Deterministic Automotive Radar Channel Modeling

A Physical Optics Approach

Ana Beatriz Neto Cerqueira

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. António José Castelo Branco Rodrigues
Prof. Dr.-Ing. Thomas Zwick

Examination Committee
Chairperson: Prof. José Eduardo Charters Ribeiro da Cunha Sanguino
Supervisor: Prof. António José Castelo Branco Rodrigues
Member of the Committee: Prof. António Luís Campos da Silva Topa

July 2020
Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.
Acknowledgments

This thesis is dedicated to everyone without whom none of this would have been possible.

First of all, I would like to sincerely thank Prof. António Rodrigues for allowing me to follow my own path and for the essential support and patience throughout all the work, and to Instituto de Telecomunicações for providing me the means for the completion of this dissertation. Also, a big thanks to Prof. Dr.-Ing. Zwick for giving me this opportunity, the guidance and the time spent.

To my family, for your constant and unconditional support. Thank you for all the opportunities you give me without question, for always believing in me and for trusting my abilities as a student and a person alike.

To Momme Rickmers, for your kindness, patience and resilience. Thank you for being by my side all the way through, always lifting me up and making me a better version of myself.

To my dearest friend and fiercest supporter Denise Centeno. It is with you that I share my insecurities, but most importantly my victories. This one is yours too.

To Família Pereira, for your care and dedication. Thank you for helping me grow as an individual and for your availability and genuine support.

Thank you, Larissa Pillay, for your incredibly present friendship and for your true understanding. I hope that our paths meet again.

To my friends from Técnico who shared this adventure with me and made it so memorable. Thank you Pedro Micaela, António Almeida, Daniel Sousa, Francisco Eiras, Miguel Cardoso and João Borrego. A special thanks to Filipa Guimarães and João Cordeiro for being my mentors and dear friends.

To Dra. Vânia, thank you for helping me rebuild my confidence, giving me the power to hear my own voice.

To everyone who accompanied me throughout this thesis and my life in general, a heartfelt thank you. I finish this work and my academic journey with a special and true sense of accomplishment and excitement towards the future.
Resumo

A condução autónoma tem-se tornado rapidamente um objectivo realmente promissor e concretizável nos últimos anos. Inúmeros esforços têm vindo a ser investidos com o objectivo de criar novas funcionalidades e algoritmos que permitam aos sensores supervisionar o ambiente dos veículos constantemente e de maneira segura. Neste aspecto, radares surgem como um dos mais relevantes tipos de sensores na indústria automóvel, graças à sua função essencial na detecção de obstáculos a longa distância e a sua robustez contra condições climáticas adversas.

Com o intuito de desenvolver funções seguras e eficientes para a implementação de condução automática e autónoma, um modelo determinístico de um canal de radar, que simule e preveja com precisão o comportamento de um radar no seu ambiente, é necessário. Uma abordagem focada na simulação determinística da dispersão (scattering) de ondas electromagnéticas é investigada, através da aplicação da aproximação da óptica física em condutores eléctricos. Uma vez que este mecanismo de propagação desempenha um papel extremamente importante em ambientes automotivos, e porque os modelos existentes para a sua simulação apresentam algumas limitações problemáticas, esta nova abordagem procura melhorar o desempenho de simulações deste tipo até certo ponto.

Através da implementação de um programa em C++ baseado num algoritmo de ray tracing, adaptado para dispersar raios no momento de scattering por uma superfície dispersiva, gráficos de RCS (radar cross section) de variados objectos standard são obtidos e comparados com expectativas teóricas e com resultados de outras ferramentas de software de simulação, como o CST Studio Suite.

Os resultados obtidos através da variação de alguns parâmetros do simulador sugerem que a análise se deve focar fundamentalmente no número de raios incidentes que representam a onda electromagnética emitida pelo radar. Apresentando algumas inconsistências no caso de múltiplos raios incidirem numa superfície plana reflectiva de grandes dimensões, a simulação pode ser, contudo, adaptada por um factor de correção que ajusta a gama de valores dos gráficos obtidos. Não obstante, para superfícies irregulares ou curvas, a abordagem da óptica física surge como uma alternativa precisa e coerente aos modelos estocásticos e não-determinísticos existentes, revelando um desvio relativo médio de 1.96%, majoritariamente causado por simplificações relacionadas com a contrução digital do modelo tri-dimensional.

Uma conclusão final estabelece que, para a simulação da reflexão por superfícies planas de grande dimensão, a simplicidade de outros algoritmos como a óptica geométrica pode ser uma vantagem. Aparar de não tão precisa como a óptica física, a estratégia simplificada da óptica geométrica é, ainda assim, mais eficiente em termos de tempo de execução e recursos computacionais do que o algoritmo implementado com o factor de correção adicional. Em contrapartida, para obstáculos altamente dispersivos como esferas reflectivas ou superfícies irregulares de objectos de grande porte, a óptica física prova ser notavelmente vantajosa. Uma adicional optimização do código, visando um muito mais curto tempo de execução e uma mais eficiente gestão de memória, é desejada.

Palavras-Chave: ondas electromagnéticas, scattering, radar cross section, óptica física, ray tracing, radar automotivo
Abstract

Autonomous driving has rapidly become a very promising and achievable technological goal in the past few years. Enormous efforts have been invested into creating new functionalities and algorithms that enable sensors to permanently and reliably supervise the vehicles’ surroundings. In this aspect, radars arise as one of the most relevant sensors in the automotive industry, given their essential function on long range detection of obstacles and their distinctive robustness against bad weather conditions.

In order to develop reliable and efficient functions for the implementation of autonomous driving, a deterministic radar channel model that accurately simulates and predicts the behaviour of a radar in its environment is necessary. An approach focused on the deterministic simulation of the scattering of electromagnetic waves is pursued, through the application of the physical optics approximation on perfect electric conducting targets. Since this wave propagation mechanism plays a major role in automotive environments, and because the existing models for its prediction present some concerning limitations, this new approach seeks to improve the performance of simulations of this kind at some extent.

Through the implementation of a program in C++ based on a ray tracing algorithm, adapted to perform the diffusion of rays at the moment of scattering by a scattering surface, radar cross sections of various standard objects are obtained and compared with the theoretical expectations and with results from other simulation software tools, such as CST Studio Suite.

The results obtained through the variation of some simulator parameters suggest that the focus of the analysis should be the number of incident rays that represent the electromagnetic wave launched from the radar transmitter. Presenting some inconsistencies for the case of multiple incident rays incident on a large planar reflective surface, the simulation can be, however, adapted by a correction factor that adjusts the range of values of the radar cross section plot. Nevertheless, for irregular or curved surfaces, the physical optics approach emerges as an accurate and coherent alternative to the existing stochastic non-deterministic methods, disclosing a medium relative deviation of 1.96%, mostly caused by simplifications on the construction of the three-dimensional digital model.

A final conclusion establishes that, for the simulation of reflection by large planar surfaces, the simplicity of other algorithms such as the geometrical optics approximation can be an advantage. Even though not as accurate as the physical optics, the geometrical optics simplified strategy is still more efficient in terms of time and computational resources than the developed algorithm with an additional correction factor. On the other hand, for highly scattering obstacles such as spheres or irregular surfaces of extended objects, the physical optics proves to be remarkably advantageous. A further optimization of the code, aiming for a much shorter running time and more efficient memory management, is desired.

Keywords: electromagnetic waves, scattering, radar cross section, physical optics, ray tracing, automotive radar
# Contents

<table>
<thead>
<tr>
<th>Acknowledgements</th>
<th>iii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resumo</td>
<td>iii</td>
</tr>
<tr>
<td>Abstract</td>
<td>v</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xi</td>
</tr>
<tr>
<td>List of Symbols</td>
<td>xv</td>
</tr>
<tr>
<td>1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Context and Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Main Concepts</td>
<td>2</td>
</tr>
<tr>
<td>1.2.1 The ray concept</td>
<td>2</td>
</tr>
<tr>
<td>1.2.2 Geometrical optics</td>
<td>3</td>
</tr>
<tr>
<td>1.2.3 Wave propagation mechanisms</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Objectives</td>
<td>8</td>
</tr>
<tr>
<td>1.4 Thesis Outline</td>
<td>9</td>
</tr>
<tr>
<td>2 Theoretical Background</td>
<td>11</td>
</tr>
<tr>
<td>2.1 Physical Optics</td>
<td>11</td>
</tr>
<tr>
<td>2.1.1 The physical equivalent</td>
<td>11</td>
</tr>
<tr>
<td>2.1.2 The physical optics approximation</td>
<td>12</td>
</tr>
<tr>
<td>2.2 Scattering Equations</td>
<td>13</td>
</tr>
<tr>
<td>2.2.1 Vector potentials</td>
<td>13</td>
</tr>
<tr>
<td>2.2.2 Field equations in the far field region</td>
<td>14</td>
</tr>
<tr>
<td>2.3 Radar Cross Section</td>
<td>16</td>
</tr>
<tr>
<td>2.4 Scattering from a rectangular plate</td>
<td>17</td>
</tr>
<tr>
<td>2.5 Ray Tracing</td>
<td>21</td>
</tr>
<tr>
<td>2.6 Direct methods of ray tracing</td>
<td>21</td>
</tr>
<tr>
<td>2.6.1 Fermat’s principle of least time</td>
<td>22</td>
</tr>
<tr>
<td>2.6.2 Image method</td>
<td>22</td>
</tr>
<tr>
<td>2.7 Ray Launching: the indirect approach for ray tracing</td>
<td>23</td>
</tr>
<tr>
<td>2.7.1 Shooting and bouncing ray image approach</td>
<td>24</td>
</tr>
<tr>
<td>2.7.2 Backward ray tracing</td>
<td>24</td>
</tr>
<tr>
<td>2.7.3 Discrete ray tubes</td>
<td>25</td>
</tr>
<tr>
<td>2.7.4 Reception spheres</td>
<td>25</td>
</tr>
<tr>
<td>2.7.5 Ray density normalization</td>
<td>27</td>
</tr>
<tr>
<td>2.8 Radar Technology</td>
<td>28</td>
</tr>
<tr>
<td>Chapter</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>3</td>
<td>Simulator Setup</td>
</tr>
<tr>
<td>3.1</td>
<td>Simulation Concept</td>
</tr>
<tr>
<td>3.2</td>
<td>Algorithm Considerations</td>
</tr>
<tr>
<td>3.3</td>
<td>Ray Tracing Method</td>
</tr>
<tr>
<td>3.4</td>
<td>Bistatic Setup</td>
</tr>
<tr>
<td>3.5</td>
<td>Monostatic Setup</td>
</tr>
<tr>
<td>3.6</td>
<td>Complex Vectors</td>
</tr>
<tr>
<td>3.7</td>
<td>Wave Polarization</td>
</tr>
<tr>
<td>3.8</td>
<td>Simulation Variables</td>
</tr>
<tr>
<td>4</td>
<td>Code Implementation</td>
</tr>
<tr>
<td>4.1</td>
<td>Ray Tracing's Recursive Nature</td>
</tr>
<tr>
<td>4.2</td>
<td>Generating Incident Rays</td>
</tr>
<tr>
<td>4.3</td>
<td>Incident Electric Field</td>
</tr>
<tr>
<td>4.4</td>
<td>Rendering Digital Models</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Reading a mesh file</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Intersection between a ray and a triangle</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Back-face culling</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Separation between rays at intersection</td>
</tr>
<tr>
<td>4.5</td>
<td>Geometry</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Global and local coordinate systems</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Radar angular sweep</td>
</tr>
<tr>
<td>4.5.3</td>
<td>Scattering of rays</td>
</tr>
<tr>
<td>4.6</td>
<td>Reception of Rays</td>
</tr>
<tr>
<td>4.6.1</td>
<td>Dimension of the reception sphere</td>
</tr>
<tr>
<td>4.6.2</td>
<td>Intersection between a ray and a sphere</td>
</tr>
<tr>
<td>4.6.3</td>
<td>Multiple rays</td>
</tr>
<tr>
<td>4.7</td>
<td>Calculation of the Scattered Field</td>
</tr>
<tr>
<td>5</td>
<td>Simulations and Results</td>
</tr>
<tr>
<td>5.1</td>
<td>Simulation parameters and considerations</td>
</tr>
<tr>
<td>5.2</td>
<td>Scattering by a rectangular planar surface</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Bistatic setup with oblique incidence</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Monostatic setup</td>
</tr>
<tr>
<td>5.3</td>
<td>Multiple incident rays</td>
</tr>
<tr>
<td>5.4</td>
<td>PEC Cube</td>
</tr>
<tr>
<td>5.5</td>
<td>Sphere</td>
</tr>
<tr>
<td>6</td>
<td>Conclusions</td>
</tr>
<tr>
<td></td>
<td>References</td>
</tr>
<tr>
<td>A</td>
<td>Pseudo-code of the simulator</td>
</tr>
<tr>
<td>B</td>
<td>Additional Plots</td>
</tr>
<tr>
<td>B.1</td>
<td>Horizontal Polarization</td>
</tr>
<tr>
<td>B.2</td>
<td>Number of Scattered Rays</td>
</tr>
<tr>
<td>B.3</td>
<td>Multiple incident rays on cube</td>
</tr>
<tr>
<td>B.4</td>
<td>Number of incident rays on sphere</td>
</tr>
</tbody>
</table>
## List of Figures

1.1 Variety of sensors present in an automotive vehicle [1] ........................................... 2
1.2 Geometrical representation of the ray concept for plane waves [2] ............................... 3
1.3 Astigmatic tube of rays [2] ...................................................................................... 4
1.4 Reflection and transmission of a wave on a planar smooth surface [3] ......................... 4
1.5 Reflection of an astigmatic tube of rays at a curved smooth surface [3] ....................... 5
1.6 Cone of diffracted rays produced by an edge of a surface ................................. 6
1.7 Diffracted rays produced by two slits on a surface .............................................. 7
1.8 2D representation of the stochastic scattering approach [2] ..................................... 8
2.1 Scattering by a perfect electric conductor (left) and its physical equivalent (right) [3] 12
2.2 Physical optics phenomena at an infinite ground plane [3] ...................................... 13
2.3 Coordinate system for radiation due to a scatterer [3] ............................................. 14
2.4 Coordinate system for scattering in the far field region [3] ........................................ 15
2.5 Perfectly electric conducting rectangular plate illuminated by uniform plane wave [3] 17
2.6 Monostatic radar cross section per squared wavelength of a rectangular plate [2] .......... 20
2.7 Bistatic radar cross section per squared wavelength of a rectangular plate for an incident angle of $\theta_i = 30^\circ$, $\phi_i = 270^\circ$ [2] ............................................................... 20
2.8 2D geometrical representation of the image method [2] ............................................ 22
2.9 Principle of ray launching [4] .................................................................................. 23
2.10 Different strategies for designing the ray tracing algorithm ..................................... 24
2.11 Reception of rays using discrete ray tubes [2] ....................................................... 25
2.12 Validation of multiple rays, using constant size reception spheres [2] ..................... 26
2.13 Tessellation of an icosahedron’s face and originated wavefronts [5] ......................... 26
3.1 Decomposition of a representative surface into small polygons ......................... 30
3.2 Two correspondent rays with the same direction .................................................... 31
3.3 Launching equidistant camera rays .......................................................................... 32
3.4 Two-dimensional representation of a radar shooting incident rays ...................... 32
3.5 Launching scattered rays from two different scattering points A and B .................... 34
3.6 Surface decomposed into rectangles by the incidence of multiple rays ................. 34
3.7 Reception sphere located at the receiver position .................................................... 35
3.8 Bistatic setup ........................................................................................................ 36
3.9 Monostatic setup .................................................................................................... 37
3.10 Phase front of a generic travelling wave [3] .......................................................... 38
4.1 Tree of rays for a scattering problem ...................................................................... 44
4.2 Transformation between the screen and the camera spaces ................................... 45
4.3 Camera local coordinate system ............................................................................. 46
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4</td>
<td>Local coordinate system of the radar and polarization components for a ray propagating along the positive z-axis.</td>
<td>47</td>
</tr>
<tr>
<td>4.5</td>
<td>&quot;Inside-outside&quot; test to determine whether a point is inside the area of a triangle.</td>
<td>50</td>
</tr>
<tr>
<td>4.6</td>
<td>Back-face culling of surfaces that face away from the radar (in red).</td>
<td>51</td>
</tr>
<tr>
<td>4.7</td>
<td>Equidistant rays at a surface perpendicular to the radar orientation.</td>
<td>52</td>
</tr>
<tr>
<td>4.8</td>
<td>Distance between rays for a generic orientation of the surface (horizontal or vertical projection).</td>
<td>53</td>
</tr>
<tr>
<td>4.9</td>
<td>Global coordinate system (blue) and local coordinate systems (red).</td>
<td>54</td>
</tr>
<tr>
<td>4.10</td>
<td>Angles of incidence $\gamma$ and $\delta$.</td>
<td>55</td>
</tr>
<tr>
<td>4.11</td>
<td>Scattering angles $\phi$ and $\theta$.</td>
<td>57</td>
</tr>
<tr>
<td>4.12</td>
<td>Dimension of the reception sphere as a function of the angle increment and the far field distance.</td>
<td>59</td>
</tr>
<tr>
<td>5.1</td>
<td>Bistatic radar cross section of a rectangular plate for an angle of incidence of 30°.</td>
<td>67</td>
</tr>
<tr>
<td>5.2</td>
<td>Monostatic radar cross section of a small rectangular plate.</td>
<td>68</td>
</tr>
<tr>
<td>5.3</td>
<td>Monostatic radar cross section of a large rectangular plate.</td>
<td>69</td>
</tr>
<tr>
<td>5.4</td>
<td>Monostatic radar cross section of a large rectangular plate with multiple incident rays.</td>
<td>70</td>
</tr>
<tr>
<td>5.5</td>
<td>Monostatic radar cross section of a rectangular plate with multiple incident rays.</td>
<td>70</td>
</tr>
<tr>
<td>5.6</td>
<td>Monostatic radar cross section of a small cube.</td>
<td>72</td>
</tr>
<tr>
<td>5.7</td>
<td>Monostatic radar cross section of a large cube.</td>
<td>72</td>
</tr>
<tr>
<td>5.8</td>
<td>Monostatic radar cross section of a small sphere.</td>
<td>73</td>
</tr>
<tr>
<td>B.1</td>
<td>Monostatic radar cross section of rectangular plate for horizontally polarized incident wave.</td>
<td>83</td>
</tr>
<tr>
<td>B.2</td>
<td>Monostatic radar cross section of rectangular plate with less scattered rays.</td>
<td>84</td>
</tr>
<tr>
<td>B.3</td>
<td>Monostatic radar cross section of small cube with multiple incident rays.</td>
<td>84</td>
</tr>
<tr>
<td>B.4</td>
<td>Monostatic radar cross section of sphere with 5 x 5 incident rays.</td>
<td>85</td>
</tr>
</tbody>
</table>
List of Symbols

Mathematical notation

$F$ Real number.

$\mathbf{F}$ Complex number.

$\mathbf{F}$ Vector.

$\hat{\mathbf{F}}$ Vector with unit length.

$\mathbf{F}$ Complex vector.

$\mathbf{F}_1 \cdot \mathbf{F}_2$ Scalar product of two vectors.

$\mathbf{F}_1 \times \mathbf{F}_2$ Cross product of two vectors.


$|\cdot|$ Absolute value.

$\langle \cdot \rangle$ Expected value.

Greek letters

$\alpha$ Field of view.

$\beta$ Phase constant of a wave, $\beta = k = 2\pi/\lambda$.

$\gamma$ Horizontal projection of the angle of incidence on the xz-plane.

$\delta$ Vertical component of the angle of incidence measured from the positive y-axis.

$\Delta$ Additional factor for the calculation of the scattered electric field.

$\eta$ Wave impedance, $\eta = \sqrt{\mu/\varepsilon}$.

$\theta, \phi$ Spherical coordinate system components.

$\theta_i$ Incident angle, measured between the direction of propagation of the wave and the surface normal.
\( \theta_{\text{obs}} \) Angular increment of the observation angle.

\( \theta_s, \phi_s \) Direction of the scattered wave in its spherical configuration.

\( \lambda \) Wavelength, \( \lambda = c/f \).

\( \rho \) Angle between two vectors.

\( \sigma \) Radar cross section.

\( \sigma_h \) Standard deviation of surface height distribution.

\( \varphi \) Phase component of the complex field.

\( \psi \) Rotation angle.

\( \Psi_z \) Phase component of the path section correspondent to the depth level \( Z \).

\( \omega \) Angular frequency, \( \omega = 2\pi f \).

\( \Omega \) Scattered electric field for the scattering direction.

**Latin letters**

\( a, b \) Dimension of a rectangular plate in the \( x \) and \( y \) directions, \( a \times b \) returns its area.

\( \hat{a} \) Generic unit vector indicating a certain direction.

\( A \) Magnetic vector potential.

\( C \) Center of the reception sphere.

\( d \) Distance between the radar and the geometric center of the target.

\( \hat{d} \) Direction of a ray.

\( D \) Largest dimension of a scattered.

\( \hat{e} \) Direction of the electric field complex vector.

\( E \) Electric field.

\( f \) Frequency.

\( F \) Electric vector potential.

\( h \) Surface height.

\( \hat{h} \) Direction of the magnetic field complex vector.
H  Magnetic field.

i, j  Index of a ray in the horizontal and vertical directions.

J  Electric current density.

k  Wave number, \( k = \frac{2\pi}{\lambda} \).

L  Correlation length of heights along a surface.

M  Magnetic current density.

n  Number of scattered rays.

\( \hat{n} \)  Normal vector to a surface.

N  Total number of rays received at the radar.

O  Origin of a ray.

P  Generic point along the path of a ray.

Q  Reflection point.

r  Distance covered by a wave or a ray from the transmission point.

R  Distance from the transmission point.

\( R_{sphere} \)  Radius of the reception sphere.

\( \hat{\text{ref}} \)  Reference unit vector.

Rx  Receiver.

s  Mean slope of an irregular surface.

t  Distance from the origin to any point along the path of a ray.

T  Transformation matrix.

Tx  Transmitter.

v  Vertex of a triangle.

Z  Depth level of the received ray.

Subscripts

0  Reference.

1  Incident.
1,2 Polarization components.

depth Active depth level.

h,v Horizontal and vertical polarization components.

H,V Horizontal and vertical directions.

i Incremental index.

max Maximum.

polygon Polygon local coordinate system.

r, θ, φ Radial, polar and azimuth direction in the spherical coordinate system.

radar Radar local coordinate system.

received Received at the radar.

S Surface.

total Overall scattered electric field.

world Global coordinate system, or world space.

x,y,z x, y and z direction in the Cartesian coordinate system.

Superscripts

i Incident.

s Scattered.

Constants

ε₀ Vacuum permittivity, $\varepsilon_0 = \frac{1}{\mu_0 \omega^2} \approx 8.854188 \cdot 10^{-12} \text{F} \cdot \text{m}^{-1}$.

μ₀ Vacuum permeability, $\mu_0 = 4\pi \cdot 10^{-7} \text{N} \cdot \text{A}^{-2}$.

c₀ Speed of light in vacuum, $c_0 = 1/\sqrt{\varepsilon_0 \mu_0} = 2.99792458 \cdot 10^8 \text{m} \cdot \text{s}^{-1}$.

e Euler’s number, $e \approx 2.71828$.

j Imaginary unit, $j^2 = -1$. 
Chapter 1

Introduction

This chapter presents the generic context and the motivation behind this thesis, followed by some brief basic concepts that introduce the project's contents. Finally, the objectives of the work developed in the scope of the thesis are provided and an outline of the whole document is defined.

1.1 Context and Motivation

Over the past few years, autonomous driving has become a major research field for engineers worldwide and a very promising technology. Enormous efforts and investments have been made in the development of new functionalities which will lead to the proliferation of autonomous vehicles and, consequently, autonomous driving, in just a few years.

With the objective of accelerating the successful development of autonomous vehicle technologies, the Defense Advanced Research Projects Agency (DARPA) launched “The Grand Challenge” [6], a competition in which teams should complete several trails with autonomous vehicles in very different environments. On the first event, in 2004, no teams were able to complete this challenge. However, right in the following year, five teams succeeded with surprising results, proving that this field is improving at an incredible pace.

In 2013, Nissan announced their intentions to commercialize multiple vehicles with autonomous driving capabilities by the year of 2020 [7].

In the near future, autonomous vehicles will emerge as a technology that will greatly improve the transport system among society, especially in terms of safety.

According to the National Motor Vehicle Crash Causation Survey, conducted by the National Highway Traffic Safety Administration, from 2005 to 2007, in the United States of America, 94 percent of all car crashes are caused by errors of human nature, mostly related to the driver's inattention, internal and external distractions or inadequate surveillance. Over thirty thousand people die from car crashes every year in the U.S. and huge amounts of money are spent in the occurrence of such accidents [8].

These alarming numbers are the main motivation for the development of self-driving vehicles and their introduction in the environment as soon as possible.

Besides preventing deadly accidents, other sociological and economical consequences arise directly from the implementation of this technology next to the society, such as the provisioning of critical mobility to the elderly and disabled, the optimization of road capacity, fuel saving and, ultimately, the reduction of green house gases.

Particularly, autonomous vehicles are able to sense and even anticipate other vehicles' braking and acceleration decisions. This way, it is possible to systematically achieve smoother and finer acceleration
adjustments, resulting in less fuel consumption, less mechanical damage and better traffic management. Other autonomous driving functionalities also allow a more efficient utilization of the road lanes and selection of routes [9].

However, despite all the potential progress, there are still some critical problems which constitute serious limitations in the employment of autonomous driving technologies to vehicles. Some fundamental challenges and tasks, such as the recognition of other vehicles and humans in the road, are extremely difficult and complex to implement on computational developed autonomous vehicles. Furthermore, the differentiation between objects with different materials should be a fundamental functionality of every sensor present in the vehicle [9].

![Figure 1.1: Variety of sensors present in an automotive vehicle](image)

The most important requirement of this technology is that cars are equipped with a variety of sensors that, together, enable a constant and reliable supervision of the surrounding environment.

As it is evident in Figure 1.1, radar sensors play only a part in the whole technology involved in self-driving vehicles. However, their role is of the most important and revolutionary character, since they are the most unaffected sensors by poor weather conditions, such as fog, snow and rain, and this is why their functioning must be as accurate and infallible as possible.

1.2 Main Concepts

In this section, some ray-optical and wave propagation concepts are introduced. A brief contextualization of the ray concept is given, along with an introduction of the most relevant propagation phenomena.

1.2.1 The ray concept

The first step to explain the ray concept is the introduction of the vector wave equations for source-free and lossless media, represented by

\[
\nabla^2 \vec{E}(x, y, z) + k^2 \vec{E}(x, y, z) = 0
\]

(1.1)

and

\[
\nabla^2 \vec{H}(x, y, z) + k^2 \vec{H}(x, y, z) = 0
\]

(1.2)

where \( k \) is the phase constant, which includes in its definition the relative complex permittivity and permeability of the propagation material [3].
The simplest solution of these wave equations is, then, given by the expressions for the electric and magnetic fields of a propagating wave [2],

\[
\vec{E}(x, y, z) = E_0 e^{-j\vec{k} \cdot \vec{r}} \hat{e}
\]

and

\[
\vec{H}(x, y, z) = H_0 e^{-j\vec{k} \cdot \vec{r}} \hat{e}
\]

where \( \vec{k} \) points in the direction of propagation of the wave and \( \vec{r} \) is the position vector.

In its turn, the relation between the electric and magnetic field vectors indicate that both are perpendicular to each other and to the direction of propagation.

Therefore, the rays, which can be represented by power flow lines, are perpendicular to the series of wavefronts resulting from the plane wave propagation. These power flow lines do not intersect each other, in any other point other than their source [10]. Figure 1.2 illustrates this concept in terms of bi-dimensional geometry.

1.2.2 Geometrical optics

Geometrical optics is a high-frequency approximation that determines the propagation of incident, reflected and transmitted waves [3].

The basic concept of geometrical optics is based on the conservation of energy flux in a tube of rays. Along the length covered by the tube, the cross section at each point, also called Eikonal surface, has an area and radiation density associated to it. Combined, these two characteristics amount to the total energy at each cross section point, which is the same at every point along the direction of the propagation of the ray [2], hence the principle of conservation of energy within the tube. Figure 1.3 shows this representation. The caustic lines PP' and QQ' represent the lines which contain all the rays of the tube, and are in principle infinite. The geometrical optics field on these lines is thus mathematically infinite and can only be defined theoretically, since infinite is not a possible empiric measure for a field [2].

Geometrical optics also approximates the scattered fields only towards specular directions, following the principle of Snell's law of reflection, as explained in subsection 1.2.3. This approximation is only acceptable in special high frequency conditions, where the geometrical optics field dominates over the scattering phenomena [3].

When modelling wave propagation with geometrical optics, each ray is traced by means of its propagation mechanisms when interacting with surrounding interfaces. The complex and vector superposition
Wave propagation mechanisms

Before modelling the propagation of radio waves, the different existent propagation mechanisms should be initially taken into account.

Besides the free space phenomena propagation and absorption, which greatly influence the field strength at each observation point of the ray propagation path, the most basic mechanisms that occur when the ray intersects an interface are the reflection and transmission at a smooth surface.

In this case, there are two media separated by an interface, and three planes to be considered, the incident, the reflected and the transmitted, as in Figure 1.4.
that is perpendicular to the plane of incidence is called perpendicular polarization, while the component that is parallel to this plane is named parallel polarization. These vectors should be manipulated separately, and their sum returns the total incident field. The same principle of decomposition is applied to the reflected and transmitted fields, and to the respective magnetic fields.

The respective angles of reflection are determined by Snell's law of reflection, in which it is stated that the reflected ray angle is equal to the incident one, \( \theta_r = \theta_i \).

The reflection coefficients are a particular characteristic of each surface material and determine the magnitude of the reflected electric and magnetic fields, relatively to the incident ones.

Equivalently, the same logic can be applied to the transmitted ray, considering Snell’s law of transmission and the transmission coefficients at the interface [2].

In the particular case of Figure 1.4, the described polarization is perpendicular.

Similarly to the example of reflection and transmission presented above, one can also determine the path traversed by a ray once it is reflected at a curved smooth surface, when the radius of the curvature of the surface is large when compared to the wavelength [3].

Geometrical optics can be applied to an incident ray that hits a curved smooth surface [3]. Recurring to Snell's law of reflection, one can determine the direction of the correspondent reflected ray, considering only the point of incidence and the curvature of the surface.

The incident ray is simply reflected at the tangential plane in its intersection point with the surface. When applied to all the rays in the tube of rays, this process results in a reflected tube of rays, which will present a different curvature and size of its cross section than the incident one [2].

The incident angle is measured between the normal of the tangential plane in the reference point of reflection \( Q_R, \hat{n} \), and the direction of the incident ray, \( \hat{s}_i \). In its turn, the reflection angle is measured between the same normal and the direction of the reflected ray, \( \hat{s}_r \), and it is equal to the incident one.

This process is illustrated in geometrical terms in Figure 1.5.

![Figure 1.5: Reflection of an astigmatic tube of rays at a curved smooth surface](image)

In the event that a surface contains irregularities in its shape such as edges, vertexes, corners or apertures, the electromagnetic mechanism that occurs is the diffraction. This phenomenon can also be considered as a correction factor for the geometrical optics approximation [3], since it includes the rays
produced by “boundary effects” on surfaces, in addition to the optical ones, produced by just reflection and transmission.

In order to calculate the fraction of energy diffracted at a surface or aperture and its direction of propagation, one can apply the Geometrical Theory of Diffraction [11], which is an extension of geometrical optics that describes diffraction phenomena.

A single incident ray produces infinite diffracted rays when it hits the edge of a surface. The diffracted rays follow the law of edge diffraction, which states that every diffracted ray makes the same angle with the edge and lies on opposite sides of the plane normal to the edge at the point of diffraction than the respective incident ray [12]. Therefore, the diffracted rays lie on the surface of a cone, with a half-angle that corresponds to the angle between the incident ray and the edge. In the case that the ray hits the edge perpendicularly, the diffracted rays are spread over a plane which is perpendicular to the edge. Figure 1.6 illustrates the directions in which diffracted rays are launched when an oblique incident ray hits the edge of a surface.

![Figure 1.6: Cone of diffracted rays produced by an edge of a surface.](image)

When a ray hits a corner or vertex of an edge, the diffracted rays originated by this phenomena satisfy the law of vertex diffraction. In this case, the diffracted ray may have any direction, independently of the direction of the incident ray, which means that the diffracted rays will leave the vertex in all directions [12].

In the presence of slits on a surface, the same law of edge diffraction applies to the diffracted rays. However, one must note that on each point beyond the surface pass two diffracted rays, one derived from each slit, meaning that multiple diffracted rays occur [12]. The field value at any point must therefore be, not only the field of the isolated diffracted ray, but the sum of the fields of all rays that pass through that point. In this case, this includes the two diffracted rays and also the ray transmitted through the surface. The scheme in Figure 1.7 shows the multiple diffracted rays generated when an incident plane wave passes through two slits on a surface.
So far, geometrical optics has been able to determine the reflected, transmitted and even diffracted fields from surfaces and apertures. However, this approach still fails to determine what exactly happens to rays when they hit complex surfaces with irregularities close to or smaller than the wavelength.

In reality, objects such as buildings, trees and cars don’t present only flat or perfectly curved smooth surfaces. The irregularities on these surfaces can have dimensions in the order of the size of the wavelength [13], which results in the scattering of energy in all directions, in what appears to be a random distribution.

An available method that can be useful to treat the scattering mechanism is the stochastic scattering approach, which can be used on surfaces with undulations, whose horizontal dimensions are large compared to the incident wavelength [2].

This method states, just like in geometrical optics, that the total field at any point of the surface can be determined as if the ray intersects the plane which is tangential to the irregularities at that point. The actual difference is that, in this case, not every ray is reflected by the same ideally smooth planar surface, but in a complex structure that presents random orientation and height at each different point of reflection for each elementary ray [2]. Therefore, according to Figure 1.8 one has to consider the representation of the incident plane wave through multiple discrete rays, which will be treated independently at the moment of scattering.

The statistical variations in the orientation of the tangential planes and their position are naturally related to the topology of the surface. These can be derived from the assumption that the surface height of most man-made materials can be approximated by a normal distribution with zero mean and standard deviation $\sigma_h$.

After some mathematical computations [2], one can obtain the slope magnitude of the irregularities for a two-dimensional isotropic rough surface from

$$s = \sqrt{\frac{< h_x'^2 + h_y'^2 >}{L}} = \frac{2\sigma_h}{L}. \tag{1.5}$$
The normally distributed variable \( h = h(x, y) \) is also described by a Gaussian auto-correlation factor, which gives the correlation between the height at two points separated by a certain distance. \( L \) is the correlation length, for which this auto-correlation equals \( \frac{1}{2} \).

By applying the stochastic scattering approach, in the condition that both the mean radius of the curvature of the random rough surface and the correlation length are large when compared to the wavelength, it becomes possible to expand the geometrical optics approximation by introducing a stochastic component [2]. However, it is important to note that when including the contribution of random scattering effects, the modelling becomes non-deterministic.

1.3 Objectives

This thesis aims to simulate the behaviour of a radar channel in a deterministic way, especially focusing on the scattering by extended three-dimensional targets.

In order to simulate any radar behaviour, one must first understand the nature of the surrounding environment and its conditions. In a real-life scenario, the automotive radar serves to detect any object or person that might be in the vicinity of the vehicle or its trajectory, in a sufficiently long range.

The most problematic fact about these scenarios is their unpredictable nature. Other vehicles, pedestrians, and even trees or buildings, all have very different attributes relating to their shape, texture and velocity. While some targets might be static and easy to recognize, others might be moving in a random trajectory and acceleration, even changing it at their will. This makes the job of the radar sensor much more complex, since it not only must detect the position of the object with a lot of precision, but it should also be able to determine whether it is getting closer or further away from the radar, and how fast.

Therefore, the primary concern when designing a radar channel should be its accuracy and consistency. The radar needs to correctly detect and interpret all the surrounding targets, without making any mistakes that might put the safety of the environment in jeopardy. For that reason, any simulations relating the radar scanning of the surrounding scene should mirror as perfectly as possible the real behaviour of the sensor. If a simulation proves to be adequate to the behaviour of a radar, one can predict its response without the necessity of running complex measurements of extended objects or scenarios, which can be extremely time and resource consuming.

This thesis focuses on the scattering mechanism from the perspective of physical optics, in opposition to the geometrical optics approach, used in the majority of radar channel simulators designed to the present day. The geometrical optics approximation proves to be very effective for reflection, transmission and even diffraction of electromagnetic waves, which are predominant phenomena in the presence of
large reflective surfaces.

However, scattering is the most dominant mechanism when the waves interact with non-ideal surfaces containing small irregularities, with random distribution and intensity. This is the case of most surfaces present in any urban or natural environment. Moreover, when defining scattering as a stochastic phenomena, the model becomes non-deterministic, which is questionable when simulating an environment whose main objective is to be extremely consistent and reliable.

Therefore, the objective of this work is to create a computer program which simulates the behaviour of a radar channel in a deterministic way, determining scattering through the application of the physical optics approach. By comparing the simulation of this mechanism for single standard targets with theoretical expected results, it can be studied whether this approach is suitable for such calculations. Additional comparisons with other existent simulation software programs might also assert whether this is an accurate algorithm for scattering of surfaces and in which scenario characteristics does the simulation perform the best.

Basing the principle of the code on following the rays throughout the scene, the code is built fundamentally around a ray tracing algorithm structure. Presenting a very similar logic as the actual function of a radar, the most fundamental structure of this code is the ray and its features. This way, the code is able to calculate the monostatic radar cross section of three-dimensional objects, which can later be compared with simulations from other programs and some theoretical calculations.

1.4 Thesis Outline

The work developed in the course of this thesis is reported in the next chapters and organized in five main sections.

Firstly, chapter 2 contemplates the theoretical background of the thesis. Starting with a clarification about the physical optics principle, its equations and the conditions of the approximation, the chapter continues by introducing the radar cross section concept and the analysis of a concrete problem for exemplification purposes. Next, the ray tracing technique is reviewed, followed by its different approaches and respective characteristics. Finally, some considerations about radar technology are explained.

In chapter 3, the configurations of the developed simulator are specified. The concept behind the simulation, the detailed conditions of the measurements and the structural architecture behind the ray tracing algorithm are explained. Moreover, the configuration of the measurements is described, along with some setup designs and a final consideration about the variables present in the code.

The code implementation is detailed in chapter 4, where the nature and logic of the algorithm are defined, as well as the intricacies related to the three-dimensional computation aspect of the program. The tracking of the rays along their paths from the transmitter to the receiver is delineated through the implementation of the ray tracing algorithm. Furthermore, the physical optics concept is applied to the electromagnetic field calculations.

The performed simulation tests and respective outcomes are compiled in chapter 5. The final results for the different analyzed scenarios are compared, discussed and interpreted. The effect of the different parameters is studied and an assessment of the quality of the program is made.

The conclusions and some final notes on the overall work are summarized in chapter 6, along with suggestions for further work on this subject. Appendix A and Appendix B contain, respectively, clarifying pseudo-codes of the developed algorithm and additional plots contemplating relevant results.
Chapter 2

Theoretical Background

This chapter presents the most fundamental concepts for the development of this project. Electromagnetic considerations are introduced, such as the physical optics approximation, the scattering equations and the radar cross section of objects. Concerning the digital processing approach, important aspects about the ray tracing method and its different considerations are studied.

2.1 Physical Optics

In this section, the physical optics concept is introduced as an approximation for solving the scattering by a perfect electric conducting obstacle. It emerges as an alternative to the induction equivalent, for the case where the induced current at the surface by the incident wave is unknown [3].

This equivalent is used to develop many useful calculation methods, such as the electric field integral equations (EFIE) and the magnetic field integral equations (MFIE). These can be simplified to a number of algebraic equations that can be solved by the use of matrix or iteration techniques. A very popular numerical technique to solve these integral equations is the moment method, still widely used in many simulation tools for the solving of scattering of electromagnetic waves [14]. However very effective in returning accurate results through the precise calculation of the induced current at a surface [3], this is a quite complex method, which involves many calculations and is difficult to implement. This is why a simpler approach, recurring to some simplifications, could be very desirable if it succeeds in returning satisfying results.

2.1.1 The physical equivalent

The calculation of the scattering by a perfect electric conducting obstacle bases itself on the concepts of the electric and magnetic field inside and outside of the volume of the scatterer, and on the induced current at its surface.

For the case where the surfaces of the obstacle are perfect electric conducting, both the electric and magnetic field inside the object are equal to zero, which indicates that the whole electric and magnetic field on the system is represented by only the incident and scattered fields outside the volume.

Figure 2.1 represents the imaginary surface $S_1$, which limits the generic volume of a scatterer $V_1$.

Over the boundary of the surface, the total tangential components of the electric field are equal to zero and the total tangential components of the magnetic field are equal to the current density induced
at the area of the surface. Therefore,

\[
\vec{J}_p = \hat{n} \times \vec{H}_1 + \hat{n} \times \vec{H}^s \tag{2.1}
\]

where \(\vec{J}_p\) is the complex vector correspondent to the induced current at the surface, \(\hat{n}\) is the unit vector describing the direction of the normal of the surface at that point, and \(\vec{H}_1\) and \(\vec{H}^s\) correspond, respectively, to the incident and scattered magnetic fields.

To introduce the equivalent, it is essential to mention that both the incident and scattered electromagnetic fields are solutions to Maxwell’s equations outside the volume \(V_1\). Therefore, it must be assumed that the same medium \(\mu_1, \varepsilon_1\) is present inside and outside of the object for the required conditions to remain valid.

Since, in this case, the induced current radiates in only one medium, one can apply the vector potential equations for the determination of the scattered electric and magnetic field as a function of the induced current. So far, this quantity is unknown and it constitutes a fairly complicated problem in scattering situations. Fortunately, some approximations make it easier to be determined, therefore allowing the calculation of the scattered fields to be more straightforward.

### 2.1.2 The physical optics approximation

In the ideal case where the obstacle is an infinite, flat, perfect electric conducting plate, the physical equivalent problem reduces itself to the one described in Figure 2.2. Here, the tangential components of the scattered magnetic field present the same phase and amplitude as the tangential components of the incident magnetic field. This problem is referred to as the physical optics, where the induced current can simply be defined by

\[
\vec{J}_p = 2\hat{n} \times \vec{H}_1. \tag{2.2}
\]

Since the incident electric and magnetic field is known in the system, the problem of having an unknown current density at the surface doesn’t exist anymore in this case, which makes it far less intricate to solve.

Consequently, the physical equivalent can be approximated to any physical optics problem where the scatterer presents electrically large dimensions, which would, in the theoretical limit, be an infinite ground plane. This situation implies that there is an approximation concerning the boundary conditions at the surface of the object, allowing the current to be approximated by

\[
\vec{J}_p \simeq 2\hat{n} \times \vec{H}_1. \tag{2.3}
\]
2.2 Scattering Equations

One way of obtaining the electric and magnetic radiated fields of a certain problem is by solving Maxwell’s or the wave equations. This strategy, although it requires just one step, can be very extensive and complex. Another approach to solve radiation and scattering problems in a more straightforward way is by calculating auxiliary vector potentials, followed by the computation of the electric and magnetic fields.

2.2.1 Vector potentials

There are two most common vector potentials, \( \vec{A} \), the magnetic vector potential, and \( \vec{F} \), the electric vector potential, respectively derived from the total electric and magnetic fields over the surface in question.

The construction of various solutions for the vector potential wave equation considering specific wave polarizations can be found in detail on [3]. The time varying solution of the inhomogeneous vector potential wave equation at a surface is described by

\[
\vec{A} = \frac{\mu}{4\pi} \iint_{S} \vec{J}_{S}(x',y',z') \frac{e^{-j\beta R}}{R} \, ds'
\]

and

\[
\vec{F} = \frac{\varepsilon}{4\pi} \iint_{S} \vec{M}_{S}(x',y',z') \frac{e^{-j\beta R}}{R} \, ds'
\]

where \( \vec{J}_{S} \) and \( \vec{M}_{S} \) are the currents due to the magnetic and the electric field, respectively, the primed Cartesian coordinates \((x',y',z')\) represent the position of the radiation point and \( R \) is the distance from each point of radiation to the observation point, as illustrated in Figure 2.3.

The electromagnetic waves radiated by antennas are spherical, but when at a considerable distance from the source, they appear to be planar and, therefore, the radial component of the electric and magnetic fields can be neglected, leaving only significant dependencies to the contribution of the angular components \( \theta \) and \( \phi \). This distance at which the wave is considered planar is called the far field and it corresponds to a distance \( r \) at which \( \beta r \gg 1 \), where \( \beta \) represents the phase constant of the wave.

At last, the electric and magnetic fields can be obtained from the vector potentials \( \vec{A} \) and \( \vec{F} \) using (2.6) to (2.9).
2.2.2 Field equations in the far field region

The radiation and scattering equations in the far field region can be deduced and even simplified, by considering the case of Figure 2.4, where the radial distance $\vec{R}$ from any point on the scatterer to the observation point is assumed to be parallel to the radial distance $\vec{r}$ from the origin to the observation point [3]. This approximation is satisfyingly accurate for cases where the radial distance $r$ fulfills the following relation to the largest dimension of the scatterer $D$ and the wavelength $\lambda$,

$$r \geq \frac{2D^2}{\lambda}.$$

The magnitudes of both vectors relate to each other by

$$R = \begin{cases} r - \vec{r} \cdot \hat{a}_r, & \text{for phase variations} \\ r, & \text{for amplitude variations} \end{cases} \quad \text{(2.11)}$$

where $\hat{a}_r$ is the unit vector that describes the direction of the vector $\vec{r}$.
Figure 2.4: Coordinate system for scattering in the far field region [3].

Thus, from (2.11), the equations for the vector potentials (2.4) and (2.5) can be written as a function of \( \vec{r} \) and \( \vec{r}' \) as

\[
\vec{A} = \frac{\mu}{4\pi} \int \int_S \vec{J}_S \frac{e^{-j\beta R}}{R} ds' \approx \frac{\mu e^{-j\beta r}}{4\pi r} \vec{N} \tag{2.12}
\]

and

\[
\vec{F} = \frac{\varepsilon}{4\pi} \int \int_S \vec{M}_S \frac{e^{-j\beta R}}{R} ds' \approx \frac{\varepsilon e^{-j\beta r}}{4\pi r} \vec{L}, \tag{2.13}
\]

where

\[
\vec{N} = \int \int_S \vec{J}_S e^{j\beta \vec{r}' \cdot \hat{a}_r} ds' \tag{2.14}
\]

and

\[
\vec{L} = \int \int_S \vec{M}_S e^{j\beta \vec{r}' \cdot \hat{a}_r} ds'. \tag{2.15}
\]

From the approximations for far field in (2.6) to (2.9), the equations for the electric and magnetic scattered fields can be obtained in a simple form as in (2.16) to (2.21).

\[
E_r \approx 0 \tag{2.16}
\]

\[
E_\theta \approx (E_A)_\theta + (E_F)_\theta \approx -\frac{j\beta e^{-j\beta r}}{4\pi r} (L_\theta + \eta N_\theta) \tag{2.17}
\]

\[
E_\phi \approx (E_A)_\phi + (E_F)_\phi \approx +\frac{j\beta e^{-j\beta r}}{4\pi r} (L_\theta - \eta N_\theta) \tag{2.18}
\]

\[
H_r \approx 0 \tag{2.19}
\]
\( H_\theta \simeq (H_A)_\theta + (H_F)_\theta \simeq + \frac{j\beta e^{-j\beta r}}{4\pi r} (N_\phi - \frac{L_\theta}{\eta}) \)  \( (2.20) \)

\( H_\phi \simeq (H_A)_\phi + (H_F)_\phi \simeq - \frac{j\beta e^{-j\beta r}}{4\pi r} (N_\theta + \frac{L_\phi}{\eta}) \)  \( (2.21) \)

Using the transformation from the spherical to the rectangular components written in the matrix form as

\[
\begin{pmatrix}
A_r \\
A_\theta \\
A_\phi
\end{pmatrix} =
\begin{pmatrix}
\sin \theta \cos \phi & \sin \theta \sin \phi & \cos \theta \\
\cos \theta \cos \phi & \cos \theta \sin \phi & -\sin \theta \\
-\sin \phi & \cos \phi & 0
\end{pmatrix}
\begin{pmatrix}
A_x \\
A_y \\
A_z
\end{pmatrix},
\]  \( (2.22) \)

the \( \theta \) and \( \phi \) components of \( \vec{N} \) and \( \vec{L} \) are given by \( (2.23) \) to \( (2.26) \).

\[
N_\theta = \iint_S (J_x \cos \theta \cos \phi + J_y \cos \theta \sin \phi - J_z \sin \theta) e^{+j\beta r \cdot \hat{\alpha}} \, ds'
\]  \( (2.23) \)

\[
N_\phi = \iint_S (-J_x \sin \phi + J_y \cos \phi) e^{+j\beta r \cdot \hat{\alpha}} \, ds'
\]  \( (2.24) \)

\[
L_\theta = \iint_S (M_x \cos \theta \cos \phi + M_y \cos \theta \sin \phi - M_z \sin \theta) e^{+j\beta r \cdot \hat{\alpha}} \, ds'
\]  \( (2.25) \)

\[
L_\phi = \iint_S (-M_x \sin \phi + M_y \cos \phi) e^{+j\beta r \cdot \hat{\alpha}} \, ds'
\]  \( (2.26) \)

2.3 Radar Cross Section

The quantity that evaluates the electromagnetic scattering by a target is the radar cross section, also frequently abbreviated as RCS. Usually represented by \( \sigma \), the radar cross section is defined as the area intercepting the amount of power that, when scattered isotropically, produces at the receiver a density that is equal to the density scattered by the actual target [3]. In other words, this concept gives an idea of the amount of scattered power in relation to the one incident on a certain area of a target, specifically, at its visible area.

There are two main representations for this measurement, considering the geometry of the problem [3]. If a three-dimensional target is measured, the result is defined as a radar cross section per unit area, hence its units given by \( dBsm \). On the other hand, if the analysis derives from a two-dimensional target, then the results return what is called a scattering width, which is equivalent to a radar cross section per unit length, expressed in \( dBm \).

Furthermore, there are two available techniques for the implementation of an RCS measurement. If a measurement is monostatic (also referred to as backscattered), it means that the transmitter and the receiver are in the same position. This is the case of a conventional automotive radar, which both transmits and receives the electromagnetic waves in the same position in space. A radar cross section can also be bistatic, and that happens when the transmitter and the receiver are in different locations.

An important consideration for the measurement of the radar cross section of objects is the assumption that the observation points are at an infinite distance from the target, which can only be approximated in reality in the far field region of the transmitter \( (2D^2/\lambda) \), where \( D \) is the largest dimension of the target [3].
For a three-dimensional target, the RCS per unit area can be calculated from the scattered and incident power, or from the electric or magnetic scattered and incident fields, such as in (2.27).

\[
\sigma = \lim_{r \to \infty} \left[ \frac{4\pi r^2 S^s}{S^i} \right] = \lim_{r \to \infty} \left[ \frac{4\pi r^2 |\vec{E}^s|^2}{|\vec{E}^i|^2} \right] = \lim_{r \to \infty} \left[ \frac{4\pi r^2 |\vec{H}^s|^2}{|\vec{H}^i|^2} \right]
\] (2.27)

The radar cross section consists of a very important parameter that characterizes the scattering properties of a target. When obtaining RCS results, RCS patterns can be obtained, which are essentially plots of the radar cross section as a function of space coordinates.

Real radar measurements of standard objects, in particular their backscattering behaviour [15], can be of great help when developing models for scattering and radar cross section simulations for high frequency approximations, such as geometrical optics or physical optics.

### 2.4 Scattering from a rectangular plate

In the presence of a flat rectangular plate, the scattering of an uniform plane wave can be determined using the concepts and equations obtained in the previous sections.

To better understand the solution of a scattering problem, the example from Figure 2.5 described in [3] is presented. A plate with dimensions \(a \times b\) and aligned with the \(xy\) plane is assumed. As a simplification, the incident wave can be chosen as TE\(_x\), which means that the electric field vector is perpendicular to the \(x\) axis. Nonetheless, the problem can be solved for any other field configuration.

![Figure 2.5: Perfectly electric conducting rectangular plate illuminated by uniform plane wave](image)

The incident electric and magnetic fields are given by the solution of the wave equation for a standing wave for a known magnitude of the magnetic field, \(H_0\), as

\[
\vec{E}^i = \eta H_0 (\hat{a}_y \cos \theta_i + \hat{a}_z \sin \theta_i) e^{-j\beta r}
\] (2.28)
\[ \vec{H} \equiv \hat{a}_x H_0 e^{-\beta \cdot \vec{r}} \]  \hspace{1cm} (2.29)

respectively, where \( \theta_i \) is the angle between the direction of propagation of the incident wave and the normal of the surface, \( \beta \) is the phase constant of the wave, with direction correspondent to its propagation direction and \( \vec{r} \) is the position vector.

Assuming a plate with finite dimensions such as the one on this example, the scattered angle is not exactly equal to the reflected one, calculated by image theory. This only happens in the theoretical case where the plate is infinite in all directions, when geometrical optics and physical optics reduce to each other \([3]\). Thus, one can use physical optics techniques to determine the scattered fields in this case.

The induced current at the surface is determined with (2.3) at the point of incidence \((x', y', 0)\), which becomes

\[ \vec{J}_S = 2\hat{n} \times \vec{H} \bigg|_{z = 0} = 2\hat{a}_z \times \hat{a}_x H_0 e^{-\beta \cdot \vec{r}} \bigg|_{z = 0} = \hat{a}_y 2H_0 e^{-\beta \cdot \vec{r}} \bigg|_{z = 0}. \]  \hspace{1cm} (2.30)

In this case, both \( x \) and \( z \) components of the induced current are zero, assuming this vector the direction along the \( y \) axis.

Since the plate is perfectly electric conducting, the induced current due to the electric field, \( \vec{M}_S \), is zero, which is why only \( N_\theta \) and \( N_\phi \) need to be calculated, using (2.23) and (2.24) and resulting in

\[ N_\theta = 2abH_0 \cos \phi_s \sin \phi_s \int_{-\frac{b}{2}}^{\frac{b}{2}} \int_{-\frac{a}{2}}^{\frac{a}{2}} \left[ e^{-\beta \cdot \vec{r} + \beta \cdot \vec{r}' \cdot \hat{a}_r} \right] dx'dy' \]  \hspace{1cm} (2.31)

and

\[ N_\phi = 2abH_0 \cos \phi_s \int_{-\frac{b}{2}}^{\frac{b}{2}} \int_{-\frac{a}{2}}^{\frac{a}{2}} \left[ e^{-\beta \cdot \vec{r} + \beta \cdot \vec{r}' \cdot \hat{a}_r} \right] dx'dy' \]  \hspace{1cm} (2.32)

where \( \phi_s \) and \( \phi_s \) represent the radial components of the scattering point \((x', y', z')\) when expressed in spherical coordinates.

Since

\[ \int_{-\frac{c}{2}}^{\frac{c}{2}} e^{\gamma z} dz = c \left[ \frac{\sin \left( \frac{\gamma c}{2} \right)}{\frac{\gamma}{2}} \right] \]  \hspace{1cm} (2.33)

holds for any integral of this kind, the integrals in (2.31) and (2.32) are explicitly solved, reducing to

\[ N_\theta = 2abH_0 \cos \theta_s \sin \phi_s \cdot \frac{\sin \left( \frac{b}{2} \beta \cdot X \right)}{\frac{b}{2} \beta \cdot X} \cdot \frac{\sin \left( \frac{b}{2} \beta \cdot Y \right)}{\frac{b}{2} \beta \cdot Y} \]  \hspace{1cm} (2.34)

and

\[ N_\phi = 2abH_0 \cos \phi_s \cdot \frac{\sin \left( \frac{b}{2} \beta \cdot X \right)}{\frac{b}{2} \beta \cdot X} \cdot \frac{\sin \left( \frac{b}{2} \beta \cdot Y \right)}{\frac{b}{2} \beta \cdot Y} \]  \hspace{1cm} (2.35)

where \( X \) and \( Y \) are respectively the \( x \) and \( y \) components of the inner product \((-\beta \cdot \vec{r} + \beta \cdot \vec{r}' \cdot \hat{a}_r)\).
In this specific example, given the geometry of the problem, 

\[ X = \sin \theta_s \cos \phi_s \]  

(2.36)

and 

\[ Y = \sin \theta_s \sin \phi_s - \sin \theta_i. \]  

(2.37)

Finally, the electric and magnetic scattered fields are obtained by applying (2.16) to (2.21). Furthermore, the plane that contains the maximum scattered field is the one defined by \( \phi_s = \pi/2, 3\pi/2 \) and \( 0 \leq \theta_s \leq \pi/2 \), which corresponds to the plane parallel to the incident plane. In this situation, the electric field is given by 

\[ E_s^r \simeq E_s^\phi \simeq 0 \]  

(2.38)

\[ E_s^e \simeq -j\eta ab \beta H_0 e^{-j\beta r} \frac{\sin \frac{\beta b (\sin \theta_s - \sin \theta_i)}{2}}{r \cos \theta_s} \frac{\sin \frac{\beta b (\sin \theta_s - \sin \theta_i)}{2}}{r \sin \theta_s \sin \theta_i}. \]  

(2.39)

From (2.27), the RCS is given only as a function of the dimensions of the plate \( a \) and \( b \) and the wavelength \( \lambda \), since the wave propagation constant can be described as \( \beta = \frac{2\pi}{\lambda} \). The RCS is, thus, 

\[ \sigma = \lim_{r \to \infty} 4\pi r^2 \left| \frac{E_s^e}{E_i^e} \right|^2 = 4\pi \left( \frac{ab}{\lambda} \right)^2 \cos^2 \theta_s \left[ \sin \left( \frac{\beta b (\sin \theta_s - \sin \theta_i)}{2} \right) \right]^2 \]  

(2.40)

In the case of monostatic radar cross section, or backscattering, the angle of scattering is always the same as the angle of incidence, since the observation point is the same as the transmission point. Therefore, (2.40) simplifies to 

\[ \sigma = 4\pi \left( \frac{ab}{\lambda} \right)^2 \cos^2 \theta_i \left[ \sin \left( \frac{\beta b \sin \theta_i}{2} \right) \right]^2 \]  

(2.41)

The analysis of the dependencies between the wavelength \( \lambda \) and the dimensions of the plate \( a \) and \( b \) on (2.40) concludes that, when \( b \gg \lambda \), the maximum is located at the specular direction, where \( \theta_s = \theta_i \). This shows the conformity of this technique with geometrical optics for the case of scattering by a plate of large dimensions.

This can be proven, in this case, by the influence of the \( \sin z/z \) function. The maximum of this plot is more accurate and very clear for large dimensions on the plate when compared to the wavelength.

A plot of (2.40) can be found in Figure 2.6 where the dimensions of the plate were chosen to be \( a = b = 5\lambda \). The division of the equation by the square of the wavelength lets the results be independent of \( \lambda \).

As it is observed in the plot, the maximum of the backscattering calculation is located on the direction of normal incidence on the plate, for \( \theta_i = 0^\circ \).

For the case of the bistatic radar cross section of the same plate, as in Figure 2.7 but an angle of incidence of \( \theta_i = 30^\circ \), the maximum occurs at \( \theta_s \simeq \theta_i = 30^\circ \).

Based on the simplicity of this method, physical optics shows to be quite useful when evaluating the radar cross section of objects. When aiming to calculate the scattering by an extended target, one must note that scattering contributes very significantly in most occasions and should therefore be determined.
Figure 2.6: Monostatic radar cross section per squared wavelength of a rectangular plate [2].

Figure 2.7: Bistatic radar cross section per squared wavelength of a rectangular plate for an incident angle of $\theta_i = 30^\circ$, $\phi_i = 270^\circ$ [2].
One way of tackling this situation, which is suggested and investigated in this thesis, is to take every finite big surface of a complex target and divide them in many small polygonal surfaces. The scattering at these small individual fragments of surface can be determined exactly as in the example above. Afterwards, the calculations can be repeated for each surface and combined, ultimately determining the overall scattered field by the extended target [16].

2.5 Ray Tracing

So far, some strategies to model ray-optical wave propagation in the presence of various obstacles have been mentioned and briefly explained. However, one question still remains about how to simulate the trajectories when the path taken by the rays between the transmitter and the receiver is unknown.

One method that can be used to solve this problem is the ray tracing. This method is composed essentially by two calculations. First, the determination of the ray trajectory between the transmitter and the receiver, and second, the calculation of the field strength on the receiver, based on the propagation phenomena endured by the ray during this trajectory [2]. Since the latter involves the “tracing” (or following) of the ray during the path, which is determined on the former, usually both stages are inherently dependent on each other and should therefore be performed simultaneously.

While purely theoretical or purely empirical models are only valid in constrained environments, the ray tracing model appears as a generic deterministic propagation modeling and simulation tool. The ray tracing technique, when combined with the solving of Maxwell’s equations and respective approximations in the high frequency regime, is able to provide estimates for path loss, angle of arrival and departure, time delay, and channel characteristics such as Doppler shift, power delay profile and angular power spectrum [10].

By employing the ray concept introduced in subsection 1.2.1, the ray tracing process uses basic algorithms to describe electromagnetic waves using several known propagation mechanisms. In the case of free space propagation, for instance, there is only one single ray which is defined by a straight line that is originated in the transmitter and presents a certain direction. When simulating the case of reflection by a perfect electric conducting surface, on the other hand, two rays are necessary, the incident and the reflected. The latter has its origin located in the reflection point where the first one hits the surface and its direction is determined as a function of the direction of the incident ray, following the fundamental law of reflection.

However, in a more complex environment such as the urban one, where automotive radars are most required, mainly exist many different paths (called multipaths) between the same source and destination, which are subjected to many interactions with several different interfaces within their trajectory [10].

To determine these paths can be an exhaustive calculation, especially due to the large amount of rays that interact simultaneously on a scene, but some basic algorithms have been developed which can actually perform ray tracing in a fairly simple way. The methods that can be applied on the rays to implement their tracing can be either direct or indirect, and both are explained in the following sections.

2.6 Direct methods of ray tracing

The direct methods of ray tracing consist in the use of algorithms that allow a determination of the exact ray paths between the transmitter and the receiver points. These simple and straightforward algorithms constitute a good strategy to develop very simplified ray tracer technologies, given that they are so easy to implement.
However, these methods present some significant disadvantages and limitations, as explored further, proving them to be insufficient at times, especially when addressing complex propagation mechanisms in elaborate scenarios.

Some different methods, called indirect, might help filling these gaps in the ray tracing problem. It remains to decide which type of method is more appropriated and efficient for each specific objective.

Two examples of direct methods of ray tracing are given and briefly explained.

### 2.6.1 Fermat’s principle of least time

Fermat’s principle of least time is one of the algorithms used to determine the path that a ray covers from one point to another in a propagation environment. It defines that this path is determined by the route that consumes the least possible amount of time.

This property of minimum optical length allows the determination of optical distances with sufficient accuracy when the course of the rays is only approximately known [17]. On the case of ray tracing, the main objective is to determine not only the distance traversed by the ray, but also the path that the ray takes in real circumstances, and this can be answered by this method in a very effective way.

Also, the course of the rays determined by this process leads to the correct construction according to laws of geometrical optics [17]. The laws of reflection, transmission and diffraction can be derived from it, and it also provides bases for the image method, commonly used to model reflection phenomena [10].

### 2.6.2 Image method

To determine the trajectory of a ray reflected at a plane surface, the image method states that the path can be defined by three points, $T_x$, $R_x$ and $Q$ [10], illustrated in Figure 2.8. The two first ones are known and they correspond to the transmitter and receiver locations respectively, but it remains to calculate the intermediate point $Q$, where the reflection occurs.

Firstly, one should make the assumption that the surface of incidence is approximately infinite [3], which means that it is wide enough in all directions so that the most predominant effect on the wave is the specular reflection.

The image method consists of introducing virtual points (images) that account for the reflections. In Figure 2.8 the image of the receiver point $R_x$ is determined by its symmetrical point with respect to the reflection surface.

Figure 2.8: 2D geometrical representation of the image method [2].
This image is imaginary and it is created for calculation purposes only. When combined with the real points, it forms an equivalent system that replaces the actual one, resulting in the same fields as in the real situation [3].

To trace the ray path, one should then connect the transmitter point $Tx$ and the image $Ri$, defining the point of intersection with the surface, $Q$. Finally, the trajectory is defined by the correct points.

In an absolutely equivalent way, one can also create the image of the transmitter point and connect it to the receiver to calculate the point of reflection.

The nature of this method allows it to be applied recursively to trace rays that withstand multiple reflections. However, in a realistic scenario, the exclusive use of the image method might be rather inefficient in terms of computational speed, due to the large number of reflective surfaces that might exist [10]. This is considered a major disadvantage, especially when the objective of the model is to perform ray tracing in real time. This is a very ambitious objective for such models, but there are, in fact, indirect methods of ray tracing that enable a much faster computational process of the data.

### 2.7 Ray Launching: the indirect approach for ray tracing

Ray launching is the name given to the indirect approach of ray tracing and it consists on the shooting of rays from the transmitter in arbitrary directions and their tracing until they eventually hit the receiver or until they surpass a certain level of attenuation [2].

Also known as the shooting and bouncing ray (SBR) method, this algorithm traces each ray launched from a source location with the purpose of determining whether it reaches the receiver or not, as depicted in Figure 2.9.

![Figure 2.9: Principle of ray launching](image)

The first step of this method is the process of launching the rays. These can be sent out from the receiver in arbitrary or specific directions. However, it is usually required that all the rays are as uniformly distributed as possible so that each ray carries similar power for an isotropic source [10].

The second step includes the actual tracing of the ray. Generally, in an urban environment, the complex geometry and extensive amount of surfaces lead to many intersections of each ray with many objects in the scene [10]. Then, the wave propagation is calculated according to the chosen and adequate mechanism, originating, for example, reflected, diffracted or even scattered rays [2]. This trace ends when the ray finally hits the receiver or, instead, after it surpasses a maximum level of attenuation.

Finally, at the receiver, all incoming rays are combined as contributions, creating the total received field at this point. Normally, only a few of the traced rays reach the receiver [2], which is why this approach of ray tracing is referred to as indirect.

The most significant advantage of ray launching when compared to the direct methods is its applicability in curved geometries. For this reason, so many investigations in wave propagation modelling in
tunnels have used this method, due to the curved nature of the walls. Nevertheless, direct and indirect methods are not mutually exclusive, since they both present advantages for different kinds of problems.

### 2.7.1 Shooting and bouncing ray image approach

The principle of ray launching to validate rays that reach the receiver can be combined with the precision of the trajectory calculation by the image method, resulting in a hybrid and more accurate method, called the shooting and bouncing ray image approach.

This is a deterministic model proposed in [18] and it is introduced as a modified shooting and bouncing ray method. Fundamentally, it consists on finding the equivalent sources, also called images while applying the ray tracing method on each ray tube, and coherently summing the received complex amplitudes of all contributing rays, as stated in the conventional shooting and bouncing ray algorithm.

To apply this technique along with the image method, one should note that an approximation for curved surfaces should be assumed. Since image theory only applies to planar surfaces, the surrounding environment can be approximated by the decomposition of the surfaces into small planar triangular facets [18]. The good agreement between the results from simulations run with this method and the theoretical results of canonical examples shows that this approach provides a good approximation for high-frequency radio propagation inside tunnels where reflection is dominant.

### 2.7.2 Backward ray tracing

The main disadvantage concerning ray launching is that only a fraction of the total amount of rays that are launched from the source eventually reach the receiver. This makes the process highly inefficient, not to mention time consuming.

Backward ray tracing arises as a solution for this problem, making it more efficient [19]. In this case, the rays are launched from the receiver, usually represented as an eye, as illustrated in Figure 2.10.

![Figure 2.10: Different strategies for designing the ray tracing algorithm.](image)

Each ray is then traced from the receiver into the scene. If it hits an object, the way to find how much energy is carried in that ray at that point, is to trace another ray from the intersection point towards the transmitter. It is possible that this ray encounters another object in its path, or even reach the transmitter. If a ray does not eventually reach the transmitter, then it means that the original intersection point is in a shadow, because it is not visible by the transmitter in the first place [20].

The strategy of reversing the ray tracing method is especially useful to quickly determine visibility between two points and mirror reflections. However, when the ray tracing problem includes more complicated mechanisms such as scattering of rays into unpredictable directions, for instance, the algorithm must be developed as a kind of compromise between both concepts [21], which will be explored in the
course of this thesis.

The ray launching concept has been introduced as an alternative approach for the ray tracing direct methods, but one underlying problem of this process still hasn’t been clarified, and that is how to determine whether a ray hits the receiver or not. To answer this question, there are two main methods that can be selected, which are elucidated further.

### 2.7.3 Discrete ray tubes

Considering the definition of a tube of rays introduced by geometrical optics, it is designated that each tube is delimited by up to four lines that define one reference ray per tube in their center, which represents its correspondent locally plane wave.

Assuming this, determining whether a receiver is hit by a tube of rays or not is extremely simple. If the receiver point falls within the discrete tube of rays at some point, it is directly inferred that the ray tube hits the receiver, hence contributing to the received field [2].

A two-dimensional representation of this phenomena is presented in Figure 2.11. Here, it is possible to solve the propagation path of the tube of rays through the implementation of image theory.

![Figure 2.11: Reception of rays using discrete ray tubes][2]

Not only is this method quite simple to implement, but it is also fairly efficient when it comes to determining if the receiver is within the cross section of a tube of rays. However, should this approach be applied by itself, the resulting received field would be most certainly wrong in most cases. As it is observed in Figure 2.11 one receiver is hit by two tubes of rays, which coincidentally belong to the same incident plane wave. This phenomena is referred to as multipath and it is inevitable in a complex scene with several surfaces being hit by rays. In this situation, it is essential to determine how many multipaths exist, so as to properly calculate the received field, which can constitute a quite overwhelming complication.

### 2.7.4 Reception spheres

Alternatively to considering tubes of rays, one can assign to each receiver a certain volume instead. In most cases, the chosen shape for such volumes is a sphere, since it is simple to determine its geometrical intersection with a ray. Fundamentally, if a ray eventually intersects the sphere that represents the receiver point, it means that it reaches the receiver.

The biggest challenge associated with this approach is the determination of the size of the sphere, since too large dimensions result in several identical rays being interpreted as different, and too small dimensions can lead to some essential propagation paths being neglected [2].
Moreover, Figure 2.12 proves that the number of validated rays depends on the distance at which the receiver is from the transmitter, since for a similar sphere size, more rays hit the sphere when this is located closer to the ray transmitting point.

![Figure 2.12: Validation of multiple rays, using constant size reception spheres [2].](image)

One way to solve this situation is to create adaptive reception spheres [2], where the ray is validated only if its distance to the center of the sphere is less than half the separation between rays at that particular point. The smaller the angular ray separation at the transmitter, the more probable it is for the validated ray to pass directly through the center of the receiver.

Another angular separation process can also be achieved by the process described in [5], which describes the surface of the transmitter as a regular icosahedron with tessellated triangular faces. Twelve of the launched rays are derived from the polyhedron’s vertices, creating pentagonal wavefronts, but the majority of the rays are launched from the intersection points between triangular facets, resulting in hexagonal wavefronts, as illustrated in Figure 2.13. Therefore, by the geometry of hexagonal shapes, one can calculate the radius of the reception spheres as a function of the angular separation between rays and the distance of the observation point to the transmitter [2].

![Figure 2.13: Tessellation of an icosahedron’s face and originated wavefronts [5].](image)

Since the reception spheres are round and the wavefronts hexagonal, there will be a small superposition between spheres. So, to eliminate multiple received rays, each ray should be identified.

This procedure can be implemented with any chosen geometry, not only for the receiver, but also for the shape of the wavefronts. Nonetheless, the identification of the rays is essential for a correct determination of the received field and should always be implemented.

This identification can be based on the number of reflections, delay time and angle of transmission of each ray [2]. If two received rays present the same conditions for the three criteria, it is concluded that they ultimately correspond to the same ray and should only be validated as one.
Both methods introduced above are assumed to be applied in the presence of planar surfaces. When applied on curved or irregular surfaces, discrete tubes of rays can either converge or diverge, whether the interface is concave or convex, respectively. Thus, these models might be insufficient when solving the wave propagation phenomena at such interfaces.

Consequently, another strategy can be implemented which enables the validation of launched rays at the receiver in other types of surfaces. In [2], an approach for curved surfaces is suggested, with the objective of applying it to tunnel environments.

2.7.5 Ray density normalization

As it was previously concluded, both methods of discrete ray tubes and reception spheres rely on a single ray to reach the receiver and contribute to the received field. However, this is sometimes not the case, especially in the presence of convex or concave surfaces, and a method that allows the determination of the number or repeated rays at the receiver is advantageous.

Reception spheres allow an additional process of normalization which is also valid in surface geometries other than planar [2]. The ray density normalization (RDN) assumes, a priori, that several multiple rays (which represent the same wave) are found on the same physical propagation path. Thus, there is a clear distinction between this method and the previously stated ones that follow classical geometrical optics, as in this case each locally plane wavefront is represented by multiple ray contributions.

The fundamental concept of ray density normalization has the purpose of determining the number of rays that reach the receiver from the same propagation path. For this, the concept of ray density is introduced, which consists on the number of rays per unit area [2]. To directly calculate the total number of incident rays on a sphere, one can simply multiply the ray density by the area of the surface of the sphere which is visible from the ray’s perspective.

Equivalently, the same logic applies to the number of rays radiated from a transmitter, in order to calculate the ray density at a given point of the ray’s trajectory. However, when coming across non-planar surface geometries, the relation is not trivial anymore, since the trajectory of the rays can be altered and, consequently, the separation between them.

In the case of curved geometries, the principle of geometrical optics for reflection in such types of surfaces can be applied, as introduced in subsection 1.2.3 and calculated in [3]. In [2], the formula that calculates the ray density after reflection in a curved surface is inferred as a function of the incident ray density, the radius of curvature of the surface after reflection and the distance to the transmitter. Other geometries of surfaces can be analyzed in analogy to the curved one, as long as there is a way of calculating the ray density normalization after the interaction between the ray and the surface.

In the event that a ray hits the receiver, the theoretical number of rays that reach the receiver through the same propagation path can be calculated by multiplying the ray density at that point by the visible area of the receiver normal to the propagation direction of the ray. Of course, these are multiple rays, which means they have the same physical properties, such as direction of arrival, delay and number of reflections [2]. Therefore, the ray density normalization process consists of weighting these rays in a way that the combination of the calculated number of rays leads to the same result as if only one single ray would have reached the receiver. This normalization can be implemented by two different approaches [2]. Either by assigning a certain electric field to each ray (field trace) or by analyzing its power (power trace), the values of all contributions at the receiver are summed and normalized for a single propagation path, always depending on the ray density.
As a final consideration, it should be noted that, in order to apply this principle, a considerably large number of rays should be used [2]. The calculations to determine the theoretical number of validated rays return a real valued number, but the actual registered rays should be an integer. The number of launched rays should therefore be sufficiently large to minimize the error introduced by the weighting.

2.8 Radar Technology

For a better understanding of the radar operation, essential for the adequate simulation of its behaviour, a brief introduction on the basic concepts of radar technology is included.

The radar electronic system consists of a transmitter, a receiver, both respective transmitting and receiving antennas, and a processor [16]. In most radars, the transmitter and the receiver are located in the same place, since they are meant to perform backscattering measurements. To obtain bistatic measurements of targets, a system with two different radars can be arranged, where one works as a transmitter and another as a receiver.

The transmit antenna emits a beam of electromagnetic waves in the desired direction. The directivity of the antenna is related to the field of view of the radar.

Once the emitted wave hits an object, it is scattered, producing what is called an echo-signal. The receiving antenna gathers the signal and the receiver delivers it to the processing stage, where the signal is amplified and converted into legible values or diagrams. After processing the received waves, the radar measures the distance to all the detected objects in its angular range. Furthermore, with the evaluation of the magnitude and direction of each received electromagnetic wave, the radar system is capable of determining not only the position of the target in space, but also its size, shape, velocity and direction of motion [16].

The electromagnetic waves used in radar technology are in the high frequency radio range. More specifically, collision avoidance radar systems, used in autonomous driving have been assigned the frequency range between 76 and 77 GHz by the Federal Communications Commission [22]. This specific range presents a long detection range, better spacial position resolution and less sensitivity to types of roads and weather, which makes it extremely well suited for this kind of systems.

Monostatic measurements provide the radar cross section of targets as a function of the observation angle. From these sets of results, one can additionally determine the object’s most relevant scattering centers. On the specific case described in [15], these results are obtained by performing successive measurements while rotating the vehicles up to 180°, obtaining results in terms of the azimuth angle.

The radar cross section patterns confirm that each type of vehicle presents certain characteristics, reflected by its backscattering behaviour. The larger the vehicle, the larger the corresponding radar cross section. In the same line of thought, for vehicles like vans or trucks, the radar cross section is much more concentrated at azimuth angles around 90°, since its side structure can almost be approximated by a large reflective plane, like in the case of specular reflection for geometrical optics [15].

Associating the most common obstacles in the automotive radar environment, such as different types of vehicles and pedestrians, with a specific backscattering pattern, allows a fast and effective recognition of each target’s shape and its almost immediate identification. Radars are, therefore, not only capable of determining the position of different objects in space, their velocities and direction of movement, but also identifying different varieties of targets and recognizing their nature, consequently significantly broadening the possibilities of autonomous driving.
Chapter 3

Simulator Setup

In this chapter, the concept of the implemented simulator is determined. The most important considerations about the algorithm are outlined, along with the configuration design of the simulation environment. Finally, the monostatic and bistatic simulation setups are described and the simulation variables are specified.

3.1 Simulation Concept

The objective of the simulator designed in this thesis is to calculate the scattering of an extended target and determine its radar cross section pattern as a result.

The scattering is calculated through the application of the physical optics approximation. This simulator is intended to test the quality of the results in comparison with other available simulators based on different techniques, evaluating whether this approach is appropriate and which aspects condition its operation.

The program is based on the ray tracing algorithm in one of its many forms. Usually, these types of algorithms that involve image processing are implemented at the graphics processing unit (GPU) level, for a faster computer processing and a better memory management [23]. However, this program is implemented from scratch and it is beyond the scope of this thesis to acquire the knowledge to manipulate this programming language. Therefore, the code is implemented for testing purposes in C++ language, in a fairly simple and non-optimized way. Should the results prove to be satisfying and promising, an optimization of the code is vital, including possibly its implementation on the GPU.

The analyzed target or overall scenario is represented in the simulator program as one or more three-dimensional digital models that approximate the actual shape of the objects in real life. As expected, the model is composed by surfaces of variable shapes, sizes and materials.

The idea behind this simulation concept is that, when hitting the surface, each ray is responsible for the surface area that belongs to its respective cross section. One can think of this method as a decomposition of the whole surface into smaller shapes that, together, amount to the whole area of the surface in question. Figure 3.1 illustrates a generic surface that is hit by several rays and decomposed into representative areas. Each incident ray induces current at the section of surface correspondent to its area, being the scattered rays subsequently launched from there back into the surrounding environment, in all directions.

By assuming that all the surfaces are flat and perfect electric conducting, the approximation is valid and the scattering by each small polygon can be calculated. By determining all the scattered fields
for every decomposition section through the physical optics approximation, the scattering of the whole object can be obtained [18].

Figure 3.1: Decomposition of a representative surface into small polygons.

### 3.2 Algorithm Considerations

A crucial aspect of implementing a procedure that relies on an approximation is that it introduces several assumptions which need to be considered and conditions that must be fulfilled for the concept to be valid in the first place.

As stated above, the first essential condition required by this concept is that all the objects in the scene have perfect electric conducting surfaces, even though sometimes, in a real environment, this does not happen. Scattering can be calculated in the presence of different materials, but not through the physical optics approximation. This, of course, has influence on the final results of the simulations preformed by this code. Nevertheless, it is still of interest to analyze the scattering by perfect electric conductors or other objects whose materials and properties can be approximated by perfect electric conducting properties. In the eventuality that the results are satisfactory for these types of targets, additional considerations can be added to the algorithm to make it suitable for any other surface properties.

Another relevant requirement of the simulator configuration is that the targets on the scene are located in the far field of the radar. As settled in section 2.2 and section 2.3, the calculations of the scattering equation and radar cross section involve approximations that are only valid when the target is at a certain distance from the transmitter, depending on the relation between its dimensions and the selected wavelength. Furthermore, the rays launched from the radar should also travel a minimum distance before reaching any target in order to be approximately parallel to each other. Only in this case, they comprise a locally plane wave, featuring straight wavefronts, instead of curved ones that represent a spherical wave in the vicinity of the transmitter point. For these reasons, the position of the radar during the simulation must be designated so that a reasonable distance to the targets is assured at all times.

Finally, considering the decomposition of the surfaces into smaller polygons, it is important to have an idea of the amount of rays that should hit each surface, or in other words, which dimensions should the decomposition polygons assume.

Each decomposition polygon is defined by the area of the surface that belongs to each incident ray. Thus, the size of the polygons is directly connected to the ray density. If more rays are launched per unit area from the transmitter, more rays will intersect each surface and thus the resulting decomposition polygons will present smaller dimensions.

As concluded in section 2.2, the radar cross section of a rectangular perfect electric conducting plate derived from physical optics varies with the dimensions of the plate. Not only is the maximum value
consistently dependent on the area of the surface, but also for angles other than the specular one, the oscillation rate is very susceptible to the length of the plate.

The magnitude of the scattered field varies as a function of the scattering angle. This entails that, when two rays hit the surface at different points, the scattered rays will match their field magnitude in terms of their direction and not their position in space. Figure 3.2 demonstrates how two scattered rays originated from two different polygons with the same area have the same field magnitude when they present the same direction, but since they are parallel, they will never be received at the same point and therefore, the field is not properly calculated. A possible countermeasure to this problem is to launch the least possible amount of rays, in order to minimize the decomposition of the surface, consequently minimizing the error associated with this operation. Minimizing the number of incident rays also significantly contributes for a smaller compilation time, which already constitutes an enormous advantage.

![Figure 3.2: Two correspondent rays with the same direction.](image)

On the other hand, because the concept assumes that the same medium is inside and outside of the surface of incidence, it is best that the surface is divided into many small polygons, so that the majority of them are actually surrounded by the same medium and only a fraction of them is actually on the edge of the surface, minimizing the error associated with the approximation derived from the edge effects. In the same line of thought, the induced current is assumed to be constant in the area of incidence, which means that a smaller area will lead to more accurate results. Additionally, in the presence of curved or irregular surfaces, it is decisive to have small polygons, in order to obtain a more approximate representation of the object’s shape.

This means that a compromise should be taken into account between the size of the polygons and the accuracy of the results. Varying the number of rays during simulations should give an idea of the degree of influence that this parameter has on the results.

### 3.3 Ray Tracing Method

Now that the concept of the simulator has been clarified, the logic behind the implementation of the program can be described. The general algorithm bases itself on the ray tracing method, more specifically, the ray launching technique. In its original definition, ray tracing is used to exclusively compute the
visibility between points. It is mostly used to render three-dimensional images, but when complemented with algorithms that compute propagation of light through space, for example, this technique determines the color of a point in the scene \cite{24}. Analogously, if implemented along with an algorithm that computes the scattering phenomena of electromagnetic waves, the program is able to determine the scattered field at a point of the scene. The ray tracing method is systematized in this section.

The operation of an actual radar commences when one incident plane wave is launched from the transmitter point. Equivalently, in the program, a beam of incident rays, representing the incident wave is launched and traced throughout its propagation path.

These incident rays should be launched in the direction that the radar is facing and they should be evenly distributed in space, in order to correctly represent an isotropic source \cite{20}. Usually, in ray tracing the rays are equidistant, which means that they are launched from a point through a grid with equally sized polygons, just like the one depicted in Figure 3.3. A two-dimensional representation of incident rays being launched from the radar is described in Figure 3.4. As it is evident, the rays consistently present the same distance between each other throughout the whole path that the wave covers, ultimately presenting planar wavefronts, just like in the case of a plane electromagnetic wave. Since this method is similar to the operation of a camera, the rays that are distributed this way are called camera rays.

Another way of launching rays into a scene is to distribute them evenly in terms of angular distance, as mentioned in \text{subsection 2.7.4}. This approach lets the transmitter be represented by a polyhedron and tessellates its surfaces into small triangles. Even though the rays are indeed evenly distributed in terms of angular direction, their wavefronts assume different geometries. This means that the tubes that represent the rays have different shapes and, therefore, different areas, which significantly complicates the process of ray tracing. Additionally, it is not efficient to launch the rays in every direction around the transmitter, since the radar only launches rays directed to its front side.
Therefore, although an angular even distribution could result in more accurate results, the equidistant rays appear as a more logic and resourceful approach to this case, especially because the measurements are performed only in the far field region, where one solution reduces to the other.

The next stage of the ray tracing algorithm deals with the arrival of the incident rays at the target. Depending on the size and shape of the object, but also on the number of rays and the angular range of the incident beam, some of the rays that are launched from the radar position might never reach the object, eventually not withstanding an intersection with any surface on the scene. These rays will not contribute for any further calculation and so it is important for the efficiency of the algorithm that they are quickly dismissed. The tracing of these rays takes time and memory to process and the computing should therefore not be continued for a ray that is not useful.

To understand how the incidence points are determined, the configuration of the three-dimensional digital model should be primarily understood. In computer graphics, geometries of any kind are most commonly represented by polygons meshes. This means that an object is composed only by polygons, that correspond directly to its faces. One face can be defined by three or more vertices and when combined together, the polygons create complex polyhedral shapes [25]. Three-dimensional digital models are always composed by planar surfaces, even when they appear to represent smooth curved objects such as spheres. In order to make a surface appear less "blocky", the model should be composed by a larger amount of smaller polygons.

With the objective of accelerating the computation and simplifying the calculations, meshes are exclusively composed by triangular facets. Any generic polygons representing the object's faces can be triangulated, which is the name given to the process of decomposing a polygon into triangles [26]. Therefore, any typical digital file that stores polygon meshes represents them in a series of triangular facets, each defined by a set of three vertices and a normal unit vector facing the outside of the model.

Thus, calculating the intersection between a ray and a face of the model consists simply on determining the geometrical three-dimensional point of intersection between a line and a triangle. A computation for every facet belonging to the model tests if the intersection point with an incident ray exists, determining whether the given ray is useful for the algorithm or not.

After hitting a surface, the ray is then scattered back into the same medium where it was propagating before. This is the stage where wave propagation algorithms must take part along with ray tracing, in order to compute the trajectories of the rays after colliding with the objects of the scene.

In the case of reflection by image theory, the calculation of the trajectory of a reflected ray can be very straightforward, especially because one incident ray results in the launching of another single reflected ray. However, scattering is a more complex mechanism. The area of the surface affected by the incident ray works as a source and the scattered wave assumes a spherical configuration, propagating in all directions. For this reason, it is essential for the correct simulation of this phenomena that for each incident ray that hits a surface, a whole new set of scattered rays is generated.

The scattered rays can be launched from the intersection point between the incident ray and the surface in the same way that the incident rays are launched from the radar position, as illustrated in Figure 3.5. Naturally, due to the limited amount of rays that can actually be launched and digitally processed, each ray works as a kind of sample and therefore the directions in which the scattered wave propagates are limited. It is thus beneficial that a large amount of rays is launched at the moment of scattering and that the angular range of the beam is broad, in order to approximate the propagation of a spherical wave, returning, subsequently, more accurate results.

It is also important to mention that, given the chosen generation algorithm for incident rays, the wavefronts consistently assume the shape of rectangles. When hitting a surface, the dimension of the area
for which the ray is responsible is given by the separation between the rays along the horizontal and vertical directions, referred to in Figure 3.6 as $a$ and $b$, respectively.

Figure 3.5: Launching scattered rays from two different scattering points A and B.

Figure 3.6: Surface decomposed into rectangles by the incidence of multiple rays.

The last stage of the algorithm is related to the reception of the scattered rays and their processing with the objective to calculate the radar cross section of the analyzed object.

As suggested in subsection 2.7.4, the receiver should be represented as a volume in space, more specifically, a sphere, for faster and simpler calculations. A generic representation of a reception sphere working as a receiver of scattered rays is shown in Figure 3.7. Just like when determining the intersection of an incident ray at the model, the principle of determining whether a scattered ray is received at the radar is by calculating the intersection point between the scattered ray and the reception sphere. Since it is simply the intersection between a line and a sphere, this calculation is very straightforward, as long as the center point and the radius of the sphere are known.

The sphere should be centered at the point that defines the transmission of the incident rays, which is also the center of the radar. Its radius should be dependent on the distance to the model and the separation between the received rays at that point. Logically, the dimensions of the reception sphere influence the number of rays that are actually received and processed, thus influencing the final results. The diameter of the sphere should be large enough so that it assuredly receives at least one ray from every scattered wave generated at the object. On the other hand, it shouldn’t be too large so that many multiple rays from the same wave need to be disregarded and therefore traced in vain.

If a ray does not have a valid intersection with the reception sphere, it means it is not yet received at the radar and should, therefore, continue to be traced to test whether an intersection with another
surface exists. This process should be successively repeated until a condition for the termination of the ray tracing is met, either by reaching the receiving point or by surpassing a certain level of attenuation.

Finally, when the tracing is finished and all the received rays are stored, the program determines the received scattered field by summing the received scattered fields from all received rays. The sum of all contributions amounts to the total scattered field and the radar cross section of the object can be calculated.

3.4 Bistatic Setup

The radar cross section of an object can be measured in two different ways depending whether the system presents a bistatic or a monostatic configuration. This section describes the configuration of a system that measures the bistatic radar cross section of a target.

Usually, to obtain relevant plots of radar cross sections of objects, measurements are performed in a way that all the sides of the object are detected. This can be achieved either by rotating the object around its center or by changing the position of the radar in a circle around it. The first strategy can be a quite demanding process, especially if the target presents very big dimensions. This is an extremely challenging process not only in real life for needing to move a very heavy and large mass, but also in computational terms, if an enormous amount of polygons depicting the surfaces of the object need to be transformed. Thus the second approach is the one chosen for the setup of this simulator, changing the position of the radar as desired while the position and orientation of the model remains unchanged.

The bistatic configuration is based on the assertion of an incident angle. The angular sweep of the bistatic simulation is given by the scattered field, which is received in different angles in relation to the object. Therefore, in a system of this kind, there is a defined position for the transmitter and a succession of positions for receivers [27].

In Figure 3.8a, the transmitter position is located in point $C$, while all the points $A$ through $E$ work successively as receivers. The reception of scattered field at all the receiving points gives the angular sweep of the measurement and results plotted as a function of the observation angle. Assuming that, in this case, the reference angle is the line that connects point $C$ to the geometrical center of the target, this configuration determines the bistatic radar cross section for an angle of incidence of 0 degrees.

The system configuration illustrated in Figure 3.8b measures the bistatic radar cross section for an
incident angle different than 0 degrees, being the incident wave oblique to the reference direction. In this case, point $B$ marks the position of the transmitter, while the receivers are located in points $A$ through $E$.

![Diagram of bistatic setup](image)

(a) Transmitter in point C, $\theta_i = 0^\circ$.
(b) Transmitter in point B, $\theta_i \neq 0^\circ$.

Figure 3.8: Bistatic setup.

It is important to note that the bistatic system configuration only analyzes the target from one perspective. Since there is only one transmission point in the whole simulation, the algorithm runs merely in one cycle, being the sweep given by the reception of the rays at the several receivers, but only corresponding to one single incident transmitted wave. The configuration is thus of quite simple character, but it does not detect the object from all fronts and therefore the obtained scattering pattern is not sufficient to characterize the target completely. This is why the results of the monostatic measurement are so important, because this type of configuration is capable of determining the radar cross section of an object at all its fronts, associating it with a meaningful scattering pattern.

### 3.5 Monostatic Setup

The premise of the monostatic configuration is that the transmitter and the receiver are located in the same position at all times [27]. This directly implies that the system only needs one radar to operate, which is in charge of both the transmission of the incident wave and the reception of the scattered rays.

In this case, the angular sweep is achieved by changing the direction in which the incident wave reaches the object. As mentioned in the previous section, this can be assured by successively changing the position of the radar, as shown in Figure 3.9. The measurement depicted in Figure 3.9a shows the radar located in point $B$, which both transmits and receives the processed rays along their paths. Another measurement is illustrated in Figure 3.9b where point $C$ is the radar position.

Concerning this setup, it is evident that for an angular sweep to be obtained, several different measurements need to take place, and in each particular one, only one point of the angular sweep is relevant. After determining the scattering results for points all around the target, a complete pattern for the radar cross section of the object can be obtained.

The angular separation between two consecutive radar positions is referred to as angular resolution and this quantity defines how many observation angles are represented in the radar cross section plot.
It is typical that, for a target that represents some kind of symmetry, only a fraction of the angular sweep is actually carried out, since the results are similar for symmetric parts of the object. This greatly simplifies the calculations and reduces the computation time.

The results for both system configurations are different and point out different aspects of the scattering properties of objects. Depending on what kind of characteristics are to be analyzed, a certain setup might be more appropriate than the other. To study the backscattering pattern of targets, the monostatic configuration is the most common given that its operation is identical to the behaviour of a single radar. However, it can be useful, in some situations, to compare bistatic radar cross section plots of targets and so simulations with this kind of setup might be convenient.

3.6 Complex Vectors

Introducing the implementation of the program, more specifically the representation of the electric and magnetic fields of all rays traced in the scene, it is expected that vectors play an essential role in the construction of the program. Not only do they represent the direction of the electric and magnetic field for each ray, but they also contain the information related to the magnitude of the fields, which is described by a complex value.

The basic structure of the program is, therefore, the complex vector, generically described as \( \vec{V} \), defined by a complex number \( z \) referring to its magnitude and a vector \( \vec{v} \) describing its direction, as in

\[
\vec{V} = z \cdot \vec{v} = \begin{bmatrix} a + jb \\ c + jd \\ e +jf \end{bmatrix}.
\] (3.1)

In order to easily and efficiently implement the necessary vectors in the code, a C++ template library called Eigen is used [28]. This library provides predefined structures that allow a very simple representation of vectors and matrices. Furthermore, it supports matrix and vector operations and geometrical transformations, which are essential in ray tracing problems.
While implementing the electromagnetic equations on the program, it is essential that all the calculations strictly follow a generic notation, especially concerning the vector arithmetic. In the scattering problem depicted in Figure 2.5, the electric field of the incident wave can be manipulated by some basic geometry theorems and very intuitively given by

\[ \vec{E}_i = E_0 (\hat{a}_y \cos \theta_i + \hat{a}_z \sin \theta_i) e^{-j\beta(y \sin \theta_i - z \cos \theta_i)}. \]  

However, this is not a good way of treating the problem, since this equation can only be applied to this specific geometrical configuration. Instead, the equation should be structured in a universal way and this is achieved by the use of vector arithmetic. The generalization of this equation has two parts.

First, the phase component of the complex magnitude of the electric field vector assumes that the vector that defines the position of the observation point, \( \vec{r} \), coincides with the direction of propagation of the wave. As demonstrated in Figure 3.10, this is not always verified, especially when dealing with waves defined by more than one ray.

Therefore, the phase component of the electric field should be based on both vectors, \( \vec{r} \) and \( \hat{k} \), and the inner product between them, which translates to a geometric projection of one vector onto the other.

Secondly, the direction of the electric field vector \( \vec{E}_i \) should not be dependent on the coordinate system axis and should have its own definition, generally referred to as \( \hat{e}_i \).

Finally, a generic equation can be used to characterize the incident electric field at any point

\[ \vec{E}_i = E_0 e^{-j\beta \vec{r} \cdot \hat{e}_i}. \]  

Another important matter about complex vectors is the process in which they are manipulated. As mentioned before, the complex vectors are composed by a vector that defines their direction and their magnitude which is a complex value. In this program, all the operations concerning complex vectors should treat these two components separately for simplification reasons.

A very recurrent operation of this kind is to determine the direction of a vector that is orthogonal to other two vectors. This happens, for instance, when the direction of the magnetic field, \( \hat{h} \), needs to be found, knowing the direction of propagation, \( \hat{k} \), and the direction of the electric field, \( \hat{e} \). In this case, the
relation is established by

\[ \hat{h} = \hat{k} \times \hat{e}. \]  

(3.4)

The cross product between two complex vectors in this program is, therefore, always calculated in two independent steps. On the one hand, the direction of the resultant vector is determined for the geometrical purpose of the problem, by applying the cross product to the unit vectors. On the other hand, the magnitude must be calculated based on the definition [29] of the cross product

\[ \hat{k} \times \hat{e} = |\hat{k}| \cdot |\hat{e}| \cdot \sin \rho \]  

(3.5)

where \( \rho \) is the angle between the vectors and \( 0 \leq \rho \leq \pi \).

The magnitude of the resultant complex vector is thus determined by multiplying both complex magnitudes with each other and to the factor of \( \sin \rho \).

### 3.7 Wave Polarization

An extremely important setting of the developed program is the polarization of the incident wave that is launched from the radar. Although the effects of this parameter on the scattering of the waves are not the main focus of this thesis, it is very interesting to implement this feature in order to allow the simulator to give results for any kind of polarized incident wave and, later on, being able to compare the results with actual measurements made with real radars, which also operate with waves that present different kinds of polarization settings.

In order to represent all the possible types of polarization of electromagnetic waves, the electric and magnetic field of every wave, incident and scattered alike, needs to be characterized by, not only one, but two complex vectors, each corresponding to one polarization component. This method is based on the Jones vector representation of the polarization states [30]. In other words, the electric field is decomposed into two orthogonal polarization components, which are manipulated independently throughout the whole ray tracing algorithm. The same principle applies to the magnetic field complex vector. Therefore the fields are represented as

\[ \vec{E} = E_1 \hat{e}_1 + E_2 \hat{e}_2 \]  

(3.6)

and

\[ \vec{H} = H_1 \hat{h}_1 + H_2 \hat{h}_2 \]  

(3.7)

where \( \vec{E} \) and \( \vec{H} \) are respectively the electric and magnetic fields, the complex numbers \( E_1, E_2, H_1 \) and \( H_2 \) are the electric and magnetic magnitudes of both polarization components, and the unit vectors \( \hat{e}_1, \hat{e}_2, \hat{h}_1 \) and \( \hat{h}_2 \) define the directions of each polarization component.

The polarization components can assume any direction, as long as they are orthogonal to each other and to the direction of propagation. The correspondent electric and magnetic polarization components also need to be orthogonal to each other, and so \( \hat{e}_1 \) is orthogonal to \( \hat{h}_1 \) and \( \hat{e}_2 \) is orthogonal to \( \hat{h}_2 \).

In this program, the polarization components were chosen to be aligned with the horizontal and vertical axes directions. The implementation of the polarization components as a function of the wave’s propagation direction is fully explained in the following chapter.

Three main polarization categories exist, them being linear, circular or elliptical, depending on which shape is described by the electric field vector as a function of time. Considering the electric field complex
vector $\vec{E}$, the different polarization configurations are achieved by designating certain absolute values, $E_1$ and $E_2$, and phases, $\varphi_1$ and $\varphi_2$ for the complex magnitude of the two polarization components [3], as introduced in

$$\vec{E} = \hat{e}_1 \cdot E_1 e^{j\varphi_1} \cdot e^{-j\beta \vec{r}} + \hat{e}_2 \cdot E_2 e^{j\varphi_2} \cdot e^{-j\beta \vec{r}}. \quad (3.8)$$

One way of defining a linear polarization is to define only one component, meaning that either $E_1 = 0$ or $E_2 = 0$. Another way is to assure that, independently of the absolute values, the components are in phase or in phase opposition with each other and therefore $\varphi_1 - \varphi_2 = m \cdot 180^\circ, m \in \mathbb{Z}$.

To represent a circular polarization, both components need to have the same absolute value, so $E_1 = E_2$. Additionally, the phases of the two polarization components need to be separated by $90^\circ$. Therefore, $\varphi_1 - \varphi_2 = (2m - 1) \cdot 90^\circ, m \in \mathbb{Z}$.

It is important to note that the linear and the circular polarizations are specific cases of the elliptical polarization, when the magnitude and the direction of the electric field vector forms a straight line and a perfect circle, respectively. Thus, the elliptical polarization is characterized by any other situation for which the conditions disclosed above do not validate.

In the solving of a scattering problem, as it is expected from this simulator, the two polarization components are stored and manipulated in a totally independent way. The polarization of the scattered waves is directly determined by calculating the scattering for both components separately. Since this is a linear problem, all calculations are performed for both components in the exact same way and, in the end, the total electric field can be obtained by summing them in their complex vector form, by applying the superposition principle.

### 3.8 Simulation Variables

Before clarifying the specific implementation of the algorithm, the parameters of the simulator must be enumerated, so as to give an idea of the variables and the specific structures that are necessary in the development of the program.

The most important category of variables is the one that is designated by the user. These variables store parameters that determine the course of the simulation and, consequently, its results. The parameters chosen by the user are:

- **Far field region**, which determines the distance at which the radar is positioned in relation to the geometrical center of the model. With every different position of the radar, in order to produce an angular sweep around the object, the radar is placed in successive positions all at the same distance from the radar, amounting finally to a circular shape around the model;

- **Angle increment**, corresponding to the angular resolution of the simulation, or the number of observation samples the radar cross section plot contemplates in the end. This determines in how many points the radar will be placed around the object, meaning that a smaller increment will lead to a larger number of samples and therefore a more detailed measurement;

- **Range of observation**, translated by the angular spread in which the radar positions are distributed around the object;

- **Magnitude of the incident electric field**, which is described by a horizontal and a vertical component and is ultimately represented by two real values;
• Phase of the incident electric field, that subsequently determines the type of polarization of the incident electromagnetic wave. It is necessary to establish two phases, one for each polarization component;

• Frequency, corresponding to the frequency at which the radar operates, and therefore the frequency of the incident wave. The recommended range of values for this parameter lies between 76 and 77 GHz.

There are some additional parameters which are chosen by the user, but these ones are relative to the ray tracing method in particular. However essential to the nature and accuracy of the results, these variables solely affect the geometrical calculations, more specifically, the ray tracing component of the algorithm, and they are:

• Number of incident rays launched from the radar, described by two variables, one for each direction of the camera, vertical and horizontal;

• Field of view of the incident ray generation camera, which is the range of the beam of rays that is transmitted from the radar;

• Number of scattered rays launched from a scattering point, also described in terms of vertical and horizontal orientations;

• Field of view of the scattered ray generation camera, which is the range of the beam of scattered rays;

• Offset, describing a threshold value used in calculations where small margin of error is critical, such as when determining whether a ray is immediately on the inside of a surface or the outside;

• Maximum depth, which consists of a natural number that represents the maximum desired number of reflections, or interactions, with surfaces. When a ray surpasses this number of scattering points along the way, it is considered too attenuated to have any effect on the following calculations and it is, therefore, neglected.

An essential basis element for an organized design of this code is the class. A class is a well defined data structure with several attributes, which are variables or functions that belong to the class. An object is an instance of a class. Two distinct classes were implemented in this simulator. The first, denominated Ray class, contains the information relative to each ray that is traced during the simulation and its features are:

• Absolute value of the horizontal polarization component, represented by a complex number;

• Direction of the horizontal polarization component, which is stored in a three-dimensional unit vector;

• Absolute value of the vertical polarization component, a complex number;

• Direction of the vertical polarization component, a three-dimensional unit vector;

• Origin, described by a three-dimensional vector that depicts the Cartesian coordinates of the point where the ray is generated;

• Depth, which is the number of scattering points that the ray has endured since its origin at the radar;
• Ray initialization function, a function that initializes an object of a Ray class, attributing the designated values to the corresponding attributes.

The second class implemented in this program, designated Surface class, concerns the parameters relative to the surfaces of the digital model and its essential functions, which are:

• First vertex, characterized by a three-dimensional vector which corresponds to the Cartesian coordinates of that point in space;
• Second vertex, a three dimensional vector;
• Third vertex, a three dimensional vector;
• Normal vector, a three-dimensional unit vector, depicting the coordinates of the direction of normal to the surface.
• Surface initialization function, which initializes a Surface object, attributing the features of the surface to the correspondent attributes;
• Intersection with ray function, a function that returns the three-coordinate vector that describes the geometrical point of intersection between a ray and the respective surface.

Finally, the overall variables describe parameters that are essential to every calculation in the algorithm and should therefore be initialized in the beginning of the code. They are not constant throughout the course of the simulation, most of them changing, in fact, with every successive position of the radar. The overall variables are:

• Radar position, represented by a three component vector, describing the Cartesian coordinates of the position of the radar in each measurement loop;
• Angle of incidence, the angle at which the incident wave hits the object, in relation to the reference direction. This parameter is described by two angles, one relative to the horizontal reference direction and another to the vertical;
• Default orientation of the radar, which corresponds to the direction at which the radar points to, while reaching the object with an angle of incidence of 0°;
• Reference unit vectors, that determine the direction of both polarization components. In this case, it was established that one would be aligned with the vertical axis direction and another with the horizontal one;
• Incident complex electric field, which is determined as the aftermath of the absolute value and the phase of both polarization components asserted by the user. It is represented as a three-dimensional complex vector;
• Radar dimension, described by the diameter of the reception sphere that represents the radar as a receiver;
• Received rays, which consists of an array of Ray objects, instances of the Ray class. Only the rays that are actually received by the reception sphere are stored in this array;
• Total scattered electric field, which is a complex vector representing the cumulative sum of the scattered electric field. In every different position of the radar (in every measurement loop), the sum of the electric field of the received rays at that specific radar position is added to the variable, by a complex vector sum.
Chapter 4

Code Implementation

In this chapter, the logic behind the construction of the algorithm is defined. The tracking of the rays along their path from the transmitter to the receiver is delineated through the implementation of the ray tracing algorithm, finishing with the calculation of the scattered field that is received.

4.1 Ray Tracing’s Recursive Nature

The basic idea behind the tracing of rays throughout a scene is that each ray propagates until it intersects with a surface, generating at that moment a new set of rays that are launched from the intersection point. These new generated rays are subsequently traced through space until a new surface is hit. This process repeats itself as long as the propagating rays keep on intercepting surfaces along their path.

Because one ray generates many other rays which are traced successively, this means that the algorithm has a recursive nature, a function that calls itself while the code runs.

Unlike the simple case of reflection and transmission, which simply generates two secondary rays from one primary ray, the scattering effect presents a more complex procedure. While applying the backward tracing for the reflection and transmission phenomena, the process to determine the direction of the incident ray given the reflected one is elementary. However, for the scattering case this is not a trivial solution, since a set of several secondary camera rays is launched from the point where the primary incident ray hits the surface, making it nearly impossible to deduce with sufficient accuracy the direction of the incident ray from any of the scattered ones.

For this reason, it is more advantageous for the program to follow the actual path of the rays, from transmitter to receiver. Should the rays be traced in the opposite direction as in backward tracing, when arriving at the transmitter, the path of the ray would be known, but the scattered field could not be determined. When the rays are traced in the forward direction, on the other hand, the direction of the induced current vector at the area of the surface can be passed as an argument of the ray tracing function, ultimately allowing the determination of the scattered field at the receiver. This way, the path of each ray is discovered and, when finally received by the reception sphere, the scattered field can, too, be determined. Therefore, the forward approach of the ray tracing is selected for this simulator.

At the first level of recursion, equivalent to the depth level 0, an incident ray that is launched from the radar is scattered by a surface. This effect generates a set of $n$ secondary rays. To this level of recursion is given the depth value of 1. For depth 2, the next level of recursion, each of the secondary rays produces $n$ more scattered rays, thus producing a total of $n^2$ rays. In depth level 3, $n^3$ rays are
generated. Because the number of rays increases exponentially with the depth, the computation time also escalates exponentially with the number of scattering stages [31].

By following the generation of new secondary rays along the rendering of the scene, a ray tree can be designed as represented in the example of Figure 4.1. It is most common in computer graphics for the tree to have only two branches of secondary rays for each primary ray [32]. The complexity and extent of the tree that originates from the scattering shows exactly how elaborate this algorithm is and how much more memory and processing capacity is needed to trace all the rays.

In order to manage the computation time and prevent the recursive function from running in an infinite loop, a maximum depth must be assigned, which translates to a maximum number of allowed scattering points along the path of one ray and, consequently to a maximum number of recursions. Once the maximum depth is reached, no more scattered rays are generated.

4.2 Generating Incident Rays

Computing rays of any kind requires, first of all, an understanding of how to define a ray in a geometrical perspective.

A ray can be thought of as a half-line, characterized by only two variables, a three-dimensional vector that depicts the coordinates of its origin and a three-dimensional unit vector that describes its propagation direction. The parametric equation that defines a ray is

$$P = O + t \cdot \hat{d} \quad (4.1)$$

where $\hat{d}$ is the direction unit vector, $O$ is the origin and $t$ is the distance from the origin to any point $P$ along the path of the ray. Since the rays propagate in a single direction, exclusively points that belong to the “front” of the origin are significant and, therefore, only positive values of $t$ are considered valid.

The selected algorithm for generation of rays in this program is based on the operation of a camera, in a way that the origin of the rays corresponds to the origin of the camera and the direction of each ray is given by tracing a line from the origin through the center of each camera pixel.

By convention, the camera’s origin is located in the origin of the coordinate system and the camera is oriented toward the negative z-axis [33]. However, given the geometrical configuration of the physical optics approximation represented in Figure 2.5, it is better to chose the orientation of the beam of rays as the positive z-axis instead. The reason for this is to simplify the geometrical transformation that the rays need to endure when being launched from a scattering point, which will be discussed further in this chapter. The pixels are distributed on a plane one unit away from the origin. This is the default
configuration of the ray generation algorithm, which is represented in terms of coordinates relative to the
cordinate system of the camera, also called the camera space [33].

Observing the camera pixels represented in the example of Figure 4.2a, the pixel coordinates are
given as a function of the image width and height, corresponding to the number of rays on the horizontal
and vertical directions, respectively.

Because the screen space, represented in Figure 4.2a has its origin at the upper left corner of the
grid and a dimension of 1, to transfer the coordinates into the camera space depicted in Figure 4.2b, a
multiplication by 2, a shift of 1 unit and a mirroring of the x- and y-axis is necessary. An additional shift
of 0.5 units is considered to let the ray pass through the center of the pixel.

(a) Pixel coordinate points in the screen space. (b) Pixel coordinate points in the camera space.

Figure 4.2: Transformation between the screen and the camera spaces.

The resultant pixel coordinates are, then, given by

\[ \text{Pixel}_x = 1 - 2 \cdot \frac{i + 0.5}{\text{ImageWidth}} \] (4.2)

and

\[ \text{Pixel}_y = 1 - 2 \cdot \frac{j + 0.5}{\text{ImageHeight}} \] (4.3)

where \( i \) and \( j \) are, respectively, the index of the pixel in the horizontal and vertical orientations.

To make sure that the pixels are always square, even when the width of the image is different than
its height, a normalization must take place, where the image aspect ratio is calculated through

\[ \text{imageAspectRatio} = \frac{\text{imageWidth}}{\text{imageHeight}}. \] (4.4)

This factor is not essential to the program, since the developed simulator operates correctly for any
dimensions of the pixels responsible for each ray, independently if they are square or rectangular. How-
ever, if considered, this normalization factor must be introduced into the expression for determining
\( \text{Pixel}_x \).

As a final consideration, the field of view of the camera, in this case translated into the angular range
of the radar, must be designated. A factor of \( \tan \alpha/2 \) is introduced in the expressions for both pixel
expressions, where \( \alpha \) is the total angular range of the beam of rays, in radians.

Therefore, the final expressions for the pixel coordinates are

\[ \text{Pixel}_x = \left( 1 - 2 \cdot \frac{i + 0.5}{\text{ImageWidth}} \right) \cdot \text{imageAspectRatio} \cdot \tan \alpha/2 \] (4.5)
and

\[ Pixel_y = \left( 1 - 2 \cdot \frac{j + 0.5}{\text{ImageHeight}} \right) \cdot \tan \alpha / 2. \]  

(4.6)

Thus, because the grid of pixels is located one unit away from the origin of the system along the positive z-axis, the pixel coordinates in the camera local coordinate system are finally given by

\[ (Pixel_x, Pixel_y, 1). \]  

(4.7)

When normalized, this vector directly describes the direction of the ray generated by the respective pixel of the camera.

The origin and the direction unit vectors can, then, be stored into the attributes of the ray class object that defines the ray. Each generated ray corresponds to one object of the class, featuring its own specific attributes.

![Camera local coordinate system.](image)

Figure 4.3: Camera local coordinate system.

It is essential in this program to understand that this ray generation algorithm is implemented in the local coordinate system of the camera, illustrated in Figure 4.3. To generate rays oriented in different directions and originated in different points, such as the radar position or different scattering points to perform the scattering of secondary rays, a geometrical transformation must be performed which is introduced and analyzed further in the course of this chapter.

### 4.3 Incident Electric Field

When assigning the electric field to the incident wave launched from the transmitter, the first thing to consider is how to assert its polarization configuration. As mentioned in section 3.8, both the magnitude and the phase of both polarization components can be selected by the user before the running of the simulation. However, it is essential that the code is programmed to attribute a specific orientation to these components, as a function of the direction of propagation of the ray.

In this program, the polarization components are chosen to be oriented along the horizontal and the vertical directions, in relation to the vector that describes the direction of propagation of the ray, identified here as \( \hat{d} \). The only prerequisite for the configuration to be valid is that both polarization components are orthogonal to each other and to the direction of the ray.
A reference unit vector is first established, with the objective of applying a consistent formula to every ray that allows the correct determination of the direction of its polarization components.

Considering a ray propagating in the direction along the z-axis $\hat{d} = (0, 0, 1)$, as in Figure 4.4, the horizontal component is chosen to be aligned with the x-axis direction. Therefore,

$$\hat{e}_h = (0, 0, 1) \times \hat{r} \hat{f}_h = (1, 0, 0)$$

which implies that

$$\hat{r} \hat{f}_h = (0, -1, 0).$$

In the same line of thought, the vertical component is aligned with the y-axis direction and thus

$$\hat{e}_v = (0, 0, 1) \times \hat{r} \hat{f}_v = (0, 1, 0)$$

which means that

$$\hat{r} \hat{f}_v = (1, 0, 0).$$

By performing the cross product between the direction of a ray and the reference direction just determined, the respective polarization component, horizontal or vertical, is assured to be placed somewhere in the plane that contains the two vectors $\hat{d}$ and $\hat{e}_h$ or $\hat{e}_v$, accordingly. This ultimately ensures that the horizontal component is always orthogonal to both the ray direction and the vertical axis, while the vertical component is orthogonal to the ray propagation and the horizontal axis direction. With both reference directions defined for a generic situation, it is evident that these expressions apply for any specific direction of propagation, as long as the calculations are performed in the local coordinate system of the ray generation camera which is in this case represented by the radar.

Only after determining the direction of both polarization components for each ray, must these vectors be transformed into the global coordinate system through the use of the correspondent transformation matrix.

Following the transformation of the directions into the global coordinate system, these vectors are, then, stored as attributes of the ray. The magnitude of each polarization component, consisting of an absolute value and a phase, is automatically stored into the object’s attributes of the ray class, since these are parameters selected by the user and, therefore, don’t need to withstand any additional calculations.

Figure 4.4: Local coordinate system of the radar and polarization components for a ray propagating along the positive z-axis.
4.4 Rendering Digital Models

This section clarifies the role of the object in this program, how to represent it in a digital three-dimensional model and how to store and interpret its information.

Digital files must be studied in order to understand their structure and the intersection between a ray and a triangle must be implemented to find the point where each ray hits the analyzed model. Additionally, a small algorithm assesses which surfaces of the object are useful to the ray tracing and which can be neglected. Finally, the computation of the area of each decomposition polygon is designed, based on the distance between consecutive rays.

4.4.1 Reading a mesh file

Three-dimensional models of objects are digitally represented as meshes, more specifically, polygon meshes. The most common and efficient way of storing the information of these meshes is through the decomposition of every face of the object into triangles [26]. Subsequently, the way of storing a mesh in digital support is by using a file that contains all triangle facets that compose the model.

Even though many different file formats exist for this purpose, the meshes in this code are chosen to be saved in stereolithography (STL) files. STL files can be exported from any common software that creates three-dimensional polygon meshes of models, which motivates their wide application in three-dimensional mesh representation and manipulation programs. Additionally, not only is this file structure very organized and easy to process, but it can also be represented in both ASCII and binary versions, being the latter more compact and, therefore, more efficient [34].

Since this simulator is programmed to read files in the binary STL format, it is important to make sure that this is the exact format of the input file that represents the model to be analyzed. For this reason, an eventual conversion of format might be necessary, which does not actually constitute a problem, since many software programs are able to directly and quickly convert any mesh file format into STL binary.

The file structure [35] starts with an 80-character header, which simply identifies the model and can be ignored for the purpose of this program. After this, a 4-byte unsigned integer indicates the total number of triangles that compose the model. This parameter is essential for the reading of the file, since it specifies exactly how many objects of the class surface need to be created and how many loops need to be executed to read the information of all triangles that exist in the file.

Each triangle on the model is described by twelve 32-bit float numbers, which, when read, should be grouped into four sets of three floats, since each represents a three-dimensional vector. The three 32-bit numbers of each vector correspond, as expected, to the \(x\), \(y\) and \(z\) coordinates of the vectors in a standard Cartesian coordinate system. The first vector corresponds to the normal vector of the surface, which always points towards the outside of the model. The next three vectors are the three vertices that constitute the triangle, conventionally following a counter-clockwise distribution. Finally, an additional 2-byte unsigned integer, which is called the attribute byte count, ends the description of each triangle. This data is not useful for the purpose of this program and does not need, therefore, to be stored.

It is important to note that the STL binary file does not store any information relative to units of the metric system. For this reason, it is essential to make sure that the dimensions of the model are consistent with the units used in the program. In this case, the algorithm was implemented in meters, so the software where the model is designed should be in in the same units.

After reading each set of information that corresponds to a triangle in the file, a new object of the
class surface is created. The information obtained by the triangle is grouped into slots corresponding to the twelve coordinates needed to describe a surface and then converted into decimal numbers, which are subsequently stored in three-dimensional vectors. These vectors are, then, used to initialize the object of the surface class, as its attributes.

4.4.2 Intersection between a ray and a triangle

In order to find the point where a ray hits a certain surface of the model, the calculation that needs to be performed is the determination of the intersection point between the half-line that defines the ray with the triangle that corresponds to the polygonal facet of the model’s surface.

The calculation of the intersection between a ray and a triangle includes three main steps. The first consists on assessing whether the ray is parallel, or approximately parallel, to the triangle. This is determined through the evaluation of the inner product between the vector normal to the triangle plane, \( \hat{n} = (A, B, C) \), and the direction of the ray, \( \hat{d} \). If the result of the inner product returns zero, or close to zero, it means that the ray and the triangle will only intersect in infinity, which is not of interest in the case of this implementation. Therefore, in the case that the ray and the triangle are approximately parallel, it is immediately known that there is no intersection between them and, so, no further calculations need to be made.

Should the ray and the plane not be parallel, the second step is the determination of the point where the ray hits the plane that includes the triangle. This plane is described with its parametric equation

\[
Ax + By + Cz = D \quad (4.12)
\]

where \( A, B \) and \( C \) are, respectively, the \( x, y \) and \( z \) coordinates of the vector normal to the plane and \( D \) is the distance between the plane and the origin of the coordinate system with coordinates \( (0, 0, 0) \). Because all the vectors from the triangle are known and the distance between the plane and the origin is measured in a line parallel to the normal of the plane, the value of \( D \) can be found through

\[
D = (A, B, C) \cdot (v_x, v_y, v_z) \quad (4.13)
\]

where \( (v_x, v_y, v_z) \) is any of the three vertices of the triangle.

To calculate the intersection between the ray and the triangle \([36]\), the point \( P \) from \([4.1]\) is substituted in \([4.13]\), solving this equation as a function of \( t \) as

\[
t = \frac{(A, B, C) \cdot [(v_x, v_y, v_z) − O]}{(A, B, C) \cdot \hat{d}} \quad (4.14)
\]

where \( O \) and \( \hat{d} \) are the origin and the direction of the ray, respectively.

The parameter \( t \) alone is enough to determine the intersection point which is given by

\[
P_{\text{intersection}} = O + t \cdot \hat{d} \quad (4.15)
\]

It is important to note that it only makes sense to consider the intersection point and calculate its coordinates if it is located “in front” of the ray. This means that, because the ray is a half-line represented in space, if the intersection point corresponds to a negative value of \( t \), the ray does not propagate in that portion of the line and, thus, the intersection does not exist. For this reason, if \( t < 0 \), the intersection point is not valid and \( P_{\text{intersection}} \) does not need to be determined.
If it exists, the found intersection point between the ray and the triangle plane needs, at last, to be analyzed concerning its position relatively to the triangle itself. Therefore, the third and final step of this calculation consists on determining whether this point lies inside the area of the triangle or outside. This problem is easily solved by a so called “inside-outside” test [37].

The representation on Figure 4.5 depicts a generic triangle with vertices $v_1$, $v_2$ and $v_3$, the intersection point $P$, the edge $\vec{a}$ that connects vertices $v_1$ and $v_2$, and vector $\vec{b}$ which reaches $P$ from the first vertex. By performing the cross product between $\vec{a}$ and $\vec{b}$, the resultant vector $\vec{c}$ has a certain direction, orthogonal to both vectors and given by the right-hand rule [38]. Since the vertices of the triangle are declared in counter-clockwise order, it is inferred from a geometrical perspective that, when the vector $\vec{c}$ presents the same direction as the normal vector of the triangle, the point $P$ is located on the left side of vector $\vec{a}$. Analogously, when vector $\vec{c}$ has opposite direction to the normal vector, point $P$ is on the right side of the same vector.

By the definition of the inner product between two vectors, vector $\vec{c}$ is always parallel to the normal vector (because it is orthogonal to two vectors in the plane of the triangle). Furthermore, it is known that when the inner product between the normal vector and vector $\vec{c}$ is positive, both vectors are facing the same direction, which happens when point $P$ is located on the left side of vector $\vec{a}$. Consequently, when the result of the inner product between the vectors is negative, the vectors are facing opposite directions, which infers that $P$ is on the right side of $\vec{a}$.

By repeating the test for every edge of the triangle, it is possible to determine whether the intersection point $P$ is inside or outside the area of the triangle. If the vectors that form the edges of the triangle are formed in counter-clockwise orientation by default, and if the intersection point is located on the left side of every edge of the triangle, it is proved that the point is inside the triangle. If, for any edge of the facet, this condition is not verified, it is concluded that the intersection point is outside the triangle. In this case, evidently, the intersection between the ray and the triangle does not exist.

4.4.3 Back-face culling

In the case of this program, where the triangle facets of the three-dimensional model are declared in counter-clockwise order and, subsequently, stored this way, the normal that defines each facet is facing the outside of the object. This directly implies that the side of the triangle to where the normal vector is pointing towards corresponds to the outside surface of the model, and the other side is, then, an
inside surface. Therefore, a surface whose normal vector is facing the camera is said to be front-facing, whereas another surface with a normal vector facing away from the camera is back-facing [39].

Because, in this algorithm, the surfaces are considered to be perfect electric conducting, being there no transparent surfaces, there is no need to consider the back-faced surfaces of the model, since they are on the inside of the object.

For this reason, this simulator is programmed to perform what is called back-face culling, which consists on discarding all the surfaces of the model that are facing away from the radar. Since these surfaces are not visible from the perspective of the radar, their contribution for the ray tracing of the transmitted rays is inconsequential. Figure 4.6 illustrates the selection of surfaces concerning their orientation relative to the radar through back-face culling. In this example, the surfaces of the cube that are represented by blue normal vectors are considered, while the surfaces with red normal vectors are dismissed.

![Figure 4.6: Back-face culling of surfaces that face away from the radar (in red).](image)

Before testing any intersections between a ray and a surface, it is of the utmost importance to assess which surfaces are, in fact, front-facing and which can be immediately disregarded. This process of surface selection significantly improves the performance of the code in terms of both computing time and memory management, since the number of surfaces that are actually rendered reduces by a large factor for most objects’ shapes [37].

The way of detecting back-faced surfaces is to simply calculate the inner product between the normal of the surface and the propagation direction of the ray. If the result is negative, this means that the surface and the ray are facing each other and, thus, the surface is front-facing. In the case where the inner product is negative, the surface is turned away from the ray, inferring that the surface does not need to be rendered.

### 4.4.4 Separation between rays at intersection

An essential step for calculating the scattered field at a surface is the determination of its area. As stated in section 3.1, a wave is represented by discrete rays, each responsible for a certain section of the surface, correspondent to its cross section area. In order to calculate the induced current at each section of surface by each incident ray on the model, an algorithm that determines the separation between the rays at the point of intersection must be designed. This quantity is calculated in the horizontal and vertical orientation, immediately translating into the dimensions of the section that each ray affects, allowing ultimately the calculation of the area where the induced current must be determined.
The selected ray generation algorithm in this project infers that the rays are equidistant only if observed at a plane that is perpendicular to the direction where the radar is facing, such as the camera of pixels where the rays are created. For this reason, when a surface lies perpendicular to the view of the radar, as in [Figure 4.7] the determination of the separation between rays is very straightforward. The horizontal and the vertical distances between two consecutive rays, respectively $a$ and $b$, are given by

\[
a = t \cdot \text{HorizontalSeparation} \quad (4.16)
\]

and

\[
b = t \cdot \text{VerticalSeparation} \quad (4.17)
\]

where $t$ is the variable calculated in the previous section that indicates the distance from the origin of the ray to its intersection point with the surface, and $\text{HorizontalSeparation}$ and $\text{VerticalSeparation}$ are two variables determined at the moment of the ray generation as

\[
\text{HorizontalSeparation} = \frac{2 \cdot \text{ImageAspectRatio} \cdot \tan \alpha/2}{\text{ImageWidth}} \quad (4.18)
\]

and

\[
\text{VerticalSeparation} = \frac{2 \cdot \tan \alpha/2}{\text{ImageHeight}} \quad (4.19)
\]

![Figure 4.7: Equidistant rays at a surface perpendicular to the radar orientation.](image)

However, on most cases, the surfaces of the objects in the scene are not exactly perpendicular to the orientation of the radar and, therefore, this problem is more elaborate. For the calculation of the area correspondent to a generic ray on the surface of incidence, an approximation must be taken into account. The condition established in this program assuring the location of the scatterer in the far field of the radar allows the wave to be considered approximately planar. Subsequently, because the rays hit the surfaces of the object in the far field, they can be considered to be approximately parallel to each other.

Given the geometry of the problem in the vicinity of the surface, shown in [Figure 4.8] the horizontal and vertical separations between two consecutive rays for any orientation of the surface, $a$ and $b$ respectively, can be easily determined by

\[
a' = \frac{a}{\cos \theta_H} \quad (4.20)
\]
and

$$b' = \frac{b}{\cos\theta_V} \quad (4.21)$$

where $\theta_H$ and $\theta_V$ are the angles between the vector normal to the surface and the opposite vector of the direction of the ray, $-d$, in its projections onto the horizontal xz-plane and the vertical yz-plane.

The determination of the angles is, therefore, calculated based on the two-dimensional projections of these vectors onto the xz- and the yz-plane, where the expressions are exactly the same for both.

![Figure 4.8: Distance between rays for a generic orientation of the surface (horizontal or vertical projection).](image)

The estimation of an angle between two vectors is based on the definition of the arc-tangent function. In code, for a correct distinction of angles larger and smaller than $\pi$ radians, the function $\text{atan2}(y, x)$ is used. This function returns the result of $\arctan \frac{y}{x}$ for values $-\pi < \theta \leq \pi$.

Therefore, the horizontal and the vertical angles $\theta_H$ and $\theta_V$ are given respectively by

$$\theta_H = \text{atan2}(-n_y, -n_z) - \text{atan2}(d_y, d_z) \quad (4.22)$$

and

$$\theta_V = \text{atan2}(-n_x, -n_z) - \text{atan2}(d_x, d_z) \quad (4.23)$$

where $(n_x, n_y, n_z)$ and $(d_x, d_y, d_z)$ are, respectively, the Cartesian coordinates of the normal vector of the surface and of the direction of the ray in question.

### 4.5 Geometry

This section explains how the geometry of the simulator is conceived, including the relative position of the different components of the system, as well as the implementation of two local coordinate systems. The radar angular sweep procedure and the scattering of rays from model’s surfaces are introduced and the necessary transformations to perform these operations are clarified.
4.5.1 Global and local coordinate systems

The configuration designed for the radar system depicted in this simulator is described by a coordinate system with its origin at the center of the model, defined by the blue axis in Figure 4.9. To this system of coordinates is given the name of global coordinate system, or world space, which is the space common to all objects in the scene.

Because it is more efficient to select and change the location of the radar for each angle of incidence than to perform a transformation of every triangle facet that constitutes the model, the own model's coordinate system is chosen as the actual world space. This significantly minimizes the number of geometrical transformations and speeds up the computation time. It is important, however, that the object is centered around the origin of its own coordinate system, in order to assure that this is the selected geometry that the program was designed for.

Given the selected algorithm for generation of rays, both incident and scattered, two additional local coordinate systems are implemented. The local coordinate system is a model space where all the coordinates are relative to the own object that it represents. In this case, one model space corresponds to the radar local coordinate system and another to the polygon local coordinate system, relative to the area of the surface which is hit by an incident ray at the moment of scattering. Both the radar and polygon spaces are represented by sets of red axes in Figure 4.9.

These two spaces are based on the same camera ray generation local coordinate system, differentiating only in the parameters that describe the generation of the rays. While the radar space generates incident rays, the polygon space generates scattered ones, which can be created with different specifications.

A transformation is the process of changing elements, either points or vectors, from one space into another. This is achieved by multiplying these elements by a transformation matrix, that describes the transformation.

The elementary available transformations are translation, scale and rotation around the three axis that describe the system of coordinates, each associated with a transformation matrix that describes it. By multiplying successive matrices, different transformations can be performed consecutively. Considering that matrix multiplication is not commutative and the matrices are composed by column vectors, the chain of transformations is read from right to left, being the result a single matrix that encodes the full transformation.
Each local coordinate system is associated with a transformation matrix that transforms its elements into the global coordinate system [41]. Therefore, for this program, two matrices need to be implemented, one that transforms the radar coordinates into global, called radar-to-world matrix, and another that transforms the polygon coordinates into global, referred to as polygon-to-world matrix. The determination of these matrices is explained in the following subsections.

4.5.2 Radar angular sweep

The angular sweep of the radar cross section simulation is achieved by changing the position of the radar around the model, at a constant distance from its origin, larger than the far field distance. A loop is, therefore, implemented in the code, where the radar cross section is calculated for each radar position.

Depending on the angular range of observation and the angle increment, two angles of incidence can be described, in the horizontal and vertical projections, respectively $\gamma$ and $\delta$. As presented in Figure 4.10, the angles are measured as in a standard spherical coordinate system, with the exception that $\gamma$ is measured on the xz-plane and $\delta$ is measured from the y positive axis.

\[ \text{RadarPosition}_x = \text{FarField} \cdot \sin \delta \cdot \sin \gamma \]  
\[ \text{RadarPosition}_y = \text{FarField} \cdot \cos \delta \]  
\[ \text{RadarPosition}_z = \text{FarField} \cdot \sin \delta \cdot \cos \gamma \]

According to the angles of incidence, which change for every loop of the sweep, and depending on the far field distance, $\text{FarField}$, selected by the user, the position of the radar can be found in Cartesian coordinates. The radar position is given by

\[ \text{RadarPosition} = (\text{RadarPosition}_x, \text{RadarPosition}_y, \text{RadarPosition}_z) \]

where the three coordinates are determined as in (4.25) through (4.27).

In order to determine the radar-to-world transformation matrix, the orientation of the three axis of the radar local coordinate system need to be found in terms of global coordinates. Because the incident ray generation algorithm launches the rays in the positive z-axis direction, it is inferred that the z-axis of the radar space, $\hat{z}_{\text{radar}}$, corresponds to the opposite direction of the vector that connects the origin of the
world space to the radar position, as demonstrated in Figure 4.9. Therefore, the z-axis of the local radar system is given by

\[
\vec{z}_{\text{radar}} = -(\text{Radar Position}_x, \text{Radar Position}_y, \text{Radar Position}_z)
\]  

(4.28)

which is subsequently normalized to a unit vector with the same direction, \(\hat{z}_{\text{radar}}\).

To calculate the orientation of the x-axis of the radar space, the projection onto the xz-plane of the angle between the global z-axis and the radar z-axis must be assessed. The generic matrix that describes the rotation of a system around the y-axis [40] is given by

\[
T_{\text{rotation}} = \begin{bmatrix} \cos \psi & 0 & \sin \psi \\ 0 & 1 & 0 \\ -\sin \psi & 0 & \cos \psi \end{bmatrix}.
\]

(4.29)

For this specific case of transforming the global x-axis into the radar x-axis, the rotation angle corresponds to \(\psi = \pi + \gamma\). The shift of \(\pi\) radians derives from the rotation of the z-axis of 180°.

The x-axis of the radar local coordinate system is thus determined as

\[
\hat{x}_{\text{radar}} = T_{\text{rotation}} \cdot (1, 0, 0).
\]

(4.30)

Finally, the y-axis of the radar local coordinate system is the result of

\[
\hat{y}_{\text{radar}} = \hat{z}_{\text{radar}} \times \hat{x}_{\text{radar}}.
\]

(4.31)

The transformation matrix that describes the conversion of geometric elements from the radar space into the world space [40] is given by

\[
T_{\text{radar}} = \begin{bmatrix} \hat{x}_{\text{radar}.x} & \hat{y}_{\text{radar}.x} & \hat{z}_{\text{radar}.x} \\ \hat{x}_{\text{radar}.y} & \hat{y}_{\text{radar}.y} & \hat{z}_{\text{radar}.y} \\ \hat{x}_{\text{radar}.z} & \hat{y}_{\text{radar}.z} & \hat{z}_{\text{radar}.z} \end{bmatrix}.
\]

(4.32)

Since the origin of the incident rays is known to be the radar position, only the direction of the generated rays and the direction of their polarization components need to be transformed. Because these parameters are all vectors, only the rotation effect of the transformation is significant, being the translation neglected, which simplifies these calculations. By multiplying all vectors by \(T_{\text{radar}}\), their directions are directly given in global coordinates.

### 4.5.3 Scattering of rays

The generation of the scattered rays uses the same algorithm as the generation of incident ones, but presenting different specifications for the number of launched rays and the field of view, which are selected by the user. Analogously, the strategy to determine the transformation matrix that converts the polygon space into the world space is the same as the one implemented for the radar space, only with different criteria, specific for the scattering of rays.

The equations for the determination of the scattered electric field by a perfect electric conducting rectangular surface are designed for a system configuration where the surface, or in this case, the polygon is aligned with the xy-plane and its normal vector is aligned with the positive z-axis. In order to apply these equations correctly, all the calculations involving the electric field are performed in the polygon local coordinate system. The angles that define the spherical coordinates for the direction
of each scattered ray are given by the configuration presented in Figure 4.11 where $\phi$ is the angle measured on the xy-plane and $\theta$ is measured from the z-axis.

![Figure 4.11: Scattering angles $\phi$ and $\theta$.](image)

Evidently, the origin of the generated scattered rays corresponds to the point where the incident ray hits the surface and this, naturally, is the origin of the polygon local coordinate system. However, it is important to consider that some small errors might be associated with the calculations performed in the code for the determination of every point. Since it is essential that the scattered rays are actually launched from outside the model’s surface, a small offset value is added in the direction of the normal vector to assure that the ray tracing is not affected by minor arithmetic approximations.

The camera that generates the scattered rays is pointing towards the direction of the normal vector of the surface at the intersection point. From the algorithm of ray generation, it is inferred that the z-axis of the polygon space is, then, aligned with the normal vector of the polygon, which means that

$$\hat{z}_{\text{polygon}} = (\text{Normal}_x, \text{Normal}_y, \text{Normal}_z)$$

(4.33)

where $\text{Normal}_x$, $\text{Normal}_y$ and $\text{Normal}_z$ are the coordinates of the normal vector.

The orientation of the x-axis of the polygon space is obtained by analyzing the angle between the polygon’s and the global z-axis, on its projection onto the horizontal xz-plane.

The matrix that defines the rotation of the x-axis around the y-axis is the one in (4.29), where the rotation angle is, in this case, given by

$$\psi = \arctan \frac{\hat{z}_{\text{polygon}} \cdot \hat{x}}{\hat{z}_{\text{polygon}} \cdot \hat{z}} - \arctan \frac{\hat{z}_{\text{world}} \cdot \hat{x}}{\hat{z}_{\text{world}} \cdot \hat{z}}$$

(4.34)

The x- and y-axis directions of the polygon local coordinate system are, therefore, calculated in the same way as for the radar coordinate system where

$$\hat{x}_{\text{polygon}} = T_{\text{rotation}} \cdot (1, 0, 0)$$

(4.35)

and

$$\hat{y}_{\text{polygon}} = \hat{z}_{\text{polygon}} \times \hat{x}_{\text{polygon}}.$$  

(4.36)

The transformation matrix is, finally, obtained by joining the directions of the local coordinate system axes in a matrix format, as in (4.32), for the case of the radar-to-world transformation. Once more, only the direction of the scattered rays needs to be transformed from the polygon into the world space,
which is accomplished by simply multiplying these vectors by the transformation matrix that described
the polygon local system of coordinates.

4.6 Reception of Rays

From the perspective of the reception of scattered rays during the simulation, the radar is represented as
a three-dimensional volume with the shape of a sphere. As expected, in a monostatic system configura-
tion, the reception sphere is located at the radar position in each loop of the angular sweep, changing its
center coordinates for every new angle of observation. Since the angle increment between observations
is constant throughout the simulation, however, the diameter of the sphere is constant, which constitutes
an advantage. The following subsections systematize the determination of an adequate dimension for
the radar when operating as a receiver, how to assert whether a certain ray is received by the radar and
the elimination of multiple rays.

4.6.1 Dimension of the reception sphere

The radius of the sphere that describes the radar as a receiver is calculated in the beginning of the code
and it depends only on the far field distance and the observation angle increment between successive
observations, both parameters selected by the user. Because the initial purpose of these simulations is
to evaluate the performance of the studied algorithm, it is desirable to receive and analyze the maximum
possible number of rays that are traced throughout the simulation. That is why, in this case, the reception
spheres are designed to cover the model in all directions, but without superposing each other.

The parameters contemplated in Figure 4.12 are easily calculated through trigonometric functions, returning

\[ d_1 = d_0 \cdot \cos \theta_{obs} \]  
\[ l' = d_0 \cdot \sin \theta_{obs} \]  
\[ z = d_0 - d_1 \]  
and finally

\[ l = \sqrt{z^2 + l'^2} \]

where \( \theta_{obs} \) is the angle increment of the observation angle and \( d_0 \) is the far field distance, at which
the radar is located from the geometric center of the model.

Consequently, the radius of the reception sphere is given for each simulation by

\[ R_{sphere} = \frac{l}{2} - offset. \]

The offset is included in the expression with the objective of assuring that the sphere volumes do not
intersect. This factor prevents the same ray from being received at two different observations.

Because they are spheres, the receivers have a constant dimension on their horizontal and vertical
dimensions. For that reason, it is important to note that, if there is an angular sweep in both \( \phi \) and \( \theta \), the
angle increment should be the same in both directions. This guarantees that the diameter of the sphere
is constant throughout the whole simulation and that there is no superposition of reception spheres in any way.

### 4.6.2 Intersection between a ray and a sphere

An essential step of the ray tracing algorithm is to determine which rays are received at the radar, since only the ones that actually reach the receiver are useful for the simulation of the radar cross section result. To assess whether a certain ray hits the receiver, the program must compute the point of intersection between the ray and the sphere that represents the radar for that system configuration. This is a quite simple calculation, if the analytical solution for the problem is considered, where the ray is represented by a half-line with specified origin and direction, and the radar is a sphere for which its center coordinates and radius are known [42].

Similarly to the intersection between a ray and a triangle, the variable $t$ from the parametric equation that describes the ray is the solution of the problem which must be found.

The parametric equation of a sphere with radius $R_{sphere}$ and center Cartesian coordinates $C = (C_x, C_y, C_z)$ is

$$
(x - C_x)^2 + (y - C_y)^2 + (z - C_z)^2 = R_{sphere}^2
$$

where $x$, $y$, and $z$ are the Cartesian coordinates of any point belonging to the surface of the sphere.

By substituting the equation of the ray from (4.1) into the sphere parametric description, a quadratic function [42] is obtained

$$
|O + t \cdot \hat{d} - C|^2 - R_{sphere}^2 = 0
$$

where $t$ is the unknown variable.
Therefore, if the considered quadratic function is of the form

\[ f(x) = ax^2 + bx + c \]  \hspace{1cm} (4.44)

then, the parameters in this case are

\[ a = |\vec{d}| = 1 \]  \hspace{1cm} (4.45)

since, in this program, the direction of the ray is consistently normalized, being its norm, naturally, unitary,

\[ b = 2 \cdot \vec{d}(O - C) \]  \hspace{1cm} (4.46)

and

\[ c = |O - C|^2 - R_{sphere}^2. \]  \hspace{1cm} (4.47)

The solution for the equation is, then, given by the calculation of the roots of the quadratic function as

\[ t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \]  \hspace{1cm} (4.48)

where the factor \( b^2 - 4ac \) determines the number of solutions for \( t \) according to its sign. If the factor is negative, the solutions are imaginary and, therefore, there is no actual intersection between the ray and the sphere. When the factor equals zero, there is one solution and, thus, one point of intersection, while if it returns a positive value, there are two solutions, which means the ray intersects the reception sphere in two points.

Because the ray only propagates in one direction from the origin, negative values of \( t \) are disregarded from this calculation. An intersection does not exist when both solutions are negative and also if one root is positive and another negative, because that implies that the origin of the ray is inside the receiver and, therefore, the ray has no effect on the simulation.

The ray hits the receiver when there is only one solution for \( t \) or if both solutions are positive. In the latter case, it is inferred that the point of intersection is the one correspondent to the smallest value of \( t \), since it is the closest to the origin and subsequently the first to be intersected by the ray.

### 4.6.3 Multiple rays

Besides the determination of the intersection between a ray and the reception sphere, there is another matter which is essential to assert the reception of a ray at the receiver, concerning the ray density normalization concept introduced in subsection 2.7.5. Because many rays belonging to the same scattered wave might be received at the radar, the number of rays that hit the receiver must be taken into account and a strategy must be implemented in order to correctly determine the scattered field contribution of the wave.

One potential solution for this problem is to, while computing the ray density at the point of reception, determine how many rays are included in the section of the tube of rays that corresponds to the area of the receiver. The total sum of the scattered electric field of all ray contributions can, then, be divided by the number of received rays that belong to the same wave. This is basically the strategy proposed by the ray density normalization method. Although quite adequate, this approach implies that a large number of rays must be traced, to achieve an accurate result. Because that might not be the case for...
every simulation, another method is applied that resides on the elimination of multiple rays.

This approach consists on, at the moment that a ray is known to intersect the reception sphere, assessing whether this ray is repeated for the wave that it belongs to. Based on the attributes of the ray in question, it is possible to compare it with the rays already received and determine if this ray is unique for a wave.

The attributes that are compared are the depth of the ray and its origin point, which are identical for every ray that constitutes the same wave, but different for any other wave.

If a ray presents both the same depth and origin as another previously received ray, it is immediately discarded. If the attributes are different than for the rest of the received rays, the ray is confirmed to be unique for the wave that it represents and, therefore, stored as a received ray, along with its attributes for further comparison.

Regarding this approach, only the first ray of each wave is considered, while in the case of the ray density normalization all rays that intersect the radar are taken into account and then normalized. This might establish a disadvantage, but it can also be managed by reducing the angular distance between measurement points, therefore reducing the dimension of the sphere and, consequently, minimizing the disparity of scattered electric field intensities between consecutive multiple rays.

Additionally, this method is more efficient in terms of memory management and computation time, since it needs to store less rays and their information, and because there is no need to determine the ray density and number of rays at the section correspondent to the area of the receiver.

4.7 Calculation of the Scattered Field

The final stage of the simulation algorithm consists on the calculation of the scattered electric field, in order to determine the radar cross section of the analyzed object. In Appendix A the pseudo-code of the most relevant functions implemented in the program can be found.

Given the recursive nature of the "Trace ray" function, each time a ray interacts with a surface, scattering into several other rays, a higher level of recursivity is activated, which also translates into a higher level of depth. The primary rays, which are generated at the radar and transmitted into the scene consistently present a depth of 0, while scattered rays increase their value of depth every time they encounter a surface, until eventually a ray reaches the maximum permitted depth level and is discarded.

When a ray intersects the reception sphere and is unique for its scattered wave, it is confirmed that it is received at the radar, meaning that its electric field at the receiving point must be calculated as a contribution for the total electric field received by the radar at that observation angle. This ray might present any depth between 1 and the maximum allowed depth level.

Each received ray is assumed to be represented by the path it has maintained since it was generated at the radar as an incident ray up until the point where it was received. Because the path is composed by several individual ray sections along the way, the determination of the received electric field is determined section by section, being each part of the electric field discovered with every recursion of the "Trace ray" function.

The information known about the ray at the moment of reception at the radar is only the incident electric field, which is an overall variable, unchanged throughout the whole course of the program, and the phase difference between the reception point and the origin of the current ray, which is located at the
last scattering point that the path endured. To the incident electric field is given the representation of $E_0$ and the phase component of the last section of the path is characterized by $\Psi_Z$, where $Z$ is the index correspondent to the depth level of the received ray.

After this last recursion of the ”Trace ray” function finishes, the code decreases the level of depth by 1, being activated the level of recursion $Z - 1$. This level corresponds to the section of the path previous to the last one, which is the ray that intersects the surface at the point where the last received ray was scattered from. In this level of recursion, as observed in algorithm 2, as the ray is signaled as received, the calculation of the electric field contribution of this section takes place.

For every intermediate section of the path as this one, it is possible to calculate the area of the surface where the current is induced by the respective ray, which is given by the horizontal and vertical separation between rays as $a \cdot b$. Additionally, the scattered electric field contribution can be determined using the vector potential calculation, after determining the magnetic field at the point of intersection and the induced current magnitude. Consequently, the scattered electric field for the direction that corresponds to the scattered ray can be calculated, represented as $\Omega_{\text{depth}}$, where depth is the depth of the ray traced in the active depth level. Finally, the factor $\Delta_{\text{depth}} = 2 \cdot \frac{1}{\eta} \cdot \sin \rho$ is introduced and calculated, where the multiplication by 2 derives from the physical optics approximation for the determination of the induced current at the surface, the division by $\eta$ corresponds to the conversion from electric into magnetic field magnitude and $\rho$ is the angle between the normal vector to the surface and the magnetic field vector direction, necessary to determine the magnitude of the induced current.

It is essential, however, to understand that these quantities are all determined as a function of the electric field of the previous ray at the point of intersection with the surface. For this reason, the determination of the electric field at the point of reception by the radar is also given recursively.

The electric field of the intermediate section of the path, given by the contribution of a secondary ray of generic depth level $\text{depth}$ is given by

$$\vec{E}_{\text{depth}}^s = a \cdot b \cdot \Delta_{\text{depth}} \cdot \Omega_{\text{depth}} \cdot \vec{E}_{\text{depth}-1}$$

where $\vec{E}_{\text{depth}-1}$ is the electric field calculated for the ray correspondent to the section preceding the present one.

Ultimately, the generic formula for the electric field of the path received at the radar, for a reception at depth level $Z$ is

$$\vec{E}_{\text{received}}^s = \vec{E}_0 \cdot \Psi_Z \cdot \prod_{i=1}^{Z-1} (\vec{a}_i \cdot \vec{b}_i \cdot \Delta_i \cdot \Omega_i).$$

These calculations are performed for both horizontal and vertical polarization components. After all the rays are traced and the received ones are stored along with their scattered electric field value at the receiving point, the polarization components must be added, by the superposition principle, in their complex vector form, returning the whole scattered electric field of each ray. As introduced in algorithm 1, by summing the electric field contributions of all the received rays, the overall scattered electric field received at the radar is determined as

$$\vec{E}_{\text{total}}^s = \sum_{i=0}^{N-1} (\vec{E}_{h}^s + \vec{E}_{v}^s)$$

where $N$ is the total number of received rays, and $\vec{E}_{h}^s$ and $\vec{E}_{v}^s$ are the horizontal and vertical polarization components of the electric field of each ray, respectively.

Finally, the radar cross section for the present observation angle can be calculated by using the
definition of (2.27). As in (2.40) and (2.41), the result is given by

$$\sigma = 4\pi \left( \frac{|E_{total}|^2}{|E|^2} \right).$$

(4.52)

A conversion into a logarithmic scale, in this case to the units of $dB_{sm}$, should additionally take place for a better visualization of the results.
Chapter 5

Simulations and Results

This section features the results obtained through simulations of several different scenarios, derived from the code developed through the course of this project.

After enumerating the generic parameters involved in the simulations, some plots representing the resulting radar cross section of simple objects are shown, in order to validate the code. This is achieved by comparing the obtained results with theoretical expected values along with simulations performed on CST Studio Suite [43], a software that simulates electromagnetic field. Moreover, additional standard objects are analyzed and compared with their correspondent CST simulation results.

By changing some of the simulation parameters, the effects of each simulator feature are analyzed, being the size of the decomposition polygons the most relevant and interesting setting to consider. The effects of this parameter are discussed and potential solutions for adapting the simulator are suggested.

5.1 Simulation parameters and considerations

Concerning the performed simulations for the different scenarios presented in the following sections, some of the variables remained unchanged for every measurement, while others were chosen according to the specific situation, allowing a more clear observation of their effect on the final result.

In order to simplify the plotting of the results as a function of the angle of observation, the polar angle is always chosen as the specular direction, which contemplates the maximum radar cross section. Therefore, all simulations are performed with a polar incident angle of $\theta = 90^\circ$, for both the bistatic and the monostatic configurations. The angular sweep is accomplished by varying the azimuth angle $\phi$, which in the monostatic configuration system corresponds to the observation angle.

The electric field of the incident electromagnetic wave is chosen to have linear polarization oriented in the vertical direction, with a magnitude of 1 V/m for all scenarios. The correct implementation of the polarization components is also tested for simple measurements.

Relative to the angular sweep along the azimuth direction, the angle increment between consecutive measurements is chosen to be $0.5^\circ$. This parameter is very significant for the ray tracing process, being a strong influence on the number of obtained observation samples. Because this variable directly affects the dimension of the receiver, a small angle increment leads to many angles of observation having no scattered rays received at the location of the receiver, whereas a too large increment might lead to insufficient accuracy of results.

The distance at which the radar is located in relation to the center of the three-dimensional model
is 10 meters for all simulations. This does not satisfy the condition of far field region for entirety of the analyzed objects. However, a strict validation of this condition would lead to more complex and time consuming calculations, which would gravely lessen the computation performance. The chosen value shows a good compromise, with no significant penalty on the final results.

The frequency is 77 GHz, corresponding to a wavelength of, approximately, 3.9 mm for all measurements, stated as the value used by default for long range radar technology. Nevertheless, any other wavelength and frequency can be chosen and simulated, naturally influencing the outcome of the measurements.

Considering the ray tracing parameters of the simulation, the offset is settled as $10^{-5}$, which is a recommended value for satisfactory digital calculation accuracy.

The choice of the maximum depth has no real impact on the results of the scenarios tested in this project, given the simplicity of the three-dimensional objects used. The results are, therefore, independent of the value chosen for this parameter. For the simulation of more complex scenarios, a recommended setting is 3, which is a good compromise between sufficient accuracy of results and reasonable computation time.

Finally, the number of incident rays that represent the electromagnetic wave transmitted from the radar is the variable that most influences the simulation output. The determination of the area of each decomposition polygon is pivotal for the assessment of the performance of the implemented algorithm. The angular range in which these rays are launched from the radar determines how much of the scene the radar observes and is, therefore, selected in function of the dimensions of the analyzed object.

Moreover, the number of rays that are scattered from each scattering point and the field of view in which the rays are scattered greatly influences the accuracy of the results. The effects and further considerations concerning these parameters are presented throughout the following sections.

### 5.2 Scattering by a rectangular planar surface

The first set of results obtained from the implemented simulator have the objective of testing the correct operation of the code. By comparing the resulting plots with the ones obtained through theoretical calculations for each one of the scattering of objects by the physical optics approximation, the results can be validated.

As a first approach, a single incident ray launched from the transmitting radar is enough to properly represent the incident electromagnetic wave. In the next section, the influence of multiple incident rays on the analyzed object is explored.

#### 5.2.1 Bistatic setup with oblique incidence

In order to validate the simulation of a bistatic configuration system, a perfect electrical conducting plate with dimensions 0.02 x 0.02 m, where each edge corresponds to 5 times the wavelength, is analyzed. Exclusively on this scenario, where the angle of incidence remains constant throughout the whole observation angle sweep, it is essential to assert an angle of incidence in which the transmitting radar is oriented. This angle is measured in relation to the direction of the vector normal to the plate. As an example, the angle of incidence is set at 30°.
The dimensions of the camera for generation of scattered rays at the scattering point must contemplate at least the total number of observation samples. Therefore, the chosen dimension of the scattering camera is 91 x 91, amounting to a total of 8281 scattered rays, which, associated with a field of view of 90°, proves to deliver satisfactory results.

The bistatic radar cross section of this plate as a function of the angle of observation is plotted in Figure 5.1 compared with the theoretical calculations for the same scenario.

![Figure 5.1: Bistatic radar cross section of a rectangular plate for an angle of incidence of 30°.](image)

The simulation results show a full agreement with the theoretical ones, evidencing a maximum around the observation angle of 30°. The case of scattering by reflecting plates with dimensions in the order of the wavelength, in which the physical optics approximation reduces itself to the geometrical optics calculation, is therefore successfully validated.

### 5.2.2 Monostatic setup

Nevertheless, the most interesting results concerning the conventional operation of a radar are the ones obtained from a monostatic system configuration. For this case, the validation of the code is accomplished by measuring the monostatic radar cross section of a rectangular perfect electrical conducting plate with the same dimensions as the previous scenario.

In this situation, not only the number of scattered rays launched from the scattering point, but also the field of view of the scattering generation camera must increase to achieve more accurate results. A dimension of 551 x 551 scattered rays per scattering point and a field of view of 120° are selected.

For monostatic measurements, the angle of incidence is the same as the angle of observation and their consistent variation translates into an angular sweep, which ultimately generates the plot presented in Figure 5.2.

As expected, the maximum monostatic radar cross section is located at the specular direction, corresponding to an angle of observation of 0°, with a value of -9.23 dBsm, the same as the result obtained
by the formula for a plate of dimensions $a \times b$, as in

$$\sigma_{\text{max}} = \frac{4 \cdot \pi \cdot a^2 \cdot b^2}{\lambda^2}. \quad (5.1)$$

As predicted from the literature [3], the values present more accuracy for smaller angles of incidence, evidencing a small but consistent error for angles of observation further away from the specular direction.

An interesting simulation scenario involves the introduction of different polarization configurations of the incident electromagnetic wave. The results for a linearly polarized incident wave oriented along the horizontal direction are presented in section B.1. The coherence of the results shows that the calculations involving the polarization of waves are correct and introduce no errors in the code.

In order to test the magnitude of the effects that arise from increasing the dimensions of the object while maintaining the same distance between the radar and the geometrical center of the model, a similar square plate with an edge dimension correspondent to 50 times the wavelength is simulated. The results are plotted in Figure 5.3.

For this scenario, a dimension of $951 \times 951$ rays scattered from the intersection point proves to be suitable for a good accuracy when compared with the theoretical calculations. The comparison to another simulation with less scattered rays can be found in section B.2 demonstrating the effects of this parameter on the quality of the final results. Launching only $551 \times 551$ scattered rays from the scattering point, the simulation is more susceptible to errors and the plot presents a significant loss of accuracy in some points.

With the adequate amount of scattered rays, the maximum value at the specular direction of approximately $30 \text{ dBsm}$ corroborates the result derived from (5.1). Additionally, the oscillations show a satisfactory accuracy, once again revealing a slight deviation at larger angles of incidence.
5.3 Multiple incident rays

So far, all the simulations assume a single incident ray representing the electromagnetic wave transmitted from the radar. However, the dimension of the camera that generates the incident rays in the ray tracing algorithm is the most interesting parameter in the code, due to its significant influence on the final results of each simulation.

For this reason, the same square plate with dimensions 0.2 x 0.2 meters is simulated, this time using an incident ray generation camera with 10 x 10 incident rays. Additionally, a field of view of 1.146° is asserted for this camera, in order to direct the incident electromagnetic beam exclusively on the surface of the plate, making use of the total amount of emitted rays. The results of the simulation performed for this set of variables are presented in Figure 5.4.

Given that the rays are equidistantly distributed, there are 100 identical square decomposition polygons at the surface, each with a dimension correspondent to 5 times the wavelength, as in the simulation represented in Figure 5.2. The calculation of the radar cross section acquired by summing all contributions derived from each decomposition polygon proves to be successful only for the specular direction of 0°. Because the oscillation at which the radar cross section varies with the angle of observation heavily depends on the dimension of the surface where the scattering is determined, the observed shape is the same as for the case of a single incident ray on the plate of identical size. By simply dividing the radar cross section by the number of incident rays, the corrected final results show much more conformity with the expectation in respect to the range of values, with the exception of the specular angle.

An additional simulation is performed for a similar square plate, this time with a dimension of 15 times the wavelength. By launching 3 x 3 incident rays, with a field of view of 0.344°, each decomposition polygon has, once more, a square shape with dimension correspondent to 5 times the wavelength. The results for this scenario are displayed in Figure 5.5.

As expected, the radar cross section presents precisely a third of the oscillation frequency of the
expected results. It is important to observe that the division of the final values by the number of incident
rays efficiently and accurately corrects the local maximums of the plot. Concerning the range of values,
every third local maximum of the expected plot is correctly simulated.

On the other hand, the performed correction does not change the oscillation frequency, which still
contributes for an inconsistency of the final results. This infers that not only the number of incident rays,
but also the dimension of the whole surface must be taken into account when correcting the calcula-
tions. If, like in most practical cases, the dimension of the surface is unknown, only a shift of the plot is possible, which is why the implementation of a better fitting correction factor is very complex.

For surfaces where the physical optics approximation is reduced to the geometrical optics, such as this one, where the clear majority of the scattered electromagnetic field is directed towards the specular direction, this method presents some significant inconsistencies. However, because the geometrical optics approximation is more accurate for larger dimensions of the reflective plate, for objects where the surfaces are smaller, or curved, the geometrical optics fails to properly determine the occurring scattering effects. In situations where there is no concentration of the scattered field towards a specific direction, but the radiation is scattered in all directions, the physical optics is expected to perform better and deliver more adequate results.

To test this possibility, two standard three-dimensional objects are simulated in the following sections. Firstly, a cube represents a more complex scenario where the geometrical optics effects are predominant and thus this algorithm is expected to perform as the previous case. Secondly, a reflecting sphere tests the effects of the physical optics where the scattering mechanism is most visible and is, therefore, expected to be more adequate and less prone to inconsistencies.

5.4 PEC Cube

For the simulation of a perfect electric conducting cube, six reflective plates are assembled together in order to form the desired shape. Because the cube is chosen to have dimensions of 0.02 x 0.02 x 0.02 meters, each face has an edge corresponding to 5 times the wavelength and a square shape.

This time, the simulation is performed with a total range of observation of 180°, in order to properly observe the symmetry involved in the measurement. Furthermore, a number of scattered rays launched from each scattering point of 551 x 551 and a field of view of the scattering camera of 90° are selected.

The radar cross section of the described cube for one single incident ray is plotted in Figure 5.6.

The results are now compared with simulations performed on the electromagnetic field simulator CST Studio Suite. Since, for small objects, this software uses an integral equation solver, the results show small approximations around the local minimums which become more noticeable for observation angles further away from the specular direction. However, the plot agrees with the expectations in general, evidencing satisfactory values particularly around the specular directions, which are coherently situated at angles perpendicular to the cube’s faces.

On section B.3, the results for a simulation of the same cube with 2 x 2 incident rays are presented. The predicted effects caused by the multiple incident rays are corroborated. Not only do the results show a consistent shift correspondent to multiplying the values by a factor of 4, but also the shape of the plot is clearly varying with half the frequency as the results obtained on CST, which is expected given the number of rays incident on the model.

Therefore, an accurate and adequate measurement is obtained when using one single incident ray, while when applying multiple incident rays, a correction factor involving an adaptation of the values and of the oscillation frequency must be determined and applied.

An additional simulation on a larger cube is performed, in order to test the robustness of the code and the comparison of results with a different CST solver. On this case, a cube with dimensions 1 x 1 x 1 meters is designed. By selecting one single incident ray, 551 x 551 scattered rays and 120° of field of view of the scattering camera, the results shown in Figure 5.7 are obtained.
Here, the repercussions of the size of the plates that constitute the cube are evident. Larger plates lead to a much clearer reflection effect, which dominates over the scattering. Moreover, the most adequate CST solver to simulate targets of large dimensions is the asymptotic solver, which presents some small inconsistencies, especially around the specular direction, where most of the scattered field is concentrated towards.


5.5 Sphere

For the simulation of the scattering effect induced by a perfect electric conducting sphere, a model that approximates the representation of a curved surface must be used. Because three-dimensional digital models are always constituted by planar triangular surfaces, in order to appear smooth, the surface of the geometry needs to contain many polygons. A larger complexity of the model leads, of course, to larger compilation and running times, given that the duration of the simulation increases exponentially with the number of surfaces present on the scene.

The three-dimensional model of a sphere is designed on CST Studio Suite, with a diameter of 0.02 meters. The initially obtained model has 548 triangular facets. The model is subsequently submitted to a reduction of resolution on MeshLab [44], a mesh manipulation software. Now, with a total of 90 triangular facets, the model's complexity is considerably reduced, which allows the running time to decrease significantly. Evidently, the accuracy of results is compromised with such simplification, but a satisfactory compromise between the resolution of the shape, which still approximates to a slightly rough sphere, and the compilation time, which is now much more reasonable, is found.

Maintaining a dimension of the scattered ray generation camera of 551 x 551 scattered rays per scattering point and a field of view of this camera of 120°, the number of incident rays is now the main variable of interest on this simulation. By establishing a dimension of 10 x 10 incident rays for the incident ray generation camera, the results plotted in Figure 5.8 are obtained.

![Figure 5.8: Monostatic radar cross section of a small sphere.](image)

In this case, the size of the triangular facets that compose the model of the sphere are much smaller than in any other model previously simulated. Since these small polygons have dimensions smaller than the wavelength of the electromagnetic wave, the scattering effect dominates over the geometrical optics specular reflection, which produces an almost uniform dispersion of the radiation in all directions from each scattering point. Because there are no large planar surfaces in this model, the radiation is not concentrated towards a specific direction anymore, but presents an approximately constant value for all
angles of observation. A correction factor relative to the value range of 74 is, in this case, given by the number of incident rays that actually intersect the geometrical model, which ultimately corresponds to the number of scattering points. After dividing the radar cross section by this factor, the simulation shows adequate and satisfactory results.

The oscillation of the plot, that was so significant for the simulations of objects with planar surfaces, has now no significant consequences over the shape of the plot.

In order to demonstrate the influence of the number of incident rays on the simulation, a dimension of the incident ray generation camera of 5 x 5 is also tested. The results of the simulation depicting these parameters are plotted in [Section B.4] where a significant loss of accuracy of the final results is observed. For scenarios where the scattering effect is dominant, not only is it advantageous to launch multiple rays on the scene, but the largest possible amount of incident rays is indeed desired, in order to detect the surfaces of the target with the most possible detail.

The compilation time of this non-optimized code is, as expected, very large and almost unbearable. For the simplified model of the sphere with a total of 100 incident rays, the running time amounts to around 1 month of uninterrupted running on a standard portable computer.

For more advanced simulations, it is often required that the results are obtained much faster, ideally aiming for real-time computing. However, while these simulations are performed exclusively for testing purposes, the optimization of the code is not a focal point of this project. Suggestions of further improvements concerning the running time of this algorithm are explored on the next chapter.
Chapter 6

Conclusions

Emerging from a necessity for more accurate and precise simulations of the radar channel behaviour, the autonomous driving industry is persistently looking for better and more efficient algorithms, which must ultimately be able to exactly predict how a radar detects its surrounding environment. Because the scattering phenomena of the electromagnetic waves is most regularly predicted by stochastic methods, the simulation results obtained from the application of these algorithms do not always show the actual detection of a real-life long-range radar. Therefore, a deterministic algorithm is advantageous and thus the physical optics approximation arises as a potential solution to be tested and interpreted.

The code implemented in this thesis determines the scattering patterns of three-dimensional objects, through the application of the physical optics approximation. By analyzing different scenarios and comparing the results with patterns obtained from other algorithms and simulation software programs, this approach is tested and conclusions can be made.

The first step of this analysis focuses on the comparison between the physical optics and the geometrical optics approximation. Conceptually, the geometrical optics approximation is very adequate for simulating large planar reflective surfaces, but loses its accuracy when the dimensions of the obstacles approximate to the order of the size of the wavelength. As observed, for objects of reduced dimensions, the pattern obtained by the physical optics is more complex than a simple reflection of radiation towards the specular direction. The electric field is no longer concentrated towards a specific direction, but radiated in multiple directions, being the effects of diffraction around the edges and scattering at irregularities even more significant.

On the other hand, the fact that the results are the same for both algorithms when analyzing a plate of large dimensions proves that both techniques reduce to each other when the edge diffraction effects are not so dominant over the reflection. The geometrical optics, given its simplicity and effectiveness, can still be used for scenarios where large planar surfaces are recurrent. However, the results suggest that, even though the physical optics is a far more complex and demanding algorithm, it performs more accurately for objects with irregularities in the order of the size of the wavelength or with smooth curved surfaces, where the scattering is more noticeable.

The fact that the final results are so heavily dependent on the dimensions of the surfaces constitutes the main concern about the physical optics approach. An observation of the resulting radar cross section obtained for objects constituted by planar surfaces with dimensions larger than the wavelength suggests that the algorithm loses its purpose when multiple rays illuminate a single surface of the model. A correction factor can be implemented, which contemplates not only the absolute values but also the oscillation frequency of the plot, with the purpose of adapting the final results. However, this seems
counterproductive, since a better way of solving this particular problem would be to use the much simpler and time effective theory of geometrical optics.

On the other hand, for a standard object such as a reflecting sphere, where the scattering effect is the most dominant, the implemented code delivers quite satisfactory results. The oscillation frequency of the plot, which is so evident and critical for the case of specular reflection, is not a concern anymore and the values show good coherence with the expected results. Furthermore, the accuracy proves to increase with a higher number of incident rays on the model, which is sensible given that more incident rays correspond to more samples of electromagnetic field, which ultimately translates to a correct reading of the contour of the object. A compromise between the number of incident rays and the program running time must, thus, be assessed for each type of scenario. With the chosen settings for the scenario of a small perfect electric conducting sphere, the final results present a medium error of approximately 0.686 dBsm, which corresponds to a relative error of 1.96%.

Based on the results acquired from the performed measurements and respective conclusions, the ideal solution for the implementation of a deterministic software able to simulate the behaviour of a radar channel would be to associate the physical optics techniques explored in this project with the geometrical optics approximation. Even though the physical optics approach shows appropriate results for three-dimensional scenarios where the scattering effect is determinant, an algorithm programmed to thoroughly analyse and interpret the surfaces present in each obstacle can fairly minimize the complexity of the simulation and consequently the running time. If a surface is curved or presents irregularities in the order of the wavelength or smaller, the physical optics approximation is to be applied to guarantee an adequate set of results. However, should an object be composed by large planar surfaces when compared to the wavelength, a more straightforward and still accurate approach would be the geometrical optics.

The next step to assure a proper operation of this simulation algorithm should be, evidently, an optimization of the code. Since it is besides the focus of this project to create an efficient and optimized algorithm, the run time eventually becomes the greatest limitation for testing the different scenarios. An even more complex scenario, resembling a real environment would immediately lead to an unbearable amount of time which is not attainable in the scope of this thesis. An implementation of the simulator on the level of the graphics processing unit, through GPU programming is, therefore, desirable and should be pursued, since it allows the ray tracing algorithm to be processed much faster and more efficiently.

Furthermore, the investigation of the effects of the physical optics approach on more complex and irregular sets of obstacles is advantageous. An observation of the impact of this technique in the radar cross section of objects and a comparison with more specific complex algorithms, such as the integral equation method and the geometrical theory of diffraction, which are already extensively used in real time simulators, should also take place. Ultimately, it will be possible to assert which kind of objects and surfaces qualify to be analyzed by the physical optics approach and which ones should be simulated with a different kind of algorithm.
References


78


Appendix A

Pseudo-code of the simulator

```
Algorithm 1: "Main" function

Input: Simulation settings
Output: Radar Cross Section of an object
1 Read mesh file and store its information;
2 Determine dimension of reception sphere;

// Radar angular sweep
3 foreach Radar position do
4     foreach Incident ray do
5         Generate ray;
6         Determine direction of electric field polarization components;
7         Transform all vectors from radar space into world space;
     // Recursive function
8         Trace ray;
9     foreach $i^{th}$ received ray do
10        $E_{total}^{i} = \sum_{m=0}^{M-1} (E_{h}^{m} + E_{v}^{m});$
11        $\sigma = 4\pi \left( \frac{|E_{total}|^{2}}{|E|^{2}} \right);$
12    Print radar cross section;
```
Algorithm 2: "Trace ray" recursive function

**Input:** Ray attributes

**Output:** None

1. if $\text{Depth} \leq \text{Maximum depth}$ then
   2. if Ray intersects the model then
      3. foreach Scattered ray do
         4. Generate ray;
         5. Determine direction of induced current at surface;
         6. Transform all vectors from polygon space into world space;
            // New recursion: increase depth level by one and jump to the beginning of the function
            Trace ray;
            // End of recursion: function continues
   7. if Ray is signaled as received then
      8. $\vec{E}_{\text{depth}} = a \cdot b \cdot \Delta_{\text{depth}} \cdot \Omega_{\text{depth}}$;
      9. $\vec{E}_{\text{received}} = \vec{E}_{\text{received}} \cdot \vec{E}_{\text{depth}}$;
   10. else if Ray hits the receiver then
      11. Determine whether ray is unique for scattered wave;
      12. if Ray is unique then
         13. $\vec{E}_{\text{received}} = \vec{E}_0 \cdot \Psi_Z$;
         14. Signal ray as received and store its information;
            // Finish current level of recursion: depth level decreases by one

82
Appendix B

Additional Plots

B.1 Horizontal Polarization

Figure B.1: Monostatic radar cross section of rectangular plate for horizontally polarized incident wave.
B.2 Number of Scattered Rays

Figure B.2: Monostatic radar cross section of rectangular plate with less scattered rays.

B.3 Multiple incident rays on cube

Figure B.3: Monostatic radar cross section of small cube with multiple incident rays.
B.4 Number of incident rays on sphere

Figure B.4: Monostatic radar cross section of sphere with 5 x 5 incident rays.