distributed throughout the area according to some statistical distribution to fully cover the propagation environment. The traditional RT algorithm is then performed for each pair of transmitting antenna and reception point and the power corresponding to each reception point is used to generate the RF map.

To obtain a precise map using this approach, a large number of receptions points must be distributed in the propagation environment to ensure that the distance between these points is as small as possible. This large discretization leads to thousands and even millions of reception points generated even in small scale areas. In order to compute the signal power received by these reception points and to produce the RF map, a reception test is performed between them and the rays to determine which rays they received. This reception test process is performed for each pair of ray and reception point to check whether the ray is received or not by the reception point. In areas where this high precision is sought, this reception test process consists of a large matrix computation that needs to be performed in order to determine the reception of rays by the reception points.

Furthermore, in large-scale scenarios, namely at city-scale, millions of rays are launched and thousands or even millions of reception points are considered depending on the level of precision sought. Along with the large number of reflections that may be needed in some scenarios, the reception test process leads to high computational load, high execution time and often memory limit exceeding errors, thus preventing the generation of the RF map. For instance, after 16 hours of execution time on an Intel Core i7 machine with 16GB memory, Matlab's RT implementation crashed due to a memory limit exceeding error and was unable to generate the map in an $0.78km^2$ area. Hence showing the inability of the discretization approach to produce highly accurate and precise RF map even at this small scale.

On the other hand, since RT was not designed for generating RF maps, all the acceleration techniques proposed in the literature only tackle the complexities related to the point-to-point estimation of the received signal power. Indeed, most of them provide solutions to the complexity of the ROI test which when naively used consists of testing between each building and each ray whether there is an intersection or not. As shown earlier in Chapter 2, these techniques work pretty well and are used for all current RT implementations. Nevertheless, since the reception test is relatively less complex in point-to-point scenarios, none of these techniques tackle the overhead incurred by this technique when it is used for RF mapping. Therefore, as the size of the environment and the number of reception points get large, even the existing acceleration techniques become inefficient in the reduction of RT complexity due to the bottleneck created by the reception test process.

Yet, in RT and more specifically in SBR, this reception test process is mandatory to determine the rays received by the reception points. Indeed, in the SBR pipeline, multiple rays are launched from transmitting antennas towards the receiving ones and go through different mechanisms before their reception. Since the exact path is not traced between the reception points and the transmitting antennas, the reception test needs to be performed between all the rays and all the reception points. After a given number of reflections and/or diffraction, no prior information exists telling whether rays are in the geographic regions or not. Instead, only the coordinates of the rays and of the reception points are known. Hence leading to this naive testing between all the rays and all the reception points to be sure that no rays and no reception points are being missed.

As explained in Section 2.4.3, this reception test consists of checking whether the reception point is inside the ray cone or not. For this, Equation (2.46) is used to compute the position of the orthogonal projection P of the reception point P on the ray. Afterwards, the condition $\|\mathbf{HP}\| \leq R_i$ is used to determine whether the ray is being received or not, where, R_i is the radius of the ray at point H computed using Equation (3.2) and $\|\mathbf{HP}\| \leq R_i$ is the minimum distance between the reception point

and the ray as illustrated in Figure 2.22.

When the discretization approach is used for RF mapping purposes, Equation (2.46) can be generalized for all the rays and all the reception points. Hence in general, **AH** and R_i are $N \times M \times K$ tensors, $\mathbf{u}$ is a $N \times K$ matrix, with N, M and K being respectively the number of rays, the number of reception points and the maximum number of reflections. AH_{ijk} is the distance between the origin of the k^{th} reflection of the i^{th} ray and the orthogonal projection of the j^{th} reception point on the i^{th} ray. Similarly, H_{ijk} is the distance between the j^{th} reception point and its projection on the k^{th} reflection of the i^{th} ray. The rays received by the j^{th} reception point are those satisfying the condition: $\|HP_{ijk}\| \leq R_{ijk}$, where R_{ijk} is the radius of the i^{th} ray regarding the j^{th} reception point at its k^{th} reflection.

The matrix computation in Equation (2.46) has a complexity linked to the number of reception points M, the number of rays launched N and the number of reflections K. To obtain a precise RF map, a large value of M must be chosen in order to reduce the distance between the reference reception points. The value of M also grows with the size of the terrain; the larger the terrain size is, the larger the value of M must be in order to ensure comprehensive coverage and hence a precise map.

On the other hand, the accuracy of the signal power estimation is tightly linked to the number of rays launched as explained in Section 3.2. Hence, to obtain an accurate RF map, one must launch a large number of rays from each antenna. Since large scale scenarios and more specifically urban areas are composed of a large number of antennas, the value of N must be extremely large in order to ensure accurate estimations of the signal power received by the reference reception points.

Finally, urban and suburban environments are composed of a large number of buildings that reflect, transmit and diffract the signal. Since the street size in those environments is very small, signal usually goes through multiple reflections, transmissions, and diffractions before it completely fades. Therefore, to ensure that the signal power estimation reflects reality, the value of K must also be large enough to account for

all these mechanisms.

All these large values of the parameters necessary to perform the reception test and to obtain an accurate and precise estimation of the received signal power leads to a high computational load and a large memory computation. Due to this load, the execution time of RT explodes despite the acceleration technique used in the previous RT processes.

Furthermore, due to the large space occupied by these parameters, the memory limit is often exceeded in complex scenarios, thus preventing to perform the reception test and consequently to generate the RF map. To avoid such issues, Matlab for instance in their RT implementation limits the coverage area around an antenna to 500 meters [65]. This is meant to limit the complexity of the RF map generation when a large number of reception points is chosen. This helps in some scenarios to limit the memory consumption and hence to be able to produce the RF map. Nevertheless, this is not always the case, since some scenarios contain a large number of antennas and taking this limited radius around each antenna does not reduce the overhead due to the number of reception points and to the number of rays launched.

Instead, the approach adopted by Matlab for their RF map generation even creates a performance issue, since the discretization in their approach is antenna-related, i.e., for each antenna, a discretization is performed in its radius regardless of the other antennas. For antennas located near to each other, this leads to redundancies in the reception points, i.e., the same reception point may be considered multiple times. This issue gets severe when there is a large number of reception points leading to a large number of useless reception points for the RF map generation. And yet, the reception test process is performed for all these points that have no effect on the produced RF map, therefore incurring a large useless overhead that may prevent the RF map generation.

Additionally, limiting the radius size of an antenna to 500 meters does not reflect the real-life behavior of antennas. Indeed, depending on the channel frequency, some

antennas can cover more than 10 kilometers around them and within this radius, hundreds of other antennas may be present. This limitation prevents seeing the effect of these antennas on distances that are more than 500 meters, while in reality this part may represent more than 99% of the overall coverage area of an antenna in the extreme case. Therefore, using such an approach to generate RF maps in areas where the antenna coverage is large, namely in suburban and rural areas, leads to inaccurate estimation of antennas' coverage in those areas.

Furthermore, with this approach, potential sources of interference are also ignored from surrounding antennas. Indeed, the effect of an antenna interfering with other antennas located more than 500 meters away is not accounted for. This leads to the generation of unrealistic RF maps showing overestimated received signal power. The larger the number of antennas is in the propagation environment, the higher is the probability of having sources of interference and the worse will be the results from Matlab's approach to resolve the complexity of the reception test process. This clearly shows the inability of the state-of-the-art raw RT implementation to be used in real-life scenarios to generate precise and accurate RF maps.

To overcome the inability of the current RT implementation to produce accurate and precise RF maps even in small scale scenarios, we propose in this Chapter an optimized RT solution able to generate accurate and precise RF maps in a reasonable time. By combining the efficiency of our site-specific ray generation technique presented in Chapter 3 and by introducing an alternative to the reception test process, we manage to divide by almost 1200 the execution time of RT with less than 2% of memory usage as compared to baseline solutions. Indeed, where current models as the one of Matlab take 16 hours of execution time and often fail due to memory limitations, our solution produces the RF map within only 50 seconds.

Instead of discretizing the propagation environment as in current RT implementation, our idea is to merely capture the footprint of rays on a given plane rather than performing the reception tests on a set of reception points. We simply perform the intersection of ray cones with a fixed horizontal plane instead of performing the

reception test on all the reception points of the map with different heights. We afterwards use a simple condition to perform the ray-plane intersection test. This helps to drastically reduce the execution time of RT and lower its memory consumption due to the removal of all the matrix operations performed in the reception test process. Moreover, with our approach, it is no longer necessary to perform the discretization of the propagation environment; rather, the reception points are considered to be all located on the same fixed horizontal plane. Thus, our method is always precise because all the reception points in the area of interest are considered and none of them is missed. Furthermore, our approach has the advantage of fully capturing all the rays without missing none of them. Indeed, with the discretization approach, depending on the distance between the reference reception points, there may be lot of rays that are missed simply because they do not pass near these points. This scenario leads to an inaccurate estimation of the signal power in those areas between reception points. Usually, an interpolation is further performed in current techniques to estimate the signal power in areas where there are no reception points based on the signal power received by their neighboring reception points. On the contrary, since we capture the footprint of rays on a given plane, we are able to remove this inaccuracy and perform an exact estimation everywhere in the area of interest. Furthermore, we remove the need of performing the aforementioned reception test process which in complex scenarios prevents the RF map generation.

Nevertheless, it is important to note that since we only capture the footprint of rays on a given plane, our alternative to the reception test only works in areas with only small variation of altitudes in the propagation environment. If the terrain contains high variance in its altitudes, our method may lead to inaccurate results. We solve this issue in Chapter 5 by proposing a RF mapping solution applicable to all types of terrains.

The rest of this current Chapter is organized as follows. Our system overview is shown in Section 4.2 as well as the condition of its applicability in a given terrain. We present the validation and the performance evaluation of our solution in Section 4.3.

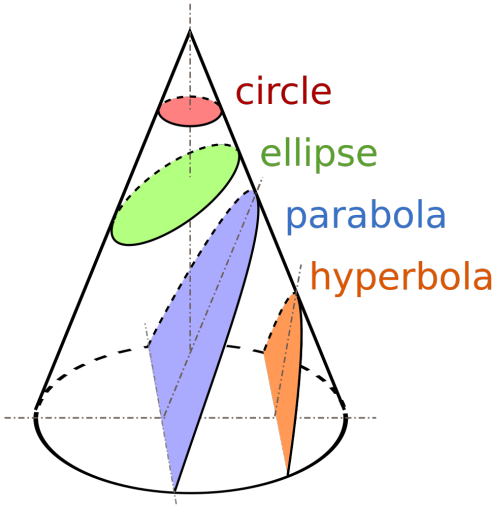


Figure 4.1: Results of the intersection of a cone with a plane. This Figure is taken from [6].

Conclusions to our work are discussed in Section 4.4.

4.2 System overview

As explained earlier, the reception test is a cumbersome process to RT. Due to the fact that, even when smaller number of rays are launched, the large number of reception points still leads to an overhead with high memory consumption and complex matrix operations. To avoid this bottleneck, we consider instead the reception points to be all located on a given horizontal plane Z_{plane}. Thus, all the reception points have the coordinates (X_i, Y_i, Z_{plane}). Fixing the plane has the advantage of avoiding the reception test between all the rays and all the reception points. Rather, an intersection test is performed between ray cones' cross-sections and the horizontal plane of reception to calculate the contribution of the different rays to the RF map. Since rays are cones, this intersection results in a conic section which is the result of the intersection between a cone and a plane. As shown in Figure 4.1, the obtained section from this intersection is either a circle, an ellipse, a parabola or an hyperbola depending on the angle of the incident ray and whether or not the plane is finite. In our approach, given that the plane is both horizontal and finite, the intersections formed when ray

cones meet the plane take the shape of either circles or ellipses. Specifically, circles are formed when the incident ray is perpendicular to the plane, whereas ellipses are formed otherwise. For the sake of simplicity, we consider the intersection with the plane to be circles since rays' cross sections are circular themselves with a well-known radius given by Equation (3.3). The consideration of ellipses would add considerable computation overhead to our approach. However, as we will see in Section 4.3, the loss in terms of accuracy due to this approximation remains negligible regarding the gain in terms of execution time and memory consumption.

Having this consideration in mind, whenever a ray intersects with the fixed horizontal plane, its circular cross-section during the intersection is taken as the footprint of the ray on the plane. As shown in Figure 4.2, a ray intersects with the plane if condition (4.1) is met. The latter consists in checking whether the distance traveled by the ray (the left part of the condition) is higher than the distance it should travel before it hits the plane (the right part of the condition).

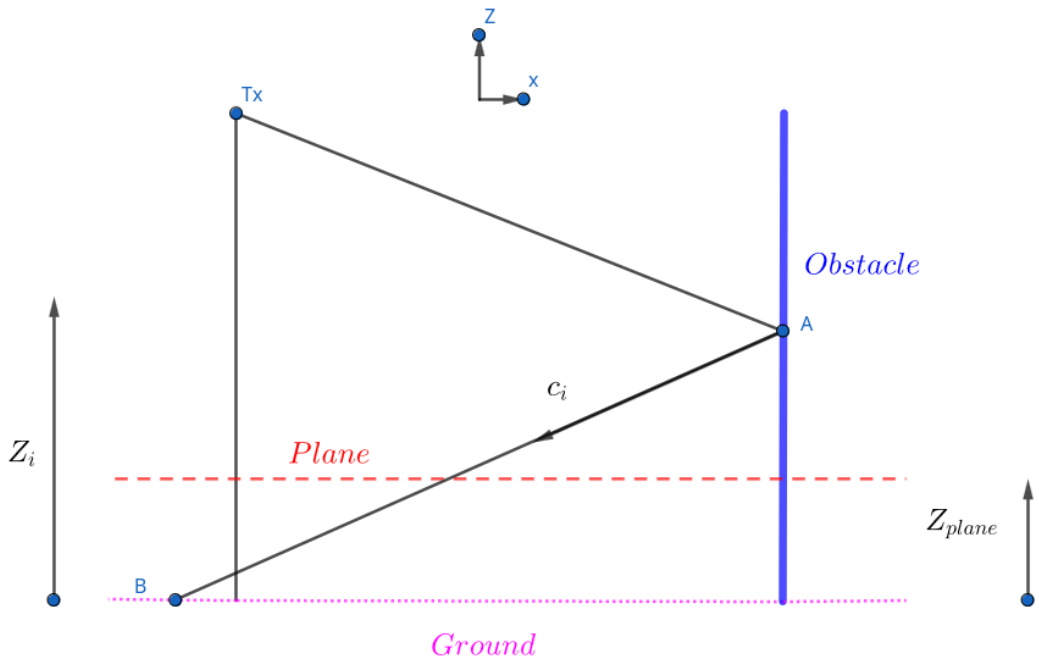


Figure 4.2: Ray-Plane intersection

$$d_i \geq \frac{Z_{plane} - Z_i}{c_i} \tag{4.1}$$

In this equation:

- $Z_i = Z_A$ is the z-coordinate in a 3D Euclidean system of the ray previous reflection point: A in the Figure. This typically corresponds to the altitude regarding the ground of the ray on the moment it hits the last obstacle before intersecting the obstacle. In case of a direct LOS, Z_i corresponds to the height from the ground of the antenna.
- c_i is the z-component of the ray's unit direction vector $\mathbf{u}$ when it moves from its previous endpoint towards the plane.
- d_i is the distance traveled by the ray between its previous end point, A and its endpoint when it hits the ground B.
- Z_{plane} is the altitude of the plane. It is the altitude at which the reception points are supposed to be on.

One important observation regarding this condition is its independence from the reception points. This highlights the fact that our method removes any need for discretizing the environment of propagation. We are therefore able to always produce a precise RF map without the need to have the reception points to be as close as possible to the traditional approach. Furthermore, there is no need to interpolate the received signal strength between the reception points to find the bitrate for unknown points. Instead, we account for every single ray that hits the plane and the footprints of these rays serve to produce the RF map. Despite the zoom level, our RF map provides the same level of high precision since it contains the real rays' footprints. The latter ensures that every single reception point in the area is considered during the generation of the RF map, which leads to a constantly high precision regardless of the targeted resolution.

On the other hand, since this condition does not depend on the discretization

approach, its complexity is quite less than the traditional reception test approach. Indeed, in scenarios where multiple rays are launched from multiple antennas, our approach exhibits only small computation. When multiple rays are present, the distance d_i in condition (4.1) becomes a vector d of size $N \times 1$, where N is the number of rays launched and $d[i]$ is the equivalent of d_i in Equation (4.1). This vector is not computed for the purpose of testing this condition, rather whenever the ray propagates from one point to another, its distance is directly computed despite the approach used. This is in fact the same behavior as the second part of the condition where the parameters Z_i and c_i are also available information that are used in RT to track the propagation of rays and Z_{plane} is simply a defined scalar. All the parameters of (4.1) being available and by simply leveraging this available information, our condition does not incur any extra memory consumption and any extra computation to compute the values of its parameters as opposed to the traditional reception test approach. In addition to this, no complex matrix operations is performed in this condition as opposed to Equation (2.46).

Note that the condition (4.1) works whether the ray is being sent from above or from below the chosen plane. Indeed, when the ray hits the plane from above, $Z_{plane} - Z_i < 0$ and $c_i < 0$, hence having a positive value. On the other hand, when the plane is hit from below, $Z_{plane} - Z_i > 0$ and $c_i > 0$ also leading to a positive value. This is particularly useful since in all scenarios rays hit the ground and are reflected back in the environment. Hence, we are able to account for the effects of the reflections due to the ground and also in scenarios where the height of the plane is higher than some antennas' height, our technique is still capable of capturing the footprints of the rays coming from these antennas.

To further reduce the computational load using our method, one may discard all the rays that intersect the plane from below. In these cases, the unit vector in the z-direction of the ray, c_i, is positive, i.e., the ray will always continue to go up. Also, since almost all the obstacles in the propagation environment are vertical (apart from the ground), there is no factor that changes the direction of the ray to bring it back

towards the plane. Thus, one may discard those rays that have already hit the plane and that have a positive c_i, which may also help to reduce again the computational complexity of the intersection test.

Finally, when all the rays are tested, our approach takes as output the information about the rays that intersect with the plane. Since, the cross-sections of the rays are considered and that these cross-sections are typically circles, these information include the center of the circles, namely the intersection point between the rays and the plane, and the radii of the rays determined by Equation (3.3).

Having these information in hand for all the rays, the next step is the computation of the power of the signal carried by those rays at the moment when they hit the plane. This power is computed using Equation (2.47) and the corresponding QoS metric, namely the SINR or the bitrate, can be computed based on the power of the signal. Once this metric at hand, the RF map showing the extent of each individual ray is generated. Each ray is located at its intersection point with the plane and the RF map shows its cross-section. Additionally, each of these rays' powers is shown in a heat map fashion, where each ray is colored by its corresponding metric's value. This resulting map is meant to show the signal coverage and the QoS of antennas in every single location of the propagation environment.

Figure 4.3 is an example of this type of map generated using our approach. It shows the continuous footprints of rays for a single antenna in the city of Nice in France for a plane height of 1.5 meters. The city center of Nice taken in this example is typically a flat zone and its coverage can be approximated by a single plane elevation. This illustration of this SINR map shows the quality of the signal coverage of this specific antenna. Due to the effects of the buildings and of the reflections, it shows rays overlapping upon each other and areas lacking coverage because this city corresponds to a dense area composed of a large number of buildings that attenuate the signal. This map is a precise estimation of the antenna's coverage, since it corresponds to reality. Indeed, when this map shows that an area is not covered by the antenna, this corresponds to reality, because the whole footprint of the ray is considered without

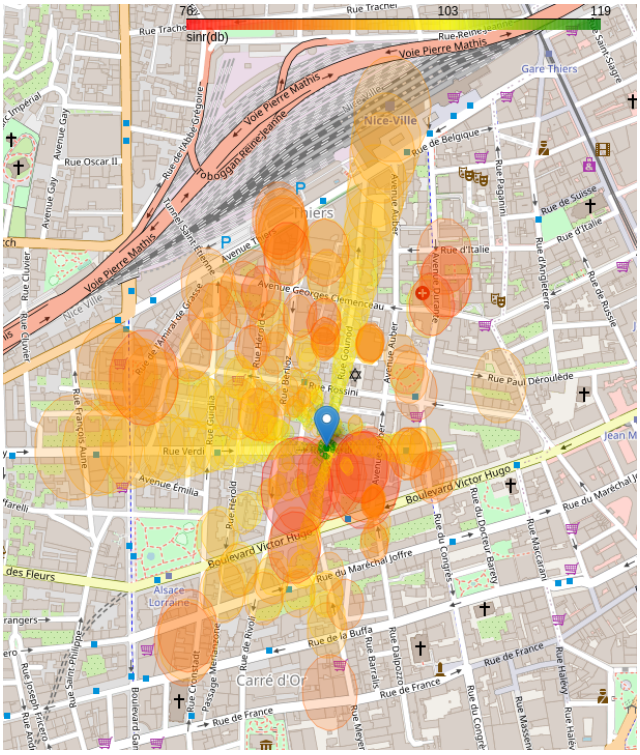


Figure 4.3: Continuous RF map of the city of Nice in France with 3 reflections

any interpolation.

When considering the fixed horizontal plane to capture the footprints of the rays and produce the RF map based on these footprints, we made the assumption that the receivers have the same height determined by the height of the plane. In our simulations we chose a plane height of 1.5 meters since this corresponds to the typical height of a person holding its mobile phone. However, this parameter can be adapted depending on the needs. It can for instance be the average altitude of the terrain or the median of the terrain's altitude variation. In the case where the variation in height is around the chosen value of the plane, our method will work perfectly due to the negligible error due to the difference in distances between the real distance towards the reception point and the estimated one towards the plane. In the case where there are large variations in the terrain, this will lead to errors in our model and specifically in scenarios where the distances traveled by rays are very large. Nevertheless, in Chapter 5 we proposed a generalization to our RF mapping solution able to account

for every single variation in the terrain.

4.3 Performance evaluation

After presenting our alternative solution to the reception test approach used in the literature, here we present both its validation and its performance evaluation. To validate our approach, we implemented a raw RT solution that uses the discretization approach and that performs the traditional reception test approach. This is meant to assess the loss in terms of accuracy that is generated by our model due notably to the approximation we made on the circular cross-section of the rays, instead of taking them to be ellipses. To do this, in the implemented raw RT, the reception points were chosen as having the same heights determined by the height of the plane in our new approach. Since our model is free from any discretization unlike the traditional RT, we made it possible, for this validation purposes, to use our model in such a way it provides the SINR for the individual receivers used in our discrete reference RT implementation. Concretely, this validation consists in checking for each reception point in the traditional RT implementation, in which area it lies is in the RF map, and consequently the rays' cross-sections it falls in. Therefore, we can obtain the power associated to the reception point in both the traditional RT implementation and our new approach and we compare the two values to assess what is the overall shift between the two models. Afterwards, we assess the computational efficiency of our new method regarding an existing state of the art solution, namely Matlab's implementation of SBR. To do this, we perform different simulations on different terrains with different settings, to assess the versatility of our model.

4.3.1 Bitrate validation

To validate our model, we performed different simulations in diverse scenarios in the city of Nice in France. This city was chosen for the simulation because the areas we considered exhibit a minimal terrain's altitude variation. Three different terrains in

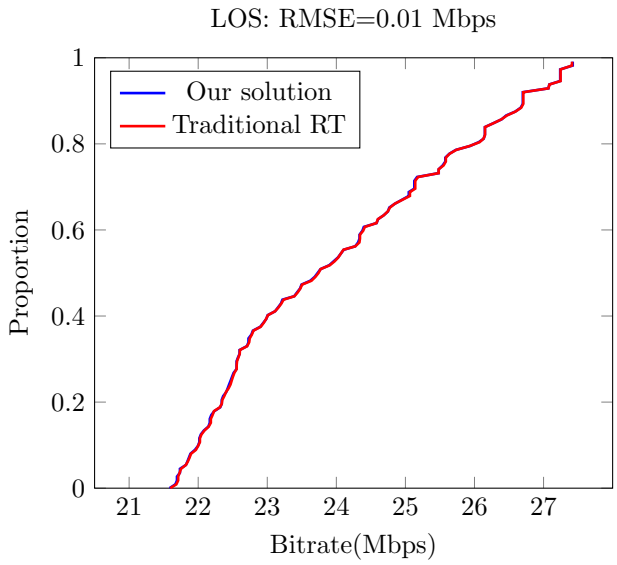


Figure 4.4: CDFs of our approach and a traditional RT: LOS

this city were considered and for each terrain the simulation was done in the case of LOS and when there are 2 and 4 reflections. This diversity is meant to show the robustness of our approach and its behavior in different conditions.

In Figure 4.4, we see that in the LOS case, our model has almost the same distribution of the bitrate as our ground truth model with an RMSE of 0.01 Mbps. This negligible RMSE means that our model has exactly the same accuracy as the traditional RT in the LOS case. In LOS the distances traveled by the rays are typically small. Since these distances are small, the error due to the estimation of the ellipse by the circle is also small. Due to this, the rays received by the reception points in the ground truth model are almost as the ones in which they lie in regarding our new solution. This helps to exhibit this similar pattern in the distribution that shows that the two models have almost the same received power and consequently the same bitrate.

On the other hand, Figure 4.5 shows the comparison of the two distributions in case of reflections, with the maximum number of possible reflections set to 2. We can see that the two models have close distributions with a slight difference in the RMSE. In NLOS scenarios, the distances are usually larger, since there exists no

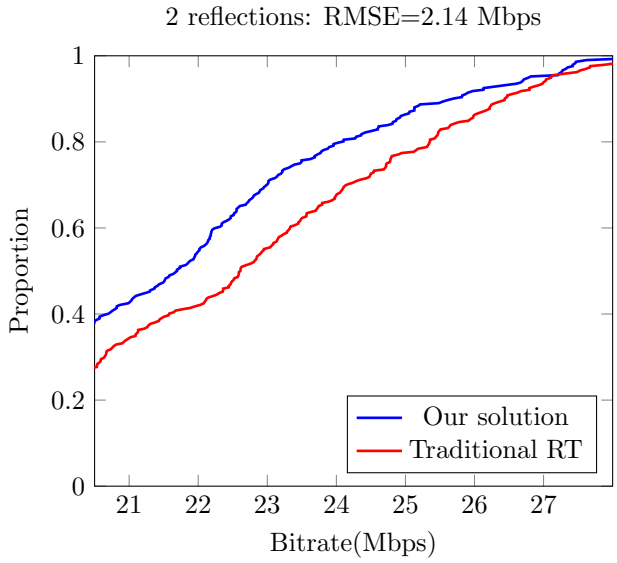


Figure 4.5: CDFs of our approach and a traditional RT: 2 reflections

direct path between these rays and the plane. Hence, these rays must go through different reflections that extend their distances before intersecting the plane. Due to these large distances, the error due to this estimation is a little bit higher than in the LOS scenario, which naturally gives rise to this slight increase in the RMSE. This difference in the distributions means that upon generating RF maps for a given terrain, our model makes an average error of about 2 Mbps for 2 reflections. However, we believe this is still an acceptable level of trade-off in terms of time savings as the following subsection will show.

Only the results for one terrain are shown in these figures, but the simulations were performed for 3 terrains in the city center of Nice in France. For these 3 terrains considered, we show the results of their simulations in Table 4.1. This table summarizes the error between our model and the ground truth for the LOS and the cases with 2 and 4 maximum reflections. The results confirm that our model is robust to the change of terrain, since the difference of RMSE in the 3 cases is small, i.e. our model keeps almost the same level of accuracy regardless of the terrain.

Apart from this validation, we perform a sensitivity study on our approach to test how its accuracy changes with the number of rays launched. Indeed, in this

	LOS	**2 reflections**	**4 reflections**
1st Terrain	0.01	2.14	2.69
2nd Terrain	0.01	2.55	3.14
3rd Terrain	0.01	2.62	3.22

Table 4.1: RMSEs (Mbps) of our model on different terrains

$\beta(^\circ)$	**0.01**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
LOS	0.01	0.01	0.02	0.04	0.05	0.05	0.07	0.08	0.09	0.1	0.12	0.13	0.12	0.16	0.18	0.19
2 reflections	2.14	2.09	2.23	2.17	2.66	2.64	2.39	2.68	2.53	2.54	2.95	2.91	2.83	2.9	2.84	2.81
4 reflections	2.69	3.15	3.26	3.21	3.25	3.68	3.63	3.7	3.67	4.04	3.73	3.82	4.36	3.8	3.84	4.31

Table 4.2: RMSE (Mbps) showing drop of performance as β increases

approach rays were launched using our site-specific ray generation technique presented in Chapter 3. This sensitivity study is meant to assess whether there is a big loss in terms of performance when only a few number of rays are launched. When few number of rays are launched, the azimuthal angular separation β is large and according to Equation (3.3), the larger will be the cross-sections of the rays. Thus, we want to assess whether this change in the radius of the rays has an impact on the overall accuracy of the model.

From Table 4.2, we can see the variation of the RMSE with respect to the azimuthal angular separation β for 3 different number of maximum reflections. We can notice from this table that smaller values of β have smaller RMSEs due to the smaller radius of the rays. Nevertheless, the difference of RMSE between the largest value of β and its lowest value is 0.18 Mbps in LOS, 0.67 Mbps with 2 reflections, and 1.62 Mbps with 4 reflections. This means that the drop of performance when going from a small to a large value of β is small. This implies that it is possible to generate an RF map with the largest value of β and get almost the same level of accuracy as the smallest ones, with the benefit of a lower execution time as will be shown in the next section. Note that the value of 15° was chosen as the maximum value of β because of the small angle approximation performed in our site-specific ray generation.

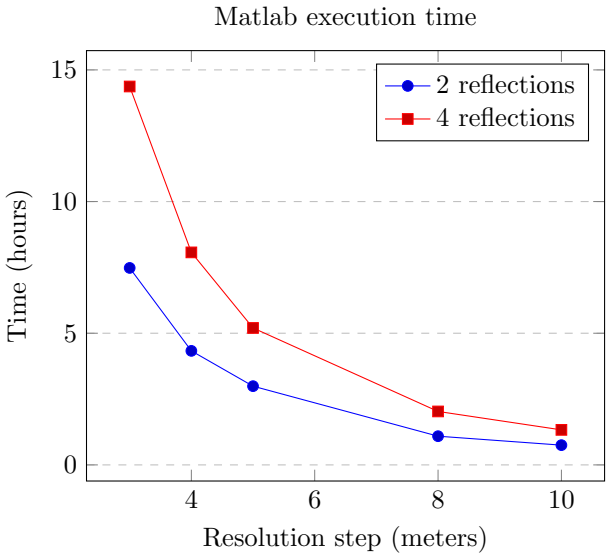


Figure 4.6: Matlab execution time

4.3.2 Execution time

For the purpose of execution time evaluation, we compare the running time of our model with Matlab's solution, which is a pure RT implementation. Matlab has two fundamental parameters while generating the RF map: the maximum angular separation (β) and the resolution step. The former must be chosen between *Low, Medium and High* corresponding to three different levels of accuracy. *Low* is the most accurate angular separation and *High* is the less accurate one, since less number of rays are launched. On the other hand, the resolution step is the maximum distance between reception points. This value of the resolution step is defined during the discretization step and the smaller it is, the larger is the number of reception points generated and the more precise the RF map is.

We performed our simulations on an HP ELITEBOOK 850 G7 laptop with 16GB of memory, and a Core i7 CPU @ 1.8 GHz. The terrain has a dimension of 1072 m $\times$ 730 m and is composed of 828 buildings in the city center of Nice in France. Based on this setup, we performed different simulations considering the accuracy and the precision of Matlab's model in LOS and with 2 and 4 reflections. On the other hand,

Resolution (meters)	**3**	**4**	**5**	**8**	**10**	**Our model**
LOS	709	422	282	129	102	5
2 reflections	12646	7310	4985	1884	1280	50
4 reflections	23198	13321	8545	3465	2234	125

Table 4.3: Mean execution time in seconds of Matlab vs Our solution

since we proved that increasing the value of β within some range does not alter the model's accuracy, we simulated our model only when β=15°, corresponding to the maximum value of the azimuthal angular separation.

Figure 4.6 gives the execution time of Matlab as a function of the resolution step in the cases where there are 2 and 4 reflections with *Low* angular separation. In the case of 4 reflections, one must wait almost 1.5 hours to obtain a less precise map and up to 15 hours to obtain a more precise map. In the case of 2 reflections, this value ranges from 45 minutes to 7.5 hours. Given the size of the terrain, this is a very high value. The trend of the graphs shows that this value will increase more and more, that is, the larger the terrain is, the longer the waiting time will be. On the opposite, our approach took 50 seconds and 2 minutes respectively for 2 and 4 reflections. This shows the ability of our model to drastically reduce the execution time of RT and to easily scale while producing the map in a reasonable time.

Table 4.3 gives a summary of the average execution time of Matlab in seconds as a function of the resolution step, compared to the time taken by our model. The values for Matlab are the averages over the results obtained for the 3 angular separations defined in Matlab: *Low, Medium and High*. This table shows that where our model takes 5 seconds in LOS to generate the RF map, Matlab's model takes on average between 102 and 709 seconds. On the other hand, our model takes respectively 50 and 125 seconds with 2 and 4 reflections while the time of Matlab is way higher. For larger environments, at country level for instance, Matlab may take days or weeks to produce the RF map making it not effective for such cases. Moreover, when taking a resolution of 2 meters, Matlab failed at producing the map in the case of reflections. This is due to the large memory consumption generated by the reception test process

and the large number of reception points. For instance, with 2 reflections and after 16 hours of execution time, Matlab crashed due to this memory issue. This shows that Matlab struggles to generate a precise map for larger environments where the number of reception points is very large, whereas our model is not very affected by memory limitations. Hence it can be applied in large scale scenarios with a lower execution time as compared to raw RT implementation as the one of Matlab.

4.4 Conclusion

In this Chapter, we introduced the generation of RF maps using RT. We demonstrated that RT processes were not designed for this purpose mainly the reception test process that accounts for most of the complexities of RT when used for RF mapping. Due to this issue in the RT design, we proved the inability of the current RT based models to generate accurate and precise RF maps at large scale in a reasonable time. To overcome this, we presented our alternative solution to the reception test process, and we demonstrated that our model could find a good trade-off between the computational complexity of RT and its accuracy, due to its ability to generate accurate and always precise RF maps within a reasonable time. Nevertheless, our method is designed only for areas with only a slight terrain variation and is not suitable for terrain with high variations. In Chapter 5 we present the generalization of this approach applicable to all types of terrains.

Chapter 5

3D Large-Scale RF Map Generation

5.1 Introduction

Ray Tracing as an accurate propagation model is very helpful for the generation of RF maps that give accurate estimation of the received signal power everywhere in a city and even a country. As explained in previous chapters, these maps are very useful for all mobile internet broadband stakeholders as it is quite helpful in taking informed decisions that are meant for instance to improve the quality of the signal.

Nevertheless, in order for these RF maps to be accurate and useful, they must be generated in a reasonable time. This is to be able to update them frequently in order for them to reflect the realities of the terrain. For instance, if these maps take a month to be generated, they may not follow the rhythm of a monthly antenna update as is the case in France. Hence, these maps will not be able to show the instant signal power received in different locations of these cities and countries where they are deployed. It is in fact this timely-based constraint that makes raw RT implementation not suitable for this purpose. Despite its accuracy, the RT propagation model and all its processes were not designed for this purpose. Thus, using them for the purpose of RF maps

generation leads to both computational complexity and accuracy/precision issues in some scenarios.

To avoid these issues, we proposed in Chapter 4 an efficient optimization to RT that, combined with our site-specific ray generation technique, was able to divide by up to 1200 the generation time of the RF map. Nevertheless, due to the complexities of modeling all single variations in terrains' altitudes, our optimization works only for terrains with only slight variations. Still, it is able to produce RF maps in those areas with a high precision despite the zoom level.

This limitation in our previous approach shows the need to generalize RF maps generation to all kind of terrains in order to be able to produce accurate and precise RF maps in every single location of a country or a city. Nevertheless, we figured out that only replacing the reception test process was not enough to produce highly accurate RF maps in a reasonable time. In fact, all the other previous RT processes and even their accelerations are not optimized for RF mapping purposes.

The visibility algorithm [35] for instance was developed to reduce the overhead incurred by the ROI test. It is meant to reduce the number of candidate tests between a transmitter Tx and a receiver Rx. Indeed, a naive intersection test consists of testing for each pair of ray and building's facet whether there is an intersection or not. By reducing the number of candidate tests, the visibility algorithm reduces the number of intersection tests performed, hence decreasing the complexity of RT.

Nevertheless, when it gets to RF mapping, many reception points are involved, leading this algorithm to suffer from its Tx/Rx dependency [34]. Indeed, the algorithm must be repeated $N \times M$ times, with N being the number of transmitting antennas and M the number of reception points. This results in many unnecessary computations performed without leveraging the inherent redundancy in the environment, thus having a longer execution time.

Therefore, using RT in its current implementation for generating RF maps in 3D large-scale scenarios is almost impossible. This is due to the high computational

load in scenarios where high precision and accuracy are needed to produce a useful map. In this Chapter we present a novel RT algorithm able to generate RF maps at large-scale while considering the 3D topography of the area of interest accurately and rapidly.. We introduce new schemes based on 3D triangulation to model the propagation environment and we leverage efficient hardware acceleration techniques to generate RF maps at large-scale.

Furthermore, instead of calculating the received signal power over a finite set of reception points and inferring the map by interpolation, we propose another reception test algorithm that is able to directly calculate the received signal power over the triangles composing the area, which allows for the consideration of all the rays hitting the surface and the direct rendering of the 3D map. All the contributions brought in this Chapter are summarized as follows:

- Introduction of a new acceleration scheme efficient for RF mapping by adapting *Embree* [37], an Intel RT engine used in computer graphics.
- Introduction of a surface topography modeling based on 3D triangulation to account for all variations in the surface's elevation and all obstacles composing the area.
- Design of a new reception test between 3D triangles and rays based on their geometries in order to account for the 3D topography and not only a plane as in Chapter 4.
- Proposition of a new RF map rendering approach based on the signal power received by the triangular mesh.

Simulations of our RF mapping solution show that our approach outperforms existing ones based on discrete reception points both in terms of execution time and memory consumption. Indeed, our approach is shown to be 50 times more efficient than traditional RT approaches to RF mapping as the one implemented in the Matlab official distribution [65]. Due to memory limitations, existing solutions are unable

to produce the RF map in urban areas involving multiple antennas and requiring a precise map. On the other hand, our solution generates a precise and accurate RF map in all types of terrains with a difference as small as 1Mbps on average in bitrate estimation compared to existing solutions. We also show that our approach preserves the accuracy of the RF map when changing the resolution step, i.e., with a 50 meters resolution step we achieve almost the same accuracy as when the resolution step gets near to zero.

Additionally, we demonstrate in this paper that the reception test accounts for more than 98% of the complexity of traditional RT approaches in urban complex areas. We then highlight how our reception test is able to tackle this issue with only a slight increase in the execution time with regards to the resolution step. We equally show how the ROI test in our solution which is based on Embree behaves independently from the number of triangles, i.e., one can choose a fine-grained size of triangles to improve the RF map precision without degrading the performance of the intersection test.

The rest of this Chapter is organized as follows. In Section 5.2, we overview the state-of-the-art and related works on the generation of RF maps. Section 5.3 is an overview of our solution with a main focus on its acceleration part. In Section 5.4 we present our RF map generation method with its novel reception test process, while simulation results and discussions are shown in Section 5.5. Conclusions are given in Section 5.6.

5.2 Related works

To mitigate RT complexity in complex environments, different sets of techniques were proposed. In those environments including a large number of buildings and featuring an irregular terrain, checking buildings' and terrain's facets hit by the rays is computationally heavy. For RF mapping purposes, the problem is even more complex because traditional RT acceleration techniques were not designed for this purpose.

This is the case for instance of the visibility algorithm explained in Section 5.1 that suffers from its Tx/Rx dependency, as it is only optimized for point-to-point scenarios.

To overcome this issue, another technique based on triangular grid approach [34] was proposed to overcome this dependency of the visibility algorithm. In this proposed technique, the vertices of the buildings serve as point clouds used to generate a triangular mesh. The obtained triangles are used afterwards to track the rays and check in which triangles the intersections with rays happen. Proceeding this way, the intersection test does not anymore depend on Tx and Rx but on the number of buildings itself, hence giving better results and accelerating the ROI test for large-scale scenarios.

However, the technique in [34] has several drawbacks. First, it is designed for a 2D scenario with a heuristic proposed to generalize it to 3D scenarios. According to this heuristic, when rays are received by a receiver in 2D, a reverse work is performed to check whether these rays hit buildings or not, taking into consideration their respective heights. Proceeding this way is a rough approximation of rays received. In fact, in 3D, rays are usually launched by an antenna placed at the center of a unit sphere in such a way to cover all azimuths and elevations; while in 2D, rays are launched in all azimuths of a single chosen elevation. It follows that all the other elevations are not accounted for. Second, due to this 2D nature of the technique, propagation mechanisms from the ground are not considered. Hilly or mountainous terrains are approximated as being flat without their elevation considered. Finally, the number of triangles used to speed up the intersection test in this approach is a function of the number of buildings' vertices. In complex environments, triangles' sizes may be close to zero, leading to a large number of small triangles used, thus yielding inefficiencies and increased complexity of the ROI intersection process.

The second aspect that is widely covered in the literature is the sampling or the discretization of the area of interest to produce RF maps as discussed in Section 5.1. Sampling is used in [23] to analyze the propagation characteristics of drone transmission at different frequencies. The simulations of the drone were performed through RT

and the RF map was displayed on a 450m $\times$ 500m terrain with a 5 meters resolution step between the reception points. In [24], the sampling approach is used to show the RT computation of the received power distribution over different antenna frequency settings in an indoor scenario with a 10 cm resolution step. Finally, in [25] the RF map is constructed for the received signal strength computed via RT at predefined reference points for indoor positioning purposes.

Sampling at high rates comes with an inherent complexity caused by the reception test as outlined in Chapter 4. This is because for each pair of ray and reception point, one must check whether the ray is being received or not. In scenarios requiring high precision as in [23] and [24], large matrix computations need to be performed in order to determine which rays have been received by which reception point.

Moreover, in large-scale scenarios, namely at city-level, millions of rays are launched and thousands or even millions of reception points are considered depending on the level of precision sought. Along with the large number of reflections that may be needed in some scenarios, the reception test can quickly get out of control. This is due to the large matrix computations that need to be performed, leading to a high execution time and a high memory consumption making it impossible to generate RF maps in some complex scenarios. Thus, one has to give up on the accuracy and/or the precision in order to be able to produce the RF map.

In [22], the discretization approach is used to show the RF map on a university campus. Authors designed a new technique in order to optimize the reception test and the field computation process. They built a triangular mesh of their area of interest using high resolution LIDAR data and ran 3D RT over it. Their approach requires for each pair of Tx and Rx, an exhaustive search among all the triangular surfaces in order to determine triangles involved in the propagation mechanisms (reflection and diffraction) between the transmitter and the receiver. In complex scenarios including large number of reception points and transmitters, this approach leads to an impractical execution time. Conscious about the computational complexity of their approach, the authors proposed an enhancement meant to reduce the computational load of

their model. Instead of performing the test with all the triangular surfaces, a subset of surfaces is a priori identified for each pair of Tx/Rx based on the distance between them. This optimization, although useful in reducing the computational complexity of their model, compromises its accuracy. Indeed, in case of high order reflections and diffraction, many triangles may be missed hence altering the real signal power received at a given geographical location.

To the best of our knowledge, none of the existing RT techniques are designed for RF mapping purposes and/or are not sufficiently optimized to meet the requirements of 3D large-scale RF map generation with account for the topography of the surface. To mitigate this, we propose in this Chapter a novel RT algorithm able to accurately and rapidly generate RF maps at large-scale while considering the 3D topography of the area of interest including buildings and elevation of the ground.

Based on a triangular grid approach, we speed up the ROI test and use the obtained triangular mesh to directly produce the RF maps. Instead of using the traditional approach to RF mapping known to be computation greedy, we introduce a new acceleration scheme by adapting *Embree* [37], an Intel RT engine used in computer graphics, to our case. To achieve this, we first triangulate the buildings and the ground taking into account the elevation of the surface and leveraging publicly available databases. Through this information, we get a model of the propagation environment in the form of a triangular mesh with the geographical locations of all the triangles as well as their elevations as compared to the sea level. This model is then fed to *Embree*, over which iterations are performed until the exact paths followed by the different rays launched by the transmitters are obtained. As Embree's kernel is designed to run directly RT on the CPU of the computer, the intersections between the rays and the triangular mesh of the terrain and of the buildings are automatically and efficiently carried out.

Moreover, Embree is simply rays and triangles dependent since the intersection test is done regardless of the positions of the reception points. We leverage this property to devise our cartography solution in a way that does not require the consideration

of receptions points. Instead of discretizing the receiving area using a finite set of reception points as it is the case in the literature, we illustrate the RF map using the same triangles colored as a function of the rays that intersect with them. By doing so, we are sure that all the rays are accounted for when producing the map and we could get rid of the cumbersome discretization approach used in the literature.

The main idea of our RF mapping solution is to directly compute the signal power received by each triangle of the terrain and to use the triangles as the basic unit to display the RF map. Since the triangular mesh captures all the details of the terrain, we are able this way to give the level of signal at every single location without performing any interpolation. Rather, we simply assign to a location the signal power of the triangle where it fits. This approach has thus the advantage of obtaining a precise map, meaning that regardless of the zoom level, the map is always able to give an estimation of the received signal power. Moreover, with our technique we account for all the rays that hit the surface, as they are all considered when computing the signal power of the triangular mesh.

Thanks to all these techniques embedded in our solution, we are able to replace the cumbersome reception test process, to provide a new RF map rendering technique based on triangles, and to propose a new power computation methodology for triangles. This allows us to reach our goal of RF map generation at large-scale, regardless of the type of terrain in a considerably shorter time than our peer state-of-the-art approaches.

5.3 System Overview

5.3.1 Real-life application

Our technique is meant to generate RF maps in real-life scenarios. To generate RF maps using our solution, three main information are needed about the propagation environment: (i) buildings' positions, (ii) antennas' information, and (iii) the eleva-

tions of the terrain. The more accurate these information are, the more accurate our RF map will be.

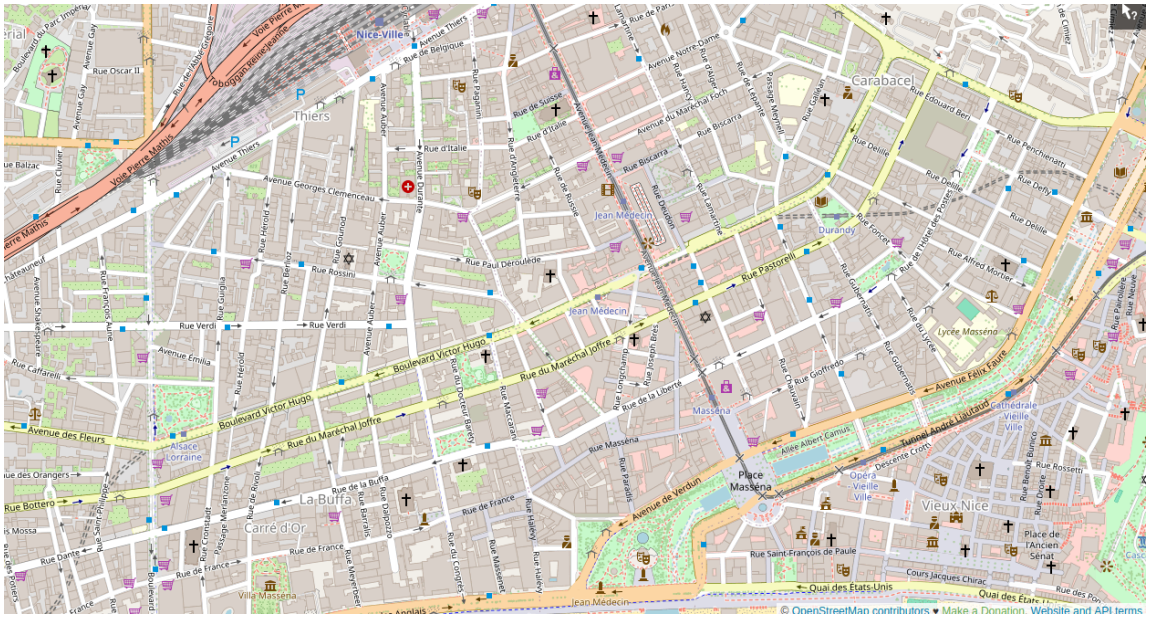


Figure 5.1: OpenStreetMap of the city of Nice in France containing buildings and streets. This Figure is taken from [7].

Depending on the country, these information may not be easy to obtain, since some countries consider these information as private, hence being challenging to use them to generate RF maps. Nevertheless, open collaborative projects such as OpenStreetMap [7] exist and they provide information about buildings present in almost all countries in the world. Yet, it is important to keep in mind that these information are put from local contributors and not from official sources. They may thus contain inaccurate descriptions of the buildings in the environment and often miss information on buildings. Still, this project remains one of the most used maps in the world, since it is free to use under open license as opposed to proprietary solutions like Google maps and Google Earth. Figure 5.1 is an example of map containing buildings and roads that can be obtained from the OpenStreetMap project. The buildings from this map can be exported in *.osm* format and can be directly fed to our RF mapping solution to produce the RF map.

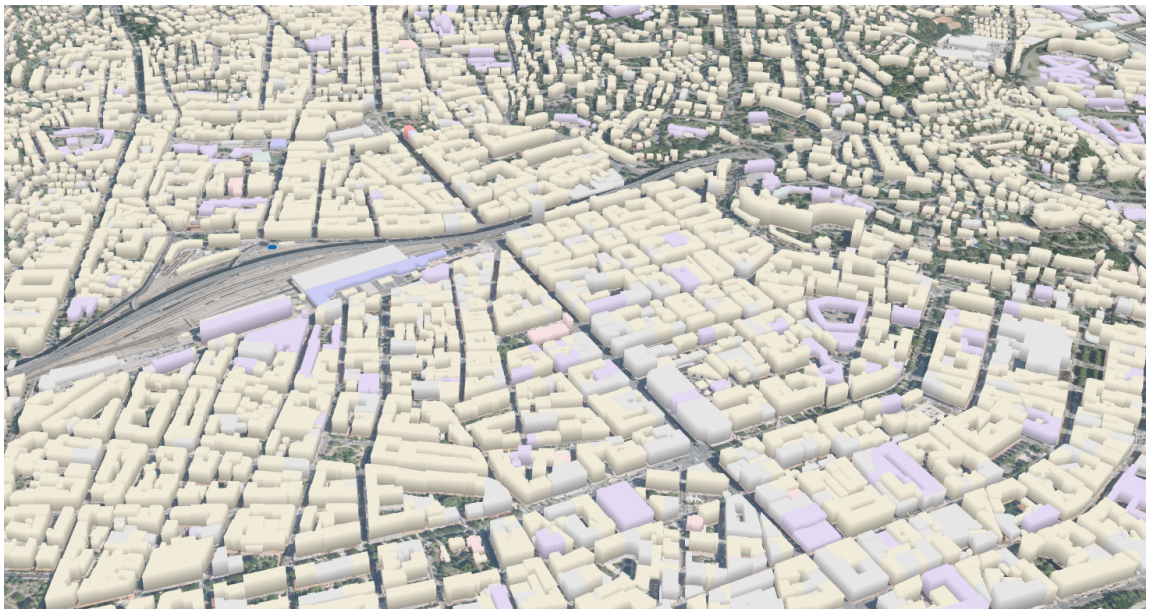


Figure 5.2: IGN's map of the city of Nice in France. This Figure is taken from [8].

Apart from this collaborative project, in some countries the government publicly releases buildings' information under open license. In France, IGN is in charge of producing and maintaining these information. IGN's databases of buildings come from aerial images, cadaster and often LIDAR. They provide detailed information about the shape of the building, its type, its height, the number of floors, materials of the buildings, the altitudes of the buildings and so on. More information on IGN's description of the buildings can be found in [9]. An example of IGN's map showing 3D buildings shapes is shown in Figure 5.2.

These rich information are frequently updated by IGN and can be easily downloaded using different GIS data formats. IGN's dataset can also be used in our solution to model the environment of propagation. This database is in fact the one we used for validating and showing the performance of our RF map solution.

It is though important to notice that these maps do not consider every single elements of a building, like windows and rooftops variations, as shown in Figure 5.2. Rather, a geometrical approximation is done on the buildings as shown in the example in Figure 5.3 explained in IGN document in [9], showing the challenges of modeling

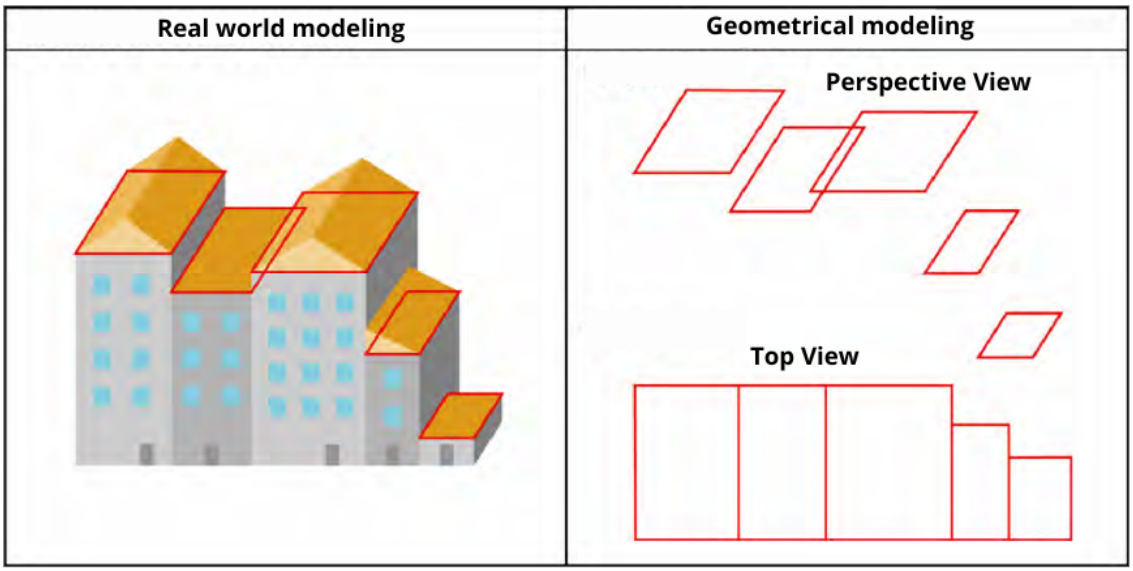


Figure 5.3: The left part of the Figure represents real-life buildings and the right part is their geometrical modeling, showing their perspective and their top views. This Figure is taken from [9] and translated from French to English.

the buildings. In this Figure, we can see that the rooftops are modeled by simple rectangles joining the two sides of the buildings and the windows are not considered at all. This is due to the geometrical complexities involved in accurately representing the various shapes of rooftops, windows, and other architectural features on buildings. Still, this remains a relatively good representation of buildings.

In addition to the information on buildings, IGN also offers a database containing the altitudes of every single location in France. The IGN's database has a 25 meters resolution over all the country. Using such map, it is possible to know the altitude of every single location in France. IGN provides this map in the form of raster data as shown in Figure 5.4. In this Figure, every single element of the raster map corresponds to an altitude value.

To ease the utilization of this altitude map, IGN implemented an API accessible using [76]. This API takes as input the latitude and the longitude of a geographical location and returns the altitude of the location based on the raster map of that location as the one in Figure 5.4.

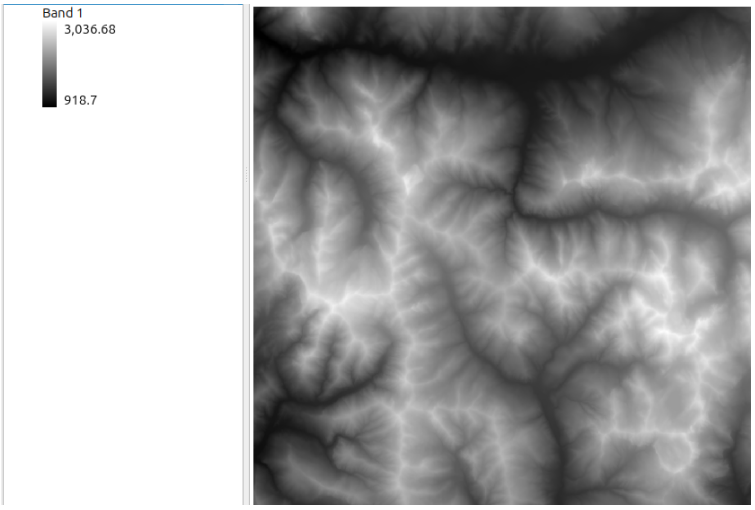


Figure 5.4: Example of IGN's raster map showing the altitude of every location in the map: Band 1 corresponds to the altitude from Black (918.7 meters) to White (3,036.68 meters).

Finally, the information about antennas is more critical to obtain depending on the country. However, this information is also publicly available in France. The ANFR, which is the French Frequency Agency Regulatory Authority releases on a monthly basis all the information on antennas in France. For each antenna, they give information about its longitude, latitude, height, orientation, type (directional or not), the frequency band used, the technology and the MNO owning it. To be transparent about all these antennas, they provide a service called Cartoradio [10] that provides to end users all the antennas in their vicinity. Figure 5.5 is an example of the Cartoradio output, showing antennas from all MNOs in the city of Nice in France.

In addition to Cartoradio, ANFR also provides its open data [77] under open license, which allows its use by everyone for free. We leveraged this open data set to deploy our RF map solution on these real-life antennas in order to produce highly accurate estimation of the received signal power.

In order for us to easily manipulate all these data during our study, we built an in-house API that gathers all this public information. Since our solution is meant to generate RF maps of a given location, the API takes as input the geographical

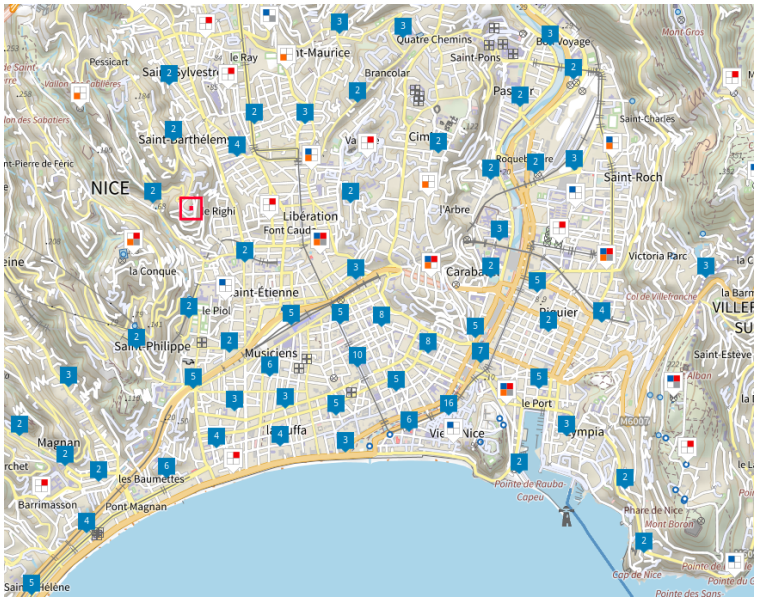


Figure 5.5: ANFR's map of the antennas in the city of Nice in France. This Figure is taken from [10].

coordinates of the center of the area of interest and the radius of interest. As output the API gives all the necessary information needed by our RF map solution, namely all the buildings and all the antennas within this radius. Different filtering parameters are available in our API as described below.

- Technology : 3G, 4G, 5G.
- Frequency: LTE700, LTE800, LTE2100, 5GNR700, 5GNR3500,...
- MNO: Orange, SFR, Bouygues, Free. These are the four major MNOs in France.
- Maximum Number of Buildings.
- Radius of the area of interest.

Using this API, we were able to run our RF map solution on real-life datasets in a handy and more controlled way. Indeed, during our simulations we applied these filtering parameters to target a specific radius size and specific MNOs for instance.

5.3.2 Topography and buildings modeling

Once we obtained the real-life datasets on buildings and antennas, another obstacle we faced was the modeling of the terrain topography. Suppose that we have an area of interest consisting of hills, valleys and other varied terrain, modeling such type of area in order to account for the variable nature of the terrain, i.e., to consider each variation in the elevation of the terrain is not an easy task [34, 36]. This is in fact the reason why in Chapter 4 we only used a fixed horizontal plane to model all the environment, limiting its use to areas with slight terrain variation.

To take into account the variation of the 3D topography of the terrain, and to account for the different buildings and obstacles over it, we proceeded this time by triangulation, an approach used in computer graphics for digitizing 3D objects. Following this approach, we represented the 3D terrain and the obstacles using a mesh of triangles able to accurately represent altitude variation. By doing so, we were able to overcome the state-of-the-art limitation concerning terrain modeling for its use in RT.

Leveraging the continuous altitude's dataset of IGN as the one in Figure 5.4 and by performing a uniform triangulation of the terrain, we were able to successfully construct 3D triangles that comprehensively capture and model fine-grained changes in altitude within the area of interest. Figure 5.6 showcases the 2D projection of the outcome of our triangulation process, where each vertex of these triangles has 3D geographical coordinates namely the latitude, longitude, and altitude.

We opted for uniform triangulation as it aligns with our specific requirements for RF map generation. The resulting triangles from this modeling technique are the ones directly used to display the RF map. Note here that as with any triangulation technique, employing smaller triangles enables to capture fine-grained changes in the terrain with greater precision. It is thus crucial to carefully regulate the size of the triangles to ensure the precision of the generated RF map. By controlling the triangle size, we can effectively account for the level of details and granularity necessary to

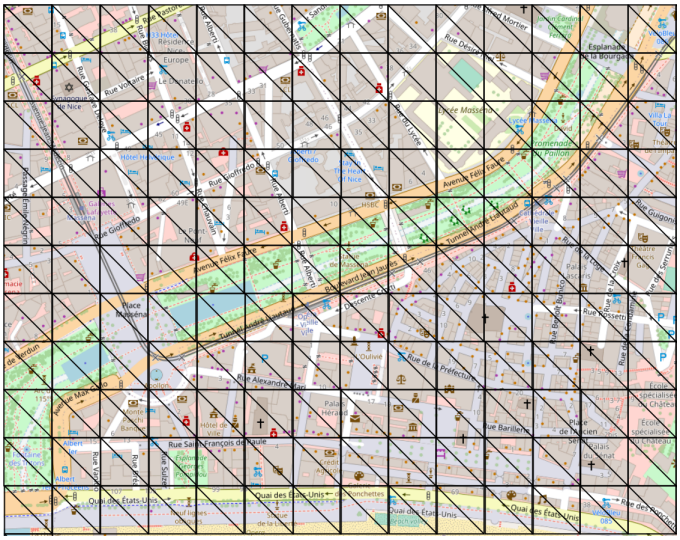


Figure 5.6: Uniform triangulation of the city center of Nice in France

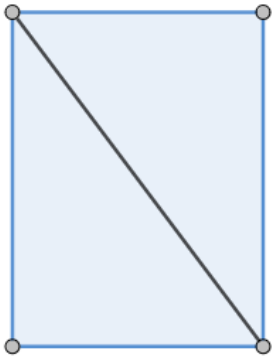
accurately represent the RF information on the map.

Additionally, we model buildings in the same fashion as the topography by the use of triangulation. When taken from IGN's database, buildings are polygons described by their shapes as in Figure 5.3. The heights of these polygons help to obtain the resulting 3D building which is elevated from the ground to its height as in Figure 5.2. This 3D representation of the buildings enables the appearance of their components: their rooftops and their facets.

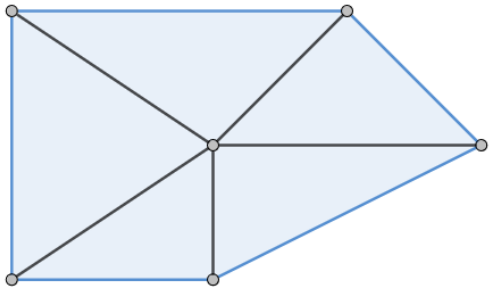
To triangulate these two components, we use the Delaunay Triangulation algorithm [78]. Delaunay triangulation is a triangulation approach that optimally arranges triangles within a given set of points, ensuring that no point lies within the circumscribed circle of any triangle, thereby maximizing the minimum angle of the triangles. This method yields to triangulated surfaces with advantageous geometric properties, such as well-shaped triangles with angles closer to equilateral.

Since the facets of buildings are vertical with a rectangular shape, their Delaunay triangulation corresponds to dividing the rectangular facet into two triangles. This division helps to capture all the information on the building's facets without any approximation. The vertices of the obtained triangles have 3D coordinates each

representing the longitude, the latitude and the elevation of the vertex as shown in Figure 5.7a. On the other hand, buildings' rooftops are horizontal with a polygonal shape. The form and size of the polygon is not known a priori as it depends on each building. There may be as many rooftop shapes as the number of buildings in the area of interest. The Delaunay triangulation is then simply applied on each individual rooftops' vertices to obtain their triangular meshes as shown in Figure 5.7b.



(a) Delaunay triangulation of a building's facet



(b) Delaunay Triangulation of a building's rooftop

Figure 5.7: Outcome of a Building Triangulation

5.3.3 Ray/Object intersection test

In large-scale scenarios involving a large number of buildings and antennas, a large number of rays are launched to ensure accurate estimation of the received signal power. These rays interact with the obstacles present in the environment through multiple reflections. A large number of ray reflections over the buildings' facets and rooftops is necessary to reflect the realities of the propagation environment. Due to the large number of buildings, complex intersection tests must be performed in order to determine the direction of propagation of rays once they hit the surface of the obstacle. For this use case, different techniques have been proposed in the literature to enhance the way these intersection tests are performed.

Nevertheless, to the best of our knowledge, none of the acceleration techniques used in the literature were designed to account for the variations of the ground when

performing the ROI test. Most of these techniques focus instead on optimizing the ray/building intersection test without any regard to ground reflections. In reality, almost all the rays launched below antennas' horizon hit the ground and are reflected back in the area of interest, hence the importance of taking this reflected energy into account when performing RT at large-scale.

To account for both buildings and the ground as obstacles, there is thus a need for a technique fast enough to account for the intersections with those obstacles. This is why we chose *Embree*, a high-performance RT library developed by Intel for x86 CPUs. Embree is primarily designed to accelerate RT computations in 3D rendering applications. The low-level kernel of Embree is designed to leverage the computational power of modern CPUs to optimize RT operations. In particular, the SIMD feature of Embree allows parallel processing of multiple data elements in a single instruction and so leads to significant performance gains. Embree uses Ray Sorting, which consists in rearranging rays based on their proximity to one another, ensuring that nearby rays are processed together. It also provides a collection of functions and data structures that enable efficient ray traversal and intersection calculations. Furthermore, the Bounding Volume Hierarchy technique used by Embree allows to organize the geometries in the area of interest into a hierarchical tree structure. Ray/Objects intersections are then performed by traversing the tree, hence reducing further the computational overhead incurred by RT.

We build upon Embree and develop a tool for a fast ROI test. When the rays and the geometries in the area of interest are given to Embree as input, we get as output the intersection points between the rays and the objects. To identify these intersection points, Embree takes as input a triangular mesh of the objects (in our case, the ground and the buildings), together with the origins and directions of rays. This property of Embree is of great interest to us because it is fully compliant with the topography triangulation scheme explained in Section 5.3.2. Since we model the terrain surface by 3D triangles, our model is automatically considered by Embree when performing the intersection tests. This triangulation constraint is also the reason why we performed

the triangulation of the buildings, both rooftops and facets. Once the intersections points between rays and objects are found using Embree, our tool calculates the reflected rays' paths using Equation (2.44) and proceeds in a recursive fashion until the maximum number of reflections is reached.

5.4 RF Map Generation

In the previous section, we presented an overview of our approach with a main focus on the models it embeds and the acceleration technique it uses to speed up RT. Since our final goal is to produce large-scale RF maps, we focus in this section on the rendering part.

As said in Section 5.2, the state-of-the art approach to produce RF maps is to sample the area of interest following a certain statistical distribution, to calculate the signal strength received by those reception points, then to interpolate it for map construction. This approach has performance, precision and accuracy issues as explained in Section 5.2.

To overcome these issues, we leverage the triangular mesh obtained from the terrain modeling scheme introduced in Section 5.3.2. Those triangles that can be seen in Figure 5.6, are directly used to produce the RF map. The signal power corresponding to each triangle is computed based on the rays it receives, both main and interference. Leveraging this power computed for every triangle, we are able to produce the RF map as a heat map of triangles, thus giving the level of signal in every single location of the area of interest at the granularity of considered triangles. This approach for rendering has the following advantages:

- **Completeness**: All the rays hitting the ground and buildings (i.e., the triangular mesh) are considered. None of them are discarded as opposed to the traditional approach where some rays are not considered because they are not received by the chosen reception points.

- **Precision**: Since the triangles cover the entire 3D space, we can capture the power received at every single location without the need for further interpolation. Instead, whatever the zoom level of the RF map is, our approach can always provide an estimate of the signal power received, which is not usually the case in the state-of-the-art approach where the power is only calculated over the set of reception points. We recall here that the precision of our approach is still determined by the size of triangles we consider. Later, we will show sensitivity results on the impact of this size on the quality of the obtained RF map.
- **Applicability in real-life**: The triangles we used are in 3D and representative of the terrain's topography. Since we are leveraging those triangles to compute the signal power received, we are able to produce accurate RF maps without any altitude approximation.
- **Scalability**: Our approach is applicable to any city or country regardless of its size. Indeed, triangles being independent from each other, we are able to treat them separately by splitting a big area of interest into smaller ones and merging them afterwards to produce the final RF map.

To achieve all these goals, a new reception test is needed to account for the geometrical interactions between the triangles and the rays. We then build upon all the rays received by triangles to calculate both the power of the main signal and the interference for each triangle and use them to color the triangles and produce the RF maps.

5.4.1 Reception Test

The reception test used in the literature to verify whether or not rays are received by the reception points is a cumbersome process. The large matrix computations performed in this process leads to high memory consumption that prevents large-scale applicability of RT. It also leads to a high execution time, thus preventing RT from producing maps in many scenarios. Since our goal is to produce large-scale RF

maps in a reasonable time, we remove this cumbersome process and propose a new one more suitable for our use case.

Our proposed test consists in leveraging the geometrical properties of the rays and the triangles they intersect with and performing spatial intersection between them. Given that rays are cone-shaped, we start by computing the radius of all the rays hitting the ground at their points of intersection. Since the cross-sections of the rays are circular, this corresponds to directly performing a spatial intersection between the two geometrical figures: circles and triangles.

Rather than going through all the complex matrix computations of the traditional RT reception test process, we propose the following test of reception for a certain ray received by a certain triangle: *a ray is said to be received by a triangle if its cross-section overlaps with that triangle.*

Leveraging the geometrical properties of rays and 3D triangles defined by their geographical locations and shapes, we avoid performing the reception test for each pair of Tx/Rx as is the case in the literature. Instead, the reception test now consists in an intelligent spatial intersection operation that accounts for the locations of rays and triangles. Due to these, we are even able to perform these operations at once, by just performing a 3D spatial join of the two geometries (rays and triangles). As result we obtain which triangle overlaps with which ray, in a rapid and efficient fashion. In this Chapter, we used Shapely [79], a geometrical tool to perform this operation.

Furthermore, our new test can easily account for the fact that a ray can be received by more than one triangle as the case in Figure 5.8. Indeed, due to the new spatial intersection test that does not only consider the center of the ray but its whole geometry, the contribution of a ray is taken into account in all the triangles it overlaps with. This property is of great interest since all the geometries of rays are accounted for, which leads to the production of accurate RF maps.

Note that this final information on the rays that are received by each triangle is helpful in the computation of the downlink signal strength and the level of inter-

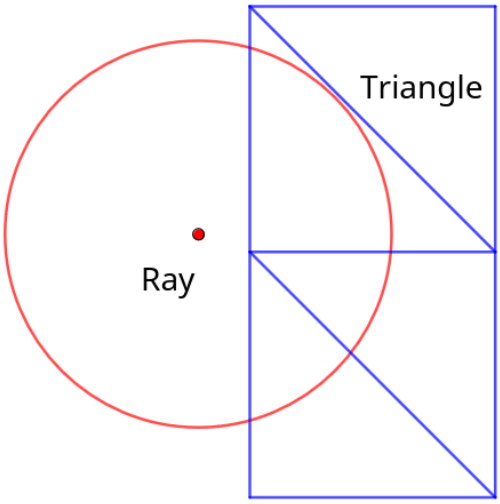


Figure 5.8: A ray cone overlapping with three triangles

ference, which can be used to calculate the SINR and an estimation of the channel capacity (bitrate) as well as producing the RF map for either of these RF metrics.

5.4.2 Power computation

Once the rays received by the different triangles are known, we compute the SINR for each of them. The intuition behind computing the power received by a triangle is to assume each triangle to be the equivalent of a reception point. The only difference is that a point corresponds to a given location and that a triangle can cover several locations. Later, we will study the impact of the triangle size on the accuracy of our solution.

To calculate the power across a triangle, we apply a similar methodology to the one used to calculate the power of a reception point while getting rid of the phase shift between received rays as opposed to the signal power formula in Equation (2.47). Indeed, in Equation (2.47), the signal power formula accounts for the phase shift since it is the power computed for single reception points. While in this current scenario, we consider the phase shift between rays as a uniform random variable due to the small value of the wavelength which is far less than the triangle's size due to the

high frequencies considered. By considering the phase of a ray as a uniform random variable in the range $[0; 2\pi]$ independent of the phase shifts of the other received rays, we can easily derive an expression for the expectation of the received power. This expectation corresponds to the average power received by a triangle.

Consider a set of received signals in the form $s_i = A_i e^{j(wt+\varphi_i)}$, with A_i, w and φ_i being respectively the amplitude, the frequency and the phase of the i^{th} signal. All these signals are supposed to be in the same frequency band, i.e $w_i = w, \forall i$. The power of the total received signal can be expressed as follows.

$$\begin{aligned} P_{rx} &= \left(\sum_{i=1}^{n} s_i\right)\left(\overline{\sum_{k=1}^{n} s_k}\right) \\ &= \left(\sum_{i=1}^{n} A_i e^{j(wt+\varphi_i)}\right)\left(\sum_{k=1}^{n} A_k e^{-j(wt+\varphi_k)}\right) \\ &= \sum_{i=1}^{n} A_i^2 + \sum_{i=1}^{n}\sum_{\substack{k=1 \\ k\neq i}}^{n} A_i A_k e^{j(\varphi_i-\varphi_k)} \end{aligned} \tag{5.1}$$

$\varphi_i, \varphi_k \in [0; 2\pi]$: uniform random variables

$\Rightarrow \mathbb{E}[e^{j(\varphi_i-\varphi_k)}] = 0, \quad \forall i \neq k.$

Hence, $\mathbb{E}[P_{rx}] = \sum_{i=1}^{n} A_i^2 = \sum_{i=1}^{n} P_i.$

It follows from Equation (5.1) that the average power received by a triangle is the sum of the powers of all the rays it receives for the same frequency band. This power P_{rx} is given by Equation (5.2) with P_i being the received power of the i^{th} ray, P_{tx} is the transmitted power, G_{tx} and G_{rx} are the antenna gains, λ is the wavelength, d is the unfolded distance crossed by the ray and R_n is the n^{th} reflection coefficient of the i^{th} ray at the n^{th} obstacle along its path given by Equations (2.48) and (2.49).

$$P_{rx} = \sum_{i=1}^{n} P_i = \sum_{i=1}^{n} P_{tx} G_{rx} G_{tx} \left(\frac{\lambda_i}{4\pi d_i}\right)^2 \left|\prod_{n=1}^{N} R_n^i\right|^2. \tag{5.2}$$

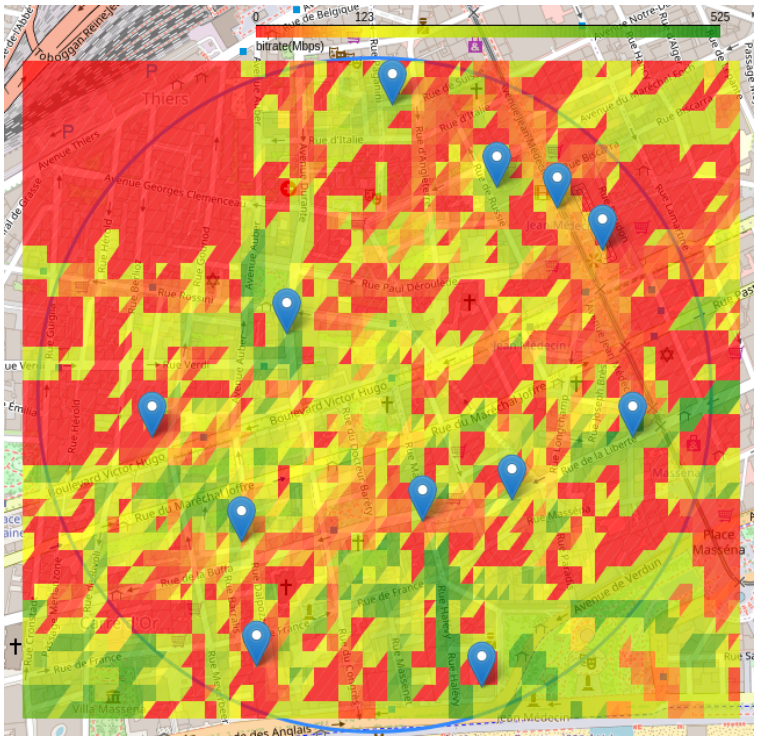


Figure 5.9: RF map of bitrate in the city of Nice in France

This power expression helps to derive the SINR, which can be computed using Equation (2.51). Once the SINR of all the triangles is computed, the RF map can be generated in a heat map fashion showing the SINR at a every single location in the area of interest. Note that with the SINR in hand, one can compute the bitrate depending on the information about the technologies used by the antennas: 3G, 4G or 5G. Additionally, from the SINR, the downlink CQI can be derived for instance, then mapped into the download bitrate at the given location using the specifications related to each wireless technology [43, 44]. For the purpose of our simulation, we opt for the Shannon capacity which gives the maximum channel capacity given the SINR value and the channel frequency bandwidth using Equation (2.52).

As a result of our implementation, Figure 5.9 gives an RF map illustrating with colors the bitrate in every single location in the city center of Nice. The colors ranging from red to green are indications of the estimated bitrate in the triangular areas. The blue marks identify the antennas used to launch the rays.

5.5 Results

We conduct several simulations using our solution to evaluate its performance and assess its capacity to generate RF maps at large-scale while considering the terrain's topography. These simulations have three main objectives: (i) validate our solution against state-of-the-art approaches, (ii) evaluate its execution time as compared to existing RT tools, and (iii) show the effect of triangles' sizes on the overall accuracy of the obtained map.

Terrain	**Number of Antennas**	**Number of Buildings**	**Center Latitude**	**Center Longitude**
T1	133	50377	43.699195	7.263854
T2	61	70744	43.596062	7.005393
T3	7	13682	43.839580	7.351883

Table 5.1: Characteristics of the three terrains

To reflect the real-life application of our approach, we conduct the simulations over three different terrains with a radius size of 5km. The characteristics of these three terrains are provided in Table 5.1. From the number of antennas and buildings, we can see that the two first ones are urban areas, and the 3rd terrain is a rural area. Figure 5.10 shows sample of buildings of the first terrain used for this simulation.

To perform our simulations, we use a Linux server featuring 32 processors and 94 Gigabytes of memory. General parameters values used in our simulations are given in Table 5.2.

Parameter	P_{tx}	ϵ_r	G_{rx}	G_{tx}	N	B
Value	40 W	5.31	2.1 dBi	0 dBi	-107 dBm	1 MHz

Table 5.2: Simulation parameters

5.5.1 Validation

We first start by validating our model and checking the correctness of the new concepts involved in its processes. We use for this a state-of-the-art traditional implementation

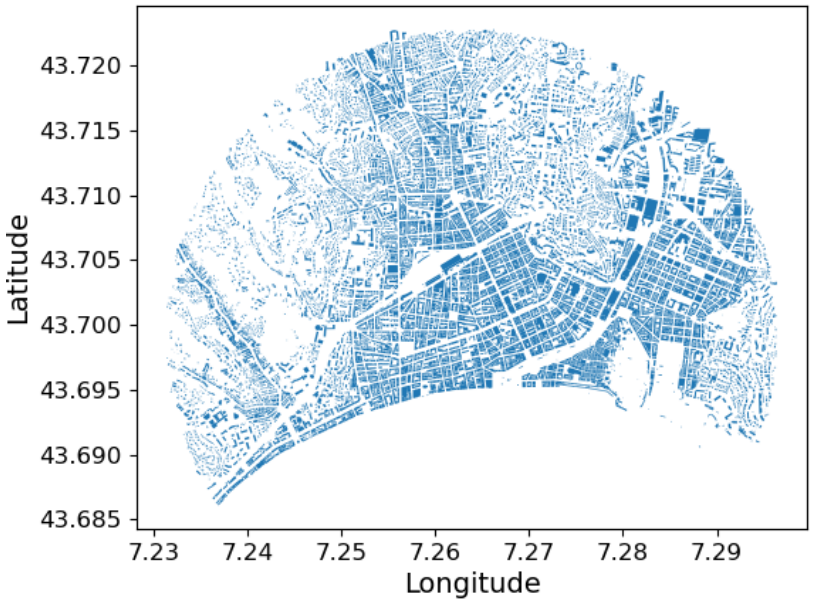


Figure 5.10: A sample of 20000 out of 50377 Buildings of Terrain 1 is shown here. This is not completely circular because the down part of the Figure corresponds to the sea.

of RT, namely the one developed by Matlab [65]. Indeed, Matlab offers a toolbox that helps to generate RF maps using the space sampling approach explained in Section 5.2. According to Matlab, one can discretize the area of interest following a uniform distribution with a given resolution step between the reception points. Since our approach is triangle-based, the received signal power of a given location is the signal power of the triangle to which it belongs to, as opposed to Matlab which directly considers discrete reception points.

To validate our model, we set the resolution step in our model to the smallest value that we could take constrained by memory limitations, which is 2 meters. This is to be as close as possible to Matlab's estimation and to mimic a reception point that is used in the Matlab's approach. Indeed, theoretically the smaller the size of the triangle is, the closer we are to the power received by a single reception point.

To assess the robustness of our approach, we perform the simulations over different settings. For all the three terrains, we vary the number of reflections and the angular separation between rays and compare our model's estimation to the one of Matlab.

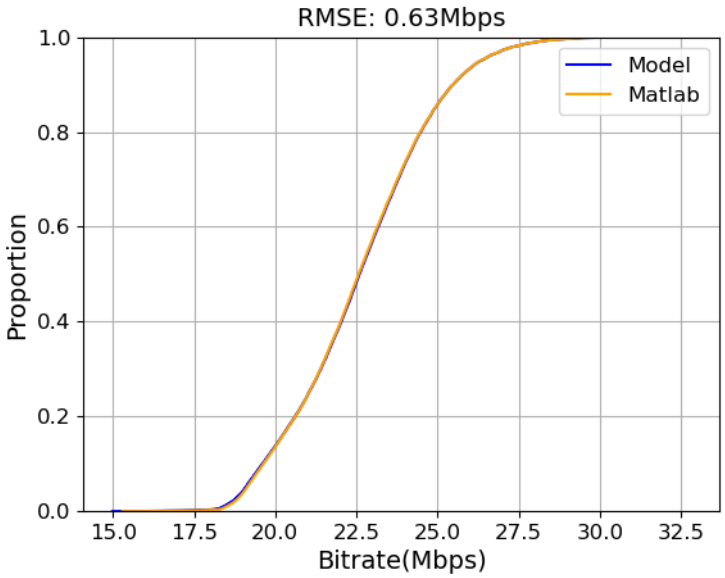


Figure 5.11: CDF Terrain 1: LOS.

For this, we choose three different ray angles known as angular separation values corresponding to Matlab's predetermined values: *large: 1.07°, medium: 0.55°, small: 0.28°*. It is important to note that small angular separation means that the distance between neighboring rays is small due to the high number of rays launched. On the other hand, three different values of maximum number of possible reflections are chosen : 0, 2 and 5. The former corresponds to a LOS scenario while the latter are NLOS ones.

Figures 5.11 and 5.12 show the CDFs of our model against the one of Matlab in respectively LOS scenario and with 5 maximum number of reflections. On the other hand, Figures 5.13 and 5.14 show how correlated they are in those two cases. In the LOS case, the two distributions overlap exactly with each other, i.e., the two estimations have the same distributions: with their bitrate value ranging from 15 to 32Mbps. The Figure shows that over the reception points considered, the percentiles values are the same for the two estimates, i.e., 25% of Matlab's and our model's estimations have a bitrate lower than 21Mbps, the two models have the same median of 22.5Mbps and so on. This property is very important since it shows that our model completely aligns with the one of Matlab even though our model uses a different

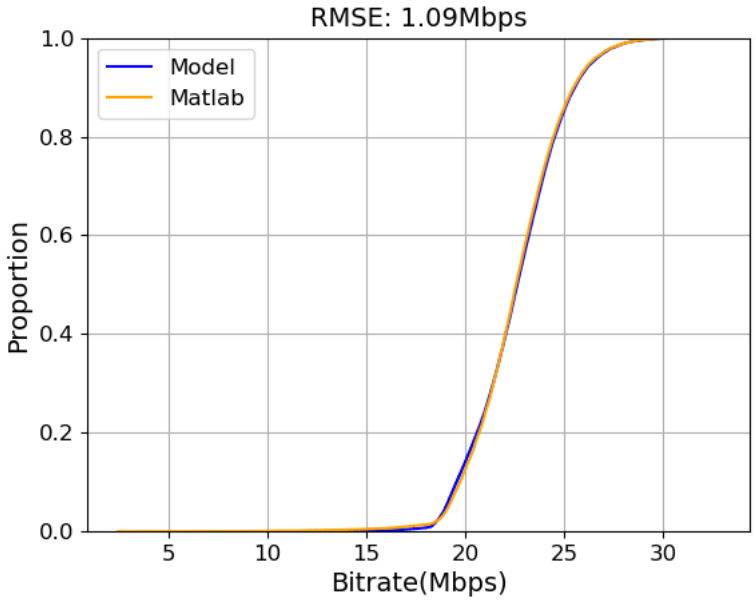


Figure 5.12: CDF Terrain 1: 5 reflections max.

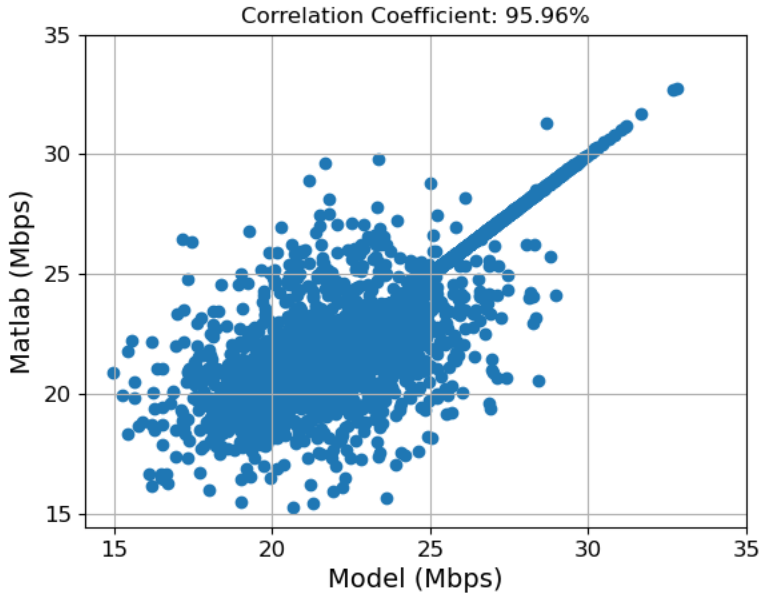


Figure 5.13: Correlation in Terrain 1: LOS.

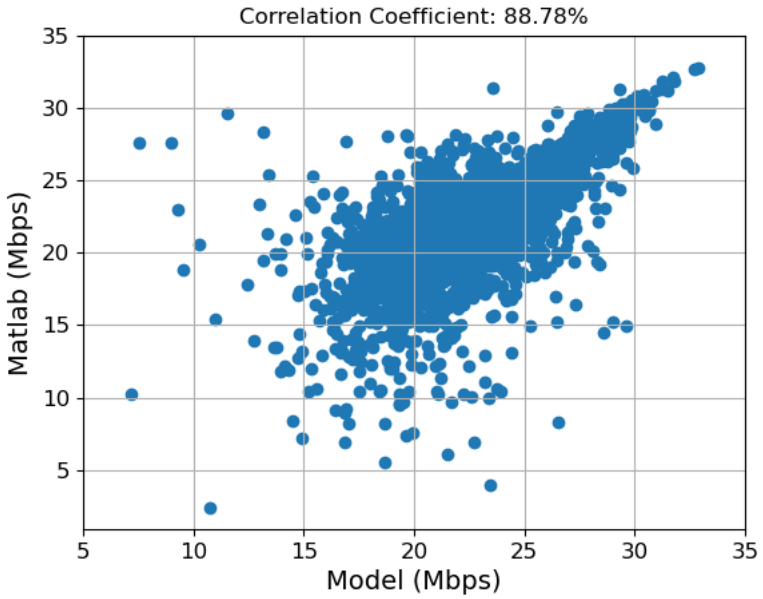


Figure 5.14: Correlation in Terrain 1: 5 reflections max.

approach to generate RF maps.

Furthermore, the RMSE value of 0.63Mbps is explained by Figure 5.13. From the latter, we can see a linear correlation of almost 96% between the two models. We can observe from the Figure that the slope of the direct line is 1, i.e., that in almost 96% of cases, our model and Matlab estimations are exactly the same. This shows the ability of our model to accurately estimate the signal power received at a given location. The Figure also shows the outliers on both sides of the main line. This slight difference is because in LOS scenarios, Matlab takes the exact path between Tx and Rx, while in our model we consider the rays intersecting the triangle in which Rx lies in. This results in a difference in the distance followed by the LOS radius, and therefore a difference in the power estimate, calculated by Equation (5.2), where the power is a function of $1/d^2$. These outliers explain this RMSE value, which sounds reasonable in view of the range of the bitrate.

Similarly in Figure 5.12 we see that in the case of 5 maximum number of reflections, the two models still maintain the same distribution, i.e., they have the same overall statistics in terms of bitrate estimations as in the LOS case. This shows that the

Terrain	**T1**			**T2**			**T3**		
Max Reflections	0	2	5	0	2	5	0	2	5
Angle: Large	1.02	1.08	1.22	1.19	1.25	1.33	0.87	0.86	0.86
Angle: Medium	0.81	0.96	1.10	1.01	1.10	1.19	0.73	0.74	0.74
Angle: Small	0.63	0.94	1.09	0.80	1.01	1.18	0.62	0.64	0.64

Table 5.3: RMSE (Mbps) on the three terrains.

maximum number of reflections does not affect the accuracy of our model and that our model is able to accurately generate RF maps including high order of reflections despite the new schemes introduced. Indeed, the error of 1.09Mbps which is mostly due to the outliers is acceptable within this range.

Figure 5.14 is the explanation of this small error. From the latter, we see that in almost 89% of cases, our model gives exactly the same estimation as Matlab. The other 11% correspond to reception points where the two models estimate the bitrate differently. These latter points seem to be distributed in a symmetric way around the diagonal. We interpret this phenomenon by a difference in the set of rays received between the two models on these points, with this difference being more pronounced on the left part of the Figure than on the right part. We believe that this difference increases with the number of reflections, which is very likely the case of the points on the left part of the Figure. Overall, our model maintains the accuracy of RT by minimizing the differences with the traditional RF mapping approaches.

The figures shown above are only for the first terrain and two use cases. Nevertheless, the simulations were performed over all the three terrains and different settings. All these studies show the same patterns in terms of distributions and errors. Table 5.3 summarizes all these findings.

From this table, we can see that all our simulations show the same trend as the results explained earlier. We can observe that the higher the number of rays launched is, the smaller is the error. The angular separation is small in those cases and as shown in Chapter 4, the smaller will be the rays' cross-sections and thus, the less difference there will be with a reception point compared to the triangle. The same thing happens with the number of reflections, when they are high, we notice a slight

increase of the error. Because a large maximum number of reflections means longer distances traveled by rays and thus larger rays will be obtained. It leads to this difference because the comparison is done with reception points.

Nevertheless, from the table we can see that despite the terrain, the error is more or less the same. This shows the robustness of our approach and its applicability over different terrains without increasing the error. Moreover, on Terrain 3 the errors with 2 and 5 maximum number of reflections are the same. As this is a rural area, a large number of reflections is not necessary, due to the limited number of buildings.

Overall, our model maintains a very good accuracy with an average RMSE that is less than 1Mbps, which is a small error regarding the range of bitrate values.

5.5.2 Execution time

In this section, we conduct a comparative analysis of the execution time between our RF mapping solution and the traditional approach presented in Section 5.2. Since Matlab allows the generation of RF maps based on the latter approach, we compared our model against it. Since this consists of comparing two different approaches, we ensure fairness by running them with comparable configurations. Indeed, in Matlab the terrain is discretized using a uniform distribution with a specific distance between the reception points. While in our model, this distance corresponds to the size of the square used for triangulating the environment. To ensure that the number of reception points in Matlab is equivalent to the number of triangles in our model, we derived a simple formula that provides the resolution step to be chosen in Matlab based on our resolution step. This formula is derived as follows.

Consider the area A of a surface of interest as the one in Figure 5.15. Since the resolution in our model is the length of the square used to obtain the triangles, the number of triangles from this Figure is given below.

$$N_{Triangle} = \frac{2A}{Res^2_{model}} \tag{5.3}$$

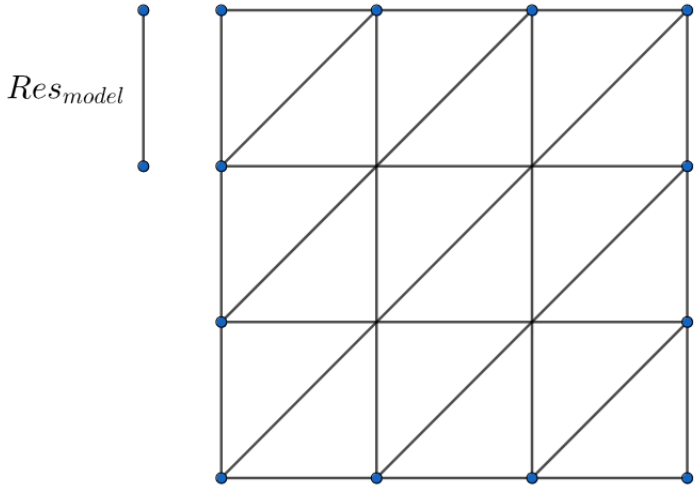


Figure 5.15: Illustration of the equivalence of the resolutions of our model and Matlab

On the other hand, the goal is to find an equivalent number of points in Matlab to the number of triangles in our model. The number of points in Matlab as the function of the resolution is given below.

$$N_{Point} = \frac{A}{Res^2_{matlab}} \tag{5.4}$$

To have the same number of points and triangles, Equation (5.3) must be equal to Equation (5.4). Therefore the resolution of Matlab as a function of our resolution is given by Equation (5.5).

$$Res_{matlab} = \frac{Res_{model}}{\sqrt{2}} \tag{5.5}$$

Afterwards, we conducted the simulations over the terrains described in Table 5.1. For all these three terrains, we assessed the execution time to generate their RF maps in both Matlab and our model. Since the resolution step and the number of rays launched determine respectively the precision and the accuracy of the map, we assess how the execution time evolves with this resolution step and with the number of rays. The resolution step was chosen in the range of 15m to 50m with a 5m step and for Matlab, the three sets of predetermined number of rays were considered: *large, medium, high* as explained earlier in Section 5.5.1.

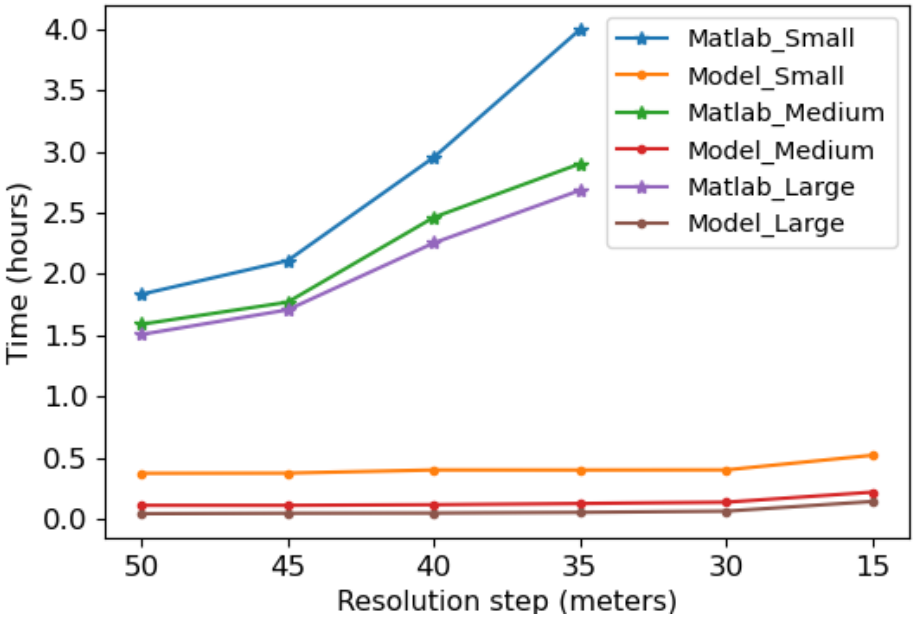


Figure 5.16: Execution Time: Terrain 1

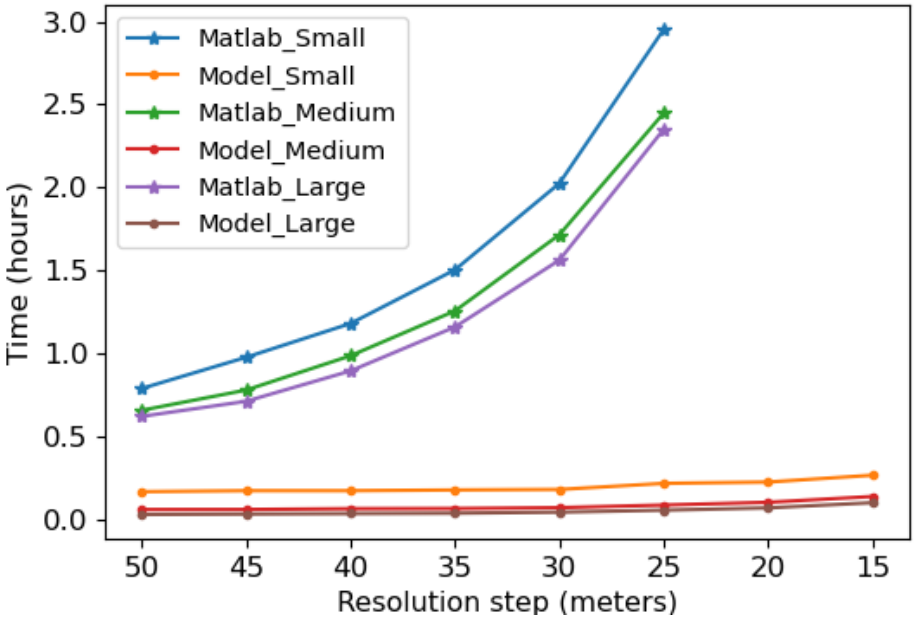


Figure 5.17: Execution Time: Terrain 2

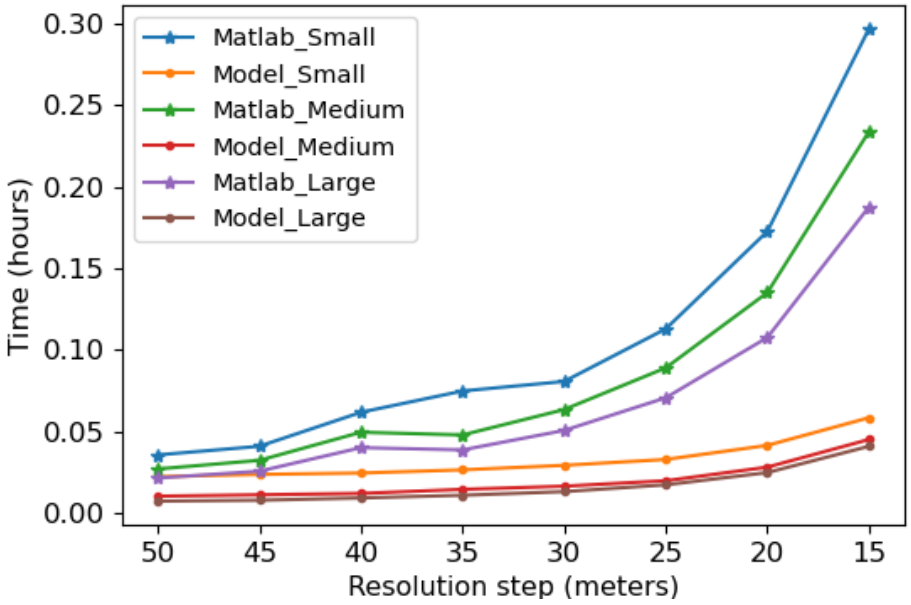


Figure 5.18: Execution Time: Terrain 3

Figures 5.16, 5.17 and 5.18 show the results of our simulation over the three terrains respectively. In each of these figures are given our model's and Matlab's execution times in hours for the three sets of angular separations as a function of the resolution step. These figures clearly show that Matlab is much more complex in terms of execution time than our model for the three terrains, the three angular separations and for all resolution steps. This means that despite the parameters chosen, our model outperforms Matlab in terms of execution time.

Additionally, Matlab shows an exponential increase in time as the resolution step becomes smaller, thus showing its inability to produce accurate maps in large-scale complex scenarios. On the other hand, our model's execution time remains almost constant regardless of the resolution step. This finding is of great interest, since there is not that much overhead incurred by our model due to the number of triangles, hence allowing the generation of high-resolution RF maps capable of precisely giving the bitrate at any location of the map using our RF map solution.

Moreover, we can notice the good reaction of our model to the number of rays, since the difference of the times between the *large* and *small* angular separations is small.

We are therefore able to produce accurate and precise maps with an execution time that is much lower than the one of Matlab. Indeed, from Figure 5.16, to obtain an accurate map with a 35 meters precision using Matlab, one has to wait more than 50 times the execution time of our model.

Moreover, in Figures 5.16 and 5.17, with respectively 133 and 61 number of antennas, Matlab's plot stops at steps 35m and 25m respectively. It stopped at those resolution steps because Matlab was unable to produce the RF map for all the three angular separations. This is due to the high complexity of the traditional approach used for RF mapping. Indeed, the large number of reception points leads to an intensive matrix calculation which exceeds the memory capacity of the machine and therefore leads to the impossibility of producing the RF map. This demonstrates that with the current RF mapping approach it is impossible to obtain precise maps in large-scale urban scenarios involving hundreds of antennas.

Furthermore, these simulations are performed on a 5km radius terrain size. This shows that on a larger scale, at an urban city-level for instance, current models will simply fail to generate accurate and precise RF map due to the large number of antennas and reception points. On the other hand, with a quasi-constant time, our model easily generates accurate and precise maps as shown in the figures. We achieve this performance because our model is solely designed for RF mapping purposes and is hence fully efficient and optimized for this.

From these figures, we can also see the impact of the number of antennas on the execution time. The maximum ratio between our model's execution time and Matlab's is 50, 43 and 5 respectively for the three terrains. This ratio decreases as the number of antennas decreases. Fewer antennas mean fewer rays launched, hence fewer matrix calculations performed by the traditional approach, thus leading to this smaller ratio value. However, even in those cases typically in Terrain 3, our model still outperforms the traditional approach and exhibits a slight increase in the execution time as opposed to the exponential increase in Matlab.

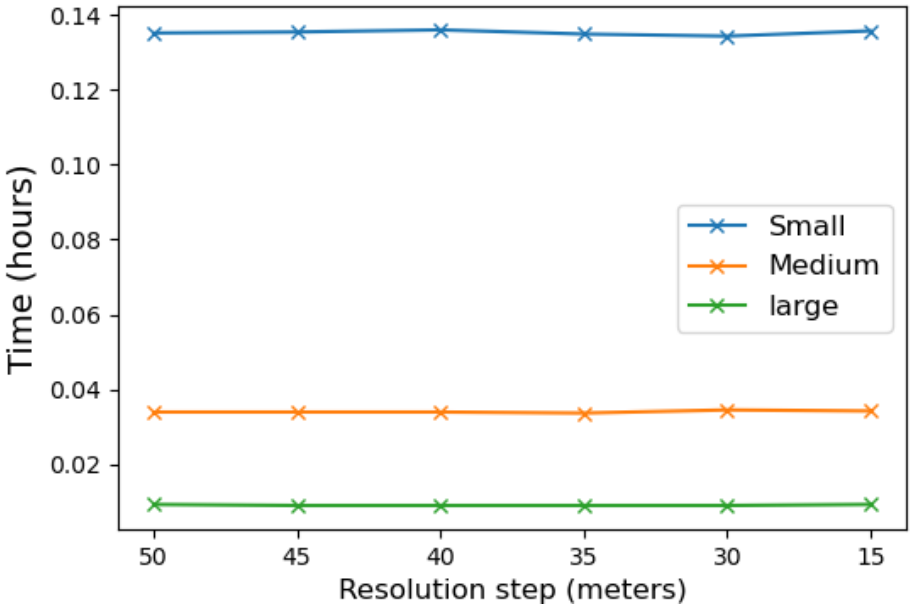


Figure 5.19: Embree Complexity: Terrain 1.

Furthermore, Figure 5.19 shows the contribution of Embree, namely the ROI test on the overall performance. From this Figure, we can see that Embree's complexity grows with the number of rays launched and is not sensitive to the resolution step. The larger the number of rays launched is, the more ROI tests Embree will perform. Hence, as the resolution step decreases, Embree's execution time remains constant. This property brought by Embree's underlying technologies explained in Section 5.3.3 helps us to choose a fine-grained resolution step without exploding the cost of the intersection test.

The combination of Figures 5.16 and 5.19 shows the efficiency of our reception test scheme introduced. Indeed, as the resolution step decreases, the execution time of Matlab is exponential due to the large matrix calculations necessary to carry out the reception test. On the other hand, the efficiency of our reception test prevents the explosion of our execution time. Instead, as the resolution step decreases, there is only a slight variation of our model's execution time in all the three terrains.

This efficiency is further emphasized when Matlab is launched with almost no reception points to decouple the cost of the reception test from the other processes.

Since Matlab requires at least one reception point, we ran the same simulation with only this reception point. When these settings are applied to Terrain 1, Matlab took 67, 59 and 57 seconds respectively for small, medium and large angular separations. These values correspond to the cost of Matlab without RF mapping, hence without the reception test. This is typically the cost of the ray launching and of the ROI test processes, which remains unchanged with the number of reception points. Hence, all the other costs incurred by Matlab come from the reception test process, showing how complex this process is. From this finding, we can see that more than 98% of Matlab's complexity comes from its reception process. This highlights the fact that traditional RT approaches were not designed for RF mapping purposes due to the complexity of the reception test process.

On the other hand, the results show the effectiveness of our reception test scheme and that it represents the majority of the gains in terms of execution time. The results shown here over Terrain 1, also hold for Terrains 2 and 3. Overall, by combining the efficiency of Embree in performing the intersection test regardless of the number of triangles and that of our new reception test scheme, we are able to outperform existing models both in terms of execution time and map precision.

5.5.3 Sensitivity Study

Our technique being based on triangles, we study here the impact of the size of the triangles, namely the resolution step, on the accuracy of the estimates of our model. For clarity and consistency, we only show here the study performed over Terrain 2, however, the other two terrains exhibit the same patterns. Figure 5.20 is obtained from Terrain 2 with the *small* angular separation and 5 maximum number of reflections. It is a box plot that shows the distribution of our model with regards to the resolution step. This aims to obtain the variation of our model on different triangle sizes.

This simulation was carried out by choosing a set of locations and their bitrate

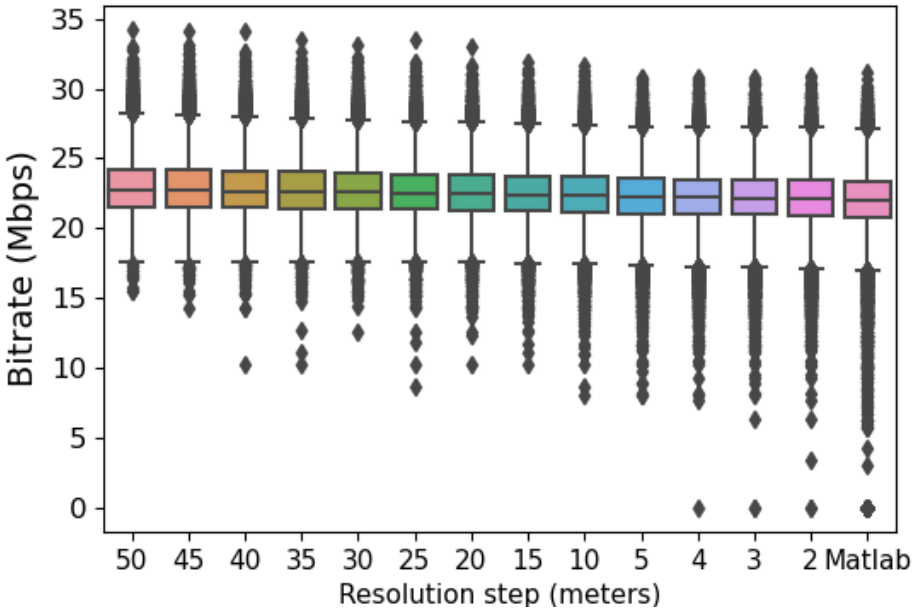


Figure 5.20: Sensitivity study on Terrain 2.

values which correspond to the bitrate of the triangle they lie in. For each resolution step, the bitrates corresponding to these points were obtained and used to plot this Figure. We have also added the Matlab distribution to this Figure in order to compare all our resolution steps with it. This makes it possible to analyze the differences between each of the resolution steps and that of Matlab. The first observation from this Figure is that all the resolution steps have the same distribution as Matlab. In addition, our model is stable to variations in the resolution step. This result is of great interest since we can choose a resolution step of 50 meters in our model without any loss in terms of accuracy while minimizing the execution time. This property shows the robustness of our model and its resilience to estimating signal power received and producing RF maps.

However, we can notice from the Figure that the smaller the resolution steps are, the lower the bitrate values we obtain, i.e., more outliers represented by dots in the box plot. This is also the case for Matlab estimates, since they consider points. Indeed, as the resolution step decreases, fewer rays are received, which leads to lower values of the bitrate. Nevertheless, our model remains stable with almost no variation in the

distributions from the largest to the smallest resolution step chosen. Therefore, our approach is almost not sensitive to the resolution step and is hence able to produce accurate maps even with large resolution steps.

5.6 Conclusion

This Chapter presents a novel RF mapping approach capable of generating accurate and precise RF maps at large-scales. Our approach outperforms existing models in both execution time and RF map precision. We prove in this Chapter that the traditional RT approaches are not adapted for RF mapping purposes and that the reception testing process used to determine the rays received by the reception points accounts for more than 98% of the total RT time in traditional approaches. To overcome this limitation, we propose a new reception test approach that exploits the geometric properties of triangles and rays and quickly determines which rays were received by which triangle. Moreover, thanks to the triangulation schemes introduced in our approach to consider the 3D topography, we propose a new ROI testing scheme that outperforms existing solutions for this process. We show that our model outperforms existing solutions by dividing the RT execution time by a factor of 50 for RF mapping purposes. Finally, we show that our approach is not sensitive to the resolution step when generating RF maps, allowing the choice of a less precise map without any loss in terms of accuracy.

Chapter 6

Conclusion and future works

6.1 Conclusion

This thesis focus was on optimizing the complexities related to the processes within RT, with a particular interest on their application for RF mapping purposes. Through the different chapters of this thesis, we proved that conventional RT methodologies are not suited for generating RF maps in large-scale scenarios. They usually suffer from a high execution time and hardware limitations that prevent them from generating such maps in large-scale complex scenarios.

One of the primary challenges highlighted in this thesis is the inefficiency of the conventional ray generation process, particularly the widely used icosahedron technique. This method, although prevalent, launches rays in all possible directions without consideration of the characteristics of the propagation environment, resulting in an unnecessary computational overhead. Moreover, existing alternatives and optimizations, while attempting to mitigate these issues, still contribute to a significant computational overhead by generating rays in all directions, lacking efficiency and adaptability.

In response to these limitations, we introduced in Chapter 3 a new solution: a site-specific ray generation technique. By leveraging information about the environ-

ment, this innovative approach optimizes the ray launching process. Our solution launches the minimum number of rays necessary to comprehensively cover the area of interest without compromising the accuracy of the received signal power estimate. Remarkably, despite launching fewer rays, compared to the state-of-the-art methods, the proposed technique maintains the accuracy of RT while significantly reducing its computational load and its memory consumption. Additionally, its remarkable speed in generating thousands of rays within seconds addresses the computational slowness of the conventional icosahedron technique.

Furthermore, we addressed in this thesis the critical challenges of generating RF maps efficiently and accurately, particularly at large scales. Our work unveiled the inefficiencies of current RT processes for this purpose, notably the reception test process, which represents a bottleneck in RF mapping due to its inherent complexities. We demonstrated that due to this complexity, current implementations of RT were unable to produce accurate and precise RF maps within reasonable timeframes.

To overcome these limitations, we introduced an alternative solution to the reception test process in Chapter 4, laying the foundation for a novel RF mapping approach. Through diverse experiments, we showcase our model's ability to consistently generate accurate and precise RF maps within practical time constraints. However, this RF mapping is only optimized for terrains exhibiting minimal altitude variation, thereby necessitating further enhancements for terrains with significant topographical diversity.

Building upon this foundational work, Chapter 5 extends the applicability of our approach to consider terrains of all types. Our RF mapping method helps to produce accurate and precise RF maps on a large-scale, surpassing existing models in both speed and precision. We proved in this Chapter the inefficiencies of RT acceleration techniques when used for RF mapping purposes. Additionally, we optimized all the processes within RT in order to solely use them for RF mapping. To achieve this, we introduced a new terrain modeling scheme based on triangulation. Leveraging this approach, we proposed a new reception technique tailored for all types of terrains and

we also enhanced the efficiency of the ROI testing. Our results demonstrate a drastic reduction in RT execution time, while maintaining its accuracy. Experimentation done over diverse terrain types showed that our approach was robust, versatile and efficient for generating large-scale RF maps.

Finally, by challenging traditional paradigms and by introducing novel techniques tailored for RF mapping purposes, this thesis not only pushes the boundaries of RT but it also paved the way for the introduction of new acceleration techniques meant to reduce the computational complexity of RT for this specific use case.

6.2 Limitations

Despite its accuracy in solving the issues related to RT complexities when used for RF mapping purposes, our solutions have some limitations.

First, the results presented in this thesis were based on simulations performed in our model and compared only against the Matlab implementation of SBR. The latter is also a simulation modeling tool and hence does not stand as a fully trustworthy ground truth model representing the real-life signal power received. Considering other RT simulation tools such as Wireless Insite [80] from Remcom and Forsk [81] from Atoll, would have added a more rigorous validation and performance evaluation.

On the other hand, a validation of our model with real-life measurement datasets was not performed. This is in fact necessary to assess the match between our model's estimation and what is really experienced by end users. This would be useful in the derivation of correlation factors that can help to move from the signal power estimated to the real signal power received while considering all the upper layers of the TCP/IP protocol stack.

Also, from Chapter 4, our new reception test algorithm only considers the circular cross-sections of rays. In reality, depending on the angle of incidence of the ray with the plane, this conic section results in either a circle or an ellipse. In some extreme

cases, mainly when rays traveled long distances before hitting the horizontal place, the difference between the ellipse and the circles can be very large. Therefore, considering further optimizations to the model that may enable the use of ellipses can help to reduce this loss in the accuracy of the model.

6.3 Future works

Having paved the way towards efficient RF map rendering in large-scale scenarios, further works can be done in order to ensure the generation of accurate and precise RF maps in reasonable time.

- **Enhancing End-User Signal Power Accuracy: Integrating Measurement Data with Ray Tracing in 4G and 5G Networks:** This is meant to ensure that the estimations done by the RT model aligns with the signal power experienced by end users. This passes by performing several real-life measurements that assess the signal power received in different network conditions within 4G and 5G networks. These measurements are meant to find correlation factors that will fill the gap between the estimations of our model and the real signal power received. In real-life scenarios, resources of a BS are shared among the different users connected to it, and these resources are re-allocated every millisecond by the BS [82]. These resources coupled with the quality of the channel decide the instant received signal power by a given end-user. On the other hand, our simulations only consider the Shannon capacity formula which assumes that a single user is connected to the BS and is having all the resources. Apart from this shift in terms of shared resources, the signal power received by end-users is also influenced by many other uncontrolled factors. The latter includes the imperfections of the physical layer and the overhead of upper layers' protocols. Also, at the moment of receiving the signal, the orientation of the mobile antenna introduces a shift between the estimated signal power and the reality [83]. Due to all these parameters, it is therefore important to compare the results of

the measurements with the ones of our model in order to calibrate the latter and ensure that an exact estimation of the received signal power is provided to end-users.

- **Integration in a cartography tool:** As mentioned in the beginning of this thesis, our goal is to provide end-users with a tool that helps them choose the best MNO depending on their areas of interest. To do this, it is important to apply our model on a country-scale for instance. This is quite challenging since it may result in further computational complexities due to this larger scale. Therefore, it is important to draw space division techniques that will allow to process in parallel city by city or region by region in order to have full coverage in the targeted country.

- **Consideration of diffraction in large-scale scenarios:** In addition to reflection, diffraction is another propagation mechanism that is often encountered in urban areas due to the large number of buildings and in rural areas due to the relief. Whenever diffraction occurs, rays are subdivided into multiple other rays that must be followed, hence adding an uncontrolled number of rays to track. This is why we decided to only consider reflection in our model, since it accounts for most of the energy carried by a ray. Still, considering diffraction will result in a more accurate model that accounts for various mechanisms when estimating the signal power received. For this, a more enhanced acceleration technique may be needed in order to account for all these increasing number of rays.

- **Coupling Ray Tracing with Artificial Intelligence to produce RF maps**: The evolution of AI has led to several works in the literature meant to accelerate RT. Some of these works run RT simulations in one area and build an AI model able to predict the path loss in different other areas [84–87]. In these works, RT is used to compute the signal power received by samples of reception points. The signal power received by these points serves as the basis to train an AI model that will be further used to estimate the signal power received by any

other reception point in that area of interest. Nevertheless, for these models to be useful and accurate enough, the RT algorithm must be performed on a large sample of reception points. As explained throughout this thesis, a large number of reception points is the main bottleneck of RT preventing the generation of RF maps in large-scale scenarios. Additionally, an AI model trained on a given propagation environment does not necessarily generalize on other terrains. Indeed, RT is said to be accurate since it accounts for every single obstacle in the propagation environment to estimate the received signal power. Having a general model meant to be applied on all types of terrains is like using an empirical model which is less accurate than the RT model.

To the best of our knowledge, none of the works in the literature associates AI within RT processes themselves in order to accelerate them for RF mapping purposes. Nevertheless, recent advances in Deep Learning are interesting paths to explore for accelerating RT processes. For instance, the Transfer Learning algorithm [88] offers models pre-trained on millions of datasets. These models are therefore very accurate and can be useful in detecting all the obstacles within areas of interest without having to retrain the model. These types of models can be particularly useful for the Ray/Object Intersection test. Indeed, an environment-aware ROI test may help to increase the speed of RT.

Furthermore, segmentation algorithms [89] used in Computer Vision have similar applications as RF mapping. These algorithms are used to classify everything present in an area of interest in a rapid fashion. These models are quite reliable in the sense that all the area is segmented without the need for much data, leading to a quite fast model. Instead of classifying objects in the area, these models can be used to show the signal power level across every single location in an area of interest in a continuous fashion. This path can be a groundbreaking revolution towards real-time RF mapping at large scale.

- **Ray Tracing and Meta Surfaces**: The 6G and beyond technologies are meant to work on Terahertz frequency bands [90, 91]. In these bands, the frequency

ranges from 0.3THz to 10THz in the electromagnetic spectrum [92], leading to the wavelength to be negligible compared to the size of obstacles. Therefore, these terahertz waves completely fade when they encounter an obstacle. To ensure coverage using this technology, an infinite number of antennas must be deployed, which is not feasible in practice. To avoid having a large number of antennas, meta-surfaces also known as Reflective Intelligent Surfaces (RIS) were introduced [93, 94]. Apart from regular surfaces that attenuate the signal and only follows the regular laws of geometrical optic and theory of diffraction, these surfaces rather reflect the signal in such a way to reach a targeted receiver despite the obstacles present in the area. In large-scale scenarios where multiple obstacles are present, it becomes quite challenging to track the multiple receivers present in the environment. To ensure signal reception by all the targeted receivers, the positioning of the RIS becomes crucial. Since RT can fully scan the environment and show areas with bad and good coverage, it is an interesting technology to help in the optimal positioning of the RIS. It can be useful to minimize the number of RIS to be deployed while maintaining comprehensive coverage.

References

[1] ARCEP. Mon reseau mobile, 2023. URL https://monreseaumobile.arcep.fr/. [Online; accessed 18-March-2024].

[2] Meiling Luo. *Indoor radio propagation modeling for system performance prediction*. Theses, INSA de Lyon, July 2013. URL https://theses.hal.science/tel-00937481.

[3] Stephen Gedney. Introduction to the finite-difference time-domain (fdtd) method for electromagnetics. *Synthesis Lectures on Computational Electromagnetics*, 6, 01 2011. doi:10.2200/S00316ED1V01Y201012CEM027.

[4] Wikipédia. Géode (géométrie) — wikipédia, l'encyclopédie libre, 2021. URL http://fr.wikipedia.org/w/index.php?title=G%C3%A9ode_(g%C3%A9om%C3%A9trie)&oldid=188739072. Last accessed 22 February 2022.

[5] Armin Tavakoli and Nicolas Gisin. The platonic solids and fundamental tests of quantum mechanics. *Quantum*, 4:293, 07 2020. doi:10.22331/q-2020-07-09-293.

[6] Wikimedia Commons. File:conic sections language neutral.svg — wikimedia commons, the free media repository, 2022. URL https://commons.wikimedia.org/w/index.php?title=File:Conic_Sections_language_neutral.svg&oldid=660011577. [Online; accessed 23-February-2024].

[7] OpenStreeMap project. Openstreetmap, 2024. URL https://www.openstreetmap.fr/. Last accessed 26 February 2024.

[8] Institut national de l'information géographique et forestière. geoportail.gouv.fr: le portail national de la connaissance du territoire mis en œuvre par l'ign, 2023. URL https://www.geoportail.gouv.fr/. Last accessed 19 March 2024.

[9] Institut national de l'information géographique et forestière. Bd topo, version 3.3, 2024. URL https://geoservices.ign.fr/sites/default/files/2024-02/DC_BDTOPO_3-3_0.pdf. Last accessed 26 February 2024.

[10] Agence Nationale des Fréquences. Cartoradio, the map of radio sites and wave measurements, 2024. URL https://www.cartoradio.fr/. Last accessed 26 February 2024.

[11] STATISTA. Mobile internet usage worldwide - statistics & facts, December 18, 2023. URL https://www.statista.com/topics/779/mobile-internet/#topicOverview. [Online; accessed 11-January-2024].

[12] Ookla. Plan, optimize, and monitor your 5g network with qos and qoe data, 2023. URL https://www.ookla.com/articles/plan-optimize-monitor-5g-q4-2023. [Online; accessed 07-December-2023].

[13] AT&T. Wireless coverage map, 2023. URL https://www.att.com/maps/wireless-coverage.html. [Online; accessed 07-December-2023].

[14] Telecom Italia. Copertura mobile tim, 2024. URL https://www.tim.it/fisso-e-mobile/mobile/mappa-copertura-mobile. [Online; accessed 11-January-2024].

[15] Orange. La carte de couverture du réseau mobile à mon adresse, 2023. URL https://reseaux.orange.fr/les-cartes-de-couverture/mobile-3g-4g-5g. [Online; accessed 11-January-2024].

[16] ZoneAdsl. Carte réseau mobile, 2024. URL https://www.zoneadsl.com/couverture-mobile/carte. [Online; accessed 11-January-2024].

[17] Ariase. Carte de couverture mobile, 2024. URL https://www.ariase.com/mobile/carte-couverture-mobile. Last accessed 01 March 2024.

[18] DegroupTest. La carte des débits internet, 2024. URL https://www.degrouptest.com/speedtest. Last accessed 01 March 2024.

[19] ZoneADSL & Fibre. Qui sommes-nous?, 2024. URL https://www.zoneadsl.com/qui-nous-sommes.html. [Online; accessed 11-January-2024].

[20] ARCEP. La qualité des services mobiles, October 26, 2023. URL https://www.arcep.fr/la-regulation/grands-dossiers-reseaux-mobiles/la-qualite-des-services-mobiles.html. [Online; accessed 11-January-2024].

[21] nperf. 3g / 4g / 5g bitrates map, 2024. URL https://www.nperf.com/en/map/FR/-/-/download/. [Online; accessed 11-January-2024].

[22] Zaid Al-Daher, Leonidas P. Ivrissimtzis, and Akram Hammoudeh. Outdoor coverage & capacity estimation in small cells with a 3d ray tracing model. In *2013 Loughborough Antennas & Propagation Conference (LAPC)*, pages 31–36, 2013. doi:10.1109/LAPC.2013.6711846.

[23] Muhammad Usman Sheikh, Kalle Ruttik, Norshaida Saba, Edward Mutafungwa, Riku Jäntti, and Jyri Hämäläinen. Analysis of drone propagation with ray tracing from sub-6 ghz upto terahertz frequencies in a real world urban environment. In *Proc. of ICOIN*, pages 169–174, 2021. doi:10.1109/ICOIN50884.2021.9333892.

[24] Przemyslaw Wozmca and Lukasz Kulas. Influence of a radio frequency on rf fingerprinting accuracy based on ray tracing simulation. In *Eurocon 2013*, pages 202–206, 2013. doi:10.1109/EUROCON.2013.6624987.

[25] Sinem Bozkurt, Ahmet Yazıcı, Serkan Gunal, and Ugur Yayan. A survey on rf mapping for indoor positioning. In *2015 23nd Signal Processing and Communications Applications Conference (SIU)*, pages 2066–2069, 2015. doi:10.1109/SIU.2015.7130275.

[26] Yoann Corre, Thierry Tenoux, Julien Stéphan, Florian Letourneux, and Yves Lostanlen. Analysis of outdoor propagation and multi-cell coverage from ray-based simulations in sub-6ghz and mmwave bands. In *2016 10th European Conference on Antennas and Propagation (EuCAP)*, pages 1–5, 2016. doi:10.1109/EuCAP.2016.7481504.

[27] Moonkyu Jo, Xilei Chen, Kyunglin Ryu, and Kwang Soon Kim. Map-based system-level performance of full-duplex in small cell networks. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 870–872, 2018. doi:10.1109/ICTC.2018.8539452.

[28] H. Suzuki. Accurate and efficient prediction of coverage map in an office environment using frustum ray tracing and in-situ penetration loss measurement. In *The 57th IEEE Semiannual Vehicular Technology Conference, 2003. VTC 2003-Spring.*, volume 1, pages 236–240 vol.1, 2003. doi:10.1109/VETECS.2003.1207538.

[29] Olaonipekun Oluwafemi Erunkulu, Adamu Murtala Zungeru, Caspar K. Lebekwe, and Joseph M. Chuma. Cellular communications coverage prediction techniques: A survey and comparison. *IEEE Access*, 8:113052–113077, 2020. doi:10.1109/ACCESS.2020.3003247.

[30] Zhengqing Yun and Magdy F. Iskander. Ray tracing for radio propagation modeling: Principles and applications. *IEEE Access*, 3:1089–1100, 2015. doi:10.1109/ACCESS.2015.2453991.

[31] Jan Reitz Moritz Alfrink and Jürgen Roßmann. Improving ray tracing based radio propagation model performance using spatial acceleration structures. In *Proceedings of the 17th ACM International Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet '21)*, Alicante, Spain. ACM, New York, NY, USA, 8 pages, November 2021. doi:https://doi.org/10.1145/3479242.3487318.

[32] Lawrence G. Sayer. *Enhanced Radio Propagation Modelling for Future Wireless Networks*. PhD thesis, University of Bristol, May 2020.

[33] M.F. Catedra, J. Perez, F. Saez de Adana, and O. Gutierrez. Efficient ray-tracing techniques for 3d analyses of propagation in mobile communications: application to picocell and microcell scenarios. *IEEE Antennas and Propagation Magazine*, 40(2):15–28, 1998. doi:10.1109/74.683539.

[34] Zhengqing Yun, Zhijun Zhang, and M.F. Iskander. A ray-tracing method based on the triangular grid approach and application to propagation prediction in urban environments. *IEEE Transactions on Antennas and Propagation*, 50(5): 750–758, 2002. doi:10.1109/TAP.2002.1011243.

[35] K. Rizk, J.-F. Wagen, and F. Gardiol. 2d ray-tracing modeling for propagation prediction in microcellular environments. *IEEE Transactions on Vehicular Technology*, 46(2):508–518, 1997. doi:10.1109/25.580789.

[36] Bernard Tamba Sandouno, Yamen Alsaba, Chadi Barakat, Walid Dabbous, and Thierry Turletti. A novel approach to mobile outdoor qos map generation. In *2023 IEEE WCNC*, pages 1–6, 2023. doi:10.1109/WCNC55385.2023.10118977.

[37] Ingo Wald, Sven Woop, Carsten Benthin, Gregory S. Johnson, and Manfred Ernst. Embree: A kernel framework for efficient cpu ray tracing. *ACM Trans. Graph.*, 33(4), jul 2014. ISSN 0730-0301. doi:10.1145/2601097.2601199. URL https://doi.org/10.1145/2601097.2601199.

[38] J Clerk Maxwell. A dynamical theory of the electromagnetic field. *Philosophical Transactions of the Royal Society of London*, 155:459–512, 1865.

[39] M. Born and E. Wolf. *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Cambridge University Press, Cambridge, U.K., 7th edition, 1999.

[40] Joseph B. Keller. Geometrical theory of diffraction∗. *J. Opt. Soc. Am.*, 52(2):

116–130, Feb 1962. doi:10.1364/JOSA.52.000116. URL https://opg.optica.org/abstract.cfm?URI=josa-52-2-116.

[41] Tammam A. Benmus, Rabie Abboud, and Mustafa Kh. Shatter. Neural network approach to model the propagation path loss for great tripoli area at 900, 1800, and 2100 mhz bands. In *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, pages 793–798, 2015. doi:10.1109/STA.2015.7505236.

[42] H.T. Friis. A note on a simple transmission formula. *Proceedings of the IRE*, 34(5):254–256, 1946. doi:10.1109/JRPROC.1946.234568.

[43] 3GPP. Study on 3d channel for lte. *document 3GPP TR 36.873, 3rd Generation Partnership Project V12.7.0*, 2017.

[44] 3GPP. Study on channel model for frequencies from 0.5 to 100 ghz. *3GPP TR38.901, 3rd Generation Partnership Project V14.1.1*, 2017.

[45] Zahera Naseem, Iram Nausheen Ahmed, and Zahwa Mirza. Propagation models for wireless communication system irjet journal propagation models for wireless communication system. *International Journal of Research in Engineering and Technology*, Volume: 05 Issue: 01 — Jan-2018:237–242, 10 2021.

[46] Kristen Rohlfs and T. L. Wilson. *Tools of Radio Astronomy*. Springer Science and Business Media, 4th edition, 2013. ISBN 978-3662053942.

[47] M. Hata. Empirical formula for propagation loss in land mobile radio services. *IEEE Transactions on Vehicular Technology*, 29(3):317–325, 1980. doi:10.1109/T-VT.1980.23859.

[48] Yoshihisa Okumura. Field strength and its variability in vhf and uhf land-mobile radio service. *Review of the Electrical communication Laboratory*, 16(9), 1968.

[49] F. Ikegami. Mobile radio communications in japan. *IEEE Transactions on Communications*, 20(4):738–746, 1972. doi:10.1109/TCOM.1972.1091218.

[50] N. Yoshikawa and T. Nomura. On the design of a small zone land mobile radio system in uhf band. *IEEE Transactions on Vehicular Technology*, 25(3):57–67, 1976. doi:10.1109/T-VT.1976.23620.

[51] Theodore S. Rappaport, Yunchou Xing, George R. MacCartney, Andreas F. Molisch, Evangelos Mellios, and Jianhua Zhang. Overview of millimeter wave communications for fifth-generation (5g) wireless networks—with a focus on propagation models. *IEEE Transactions on Antennas and Propagation*, 65(12): 6213–6230, 2017. doi:10.1109/TAP.2017.2734243.

[52] T.S. Rappaport. *Wireless Communications: Principles And Practice, 2/E*. Pearson Education, 2010. ISBN 9788131731864. URL https://books.google.fr/books?id=VmPT8B-5_tAC.

[53] M.K. Simon and M.S. Alouini. *Digital Communication over Fading Channels*. Wiley Series in Telecommunications and Signal Processing. Wiley, 2005. ISBN 9780471715238. URL https://books.google.fr/books?id=OYrDN0Q6BacC.

[54] Hassan Khosravian-Arab, Mehdi Dehghan, and M. R. Eslahchi. Generalized bessel functions: Theory and their applications. *Mathematical Methods in the Applied Sciences*, 40, 07 2017. doi:10.1002/mma.4463.

[55] Wikipedia contributors. Bessel function — Wikipedia, the free encyclopedia, 2024. URL https://en.wikipedia.org/w/index.php?title=Bessel_function&oldid=1194456667. [Online; accessed 6-February-2024].

[56] Technologies Innovation Directorate-General for Information Society: Telecommunications, Markets, Exploitation of Research, and European Commission. *COST Action 231: Digital Mobile Radio Towards Future Generation Systems: Final Report*. OPOCE, 1999. URL https://books.google.fr/books?id=G1RQyQEACAAJ.

[57] Michael Bartsch, Micha Dehler, Martin Dohlus, Frank Ebeling, Peter Hahne, Reinhard Klatt, Frank Krawczyk, Michaela Marx, Zhang Min, Thomas Pröpper,

Dietmar Schmitt, Petra Schütt, Bernhard Steffen, Bernhard Wagner, Thomas Weiland, Susan G. Wipf, and Heike Wolter. Solution of maxwell's equations. *Computer Physics Communications*, 73(1):22–39, 1992. ISSN 0010-4655. doi:https://doi.org/10.1016/0010-4655(92)90026-U. URL https://www.sciencedirect.com/science/article/pii/001046559290026U.

[58] Liping Gao. Splitting finite element methods for time dependent maxwell's equations in 2d. In *2011 IEEE International Conference on Microwave Technology & Computational Electromagnetics*, pages 395–397, 2011. doi:10.1109/ICMTCE.2011.5915542.

[59] N. N. Makwana and Avijit Chatterjee. Fast solution of time domain maxwell's equations using large time steps. In *2015 IEEE International Conference on Computational Electromagnetics*, pages 330–332, 2015. doi:10.1109/COMPEM.2015.7052651.

[60] Xing Li, Li Xu, and Bin Li. The hybridizable discontinuous galerkin time domain method to solve the 3d maxwell's equations. In *2018 IEEE International Conference on Computational Electromagnetics (ICCEM)*, pages 1–3, 2018. doi:10.1109/COMPEM.2018.8496693.

[61] Kane Yee. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14(3):302–307, 1966. doi:10.1109/TAP.1966.1138693.

[62] K.S. Yee and J.S. Chen. The finite-difference time-domain (fdtd) and the finite-volume time-domain (fvtd) methods in solving maxwell's equations. *IEEE Transactions on Antennas and Propagation*, 45(3):354–363, 1997. doi:10.1109/8.558651.

[63] Arseny Fedorov, Christina Stepa, Anna Korolkova, Migran Gevorkyan, and Dmitry Kulyabov. Methodological derivation of the eikonal equation. *Discrete and Continuous Models and Applied Computational Science*, 31:399–418, 12 2023. doi:10.22363/2658-4670-2023-31-4-399-418.

[64] Stephen Kasdorf, Blake Troksa, Cam Key, Jake Harmon, and Branislav M. Notaroš. Advancing accuracy of shooting and bouncing rays method for ray-tracing propagation modeling based on novel approaches to ray cone angle calculation. *IEEE Transactions on Antennas and Propagation*, 69(8):4808–4815, 2021. doi:10.1109/TAP.2021.3060051.

[65] Mathworks. Matlab ray tracing, 2023. URL https://fr.mathworks.com/help/antenna/ref/txsite.sinr.html. Last accessed 14 December 2023.

[66] Álvaro González. Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. *Mathematical Geosciences*, 42(1):49–64, 2010. ISSN 1874-8953. doi:10.1007/s11004-009-9257-x. URL https://doi.org/10.1007/s11004-009-9257-x.

[67] Zhengqing Yun, M.F. Iskander, and Zhijun Zhang. Fast ray tracing procedure using space division with uniform rectangular grid. *Electron. Lett*, 36(10):895–897, May 2000.

[68] Chiya Saeidi and Farrokh Hodjatkashani. Modified angular z-buffer as an acceleration technique for ray tracing. *IEEE Transactions on Antennas and Propagation*, 58(5):1822–1825, 2010. doi:10.1109/TAP.2010.2044342.

[69] Jingyu LV, Yinghua Wang, Jialing Huang, Yuxiao Li, Jie Huang, and Cheng-Xiang Wang. An improved triangular facets based angular z-buffer algorithm for im ray tracing channel modeling. In *2022 IEEE/CIC International Conference on Communications in China (ICCC)*, pages 1050–1056, 2022. doi:10.1109/ICCC55456.2022.9880630.

[70] Ingo Wald. On fast construction of sah-based bounding volume hierarchies. In *2007 IEEE Symposium on Interactive Ray Tracing*, pages 33–40, 2007. doi:10.1109/RT.2007.4342588.

[71] Zhengqing Yun, M.F. Iskander, and Zhijun Zhang. Development of a new shooting-and-bouncing ray (sbr) tracing method that avoids ray double count-

ing. In *IEEE Antennas and Propagation Society International Symposium. 2001 Digest. Held in conjunction with: USNC/URSI National Radio Science Meeting (Cat. No.01CH37229)*, volume 1, pages 464–467 vol.1, 2001. doi:10.1109/APS.2001.958892.

[72] S.Y. Seidel and T.S. Rappaport. Site-specific propagation prediction for wireless in-building personal communication system design. *IEEE Transactions on Vehicular Technology*, 43(4):879–891, 1994. doi:10.1109/25.330150.

[73] Vittorio Degli Esposti. Ray tracing: techniques, applications and prospect. In *2020 International Symposium on Antennas and Propagation (ISAP)*, pages 307–308, 2021. doi:10.23919/ISAP47053.2021.9391121.

[74] Jean-François Dufourd. Polyhedra genus theorem and euler formula: A hypermap-formalized intuitionistic proof. *Theoretical Computer Science*, 403(2): 133–159, 2008. ISSN 0304-3975. doi:https://doi.org/10.1016/j.tcs.2008.02.012. URL https://www.sciencedirect.com/science/article/pii/S0304397508001187.

[75] Wikipédia. Angular distance — wikipedia, l'encyclopedie libre, 2023. URL https://en.wikipedia.org/wiki/Angular_distance. Last accessed 26 March 2024.

[76] Institut national de l'information géographique et forestière. Calcul altimétrique (rest), 2021. URL https://geoservices.ign.fr/documentation/services/api-et-services-ogc/calcul-altimetrique-rest. Last accessed 02 February 2022.

[77] Agence Nationale des Fréquences. Bienvenue sur le portail open data de l'anfr, 2024. URL https://data.anfr.fr/. Last accessed 26 February 2024.

[78] Simena Dinas and Jose Bañón. A review on delaunay triangulation with application on computer vision. *IJCSE - International Journal of Computer Science and Engineering*, 3:9–18, 03 2014.

[79] Sean Gillies and Shapely contributors. Shapely: manipulation and analysis of geometric objects, 2023. URL https://shapely.readthedocs.io/en/stable/manual.html. [Online; accessed 18-December-2023].

[80] Remcom. Wireless insite 3d wireless propagation software, 2024. URL https://www.remcom.com/wireless-insite-propagation-software. Last accessed 29 February 2024.

[81] Forsk. Propagation modelling, 2024. URL https://www.forsk.com/propagation-modelling. Last accessed 29 February 2024.

[82] Mamoutou Diarra, Walid Dabbous, Amine Ismail, and Thierry Turletti. Cross-layer loss discrimination algorithms for mec in 4g networks. In *2021 IEEE 22nd International Conference on High Performance Switching and Routing (HPSR)*, pages 1–6, 2021. doi:10.1109/HPSR52026.2021.9481843.

[83] Yanis Boussad, Mohamed Naoufal Mahfoudi, Arnaud Legout, Leonardo Lizzi, Fabien Ferrero, and Walid Dabbous. Evaluating smartphone accuracy for rssi measurements. *IEEE Transactions on Instrumentation and Measurement*, 70: 1–12, 2021. doi:10.1109/TIM.2020.3048776.

[84] Sotirios K. Goudos, Georgia Athanasiadou, George V. Tsoulos, and Vasileios Rekkas. Modelling ray tracing propagation data using different machine learning algorithms. In *2020 14th European Conference on Antennas and Propagation (EuCAP)*, pages 1–4, 2020. doi:10.23919/EuCAP48036.2020.9135639.

[85] Aristeidis Seretis, Takahiro Hashimoto, Kun Zeng, and Costas D. Sarris. Ray-tracing driven ann propagation models for indoor environments at 28 ghz. In *2020 IEEE International Symposium on Antennas and Propagation and North American Radio Science Meeting*, pages 1029–1030, 2020. doi:10.1109/IEEECONF35879.2020.9330037.

[86] Fernado Pérez Fontán Ignacio Fernández Anitzine, Juan Antonio Romo Argota. Influence of training set selection in artificial neural network-based propagation

path loss predictions. *International Journal of Antennas and Propagation*, 2012. doi:10.1155/2012/351487.

[87] Sotirios P. Sotiroudis, Sotirios K. Goudos, and Katherine Siakavara. Deep learning for radio propagation: Using image-driven regression to estimate path loss in urban areas. *ICT Express*, 6(3):160–165, 2020. ISSN 2405-9595. doi:https://doi.org/10.1016/j.icte.2020.04.008. URL https://www.sciencedirect.com/science/article/pii/S2405959520300898.

[88] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *CoRR*, abs/1911.02685, 2019. URL http://arxiv.org/abs/1911.02685.

[89] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[90] W. Jiang, Q. M. Zhang, J. He, M. A. Habibi, S. Melnyk, M. El-Absi, B. Han, M. D. Renzo, H. D. Schotten, F. Luo, T. S. El-Bawab, M. Juntti, M. Debbah, and V. C. M. Leung. Terahertz communications and sensing for 6g and beyond: a comprehensive view. *preprint*, 2023. doi:10.36227/techrxiv.23531022.v1.

[91] Akram Shafie, Nan Yang, Chong Han, Josep Miquel Jornet, Markku Juntti, and Thomas Kurner. Terahertz communications for 6g and beyond wireless networks: Challenges, key advancements, and opportunities, 2022.

[92] Ashish Y. Pawar, Deepak D. Sonawane, Kiran B. Erande, and Deelip V. Derle. Terahertz technology and its applications. *Drug Invention Today*, 5(2): 157–163, 2013. ISSN 0975-7619. doi:https://doi.org/10.1016/j.dit.2013.03.009. URL https://www.sciencedirect.com/science/article/pii/S0975761913000264.

[93] Jialing Huang, Cheng-Xiang Wang, Yingzhuo Sun, Jie Huang, and Fu-Chun Zheng. A novel ray tracing based 6g ris wireless channel model and ris de-

ployment studies in indoor scenarios. In *2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 884–889, 2022. doi:10.1109/PIMRC54779.2022.9977575.

[94] Christoph Herold and Thomas Kürner. A concept for the efficient integration of reconfigurable intelligent surfaces into a ray tracing framework. In *2023 48th International Conference on Infrared, Millimeter, and Terahertz Waves (IRMMW-THz)*, pages 1–2, 2023. doi:10.1109/IRMMW-THz57677.2023.10299214.

[95] Zhengqing Yun and Magdy F. Iskander. Development of a practical ray-tracing program for propagation modeling. In *2021 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (APS/URSI)*, pages 1879–1880, 2021. doi:10.1109/APS/URSI47566.2021.9703773.

[96] Turf. Getting started with turf.js, 2024. URL https://turfjs.org/getting-started. Last accessed 28 March 2024.

[97] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 5.6.1 edition, 2024. URL https://doc.cgal.org/5.6.1/Manual/packages.html.

[98] Institut national de l'information géographique et forestière. Base de données topologique, 2021. URL https://geoservices.ign.fr/documentation/donnees/vecteur/bdtopo. Last accessed 14 December 2023.

[99] Geopandas. Reading and writing files, 2024. URL https://geopandas.org/en/stable/docs/user_guide/io.html. Last accessed 28 March 2024.

[100] GeoPandas project. Geopandas, 2022. URL https://geopandas.org/. Last accessed 28 March 2024.

[101] International Telecommunications Union Radiocommunication Sector. Effects of building materials and structures on radiowave propagation above about

100mhz. Technical Report Recommendation P.2040, ITU-R, August 23 2023. URL https://www.itu.int/rec/R-REC-P.2040/en.

[102] Geopandas developpers. Documentation, 2022. URL https://geopandas.org/en/stable/docs.html. Last accessed 29 March 2024.

[103] Numpy developpers. Numpy user guide, 2022. URL https://numpy.org/doc/stable/user/index.html. Last accessed 29 March 2024.

[104] Rob Story. Getting started, 2013. URL https://python-visualization.github.io/folium/latest/getting_started.html. Last accessed 29 March 2024.

[105] QGIS. Qgis, a free and open source geographic information system, 2024. URL https://www.qgis.org/en/site/. Last accessed 29 March 2024.

[106] ESRI. Arcgis overview, 2024. URL https://www.esri.com/en-us/arcgis/products/arcgis-online/overview. Last accessed 29 March 2024.

Appendix A

Ray Tracing implementation from scratch

In order to be able to enhance the processes involved in Ray Tracing by adapting them for their use for RF mapping purposes, we built an in-house Ray Tracer. Implementing this tool gave us hands on RT parameters and helped us to fine-tune them and were hence able to provide all the groundbreaking optimizations presented in this thesis.

In this appendix, we provide the details about the implementation of this RT tool and how from scratch one can build such a tool instead of only relying on current implementations. This is motivated by the fact that no work in the literature explains this implementation in a programming point of view. The only work in the literature is from [95] where a simple description of RT's implementation is provided. Here instead, we provide detailed information about implementing the Ray Tracer from scratch, by digging in depth into all the implementation of its processes. We implemented our Ray Tracer using the Python programming language, however all the concepts explained here hold for all programming languages.

Designing a Ray Tracer, passes by first modeling the concepts related to RT. Therefore in Section A.1 we present how RT's concepts can be modeled in a programming point of view. Afterwards, one has to implement all the processes of RT. The imple-

mentation of the Ray generation process is given in Section A.2, the bouncing process as well as the Ray tracking is provided in Section A.3. Finally, the Ray reception test process and the power computation are provided in Section A.4. The summary of this appendix is provided in Section A.5.

A.1 Ray Tracing's concepts

A.1.1 Antenna

Antennas are characterized by different parameters: geographical coordinates, frequency, bandwidth and gain. In a programming language, an antenna can be modeled as an array of size 6 as shown below.

$$Ant = [Latitude, Longitude, Height, Frequency, Bandwidth, Gain] \tag{A.1}$$

In the case where multiple antennas are present in the environment, the array becomes a 2D array of dimensions $N \times 6$, where N represents the number of antennas. In this case, the first element of the array corresponds to the first antenna and the N^{th} element corresponds to the N^{th} antenna. To ease this operation, one may add an indexing element to characterize each antenna. This is particularly helpful to determine from which antenna the ray is being received. Therefore, the index of the i^{th} antenna will be "i" and so on, leading to a $N \times 7$ 2D array. In Python, one may use either a *List* or an *array* to create this array of antennas.

A.1.2 Buildings

Another fundamental element in RT is building, since they are the ones that constitute obstacles. In RT, buildings can be modeled by the shapes of their rooftops and their heights, leading to a 2.5D representation of them. Considering every single details in to the building leads to a complex data structure to be managed as explained in Chapter 5.

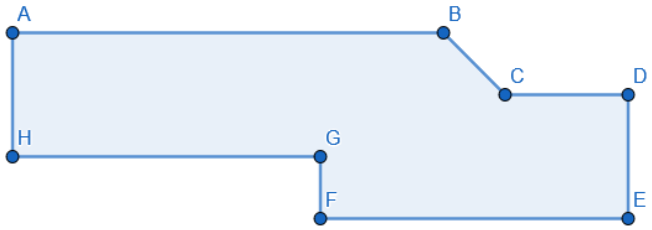


Figure A.1: Representation of a building's rooftop

Consider the building highlighted in Figure A.1. The geometry of its rooftop is simply a polygon composed of vertices linked together by edges. This polygon is defined by the geographical coordinates (latitude and longitude) of its vertices (A-B-C-D-E-F-G-H-A). Note here that, by definition a polygon has a closed-shaped, i.e., the first and the last vertices are the same. In addition to this geometry, a building is also identified by its height, its material type, and an index identifying it among other buildings. This typically corresponds to a list composed of another list and other parameters.

To represent the geometry in Figure A.1 as a Polygon, the use of a geometric library is necessary. This helps to account for the geometry as an obstacle instead of a simple vector. In Python, Shapely [79] is a library that eases the manipulation of geometries. In JavaScript, the turf.js [96] also offers this possibility. In C++, the CGAL [97] library is also necessary to perform this operation.

Instead of doing this modeling per-building, the buildings' data can be directly obtained from sources like OpenStreetMap [7] and IGN [98]. All these sources provide datasets in various Geographic Information System (GIS) data formats such as geojson, shp, and osm. A complete list of these data formats and the way to use them is detailed in [99]. The datasets taken from these sources directly contain the geometries of the buildings, thus removing the need to personally model the buildings. GIS tools like Geopandas [100] in Python help to load these datasets and offer various geometrical operations that ease the manipulation of buildings as polygons. Figure A.2 shows an example of buildings representation using the Geopandas library.

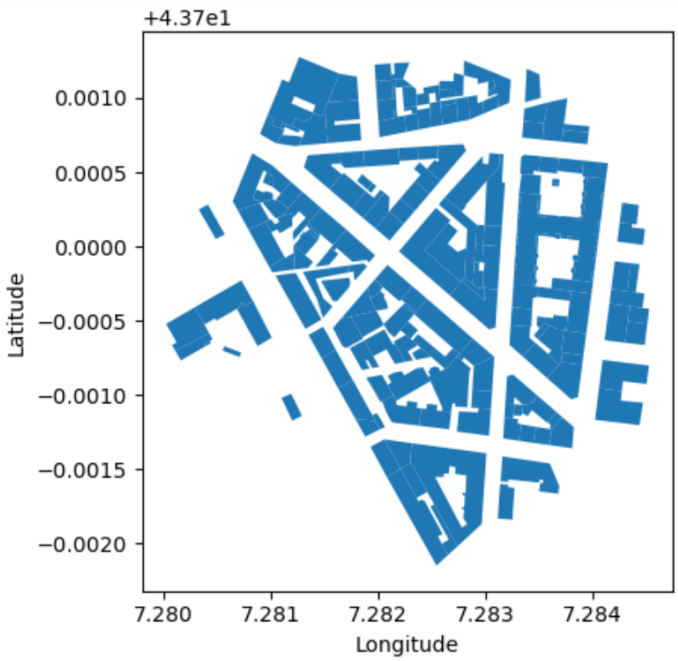


Figure A.2: Buildings represented as polygons using Geopandas

Furthermore, The Geopandas representation of some of these buildings is shown in Figures A.3, where *epsilon* and *mu* are the characteristics of the building material. A list of material electric and magnetic characteristics is given by ITU in [101].

	height	geometry	Building	epsilon	mu
0	13.4	POLYGON Z ((7.28246 43.69964 13.40000, 7.28235...	0	6.383915	8.353982
1	13.4	POLYGON Z ((7.28232 43.69973 13.40000, 7.28222...	1	8.503857	1.642572
2	13.4	POLYGON Z ((7.28289 43.69938 13.40000, 7.28276...	2	4.163493	6.821250
3	29.2	POLYGON Z ((7.28267 43.69967 29.20000, 7.28220...	3	7.661288	9.037170
4	13.4	POLYGON Z ((7.28235 43.69956 13.40000, 7.28234...	4	4.286640	8.194096
5	19.3	POLYGON Z ((7.28234 43.69954 19.30000, 7.28229...	5	5.677740	0.412107
6	29.0	POLYGON Z ((7.28277 43.69976 29.00000, 7.28278...	6	7.040668	8.838067
7	15.4	POLYGON Z ((7.28228 43.69949 15.40000, 7.28237...	7	9.893301	1.272815
8	26.9	POLYGON Z ((7.28193 43.69964 26.90000, 7.28203...	8	2.479425	1.359871
9	15.0	POLYGON Z ((7.28266 43.70016 15.00000, 7.28274...	9	9.815415	0.549075

Figure A.3: Geopandas data structure for modeling buildings.

A.1.3 Ray

In Ray Tracing, rays are the ones emitted and received by antennas. They interact with buildings and are the ones that are received by the reception point. Therefore, they represent the fundamental unit of RT as detailed in Section 2.3.3.2. In a programming point of view, a ray between a transmitting and a receiving antenna is a set of successive lines. In another words, a ray is a sequence of successive points $X_1, X_2, \ldots, X_n$ such that two successive points are joined by a line. A point X_i is described by a 1×5 vector which is $[Lat_i, Long_i, Elevation_i, Distance_{(i-1,i)}, Building]$. The first three elements of the vector are the geographical coordinates of the point, *Distance* is the distance between the previous point and the current point, *Building* is the index of the building on which the point lies in. In Figure A.4 for example, this ray from Tx to Rx is a sequence of three points forming two lines. Note here that the point at Tx has a distance of 0 because it has not moved and also the point at Rx has *None* as building index since it is not on a building. In the example provided in Figure A.4, the vector representation in a programming language is given below, where *None* is replaced by -1 to have items of same type in the array.

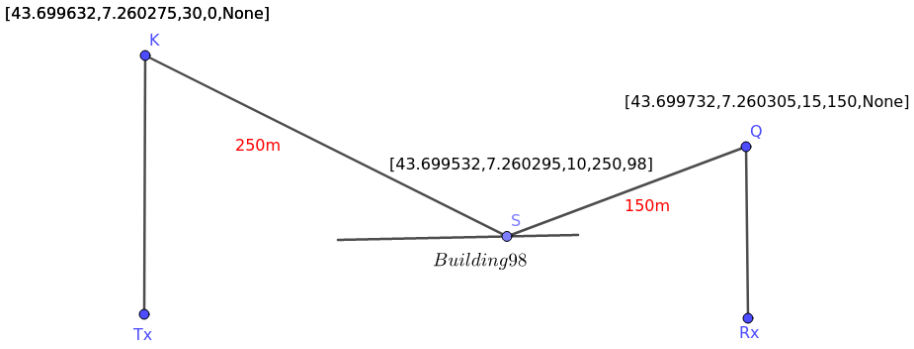


Figure A.4: Rays vectors representation

$$\begin{aligned}\text{Ray} = \big[&[43.699632, 7.260275, 30, 0, -1],\\ &[43.699532, 7.260295, 10, 250, 98],\\ &[43.699732, 7.260305, 15, 150, -1]\big]\end{aligned}$$

A.2 Ray Generation

Ray generation consists of launching rays from an antenna in order to fully cover its 3D radiation pattern. Different techniques have been introduced in the literature to perform this operation: our site-specific solution and the traditional icosahedron technique both presented in Chapter 3. In either of these techniques, the goal is to find the direction of propagation of all the rays as well as the maximum angular separation with their neighbors.

The algorithm for our site-specific ray generation is given in Algorithm 2. This algorithm is easy to implement and can be done in any programming language. The algorithm returns the direction and the maximum angular separations of all the rays, which are useful information for the Ray Object Intersection test and the reception test processes respectively.

On the other hand, the icosahedron technique is much more complicated to implement. Nevertheless, the algorithm in Section 3.2 can be implemented as a programming language function or class (in Object Oriented Programming languages) that takes a parameter or attribute n, where n is the tessellation frequency. The output of this algorithm as explained in Section 3.2 is the direction and the maximum angular separations of all the rays.

Once the directions of propagation of all the rays are found, they can be launched in the area of interest and their interaction with all the obstacles is tracked. Rays can be launched at a certain distance, d, from the antenna's geographical location $\mathbf{X_A} = [Lat_{Ant},\ Long_{Ant},\ Elevation_{Ant}]^T$, which is a column vector. The geographical coordinates of the ray (also a column vector), $\mathbf{X_R} = [Lat_{Ray},\ Long_{Ray},\ Elevation_{Ray}]^T$, at that distance is given by Equation (A.2), with $\mathbf{u} = (u_1, u_2, u_3)^T$ being the unit vector of the direction of propagation of the ray.

$$\mathbf{X_R} = \mathbf{X_A} + d \cdot \mathbf{u} \tag{A.2}$$

Since the longitude and the latitude are in decimal degrees bounded respectively by $\pm 180°$ and $\pm 90°$, the distance must also be converted to decimal degrees to ensure correctness. Approximately, $1°$ corresponds to 111km in latitude. For the longitude instead, this value is multiplied by the cosine of the latitude. Therefore, Equation (A.2) becomes Equation (A.3), with $\cdot$ being the dot product.

$$\begin{pmatrix} Lat_{Ray} \\ Long_{Ray} \\ Elevation_{Ray} \end{pmatrix} = \begin{pmatrix} Lat_{Ant} \\ Long_{Ant} \\ Elevation_{Ant} \end{pmatrix} + d \begin{pmatrix} \frac{1}{111} \\ \frac{\cos(Lat_{Ray})}{111} \\ 1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \tag{A.3}$$

Figure A.5 is an example of rays launched at a certain distance d of an antenna.

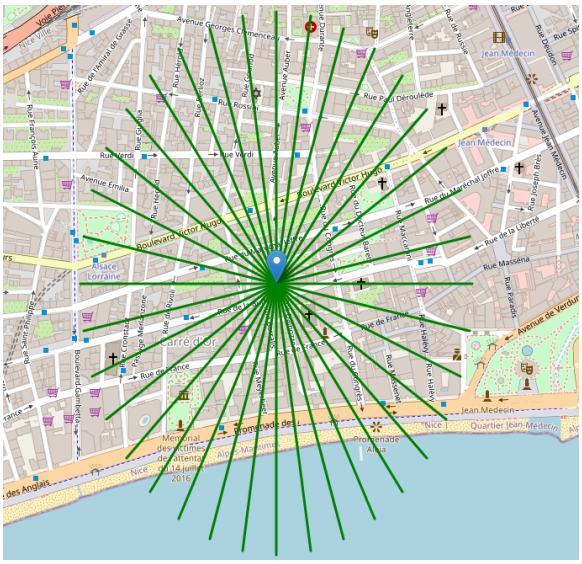


Figure A.5: Rays launched from a transmitter in 2D plane

A.3 Ray Bouncing

The bouncing process starts by choosing the maximum number of reflections K allowed for rays. For each reflection, rays are launched and their behavior is observed. In a nutshell, whenever a ray is launched, bouncing consists of checking whether or not a ray hits a building or the ground. This process is repeated K times. In order to

determine if a ray hits a building, its intersection with the building's database is done using the geometrical library used to manage buildings. In Python, this intersection test can be done using the Geopandas library. This test consists of checking all the buildings that are in a ray's path during its journey from its previous endpoint to its current endpoint when launched using Equation (A.3). Several Geopandas operations allow to do this: *overlaps, intersection,...* A complete list of these operations is found in the Geopandas documentation [102]. Once all these buildings are provided by Geopandas, a test is performed afterwards with the heights of these buildings to determine whether they are obstructions to the signal or not. In the case of obstructions, the first building that obstructs the signal is considered as being hit by the ray, and the coordinates of the ray are updated accordingly. On the other hand, if no building is hit, the intersection of the ray with the ground is tested as follows. Let d_i be the distance traveled by the ray between two points. The distance the ray should travel in order to hit the ground is: $d_G = \frac{-Z_A}{u_3}$, where Z_A is the height of the ray in its first endpoint and u_3 is the z-component of the unit vector of its direction of propagation. Finally, the ray hits the ground if $d_i \geq d_G$. If this condition is not satisfied it means that the ray hits no obstacle.

Once the behavior of all the rays is being observed, the next step is to compute the subsequent direction of rays that went through reflection. Equation 2.44 introduced in Section 2.3.3.2 gives the direction of the reflected ray as a function of its direction and the normal to the plane hit by the ray. By applying this Equation to all the rays that encounter obstacles, the process continues in a recursive fashion until the maximum number of reflections is reached. In case of no intersection with either buildings and the ground, the normal to the plane $\mathbf{N}$ in Equation (2.44) is $\mathbf{N} = [0, 0, 0]$, otherwise it is $\mathbf{N} = [0, 0, 1]$ when the ground is hit. In case of intersection with a building, the normal is found by performing the cross product between two non-orthogonal vectors on the building's facet hit by the ray. Figure A.6 is an example of plane, with two random non-orthogonal vectors $\mathbf{P}$ and $\mathbf{Q}$ belonging to the plane. The unit vector $\mathbf{N}$

is therefore given by Equation (A.4).

$$\mathbf{N} = \frac{\mathbf{P} \times \mathbf{Q}}{\|\mathbf{P} \times \mathbf{Q}\|} \tag{A.4}$$

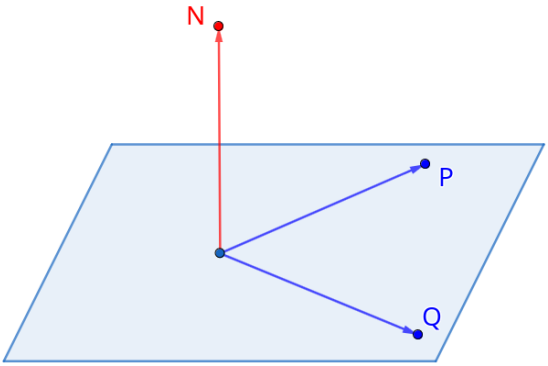


Figure A.6: Illustration of a plane and its normal

Figure A.7 gives a sample of rays that went through four reflections, it shows how they interact with buildings and also the paths followed by rays after reflection. After the maximum number of reflections is reached, the bouncing process gives as output all the coordinates of the rays, their respective directions and normals, which are useful information for subsequent processes.

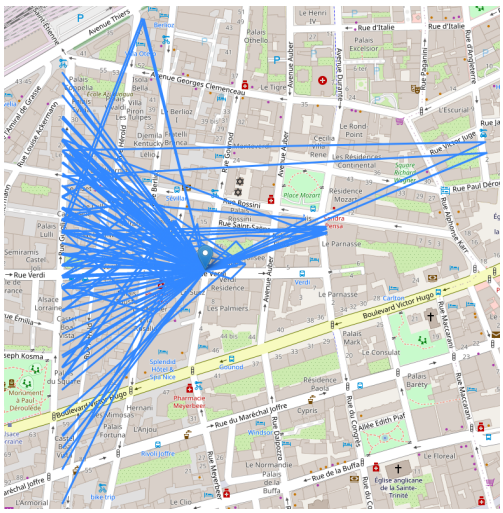


Figure A.7: Example of reflected rays from a transmitter

A.4 Reception test

Once all the rays are launched and tracked until the maximum number of reflections is reached, a reception test is performed in order to assess which rays have been received. If only one receiver is targeted, this process can be done at each reflection, otherwise it is better to wait until the maximum number of reflections is reached before performing this reception test. In either case, this reception test simply consists of checking which rays have been received by which reception point. As explained in Chapter 2 and Chapter 4, this consists of a matrix multiplication. To be told differently, it consists of operations between 2D arrays. These matrix operations are given in Equation (2.46). In Python, the Numpy library [103] can be used to perform these operations. Note here that, from the radius of the ray formula given in Equation (3.2), the maximum angular separation α corresponds to the one given by the ray generation process in Section A.2.

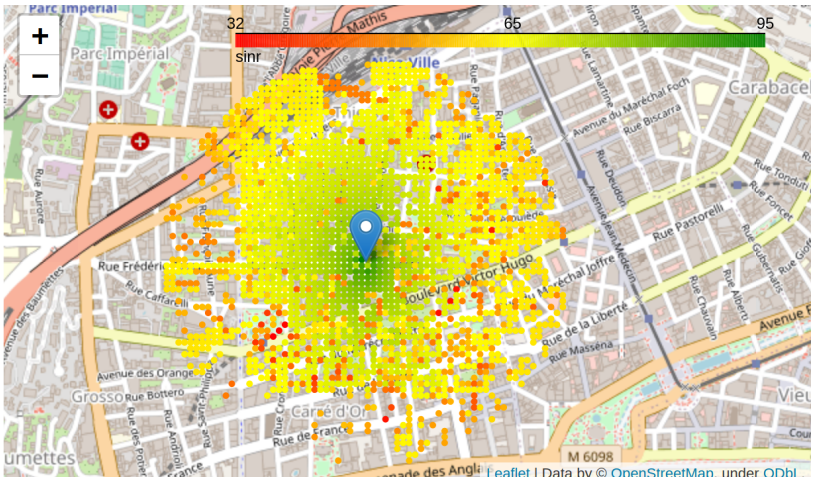


Figure A.8: Coverage map of the city center of Nice

Once this matrix operation is performed, each reception point is associated to the ray it has received and the corresponding received power is further computed. The formula for the received power is given by Equation (2.47), which requires Equations (2.48), (2.49), and (2.50). In Equation (2.50), the direction vector $\mathbf{u}$ is the one obtained during the ray launching and the bouncing process. Also, the normal vector

N is the one computed using Equation (A.4). Therefore, it is necessary to keep trace of these two vectors in the form of lists characterizing the rays.

Once the power is computed for all the reception points, the SINR can be computed using Equation (2.51) and the RF map showing the SINR can be displayed. In Python, the Folium library [104] offers hands-on tools that help displaying this map. Figure A.8 is an example of such a map generated by Folium. Apart from Folium, this map can be saved in a GIS format and be displayed afterwards in GIS tools like QGIS [105] and ArcGIS [106] for better visualization.

A.5 Summary

In this appendix, we presented how to implement Ray Tracing from scratch in a programming language. We show that a ray is the fundamental component in Ray Tracing and that it is identified by its coordinates, its directions and the normal to all the surfaces it hits. These three characteristics of the ray help to determine whether the ray is being received or not and are necessary for the power computation. Throughout this appendix, we provide hands-on tools that help in the implementation of RT either for computing the signal power received by a single receiver or to produce RF maps. This appendix is meant to help anyone willing to implement RT from scratch for their own use to do it in the programming language of their choice.