



# SIMULAÇÃO E MAPEAMENTO SUBAQUÁTICO POR IMAGING SONAR, ATRAVÉS DE TEORIA GEOMÉTRICA DE RAIOS E APRENDIZADO EM ESPAÇOS DE HILBERT

Eduardo Elael de Melo Soares

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Ramon Romankevicius Costa

Rio de Janeiro  
Março de 2017

SIMULAÇÃO E MAPEAMENTO SUBAQUÁTICO POR IMAGING SONAR,  
ATRAVÉS DE TEORIA GEOMÉTRICA DE RAIOS E APRENDIZADO EM  
ESPAÇOS DE HILBERT

Eduardo Elael de Melo Soares

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO  
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE  
ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE  
JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A  
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA  
ELÉTRICA.

Examinada por:

---

Prof. Ramon Romankevicius Costa, D.Sc.

RIO DE JANEIRO, RJ – BRASIL  
MARÇO DE 2017

Elael de Melo Soares, Eduardo

Simulação e Mapeamento subaquático por Imaging Sonar, através de teoria geométrica de raios e aprendizado em espaços de Hilbert/Eduardo Elaél de Melo Soares. – Rio de Janeiro: UFRJ/COPPE, 2017.

X, 70 p.: il.; 29, 7cm.

Orientador: Ramon Romankevicius Costa

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2017.

Bibliography: p. 63 – 70.

1. Sonar. 2. Mapping. 3. 3D Reconstruction.  
4. Underwater. I. Costa, Ramon Romankevicius.  
II. Universidade Federal do Rio de Janeiro, COPPE,  
Programa de Engenharia Elétrica. III. Título.

*Mammy.*

# Agradecimentos

A Zeus,

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

## IMAGING SONAR UNDERWATER SIMULATION AND MAPPING THROUGH RAY THEORY AND HILBERT SPACES LEARNING

Eduardo Elael de Melo Soares

March/2017

Advisor: Ramon Romankevicius Costa

Department: Electrical Engineering

This dissertation on 3D underwater simulation and mapping consists of two pieces of software that is the main objective of the research, one for simulation and another for mapping. A mapping system usually comes together with a localization algorithm in what is called a SLAM (*Simultaneous Localization and Mapping*). SLAM is an active topic of research and has remarkable solutions using laser scanners, but most of the underwater SLAM is focused on 2D maps treating the environment as a floor plan or as 2.5D maps on the seafloor.

The reason for the problematic of underwater mapping, and thus SLAM, takes part in its sensor, i.e. sonars. Contrasting to laser-based systems used outside water which are precise low-noise sensors.

The standard sensor, sonar, is composed of hydrophones which allows them to measure sound in water. Therefore, enabling them to emit and receive sound waves, resembling microphones and speakers. They are divided in passive sonars and active sonars. Passive sonars can be applied to listen its surroundings, interpreting the sounds by their spectrum.

Meanwhile, active sonars emit a beam of sound in order to perceive and measure the environment by the echo created. Active sonars are classified in two important categories, profiling and imaging, based on its beam directional gain.

Profiling have a narrow pencil shaped beam with an aperture of about 1.7 degrees, i.e. the half power point. It is meant to have a similar response of a laser scanner, despite working with sound waves. At last, they do not correlate much, since they differ greatly on noise, response time and spatial resolution.

Imaging sonars will be the focus of this work, they employ a much wider beam than profiling sonar. It is used to having around 3 degrees angle of aperture in the

vertical direction, although keeping its same angular resolution on the horizontal plane. It does not have provide precise localization of the target, but a rather more general information. Therefore, being able to infer the presence of objects below or above its horizontal plane. From this point of view, each beam gives broad information about the region it ensonifies. Thus, fusing multiple beams from different directions and angles, containing overlapping areas, is expected to provide a better outline of the environment than just using isolated sonar responses.

Another classification, besides the one mentioned above, is the multi- or single beam. Firing multiple beams at once gives a faster rate, not requiring waiting for a response before redirecting the beam to its next angular position. When working with multibeam sonar, its downside remains its market value. Contrasting to the cheaper option, the single beam sonar, which will be further studied in this thesis.

# Contents

<b>List of Figures</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Purpose and Significance . . . . .	1
1.2 Review of the Literature . . . . .	2
1.3 Objectives . . . . .	2
1.4 Methodology . . . . .	3
1.5 Work Structure . . . . .	4
<b>2 Sonar Simulation</b>	<b>5</b>
2.1 Sonar . . . . .	5
2.1.1 Physics of Sound . . . . .	6
2.1.2 Sonar Principle of Operation . . . . .	7
2.1.3 Available Models . . . . .	12
2.2 Simulation . . . . .	13
2.2.1 Techniques Overview . . . . .	14
2.2.2 Ray Theory . . . . .	23
2.3 Environment . . . . .	31
2.3.1 Modeling . . . . .	32
2.3.2 Characterization . . . . .	33
2.4 Implementation . . . . .	35
2.4.1 Algorithm . . . . .	35
2.4.2 Results . . . . .	37
<b>3 Mathematical Preliminaries</b>	<b>41</b>
3.1 Hilbert Space . . . . .	41
3.2 RKHS - Reproducing Kernel Hilbert Space . . . . .	43
3.2.1 The Evaluation Functional . . . . .	43
3.2.2 Reproducing Kernels . . . . .	44
3.2.3 Feature Maps . . . . .	46
3.3 Probabilistic Regression . . . . .	46



3.3.1	Binary Logistic Regression . . . . .	47
3.3.2	Regression on Hilbert Spaces . . . . .	48
<b>4</b>	<b>Mapping</b>	<b>49</b>
4.1	Map Representation . . . . .	49
4.1.1	Discrete Map . . . . .	50
4.1.2	Map of Features . . . . .	52
4.1.3	Continuous Map . . . . .	52
4.2	Inverse Sonar Model . . . . .	54
4.2.1	Sonar on Feature Maps . . . . .	55
4.3	Map Learning . . . . .	55
4.3.1	Stochastic Gradient Descent - SGD . . . . .	56
4.4	Implementation . . . . .	57
4.4.1	Algorithm . . . . .	57
4.4.2	Results . . . . .	57
<b>5</b>	<b>Conclusion</b>	<b>62</b>
	<b>Bibliography</b>	<b>63</b>

# List of Figures

2.1	Depiction of the working principle of an <i>active sonar</i> . The red speaker-like object represents the transducer, responsible for emitting and receiving the acoustic wave. . . . .	8
2.2	Visualization of a multipath for a high frequency short pulse (much smaller than delay times). Black vectors show the path taken by the sound wave. Red dashed vector shows the calculated distance by equation 2.7. . . . .	9
2.3	Resolution as the minimum discernible distance between ecos. . . . .	10
2.4	Farfield beam shape and its <i>beamwidth</i> (in blue). . . . .	11
2.5	Incident and reflected rays with respective angles. . . . .	24
2.6	Scattering reflection weighted. . . . .	26
2.7	BRDF and BTDF for transmitted, reflected and scattered rays. ( <a href="#">©User:Jurohi</a> / <a href="#">Wikimedia Commons</a> / <a href="#">CC-BY-SA-3.0</a> ) . . . . .	27
2.8	The source E and the virtual sources $E'$ , $E'_1$ , $E'_2$ , $E'_3$ . In this case, $E'_3$ is not a visible virtual source. . . . .	28
2.9	Source on a perfect tilling. The simple pattern facilitates computation. . . . .	29
2.10	A 2D view of a pyramidal beam tracing. . . . .	29
2.11	Materials reflective characteristics from MILLER [1]. . . . .	34
2.12	Exemple of an array for a bearing direction. Actual arrays are longer, depending on resolution. . . . .	35
2.13	Overview of the simulation algorithm. . . . .	36
2.14	Ray Tracing: Red lines are specular reflections, green lines are diffuse backscattering. . . . .	38
2.15	Sonar simulation for the box-like scene. . . . .	39
2.16	Sonar simulation for the complex scene. . . . .	40
4.1	Mapping with 10% of available beams. . . . .	59
4.2	Mapping with 30% of available beams. . . . .	60
4.3	Mapping with 100% of available beams. . . . .	61

# Chapter 1

## Introduction

Profiling sonars is the mimic laser scanner analog for underwater mapping, so most of the approaches on 3D underwater SLAM (**S**imultaneous **L**ocalization and **M**apping) focus on applying laser scanner techniques to profiling sonars, e.g. point cloud reconstruction. On the other hand, imaging sonars are usually cheaper and gives more information per sound beam. So having the possibility of using imaging sonar and benefiting from its extra information is a win-win strategy.

Besides the definition of the sonar, one should carefully look into the meaning of mapping, because, it can be interpreted in different ways, depending on the context. Taking what they all have in common, it is possible to generically define mapping as being the process of received environmental data to characterize the surroundings. The meaning characterization is where the difference lays, how the environs are going to be represented is dependent on the application.

On a SLAM system there is no intrinsic need for a human readable map. In such a system, it may be more interesting to store the map information, the characterization, only through its most representative features, if that is what matters for the localization procedure.

Still, if the map is to be seen by a human it should store more information about the environment, so that it can be displayed as a usual map, 3D or 2D depending on the case. This representation also guides how the data could be stored, e.g. if it wants to show a surface, it can be stored as a elevation map, or if one wants to see a 3D object it can be stored as a point cloud.

### 1.1 Purpose and Significance

The mapping of underwater environments is not just a part of a SLAM system. It might have importance on its own, it can be used for humans to visualize things that could not be seeing otherwise.

In the ROSA (*Robô para Operações de Stoplogs Alagados*) project, developed by LEAD/COPPETEC to ESBR, one of the goals is to make a reconstruction of the hydroelectric power plant turbine entrance to spot any underwater debris that could block the lowering of stoplogs, used to block the water flow, and then cause delays and setbacks. When it happens for a stoplock to stuck the need for a diver also incur in further human risk, so even then it is important to know the surroundings.

On a developing perspective, besides the importance for the ROSA project, it has characteristics that makes it appropriate for mapping. It has a lifting beam for inserting the stoplogs into the water that can act as stable fixation point for any sonar structure. So reducing the impact of poor a localization system on the mapping. It also has a known and not-so-complex ground truth environment (besides possible debris). Thus data collected there is suitable for testing the mapping algorithm in a real world environment, as opposed to lab testing. Although the noise environment might impair greatly the mapping, and subsequent visual reconstruction.

It is also a integral part ROV's, where it gives feedback to the operator for him to know where it is or/and what he is doing, especially because cameras do not have a very useful range. And glaringly its automated counterpart (AUV's) as a requisite for SLAM.

## 1.2 Review of the Literature

\*\*\*\*\*

1)Comment other simulators and what they miss (multipath or forwardlooking, generally). COIRAS and GROEN [2] - Simulation SAS COIRAS *et al.* [3] - GPU based SAS

2)Cite first principles confusion on references (Intensity/RMS Acoustic Pressure), Lambertian reflection, etc...

3)Comment other mappings and what they miss (3D sonar or computability usually).

\*\*\*\*\*

## 1.3 Objectives

The general objective is to map an underwater environment. But that can be done in many ways, for this thesis the main goal is to implement optimization of the visible surface based on the expected response of a sonar beam in a given direction and fuse multiple views using binary bayesian filters.

The output of the whole process will be a 3D human-readable map of the scanned environment. The optimization step might, also, supply information about the surface material, specifically its reflectance. A technique based on previously articles.

Fuse all the data on a octree structure using a robotic framework, named ROCK, and implement a visualization for the reconstructed underwater map. The data source shall be a imaging sonar mounted on a pan-tilt unit, so to provide the sonar extra degrees of freedom.

## 1.4 Methodology

The work is divided into three main branches. Each having a theoretical research followed by implementation.

The sonar model definition starts with a compilation on the description of sonar varieties, sound waves physical properties on the water, reflection, sonar directional gain and sources of noise. Then tests are realised on known environments so to fit and validate the proposed model. The test results are finally used to create the full sonar model, including metric conventions by the sonar protocol.

The second branch deal with the map filling over a discretized space, basically a Binary Bayesian Filter implementation. That is the standard state of the art technology for mapping [4].

The integration of the sonar model with the Bayesian Filter give the means to process sonar data. So data acquired on the LNDC/UFRJ tank (Laboratório de Ensaios Não Destrutivos, Corrosão e Soldagem - which loosely translated means Non-Destructive Testing, Corrosion and Welding Laboratory) and on the Jirau Power Plant, by means of the ROSA/COPPETEC project, will be processed and compared to the tank and power plant entry layout, respectively.

In a more complex endeavor, which is the third branch, a theoretical derivation for the optimization of 2D surfaces embed on 3D environments will take place. It uses the sonar response as expectation and, instead of having a fixed relation between the measured environment and the sonar response, aim to better infer the underlying geometry of the surrounds and also retrieve some information about the material's reflectivity properties. This optimization algorithm will then be integrated to the Bayesian Filter. The same data processed by the combination "Filter + Sonar Model" shall now be processed by "Filter + Optimization".

Based on the literature, even with not much similar works, the reconstruction with *a priori* sonar model will probably experience problems when reconstructing corners, shallow angle surfaces, very complex (intense multipath) or on highly noise environments. For the optimization, it shall encounter similar issues, but a less accentuated shallow angle quality degradation and an overall less blurry reconstruction.

tion.

## 1.5 Work Structure

\*\*\*\*\*

- 1) First principles -> Ray Theory (other techniques along the way)
- 2) Environments and Materials
- 3) Simulation and Results
- 4) Math to Mapping
- 5) Mapping concept
- 6) Mapping implementation and results
- 7) Conclusion / Comparison

\*\*\*\*\*

# Chapter 2

## Sonar Simulation

If you cause your ship to stop, and place the head of a long tube in the water and place the outer extremity in your ear, you will hear ships at a great distance from you.

---

*Leonardo Da Vinci, 1490*

The idea behind simulation is to have a flexible environment where the system (e.g. sonar, reconstruction model) can be tested on a variety of conditions and the ground truth is well known. It is a mature and wide spread mechanism for development of new sonar technologies [5].

Opening this chapter, it will be presented physical foundations behind sonars, and its existing technologies and models. Next describing the envisioned environmental properties, and ending with a more detailed view of the simulation technique used.

### 2.1 Sonar

Throughout this thesis one specific type of sonar will be considered, the mechanical imaging active sonar (section 2.1.3). Sonars have a common underlying principle of operation, but vary greatly on application and hardware constitution.

Sonars are, in some sense, the acoustic analog of a camera. They use sound, instead of light, to capture information about the environment. Hence, to better understand *how* they operate and *what* they are used for, it is important to have a clear concept of sound.

### 2.1.1 Physics of Sound

The phenomenon that humans percieve as sound is a pressure wave which amplitude excesses the medium's mean pressure [6]. It can be referred to as *compressional* or *longitudinal* waves, contrasting with *transversal waves*. The difference between these two kinds of waves relies on the direction of the movement of the particles, being parallel or perpendicular to the propagation of the wave, respectively[7].

On the particular, but usefull, conditions of low energy phenomena[8] (with some other suitable requirements<sup>1</sup>) the pressure pertubation wave can be described as the *D'Alembert equation*:

$$\nabla^2 \Phi - \frac{1}{c_0^2} \frac{\partial^2}{\partial t^2} \Phi = 0 \quad (2.1)$$

Where  $\Phi$  is the velocity potencial, a scalar field that helps describing the sound propagation. Its relation to sound pressure is:

$$p = -\rho \frac{\partial}{\partial t} \Phi$$

Which can be directly described as:

$$\nabla^2 p - \frac{1}{c_0^2} \frac{\partial^2}{\partial t^2} p = 0 \quad (2.2)$$

Where  $p$  is the pressure deviation from the mediums,  $\rho$  the density,  $c_0$  is the local sound speed and  $\nabla^2$  stands for the Laplace operator. These equations are only valid in free space (no source), however discrete variations of the medium are treated as boundary conditions, giving origin to reflection and refraction.

Besides pressure, sound has another important derived property: intensity. Much as the case of electromagnetic waves, sound intensity (or acoustic intensity) measures the mean value of the sound energy flux (i.e. energy rate per area):

$$\vec{I} = \overline{p\vec{v}} \quad (2.3)$$

Where  $\vec{I}$  represents the *acoustic intensity* vector,  $\vec{v}$  the *acoustic velocity* (i.e. the velocity of a particle in the medium) and overline the mean over some time period. The *acoustic velocity* can also be derived from the velocity potencial  $\Phi$  as:

$$\vec{v} = \nabla \Phi$$

When considering a wave far from its source, solutions to the equation 2.2 give rise to a *plane wave*( where the coherent wave front propagate in a plane). It makes clear the relationship between  $\vec{v}$  and  $p$ :

---

<sup>1</sup>A perfect simple fluid in an initial state of stationary homogeneous equilibrium



$$\vec{v} = \frac{p}{\rho c_0} \vec{n}_0$$

Where  $\vec{n}_0$  is the unit normal vector to the wavefront. Plugging it back to equation 2.3:

$$\vec{I} = \frac{1}{\rho c_0} \overline{p^2} \vec{n}_0 \quad (2.4)$$

This equation shows the proportionality between the *acoustic intensity* and the mean squared of the pressure. The inverse of the proportionality constant  $\rho c_0$  is called the *characteristic impedance* because it measures the degree of “resistance to propagation” of the medium.

Because the acoustic intensity (and related quantities) varies in orders of magnitude while propagating, it is common to quantify it on a logarithmic scale, specifically *decibels* (dB)[9]:

$$I_{dB} = 10 \log_{10} \left( \frac{I}{I_0} \right) \quad (2.5)$$

Here  $I_{dB}$  is the intensity measured in *decibels*,  $I$  is the intensity value and  $I_0$  a reference intensity values, usually defined somewhere near the source. In the case of reflected/refracted wave,  $I_0$  may also refer to the intensity of the incoming wave. A direct relation between the magnitude of intensity and pressure is found by applying equation 2.4 on equation 2.5:

$$I_{dB} = 20 \log_{10} \left( \frac{p_{rms}}{p_0} \right) \quad (2.6)$$

Where  $p_{rms}$  is the *rms* (Root mean squared) value of the wave’s pressure ( $\sqrt{\overline{p^2}}$ ) and  $p_0$  is a pressure value of reference, for underwater acoustics this value is the microPascal ( $p_0 = 1 \mu\text{Pa}$ )[9].

### 2.1.2 Sonar Principle of Operation

The name Sonar (*Sound Navigation And Ranging*) was originally conceived for the technique that uses acoustic waves on water for navigation, communication and detection, but nowadays it is also used for the equipment that generate/receive these sound waves.

The history of sonar has began on the year of 1490 through the statement of Leonardo Da Vinci aforementioned on the epigraph of this chapter[10]. That was the birth of *passive sonar*’s technology, where the objective is to listen (receive and process sound waves) the noise from ships, animals and other objects in an attempt to detect and recognize its origin.

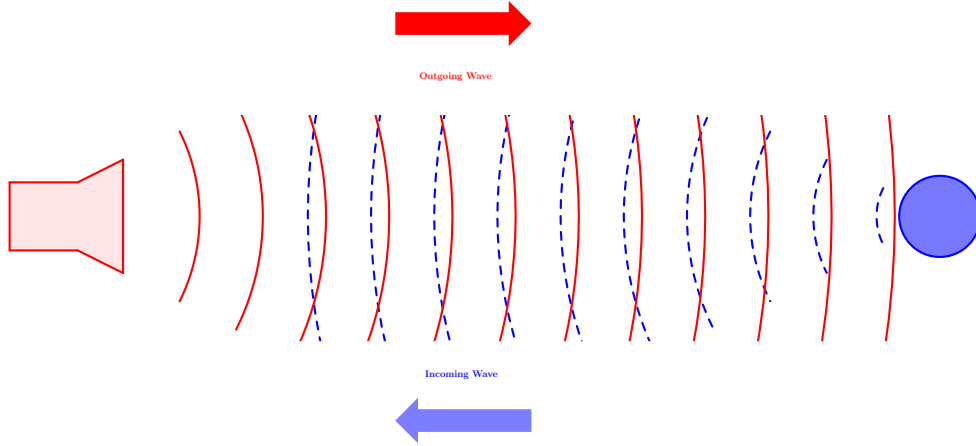


Figure 2.1: Depiction of the working principle of an *active sonar*. The red speaker-like object represents the transducer, responsible for emitting and receiving the acoustic wave.

## Active Sonar

The concept of an *active sonar*, one that emits a sound wave and detects its return (as in figure 2.1), is much recent. The loss of the *HMS Titanic* due to a collision with an iceberg during its first voyage on April 15 of 1912 [11] fostered the development of a sonar to detect objects kilometers away. Also, during World War I, Allied shipping losses to U-boat attacks further stimulated advances on techniques for uncovering of submerged enemies.

Active sonars are ranging sensor and the way they infer distance is by measuring the time between the emission and reception of an acoustic pulse (a time bounded sound wave) like on 2.1. The mean sound speed of the medium throughout the path traveled by the pulse has to be known, in order to reckon space from delay[9]:

$$R = \frac{c_0 \Delta t}{2} \quad (2.7)$$

Where  $R$  is the distance between the source and the target,  $c_0$  is the mean sound speed,  $\Delta t$  is the delay between pulse emission and reception, and the denominator 2 is a consequence of time measuring the two way trip of the pulse. When sound speed varies greatly, as in long distance travels in the ocean, special care should be taken[5], e.g. stratification of the environment by same sound speed layers.

## Multipath

Besides sound speed variation, another common issue is *multipath*. The moment a sound wave encounters an interface (e.g. sea floor, water surface or an obstacle) it does not bounces back to the source, it also undergoes refraction and reflection in

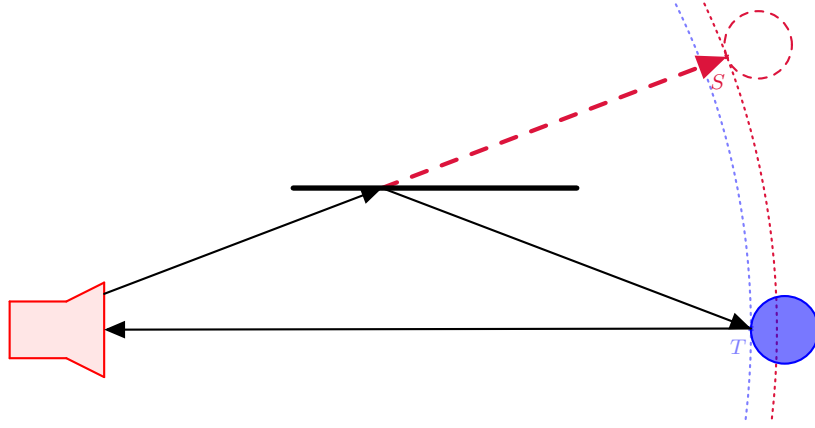


Figure 2.2: Visualization of a multipath for a high frequency short pulse (much smaller than delay times). Black vectors show the path taken by the sound wave. Red dashed vector shows the calculated distance by equation 2.7.

other directions. Thus, an echo that has traveled a longer path may also arrive, causing a naive application of equation 2.7 to predict the presence of an object further away (figure 2.2). For low-frequency stable signals, the contribution of all multipaths creates an interference pattern[9], a fact that will not be further explored.

### Sonar Resolution and Chirp Pulses

The minimum distance (or echo delay) that can be resolved by the sonar, depends on the type of pulse emitted (figure 2.3). There are two main types of pulse: *single frequency* and *chirp*[12, 13]. Some sonars use dual frequency to overcome the trade-off between reach and resolution, given that low-frequency has a longer range and high-frequency a better resolution.

For single frequency sonar, the limit resolution ( $\delta R$ ) depends directly on the pulse length ( $\Delta L$ ):

$$\delta R = \frac{c_0 \Delta L}{2}$$

However, that limitation can be overcome by the use of pulse compression (a cross-correlation filter like matched filters). In this case, a linear varying frequency signal (chirp), or similar multifrequency systems, has its resolution related to the bandwidth ( $\Delta BW$ ):

$$\delta R = \frac{c_0}{2 \Delta BW}$$

The seabed contains sediments that attenuate and retard incident sound waves, so sonars use Gaussian pulse spectrum as a measure to mitigate the impacts on temporal resolution. Although energy is lost, the Gaussian shape is nearly preserved

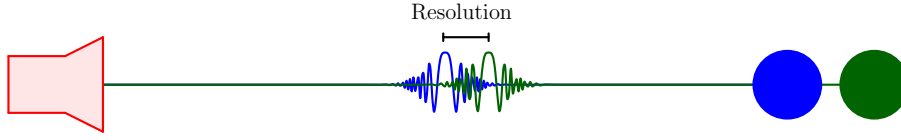


Figure 2.3: Resolution as the minimum discernible distance between ecos.

increasing sub-bottom penetration and avoiding band-width loss.

## Bearing

The direction where the echo comes from cannot be directly obtained using only one hydrophone (underwater sound transducer). There are two main elements to consider for bearing estimation: *beamwidth* and *hydrophone arrays*.

The majority of simple sonars have only one hydrophone acting as source and receiver. They cannot distinguish the direction of the incoming wave. It is possible to use its *beamwidth* to narrow down the region of the echo origin, being necessary to rotate the transducer in order to capture other directions.

The beam shape of a hydrophone is its directional gain, i.e. the ratio between the intensity of the emitted signal, in a given direction, and the maximum intensity. It also acts as proportional loss of intensity when measuring the received signal incoming from some direction. The concept of a 2-way beam shape follows directly as the net result of transmission and reception. Mathematically, it results in squaring the beam shape. All these concepts are meaningful only if sufficiently far from the source, a region called farfield[14].

The *beamwidth*, in turn, is a simplifying concept. The conventional definition is the point where intensity reaches 70% of its peak value, or  $-3\text{dB}$ . The 2-way beam shape reduces the *beamwidth* to about 72% w.r.t. the 1-way beam beamwidth. If the diameter ( $D$ ) of the transducer is large compared to the acoustic wavelength ( $\lambda$ ), the *beamwidth*( $\beta$ ) can be approximated as:

$$\beta \approx \sin^{-1} \left( \frac{\lambda}{D} \right)$$

On the other hand, there are hydrophone arrays that can infer the sound direction by relating the spacing between the transducers with the signal difference received by each of them [15, 16]. One technique is very similar to *multilateration*, simply compute the distance measured by each transducer and use this information to compute the direction of the incoming sound wave.

Another possibility is to apply signal processing by delaying the received signal

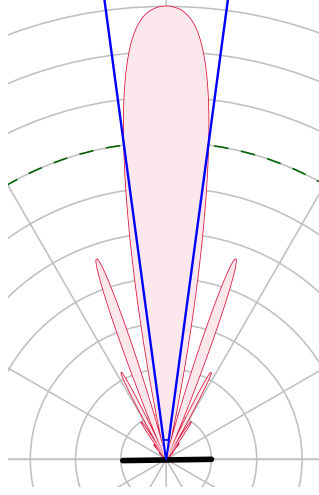


Figure 2.4: Farfield beam shape and its *beamwidth* (in blue).

from one hydrophone w.r.t. the other and adding them together. The constructive/destructive interference affectively changes the directivity of the array and, thus, can be used to find which direction gives the strongest echo. This is known as *beamforming*.

The sequence of transducers can be made into a two dimensional array (a.k.a. a grid), making it possible to detect a full 3D direction.

### TVG - Time Varying Gain

As sound waves propagates, they lose intensity through spreading and absorption. Spreading loss is usually considered to be an inverse quadratic law[5], as this is the closed surface area progression for a time-like wavefront. But for cylindrical spreading it is a simple inverse law, and for perfect plane wave there is no loss.

Absorption is conditional on the water characteristics and is modeled as a slow exponential decay. Together with spreading, they are referred as Transmission Loss (TL), in decibels (dB):

$$TL = 20 \log_{10}(r) + \alpha r$$

Where  $r$  is the wave's total traveled distance and  $\alpha$  a water dependent parameter (with order of magnitude of  $\approx 10^{-2}$ dB/m). Sonars use this equation, with a saturation around 40dB, to compensate for the echo loss[17]. And, as distance is inferred from time measurements, this compensative gain is named Time Varying Gain (TGV). CHEW and CHITRE [18] estimated TGV gain for a Tritech's Micron sonar and suggests that it agrees with the expected.

### 2.1.3 Available Models

Active Sonars, besides having a common working principle, present themselves in different models for different applications. DOBSON [19] summarized the most relevant ones.

#### Mechanically Scanning

As seen on subsection 2.1.2, it is possible to use the beam shape as a way to reduce the number of possible incoming directions for an echo. This is the idea behind a mechanically scanning sonar. As a consequence, the sonar has to rotate the hydrophone to cover a full, or partial, 360° sector.

The angular step between different hydrophone's positions is dependent on the desired resolution, smaller steps gives a better resolution, but takes longer by doing so.

1. **Profiling** - possessing a narrow conical beam shape, they are the acoustic analog of a laser scanner (although they still have a much larger apperture than a laser). Only a single echo is recorded for each angular position, either the strongest or the first to return. Typically applied for pipeline surveillance, they can spot structural differences and objects on sea floor.
2. **Imaging** - its fan shaped narrow beam covers a wider area than the profiling type, making it very usefull for navigation and obstacle avoidance on ROVs<sup>2</sup>. As its beam is wide, it usually hits the surface obliquely, receiving several echos per acoustic pulse. Each echo is displayed at a distance determined by equation 2.7 and with its strength mapped to a color scale.
3. **Side Scan** (a.k.a. towfish) - Can be either mounted on each side of a boat's hull or towed behind. With a beam shape similar to an imaging sonar, it can provide a sophisticated image of the sea floor.
4. **Echo-sounder** - mounted below a boat, it has a narrow pulse (as a profiling) with the single purpose of measuring the depth of water. It is typically applied to help with navigation or constructing depths charts.

#### Multibeam

Multibeam sonar are based on the technique of *beamforming* (described on section 2.1.2). It has several hydrophones, rendering it able to scan an underwater region with no moving parts.

---

<sup>2</sup>Remotely Operated underwater Vehicle

1. **Profiling** - similar in application to its mechanically scanning counterpart, it has multiple narrow conical beam receivers that record the signal. Instead of moving its transducer, it amplifies and processes (through *beamforming*) the received signal to identify the position of the strongest returned echo, then creating a high-speed cross-sectional profile.
2. **Imaging** - besides being much quicker than a Mechanical Imaging Sonar, it generates a crispier image of the underwater environment. It possesses a wide angle acoustic transmitter and multiple narrow beam receivers, applying *beamforming* to the received signal.

The array size of hydrophones is critical for enhancing resolution, longer arrays have a better angular resolution. To overcome physical limitations a technique known as *SAS* (Synthetic Aperture Sonar) may be applied. The transmission of several acoustic pulses in a line is used to emulate the presence of a longer array, by means of signal processing on the reception.

## 2.2 Simulation

*Computacional ocean acoustics* is the subfield of knowledge which explores algorithms that model the ocean as an acoustic medium. Works on this matter are well documented by ETTER [5]. Most of these works focus on very long range simulations, with its most important features been the ocean floor and sub-bottom region.

This work aims to reconstruct and simulate near-range partially closed environments, as those created by humans. The motivation for such a choice comes from hydroelectric power plant water intake, it is a corridor-like environment with possible obstacles on the bottom. This kind of environment is not well covered on the underwater acoustics literature, as such, some of the simulations techniques are borrowed from the closely related area of *room acoustics*.

When constructing a simulation, one has to consider the trade-off between simplicity/performance/accuracy. There are several possible techniques with different applications and assumptions, this chapter will cover the most classic ones and further explore *ray theory*, which has been used for the simulation presented here. For a more comprehensive view on this and other techniques, see LURTON [9], JENSEN *et al.* [20].

### 2.2.1 Techniques Overview

The idea behind sound simulation techniques is to solve the wave equation (eq. 2.2) considering all the physical interfaces as boundary conditions. The equation, however, cannot be analitically solved due to common present discontinuities caused by occlusions, specular highlights and other facts that result in large variations of field over small regions of the domain of integration[21].

The single most important reason that differentiates the several approaches described here is the *wave frequency*. For high-frequency (where sound speed do not vary much in a wavelength scale) geometric methods (ray theory) are justifiable and preferable (in the computational sense) [22]. In the case of low/mid - frequency or in the presence of caustics<sup>3</sup>, other wave methods (e.g. finite elements, normal modes, parabolic approximation) should be applied.

Instead of using the full wave equation, the methods work with a simplified time-independent version. It can be assumed that the equation 2.2:

$$\nabla^2 p = \frac{1}{c_0^2} \frac{\partial^2}{\partial t^2} p$$

Has a solution where time dependence is a harmonic function:

$$p(\mathbf{x}, t) = \text{Re}(\psi(\mathbf{x})e^{-i\omega t}) \quad (2.8)$$

Here  $\mathbf{x}$  is the space coordinate while  $t$  is the time. The exponential function considered is the complex exponetial with  $i$  being the imaginary unit. The real part  $\text{Re}(\bullet)$  is taken as  $p$  is real-valued, but  $\phi(\mathbf{x})$  is a complex-valued function over space.

Substituting it back into equation 2.2, gives (dropping explicit parameters again):

$$\text{Re}(\nabla^2 \psi e^{-i\omega t}) = \text{Re}(-(\frac{\omega}{c_0})^2 \psi e^{-i\omega t})$$

Defining  $k \equiv \frac{\omega}{c_0}$  (a.k.a. the wave number) and rearranging terms:

$$\text{Re}((\nabla^2 \psi - k^2 \psi) e^{-i\omega t}) = 0$$

As the harmonic  $e^{-i\omega t}$  is equivalent to a rotation in the complex plane, the equation will be satisfied for all  $t$  if:

$$(\nabla^2 - k^2)\psi = 0 \quad (2.9)$$

Equation 2.9 is known as the (homogeneous) *Helmholtz equation* and describe the time-independent part of the wave propagation. The values of  $k$  and  $\omega$  can be

---

<sup>3</sup>A region of high constructive interference that geometrically gives a point of infinity rays convegence.



physically interpreted as the spatial and temporal angular frequency of the wave. As the wave equation is linear, superposition applies, being reasonable to take into consideration one wave frequency at a time and superpose all these harmonics by Fourier synthesis[8].

Fourier synthesis is the calculation by *Helmholtz equation* of each independent frequency:

$$\psi_\omega(\mathbf{x}) = \int_{-\infty}^{\infty} p(\mathbf{x}, t) e^{i\omega t} dt$$

And reconstruct the wave equation back by:

$$p(\mathbf{x}, t) = \int_{-\infty}^{\infty} \psi_\omega(\mathbf{x}) e^{-i\omega t} d\omega$$

These equations are the inverse and forward Fourier transform, respectively.

## FEM - Finite Element Method

Finite Element Methods try to numerically find a solution to the wave equation by discretizing space, and time in some cases. It considers the equation 2.2 for inside the environment and the boundary conditions:

$$\frac{\partial}{\partial \mathbf{n}} p = -\rho_0 \frac{\partial^2 x_n}{\partial t^2} \quad \text{on the source}$$

$$c_0 \frac{\partial}{\partial \mathbf{n}} p = -\frac{1 - R_c}{1 + R_c} \frac{\partial}{\partial t} p \quad \text{on other interfaces}$$

Where  $\mathbf{n}$  is the normal direction of the surface,  $\rho_0$  is the medium density and  $x_n$  is the displacement of the acoustic membrane. The reflection coefficient  $R_c$  might depend on the interface, but as FEM are used for small frequency bands, it is a minor problem[23].

To create a linear system, the pressure function is approximated by a superposition of functions, e.g. sum of piecewise quadratic functions  $p(\mathbf{x}, t) = \sum_{i=0}^N p_i(t) \varphi_i(\mathbf{x})$ , applied to the wave and boundary equation and integrated w.r.t.  $\varphi_i$ . Giving a large set of ordinary differential equations:

$$\begin{aligned} M\ddot{p} + D\dot{p} + Kp &= Fu \\ y &= Pp \end{aligned}$$

With  $M, D, K$  being  $N \times N$  matrices,  $p$  being the vector with coefficients  $p_i$ ,  $Fu$  the input converted into a force and  $P$  some selection matrix to output the pressure

on the desired points. This model can be written and solved as generalized state-space model:

$$\begin{aligned} E\dot{\hat{x}} &= A\hat{x} + Bu \\ y &= C\hat{x} \end{aligned}$$

$$\hat{x} = \begin{bmatrix} p \\ \dot{p} \end{bmatrix}, \quad E = \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix}, \quad A = \begin{bmatrix} 0 & I \\ -K & -D \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ F \end{bmatrix} \quad \text{and} \quad C = \begin{bmatrix} P & 0 \end{bmatrix}$$

There can be made more simplifications, but it is enough to highlight the limitations of the method (For more details, refer to DEINES *et al.* [23]). The dimension of the state is related to the number of functions used to approximate  $N$  (two times without any further simplification) that depends on the frequency:

$$N = \left( \frac{nLf}{c_0} \right)^3$$

Where  $L$  is a typical dimension in the environment,  $f = \frac{\omega}{2\pi}$  the frequency and  $n$  the number of elements per wave, that should be 3 or 4 for a good approximation[23].

Given that the size of the state increases as the cube of the frequency, the technique can only be applied to low-frequency signals, which is not applicable to high-frequency sonar as envisioned by this work.

There is also a boundary method that uses surface integral form of the wave equation, but suffers from similar restrictions. FUNKHOUSER *et al.* [21] gives a brief introduction of the subject.

## Ray theory

Geometric approaches like ray theory dates back to Newton and the corpuscular theory of light. Later found to be better described as a wave, the geometric theory of light is still a very important and useful tool. The sound ray theory follows a similar path, they both apply to high frequency waves, but typical sound waves have low/mid-frequency while visible light has a much higher frequency.

To describe the ray from the wave theory, it starts by solving the Helmholtz equation (eq 2.9) with a generic complex function of space through a polar decomposition[24, 25]:

$$\psi(\mathbf{x}) = \mathcal{A}(\mathbf{x})e^{iw\tau(\mathbf{x})} \tag{2.10}$$

Here  $\tau(\mathbf{x})$  can be interpreted as the time it takes the sound to reach the location  $\mathbf{x}$  and  $\mathcal{A}(\mathbf{x})$  the amplitude of the signal at that point. Substituting it back to equation 2.9 and separating real and imaginary parts, two equations can be obtained (dropping arguments):

$$\frac{\nabla^2 \mathcal{A}}{\mathcal{A}} - (w \nabla \tau)^2 + k^2 = 0 \quad (2.11a)$$

$$2(\nabla \mathcal{A} \cdot \nabla \tau) + \mathcal{A} \nabla^2 \tau = 0 \quad (2.11b)$$

The geometric approximation is the assumption that the amplitude does not change much on the wavelenth scale, mathematically expressed as:

$$\frac{\nabla^2 \mathcal{A}}{\mathcal{A}} \ll k^2 \quad (2.12)$$

Applying this approximation to equation 2.11a and using the fact that  $\omega = kc_0$ :

$$\|\nabla \tau\| = \frac{1}{c_0} \quad (2.13)$$

The equation 2.13 is known as the *Eikonal equation* and defines the surfaces of constant phase. Equation 2.11b, called the transport equation, can then be used to find the pressure amplitude of the wave. However, when treating instensity conservation of energy can be used as will be seen later on subsection 2.2.2.

The rays are, by definition, the perpendicular lines to the wavefronts defined by equation 2.13. Which, with length parametrization, is:

$$\frac{dr}{ds} = c_0 \nabla \tau \quad (2.14)$$

Where  $r(s)$  is the path follow by the ray, and  $s$  the ray length. To verify that it is a length parametrization square equation 2.14 and use equation 2.13 to find:

$$\left\| \frac{dr}{ds} \right\| = 1$$

Showing that it is a unit norm tangent vetor. To find the ray path, first take the gradient of equation 2.13 squared:

$$\nabla(\|\nabla \tau\|^2) = \nabla \left( \frac{1}{c_0^2} \right) \quad (2.15a)$$

$$2\mathbf{H}(\tau) \nabla \tau = 2 \frac{1}{c_0} \nabla \left( \frac{1}{c_0} \right) \quad (2.15b)$$

$$c_0 \mathbf{H}(\tau) \nabla \tau = \nabla \left( \frac{1}{c_0} \right) \quad (2.15c)$$

The  $\mathbf{H}(\bullet)$  is the Hessian. Also, the derivative of equation 2.14 (divided by  $c_0$ ) w.r.t.  $s$  is:

$$\frac{d}{ds} \left( \frac{1}{c_0} \frac{dr}{ds} \right) = \frac{d}{ds} (\nabla \tau) \quad (2.16a)$$

$$\frac{d}{ds} \left( \frac{1}{c_0} \frac{dr}{ds} \right) = \mathbf{H}(\tau) \frac{dr}{ds} \quad (2.16b)$$

Using equation 2.14 and 2.15c in the r.h.s. of equation 2.16b:

$$\frac{d}{ds} \left( \frac{1}{c_0} \frac{dr}{ds} \right) = \mathbf{H}(\tau) (c_0 \nabla \tau) \quad (2.17a)$$

$$\frac{d}{ds} \left( \frac{1}{c_0} \frac{dr}{ds} \right) = \nabla \left( \frac{1}{c_0} \right) \quad (2.17b)$$

The equation 2.17b can now be integrated to give the ray path. Considering the important case of constant sound velocity  $c_0$  (as assumed elsewhere in this work):

$$\nabla \left( \frac{1}{c_0} \right) = 0$$

So, equation 2.17b (with equation 2.14) can be easily solved to:

$$r(s) = r_0 + (c_0 \nabla \tau_0) s \quad (2.18)$$

Where  $r_0$  is the ray origin,  $\tau_0$  is the gradient of  $\tau$  anywhere along the ray, and the addition is taken with  $\mathbb{R}^3$  as an affine space. This is the equation of a line, show that acoustic rays travel as a straight line on constant velocity mediums for a high-frequency approximation.

Further considerations on reflection and refraction may rely on the *Snell law* limilar to the electromagnetic case, because the eikonal equation is equivalent to the *Fermat's principle of least time*. If the first variation of the time functional is zero:

$$\delta T[C] = \delta \int \frac{1}{c_0(\mathbf{x})} \|\mathbf{x}'\| ds = 0$$

Here  $\mathbf{x}(s)$  is a parametrization of the path  $C$ ,  $\mathbf{x}'(s)$  the tangent vector at  $s$  and the whole integral  $T[\bullet]$  is the time to follow the path  $C$ . The *Euler-Lagrange equations* give:

$$\nabla \left( \frac{1}{c_0} \|\mathbf{x}'\| \right) - \frac{d}{ds} \nabla' \left( \frac{1}{c_0} \|\mathbf{x}'\| \right) = 0$$

Where  $\nabla'$  is the gradient taken on the tangent space containing  $\mathbf{x}'$ . Which become:

$$\|\mathbf{x}'\| \nabla \left( \frac{1}{c_0} \right) = \frac{d}{ds} \left( \frac{1}{c_0} \frac{\mathbf{x}'}{\|\mathbf{x}'\|} \right)$$

By considering a length parametrization  $r$  such that  $\|r'\| = 1$ , the result is the same as equation 2.17b, so all the results follow.

It is important to highlight that the single simplification from the wave equation was the high-frequency assumption [26], given at equation 2.12. A more complete description of the mathematics involving the physics behind ray theory can be found at FILIPPI *et al.* [8].

A important result, commonly applied on algorithms derived from ray theory, is the summation of energy/intensity of incoherent rays at a point. It starts by taking the norm of the intensity (equation 2.4) and applying the pressure defition to it (equation 2.8) to get<sup>4</sup>:

$$I = g_0 \mathbb{E}[|\psi|^2] \quad (2.19)$$

Where  $g_0 = \frac{1}{\rho c_0}$  (the inverse of the characteristic impedance),  $I$  the intensity and  $\phi$  the wave. For a single ray (equation 2.10) it is:

$$I = g_0 \mathbb{E}[|\mathcal{A}|^2]$$

To consider the contribution of a set of incoherent rays  $\psi_i$  (i.e. statistically uncorrelated), the total  $\psi$  becomes:

$$\psi = \sum_i \psi_i \quad (2.20)$$

With the stablshed fact that the expected value of a wave is zero (i.e. the mean pressure variation is zero), the incorrelation condition becomes<sup>5</sup>:

$$\text{cov}(\psi_i, \psi_j) = \mathbb{E}[\psi_i \psi_j^*] = 0 \quad (2.21)$$

The resulting equation of the intensity (equation 2.19) for a full wave (equation 2.20) is:

---

<sup>4</sup>Here  $\mathbb{E}[\bullet]$  means the expectation for clarity.

<sup>5</sup>Here  $\psi_i^*$  is the complex conjugate of  $\psi_i$  for better reading.

$$\begin{aligned}
I &= g_0 \mathbb{E} \left[ \left| \sum_i \psi_i \right|^2 \right] = g_0 \mathbb{E} \left[ \left( \sum_i \psi_i \right) \left( \sum_j \psi_j \right)^* \right] \\
&= g_0 \mathbb{E} \left[ \sum_i |\psi_i|^2 + \sum_{i \neq j} \psi_i \psi_j^* \right] \\
&= g_0 \mathbb{E} \left[ \sum_i |\psi_i|^2 \right] + g_0 \mathbb{E} \left[ \sum_{i \neq j} \psi_i \psi_j^* \right] \\
&= g_0 \sum_i \mathbb{E} [|\psi_i|^2] + g_0 \sum_{i \neq j} \mathbb{E} [\psi_i \psi_j^*] \\
&= \sum_i g_0 \mathbb{E} [|\psi_i|^2]
\end{aligned}$$

Which implies:

$$I = \sum_i I_i \quad (2.22)$$

This assumption of incoherent rays will be a common point among ray tracing algorithm, mainly because it implies the summation of ray intensities  $I_i$  to a single intensity. It is a reasonable assumption on mildly complex environments with diffuse (scattering) surfaces, which is explained on Ray Theory subsection [2.2.2](#).

## Normal modes

When there is a preferred direction, or a general axis symmetric medium, like the ocean (which is generally treated as a horizontally stratified medium), a cylindrical parametrization define a usefull coordinate system. The principal axis, usually depth, contain the superposition of normal modes while the other dimensions carry the traveling wave.

There are slightly different mathematical approaches to Normal Modes Theory on the literature [[5](#), [22](#), [24](#)]. The following brief description rely on the techniques of ETTER [[5](#)]. Writing the solution of equation [2.9](#) as a product of a “depth” (principal axis) function  $N(z)$  and a horizontal range function  $H(r)$ :

$$\psi = N(z)H(r)$$

Leads to a separation of variables on the ODE, with  $k_0$  as separation constant:

$$\frac{d}{dz} N + (k^2 - k_0^2) N = 0 \quad (2.23a)$$

$$\frac{d^2}{dr^2} H + \frac{1}{r} \frac{d}{dr} H + k_0^2 H = 0 \quad (2.23b)$$

The equation 2.23a describes the pressure field along the depth, it is known as the normal mode equation. Equation 2.23b, on the other hand, solely describes the traveling portion of the wave, that happens in the horizontal direction. The full solution for  $\psi$  is found by solving both equations. The equation 2.23a is a classic eigenvalue problem for the differential operator, whose solutions, including boundaries conditions, are known as Green's function  $G$  [27]. The horizontal equation 2.23b is a zero-order Bessel equation, so it may have its solution written in terms of the zero order Henkel function of the first kind ( $H_0^{(1)}$ )<sup>6</sup>. Putting these solutions together, for a monochromatic point source:

$$\psi(z, r) = \int_{-\infty}^{\infty} G(z, z_s; k_0) H_0^{(1)}(k_0 r) k_0 dk_0 \quad (2.24)$$

Where  $z_s$  is the source position on the principal axis. The evaluation of the integral as it appears is impractical, thus some simplifications are required.

The Green's function  $G$  can be expanded as a bilinear summation of ortonormal functions ( $u_n$ ), the *normal modes* [29], weighted by using their respective eigenvalues ( $k_n$ ), the natural frequencies. The real integral of equation 2.24 can be evaluated throught *contour integration* to exploit the residue of the poles present on the natural frequencies, but that comes with a price, it is necessary to chose a riemann sheet for the integral and evaluate the branch line integral separately [20, 30]:

$$\psi = \oint \sum_n \frac{u_n(z) \overline{u_n(z_s)}}{k^2 - k_n^2} H_0^{(1)}(k_0 r) k_0 dk_0 + I_{\text{branch-cut}} \quad (2.25)$$

The branch line integral  $I_{\text{branch-cut}}$  can be physically interpreted as the contrubuition of the constinuous mode spectrum (in contrast with the normal modes), representing modes that propagates through the ocean floor (being strongly attenuated) and near-field, that decays exponentially with distance. Assuming that the horizontal distance is several times the water depth, the branch line term is ignored in most theoretical developments.

Therefore, by assuming a far-field approximation ( $k_0 r \ll 1$ ), besides disregard the branch line integral, it is possible to consider a asymptotic expantions for the Hankel fuction and evaluate the contour integral, obtaining [5]:

$$\psi = g(r) \sum_n \frac{u_n(z) \overline{u_n(z_s)}}{\sqrt{k_n}} e^{i(k_n r - \frac{\pi}{4}) - \delta_n r} \quad (2.26)$$

Here  $g(r)$  is a general function of the range and  $\delta_n$  the attenuation coefficient. In practice, the summation ranges over only a bounded number of modes, but this

---

<sup>6</sup> $H_0^{(1)}(x) = J_0(x) + iY(x)$ , where  $J_0$  and  $Y_0$  are the zero-order Bessel functions of first and second order, respectively. More details on these functions, see ABRAMOWITZ and STEGUN [28]

number increases with frequency, which leaves it as undesirable for high-frequency waves.

One of the advantages of normal modes over ray theory lies on the fact that, for each source/receiver position, the ray approach have to run a full simulation of the rays, while the normal modes have a closed form (like equation 2.26) that easily adapts to new combinations of these parameters. In constrast, normal modes are constrained by the source frequency, in fact by the number of modes to be computed, and generally requires deeper knowledge of the environment. The the attenuation coefficient( $\delta_n$ ) in the ocean, for exemple, depends on water absorption, ocean sediment layer absorptions, compressional and shear attenuation on the basement, measures of the modes interactions with both sediment and basement(compressional and shear mechanisms) and statistics on the water-sediment boundary and sea surface as well.

### Parabolic approximation

The parabolic approximation replaces the Helmholtz equation 2.9, which is a elliptic partial differential equation, by a simplified parabolic version. This approximation dates back to the middle of the XXI century, in the context of tropospheric radio wave propagation.

The normal mode solution do not handle well non-stratified medium, to overcome this limitation parabolic approximation is constructed to handle slow varying mediums. The variation is made explicit by  $k = k_0 n(r, z)$ , where  $n(r, z)$  is a sound refractive index[9]. The Helmholtz equation becomes:

$$(\nabla^2 - k_0^2 n^2(r, z))\psi = 0 \quad (2.27)$$

The solutions are assumed to have a fast varying horizontal field  $H_0^{(1)}(k_0 r)$  and overall slow fluctuations  $F(r, z)$ :

$$\psi(r, z) = F(r, z) H_0^{(1)}(k_0 r)$$

Assuming asymptotic behavior to the Hankel function (similar to normal modes development) and applying it to equation 2.27:

$$\frac{d^2}{dr^2} F + 2ik_0 \frac{d}{dr} F + \frac{d^2}{dz^2} F + k_0^2 (n^2 - 1) F = 0 \quad (2.28)$$

By the assumption that  $F(r, z)$  is a slow varying field, the  $\frac{d^2}{dr^2} F$  term is neglected and equation 2.28 gives rise to the parabolic equation. LURTON [9] proceeds further by discretizing the horizontal directions, without defining explicit dependence on the Hankel function, and using an approximation for the differentials on  $z$  through



pseudo-differential operator.

The parabolic equation can then be solved numerically, but the grid size vary with frequency, becoming costly for high-frequency waves. For more details, see JENSEN *et al.* [20].

## 2.2.2 Ray Theory

Ray Theory provides a good way to treat high-frequency sound propagation, far from caustics, that retain an intuitive meaning. Introduced in subsection 2.2.1, it is very similar to its electromagnetic counterpart. Each ray carries a energy density from the source that decays as it travels through the medium (Transmission loss), as given by the equation 2.11b, when it encounters an obstacle (eg. sea floor, sea surface, man made surfaces) it (back)scatter to the source. That is the classical (non-multipath) description [1, 5, 9, 31], summarized by a sonar equation as:

$$RL = SL - DI - TL + TS \quad (2.29)$$

Each of these acronyms corresponds to energy (intensity), or energy variation, written in decibels<sup>2.5</sup>:

1. RL is the Received Level -  $10 \log_{10}([Received\ intensity]/[reference\ intensity]^7)$
2. SL is the Source Level -  $10 \log_{10}([Intensity\ at\ 1m]^8/[reference\ intensity])$
3. DI is the Directivity Index -  $10 \log_{10}([Directional\ loss]^9)$
4. TL is the Transmission loss - Intensity loss through absorption and spreading while propagating (in dB).
5. TS is the Target Strength - ratio between the intensity of outgoing and incoming rays (in dB).

Equation (2.29) does not fully describes how the simulation using ray theory should work, but gives a good insight on the elements that must be considered. SL can be inferred by the sonar power and efficiency, DI comes from the sonar beam pattern, TL is dependent on the medium, but is compensated by the sonar TVG (see subsection 2.1.2) and TS is defined by a material dependent BRDF (Bidirectional Reflectance Distribution Function), described further ahead.

---

<sup>7</sup>Intensity of plane wave with pressure amplitude of 1  $\mu$ Pa

<sup>8</sup>for an equivalent omnidirectional source

<sup>9</sup>with respect to an omnidirectional source

## Ray Interaction

For a more faithful sonar response, advanced simulation techniques based on ray theory can be applied. And, thus, requires careful consideration on the most important interactions of the rays with the interfaces, namely: **transmitted rays**, **reflected rays** and **scattered rays**, see figure 2.7.

### Transmitted Rays

When a radiant energy, in this setup represented by a ray, hits a point on a surface (or a general interface) part of the energy is absorbed, part is transmitted and, yet, another part bounces back. Transmitted rays represent the energy that goes through the interface, their intensity for each direction is mathematically defined by the BTDF (Transmitted Scatter Distribution Function)[32] (see RÖBER *et al.* [33] for an acoustic approach).

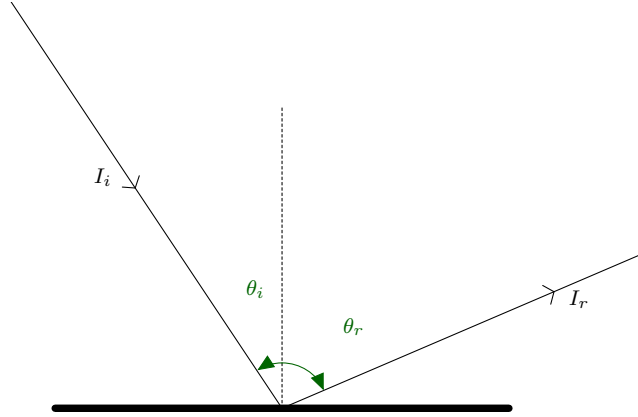


Figure 2.5: Incident and reflected rays with respective angles.

### Scattered and Reflected rays

The treatment for reflected and scattered rays starts together by means of the BRDF (Bidirectional Reflectance Distribution Function)[1, 34, 35], which works similar to BTDF. On perfect smooth surfaces, its BRDF exhibits a single direction, the specular reflection, where the ray intensity is non-zero at the same plane of incident ray and the same angle with the surface normal vector. This kind of reflection is responsible for mirror like effects, e.g. sound that hits the still water surface from within[5, 9].

The other extreme case is the perfect diffuse reflection (usually called scatter in acoustics), where a rough surface reflects same perceived energy (radiance) in all

directions, thus following a cosine law (Lambert's Cosine Law) with respect to the incident and reflected(scattered) angles [36, 37]:

$$I_r \propto I_i \cos(\theta_i) \cos(\theta_r) \quad (2.30)$$

Where  $I_i$ ,  $I_r$  and  $\theta_i$ ,  $\theta_r$  are the intensity and angle with the surface normal for the incident and reflected rays, respectively (figure 2.5). The proportionality constant is material dependent and should not exceed  $1/\pi$ , because that would violate energy conservation.

If incoming and reflected rays have the same direction, the angle is the same and the situation is called a *backscatter*:

$$I_r \propto I_i \cos^2(\theta_i)$$

There is some confusion in the literature about the cosine being or not squared[5, 9, 35, 37–39]. Possible explanations are mixing intensity information with pressure or related quantities, as intensity and radiance. This cosine law, for a single scatter, describes the Target Strength, in dB:

$$TS = A + 20 \log_{10}(\cos(\theta_i))$$

Here  $A$  is the equivalent of the proportionality constant. Under the assumption of no energy loss through transmission or absorption,  $A = -10 \log_{10}(\pi) \text{dB} \approx -5 \text{dB}$ . More realistic values vary from  $-17 \text{dB}$  for basalt ridge cliffs to  $-27 \text{dB}$  for sediment pond[40]. Transmission and reflection between multiple sediment layers at the bottom of the ocean can be treated as a single entity, subbottom scattering, that introduces a delay and a displacement of the reflection[5], that case will not be covered by the simulation procedure proposed in this thesis. A remarkable property of the scattered rays is the incoherence which makes possible to add the energy (intensity) contributions for each ray directly, according to equation 2.22.

Rough ocean surface, in the presence of wind, also backscatters, but not as a perfect scatter. Its scattering properties have been modeled in a variety of ways[39], like Kirchhoff model[40]:

$$TS = -10 \Gamma^2 \log_{10}(e) \approx -4.34 \Gamma^2$$

Where  $\Gamma = 2kh \cos(\theta_i)$ ,  $k$  is the acoustic wavenumber (equation 2.9),  $h$  is the rms height of the surface.

A general surface does not follow neither a perfect cosine law nor is a perfect reflector, but is useful to model as a compromise of both[38, 41], even some standards for acoustic measurement (e.g. ISO) split between these two types of reflection[42].

An absorption/transmission factor  $\alpha$  defines the fraction of the energy that does not reflect back, and a scattering factor  $\delta$  the fraction of the reflected energy that is scattering (figure 2.6).

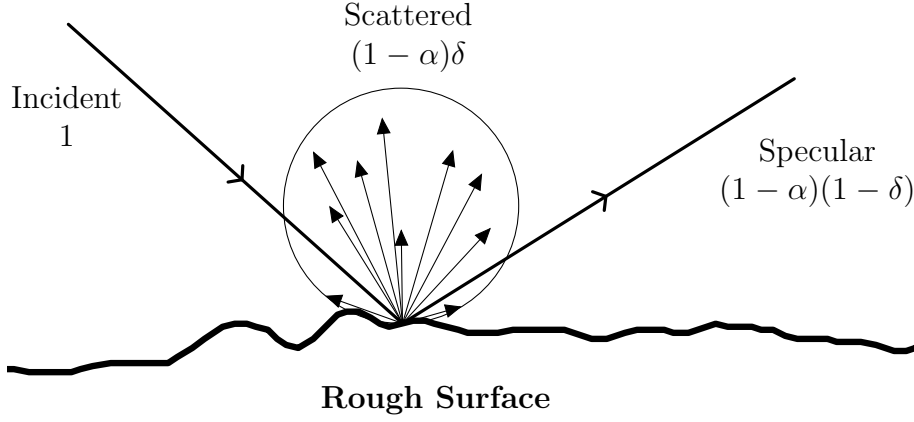


Figure 2.6: Scattering reflection weighted.

This weighting concept is explored to create a full BRDF function as  $\rho(\vec{i}, \vec{r}) = (1 - \alpha)(\delta\rho_{\text{scat}}(\vec{i}, \vec{r}) + (1 - \delta)\rho_{\text{spec}}(\vec{i}, \vec{r}))$ . For a numerical treatment, one option is to discretize the solid angles as SILTANEN *et al.* [43] describes, another is to model as computer graphics, where, instead of a single direction for the specular reflection, the specular reflection is a smooth function of the reflected direction with a peak at the actual specular direction. This smooth function creates a better transition from specular to scatter reflection. A standard model is Phong reflection[44]:

$$I_r \propto I_i \cos(\theta_s)^\nu \quad (2.31)$$

Here  $\theta_s$  is the angle between the reflected direction and the specular direction. And  $\nu$  a shininess constant describing how concentrated the specular reflection is. In the limit of  $\nu \rightarrow \infty$  it becomes a perfect mirror.

## Ray Tracing

Ray Tracing was originally an algorithm that used ray theory (described earlier in section 2.2.1) for computer graphics with a rationale very similar to the sonar equation 2.29. It traced a ray to every point on the scene from the source and the receiver, then computed the intensity of each color, ignoring multiple reflections.

A variety of derivations from the original algorithm have been conceived. Most of the methods applied to audio were developed for computer graphics rederization, but, considering sound waves with similar constraints to those applied to light (i.e. high frequency limit), it is reasonable to apply them to aurelization (sound renderization) as well.

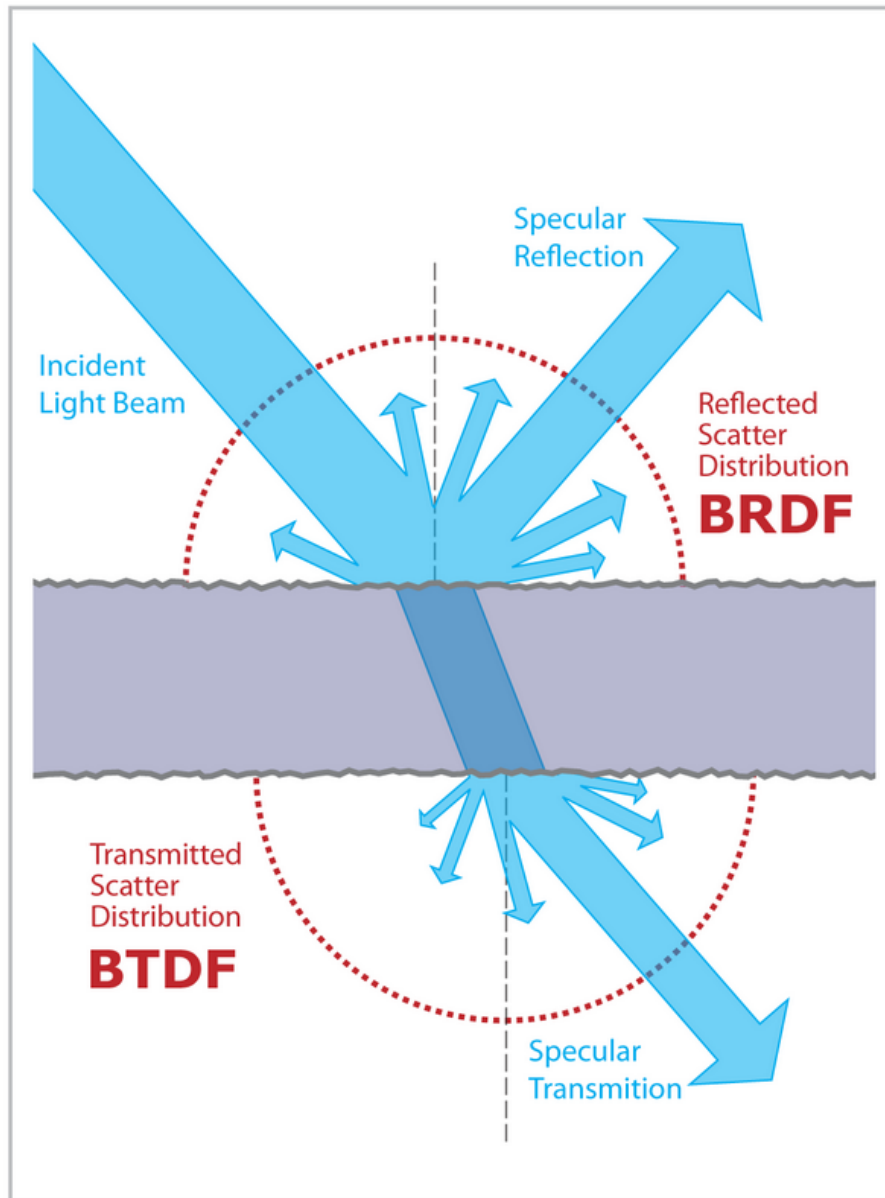


Figure 2.7: BRDF and BTDF for transmitted, reflected and scattered rays.  
 (©User:Jurohi / Wikimedia Commons / CC-BY-SA-3.0)

Among the most common techniques are **Image Source**, **Beam Tracing**, **Metropolis Transport** and **Path Tracing**. Their standard implementation assume homogeneous medium, effectively propagating rays as straight lines. An important alternative, which also overcomes sharp shadows and caustics issues, is *Gaussian Beam*, it evaluates the acoustic field at every point by adding contributions of each ray. The name comes from the fact that the weight of a ray decays in the paraxial direction as a gaussian. It is typically more computationally more costly than other ray tracing, but has applications for inhomogeneous open environments as described by TRACER *et al.* [45].

## Image Source

This method focuses on environments composed by segments on the 2D case or polygonal slices of planes on the 3D case. It works best on box-like environments, for this creates a perfect tiling of space [21] (see figure 2.9).

For each reflective surface, a virtual source is created on the reflected position of the source, these are the primary virtual sources. A virtual source will act just like a source, although its intensity is lessened by the absorption factor of the wall. When computing the sound at point, the contribution of each source (including virtual ones) are added together, the reflections are automatically taken into account by the virtual sources.

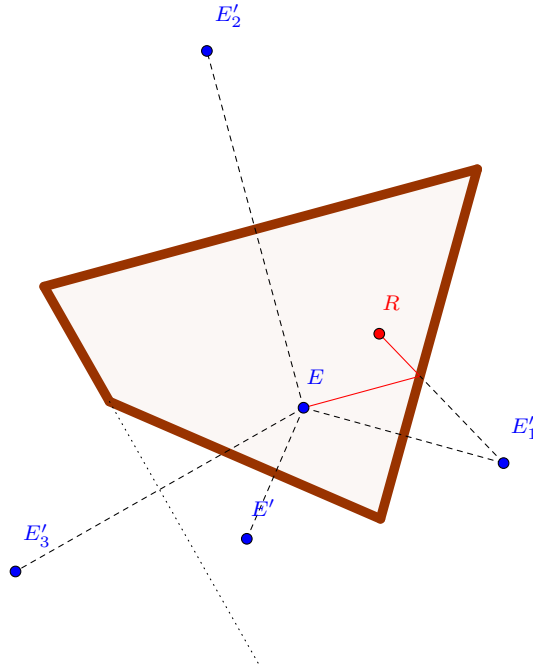


Figure 2.8: The source  $E$  and the virtual sources  $E', E'_1, E'_2, E'_3$ . In this case,  $E'_3$  is not a visible virtual source.

The procedure of creating the virtual sources may be repeated many times for

each new virtual sources, thus more reflection are effectively calculated. The visibility of a virtual source must be validated before adding its contribution.

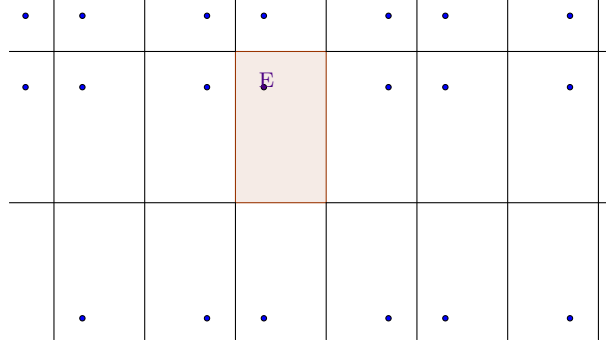


Figure 2.9: Source on a perfect tiling. The simple pattern facilitates computation.

The method does not directly implement the idea of sound scattering, but there are some possible extensions[46]. In practice, for more complex environments, only a few early reflections are generated. The number  $O(n^r)$  of sources, where  $n$  is the number of surface planes and  $r$  the number of reflections, that must be created and validated grows exponentially fast.

## Beam Tracing

Beam Tracing uses pyramidal beams, a collection of rays, instead of single infinitesimal rays to calculate reflections [21]. The solid angle on the source that describes available sound directions can be subdivided in pyramidal beams without overlap, avoiding sampling issues. These pyramidal beams exploit the spatial relation between nearby rays, being able to consider the propagation path without skipping any ray.

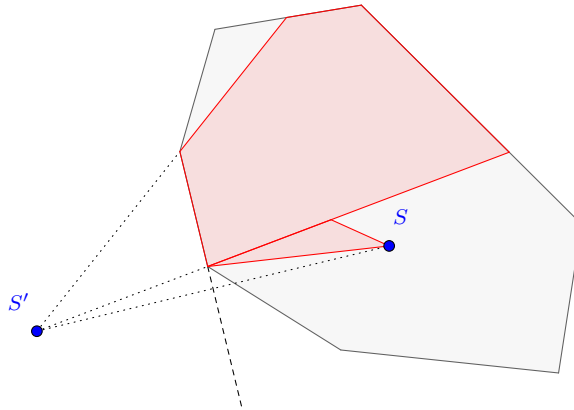


Figure 2.10: A 2D view of a pyramidal beam tracing.

The algorithm starts by subdividing the source sound directions into pyramidal regions. For each pyramidal beam, intersections with environment walls are calculated

from the first encounter to the last, clipping the beam so no shadowed regions are considered. The resulting polygons of each intersection act as a new virtual source, constructed by mirroring the source as in the Image Source method. Beam Tracing, then, progresses by repeating this procedure on each virtual source, until the desired precision (number of reflections) is achieved.

It does not suffer from visibility computation issues as Image Source, but is less efficient for highly structured box-like environments. Despite working well on simple scenes (with or without occlusions), it still not handling well curved surfaces.

## Path Tracing

Path Tracing is a recursive solution to the wave equation simplification used in ray theory. That can be written, in terms of the radiance  $\ell$  (the ray energy - acoustic energy per unit of surface area per unit of solid angle), as an integral equation, the *room acoustics rendering equation* [43]:

$$\ell(\mathbf{p}, \Omega) = \ell_0(\mathbf{p}, \Omega) + \int_{\mathcal{G}} R(\mathbf{x}, \mathbf{p}, \Omega) \ell(\mathbf{x}, \vec{\mathbf{x}\mathbf{p}}) d\mu(\mathbf{x}) \quad (2.32)$$

Where  $\mathcal{G}$  is the two dimensional subset of  $\mathbb{R}^3$  comprising all surfaces,  $\mathbf{p}$  is a point over a surface,  $d\mu(\mathbf{x})$  is the surface area differential at point  $\mathbf{x}$ ,  $\Omega$  is the outpoint energy direction,  $\ell_0$  the intrinsic surface radiance (a source point),  $\vec{\mathbf{x}\mathbf{p}}$  the unit vector in the direction to  $\mathbf{p}$  from point  $\mathbf{x}$ . The function  $R(\bullet, \bullet, \bullet)$  is the *reflection kernel*, an enhanced version of a BDRF. For a non-obstructed path between  $\mathbf{p}$  and  $\mathbf{x}$ :

$$R(\mathbf{x}, \mathbf{p}, \Omega) = \mathbf{a}(\|\mathbf{p} - \mathbf{x}\|) \rho_{\mathbf{p}}(\vec{\mathbf{x}\mathbf{p}}, \Omega) \cos(\theta_i) \cos(\theta_r) \quad (2.33)$$

When there is no visibility from  $\mathbf{x}$  to  $\mathbf{p}$ ,  $R(\mathbf{p}, \mathbf{x}, \Omega) = 0$ . Here  $\rho_{\mathbf{p}}(\vec{\mathbf{x}\mathbf{p}}, \Omega)$  is the BDRF at the point  $\mathbf{p}$  with incoming direction  $\vec{\mathbf{x}\mathbf{p}}$  and outgoing direction  $\Omega$ ,  $\theta_i$  and  $\theta_r$  are the incoming and reflected angles with surface's normal at point  $\mathbf{p}$  for directions  $\vec{\mathbf{x}\mathbf{p}}$  and  $\Omega$  respectively. Factor  $\mathbf{a}(\|\mathbf{p} - \mathbf{x}\|)$  is the decay dependent on the distance  $\|\mathbf{p} - \mathbf{x}\|$  caused by spreading and absorption, usually a inverse squared times a exponential factor (commented on subsection 2.1.2).

The solution to equation 2.32 is a Neumann series:

$$\ell_{n+1}(\mathbf{p}, \Omega) = \int_{\mathcal{G}} R(\mathbf{x}, \mathbf{p}, \Omega) \ell_n(\mathbf{x}, \vec{\mathbf{x}\mathbf{p}}) d\mu(\mathbf{x}) \quad (2.34a)$$

$$\ell(\mathbf{p}, \Omega) = \sum_{n=0}^{\infty} \ell_n(\mathbf{p}, \Omega) \quad (2.34b)$$

For the actual computation, the summation on equation 2.34b is truncated at some reasonable value of  $n$ , which is the number of reflections being considered, and



the integral on equation 2.34a can be approximated using, for exemple, a Monte Carlo method applying importance sampling w.r.t. the function  $R(\bullet, \bullet, \bullet)$  (MUNJAL *et al.* [47] describes briefly such an algorithm). When scattering is not strong enough, it is a commom practice to consider only specular directions.

Besides the success of computer graphics using the same technique, in the end of the last century there were still some concerns regarding the theoretical validity, manly because it was shown to be uncomputable [48]. That is, it was impossible to say if a ray would ever reach a certain point. Later, this decade, it was shown by BLAKEY [49] that, if one considers computational finite precision, it can be proven to be computable.

In the case of sonar, the only important point to measure the sound intensity is on the sonar itself. A simplified approach for the integral on equation 2.34a is to consider only the specular direction (where it is supposed to have greater radiance) and the scattered direction into the sonar, assuming the other scatterd directions will die out without much affecting the sonar response.

## Metropolis Transport

Metropolis Light Transport (MLT) is a incremental development over Path Tracing. It records the path taken by a ray as a tree where each reflection point is a node. By adding, removing or changing nodes, it is able to better explore the space without ruling out the work done to produce a path from source to detector.

The name comes from the use of Metropolis sampling method to explore the space, which degenerate paths that are small variations of the original one. A change in the path may be accepted or regected (as in the original Metropolis method) and the decision strategy ensures a ergotic, unbiased and generally applicable result.

At the cost of added complexity, MLT gives better results specially around caustics and difficult to reach reagions (e.g. through a narrow apperture) without degrading performance. The original paper by VEACH and GUIBAS [50] explains with great clarity the math behind as well as the results and interpretations.

## 2.3 Environment

Extensive literature have been written on ocean environment, from modeling its behaviour to measuring its properties. Different simulation techniques have also been explored[5]. The modeling presented here, althouth simple, completely suits the needs of a ray tracing technique, presented on section 2.2.2.

### 2.3.1 Modeling

Borrowed from computer graphics, the modeling properties of a scene objectes are the same for light and sound (given the high frequency limit for which ray theory is applicable). Two distinct factors are modeled, one is geometric, which defines the shape of the object, the other is acoustic, expressing how does it interact with sound.

For the geometric part, two basic functions have to be provided: **intersection** and **normal**. **Intersection** takes a ray, defined by a origin point and a direction, and outputs the distance to the first intersection point with the object. If no intersection point is found, the distance is defined to be infinity. **Normal** receives a point on the surface of the object and return the normal vector at such a point. Algorithm 1 exemplifies the **intersection** for a rectangle, the rays origin **O** and direction **D** are matrix with the concatenated information of all those whose intersection ought to be calculated.

---

#### Algorithm 1 Intersection for Rectangle

---

```

function INTERSECTION(O, D)                                ▷ O is ray origin, D is direction
     $\Delta \leftarrow center - \mathbf{O}$                                 ▷ center is the rectangle center
     $n \leftarrow \vec{s_1} \times \vec{s_1} / \|\vec{s_1} \times \vec{s_1}\|$           ▷  $\vec{s_1}$  and  $\vec{s_1}$  are the rectangle's half-sides
     $T \leftarrow [\vec{s_0} \ \vec{s_1}]^\dagger$                         ▷  $\dagger$  is pseudoinverse
     $d \leftarrow \Delta \cdot n$                                     ▷ distance to intersection point
     $P \leftarrow \mathbf{D}d - \Delta$                                 ▷  $P$  are the intersection with the rectangle's plane
     $R \leftarrow T \cdot P$                                       ▷  $R$  are the intersection described on  $[\vec{s_0} \ \vec{s_1}]$  basis
    for all  $i \in [0, \dots, \text{size}(d)]$  do
        if  $d_i < 0$  or  $|R_{i,0}| > 1$  or  $|R_{i,1}| > 1$  then
             $d_i \leftarrow \infty$                                 ▷ Check ray direction and if hit within rectangle
        end if
    end for
    return  $d$ 
end function

```

---

Any surface can be approximated by triagulation and have these functions more easily defined, but it is interesting to directly define for some geometric primitives. Plane, rectangle, sphere and cylinder were developed for this work.

Two environments were constructed using these four geometric primitives. One box-like for the reconstruction part of this thesis, which is simple enough to study the properties of the mapping. Another, more complex and inspired on a water entrance of a hydroelectric powerplant, that exhibits a richer sonar response with sound multipath and directional gain playing a more important role.

## Box-like Environment

For the **box-like structure**, five planes were used thus determining a semi-infinite box with 8 meters width, 10 meters length and the bottom 3 meters from the origin. All planes are defined by a point and its normal vector.

Plane n <sup>o</sup>	Point	Normal Vector
0	(0, 4, 0)	(0, -1, 0)
1	(0, -4, 0)	(0, 1, 0)
2	(5, 0, 0)	(-1, 0, 0)
3	(-5, 0, 0)	(1, 0, 0)
4	(0, 0, -3)	(0, 0, 1)

Table 2.1: Five planes defining box-like environment walls.

## Complex Environment

The more **complex scene** is composed of 5 rectangles, 2 planes and a sphere representing, respectively, 5 concrete walls, river floor and still surface water and a half-spherical mountain of sediments. Rectangles are defined by a central point and two perpendicular vectors, the half sides, and spheres by a center and radius.

### 2.3.2 Characterization

Instead of defining a full BRDF (explained on section 2.2.2), three parameters are considered: **diffusion coefficient**, **specular coefficient** and **shininess**. All three parameters may change at every point on the surface of an object, thus defining a texture, but for the sake of simplicity only constant values over the surface were considered.

The **diffusion coefficient** and **specular coefficient** are, respectively, fractions of incident energy over a surface patch that reflects diffusely (as a lambertian reflector) and specularly. Reflections near specularly are weighed as Phong reflection for a less unrealistically abrupt change in reflection intensity. The **shininess** is the Phong parameter. These concepts are described in section 2.2.2.

The actual values used came from a collection of sources in addition to experimentation and tacit knowledge (from previous sonar use), as these are difficult information to find in the literature.

For concrete, CHAUDHARI [12] studies the **reflection coefficient**, sum of **diffusion coefficient** and **specular coefficient**, to characterize the concrete's quality following earlier measurements of LESLIE and CHEESMAN [51]. The table provided on the article can be used in the other direction, to simulate such a concrete quality. The individual values of **diffusion coefficient** and **specular coefficient**

still have to be determined and come from a educated guess based on considerations of smooth by ETTER [5], which claims that most of the energy goes as specular reflection for smooth surfaces. For simulation purposes it has been considered values between 80% to 95% of the reflected energy to be specular.

Quality Of Concrete	Reflection Coefficient
Very good	0.76 or above
Good	0.69-0.74
Questionable	0.62-0.69
Poor	0.48-0.62
Very Poor	0.48 or less

Table 2.2: Quality of Concrete and Reflection Coefficient. (CHAUDHARI [12], LESLIE and CHEESMAN [51])

ETTER [5], also, provides equations relating wind speed with water surface reflection coefficient, which varies according to its roughness caused by the wind. For a still water there is amost no transmitted energy and all reflected energy is specular. For other materials, estimated values come from models as the one provided by MILLER [1] on figure2.11.

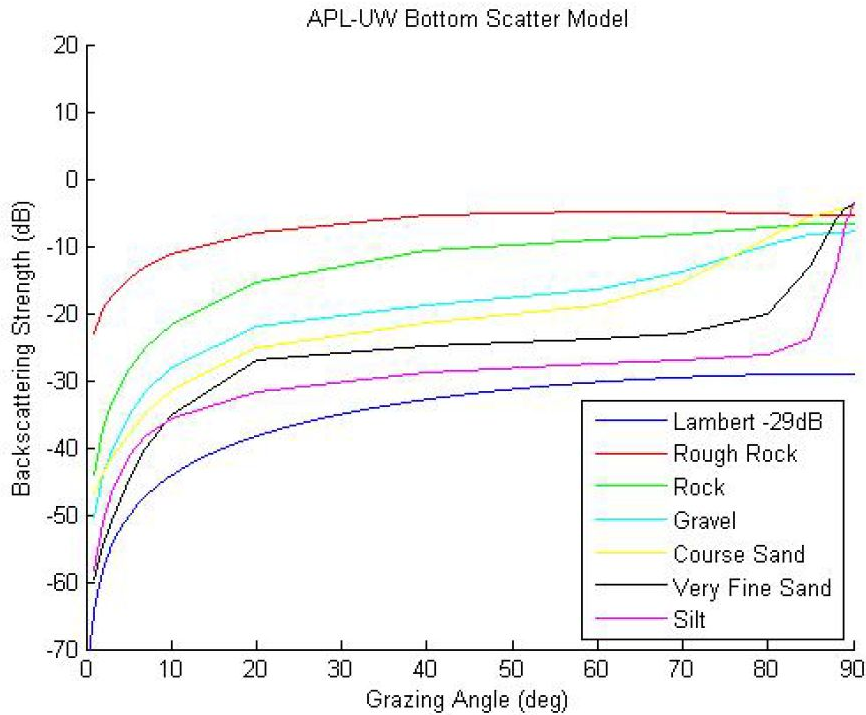


Figure 2.11: Materials reflective characteristics from MILLER [1].

## 2.4 Implementation

### 2.4.1 Algorithm

The simulation follows the ray tracing technique outlined by BELL and LINNETT [52] for side scan sonar, but applies it to a forward looking sonar imaging sonar (section 2.1.3). It also uses a noising adding step as suggested by COIRAS *et al.* [3] with statistics provided by MAUSSANG *et al.* [53]. No movement induced distortion was considered, some approaches to add this feature are available on BELL *et al.* [54], BORAWSKI and FORCZMAŃSKI [55]. Also, spreading and absorption losses are ignored, assuming they are compensated by TVG (see section 2.1.2).

Sonar parameters follow a Tritech's Micron sonar[56] information as output power, dynamic gain, beam step and sensibility were found on official Tritech's documentation[56, 57]. The directional gain was measured by the National Physical Laboratory, UK.

Simulation's output is, just as on the sonar, a sequence of arrays with values between 0 and 255. Each element of the sequence is a bearing, direction of the emitted sound pulse, and the array's components are the bins' values, sound intensity received at some range of distances (calculated from echo delay).

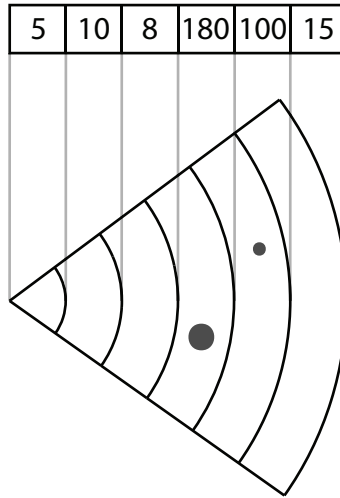


Figure 2.12: Exemple of an array for a bearing direction. Actual arrays are longer, depending on resolution.

The algorithm implementation uses the programming language Python with the mathematical library NumPy, specially for efficient linear algebra. Most of the treatment uses linear algebra to treat batch of rays at once.

Flowchart of figure 2.13 describes the simulator logic. It starts by computing ray directions spread over a sphere centered at the sonar with uniform density, otherwise it would bias the ray trace. Not all directions are actually computed because some directions have very low gain, so rays in these directions have almost no energy,

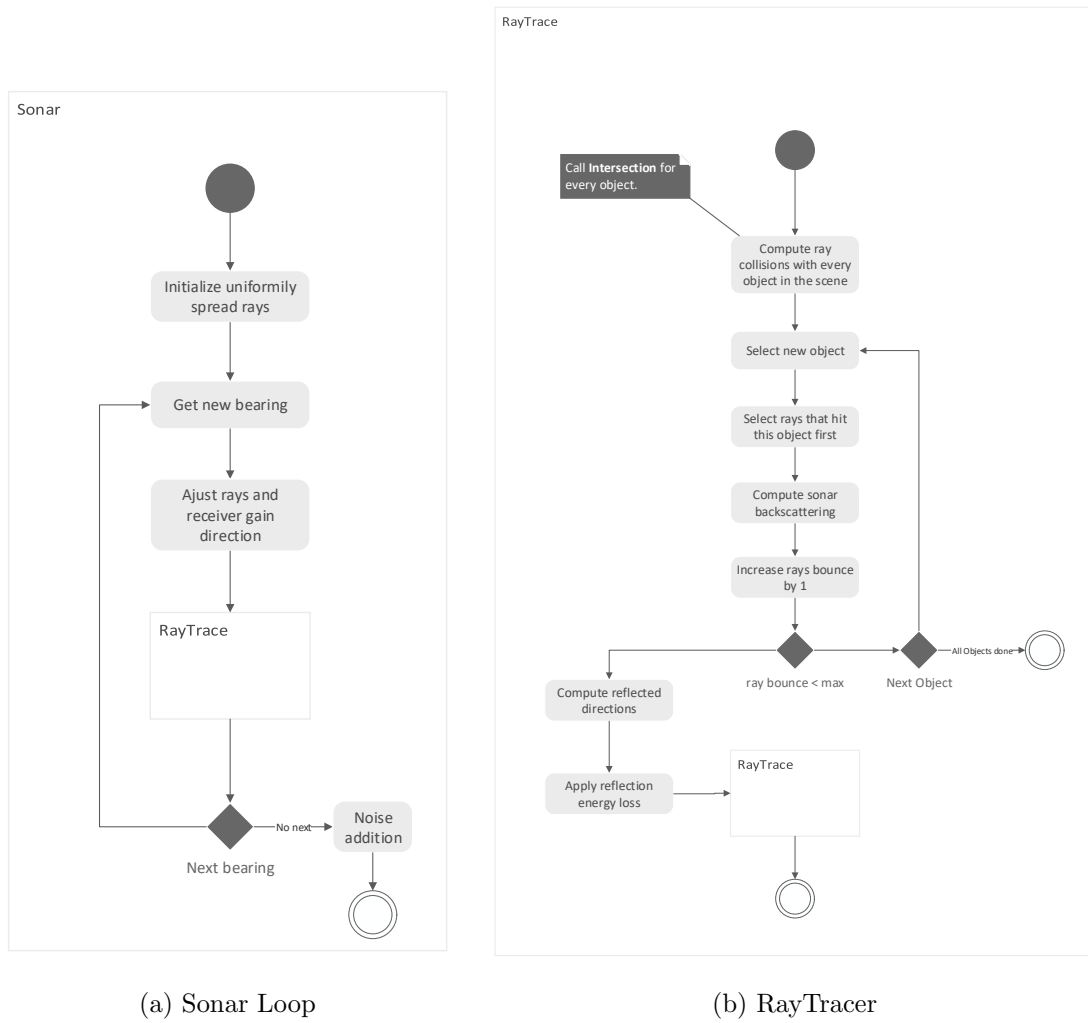


Figure 2.13: Overview of the simulation algorithm.

they can be discarded. To compute such a uniformly distributed directions apply transformation described by algorithm 2 for  $[-\alpha, \alpha]$  and  $[-\beta, \beta]$  the vertical and horizontal angular span, respectively, and  $N$  the desired number of rays. Results on section 2.4.2 use  $\alpha = 30^\circ$  and  $\beta = 3^\circ$ , approximately the values for which Micron cannot detect the echo.

The sonar bearing pace is adjustable and, following Tritech’s Micron configuration, it was set to  $1.8^\circ$ , thus, doing a complete scan on 200 steps. For each step, ray directions are changed (by a rotation) to match new bearing. Received gain is calculated w.r.t. the front direction (bearing), as the bearing changes the gain is automatically updated. Rays always carry 2 informations: its actual intensity (disregarding distance traveled decay) and its total traveled length.

A new bearing position invoke ray tracer algorithm. It begins by calling the **intersection** function (described at section 2.3.1) for each object in the scene, passing all rays as argument. Then it loops again on every object, but now only focus on the

---

**Algorithm 2** Rays Uniform Direction

---

```
procedure UNIFORM DIRECTION( $\alpha, \beta, N$ )  
   $d\theta \leftarrow 2 \cos(\pi/2 - \alpha)$   
   $d\phi \leftarrow 2\beta$   
   $\rho \leftarrow \sqrt{\frac{N}{d\phi \cdot d\theta}}$  ▷ Estimated density  
   $N_\theta \leftarrow \lceil \rho \cdot d\theta \rceil$   
   $N_\phi \leftarrow \lceil \rho \cdot d\phi \rceil$  ▷  $U(\mathbf{x})$  generates  $\mathbf{x}$  uniform samples over  $[0, 1]$   
   $\theta \leftarrow \arccos(d\theta \cdot (2U(N_\theta) - 1))$   
   $\phi \leftarrow d\phi \cdot (2U(N_\phi) - 1)$   
  for all  $(\theta_i, \phi_i) \in \theta \times \phi$  do  
     $x_i \leftarrow \sin(\theta_i) \cos(\phi_i)$   
     $y_i \leftarrow \sin(\theta_i) \sin(\phi_i)$   
     $z_i \leftarrow \cos(\theta_i)$   
     $v_i \leftarrow (x_i, y_i, z_i)$   
  end for  
  return  $v$   
end procedure
```

---

rays that have the object as first hit, and compute the backscattering to the sonar. Backscattering strength calculation use Lambert and Phong scatterings as described on section 2.2.2 and material parameters listed on section 2.3.2. This strength is add to a bin (see figure 2.12) whose position is calculated as half the full distance travelled by the ray (including previous reflections) plus a small gaussian noise. It proceeds to calculate reflection if the number of computed reflection for the ray does not exceed a maximum value (set to 5). Reflection are simple linear transformations that depends on the surface's normal, obtained via **normal** function (section 2.3.1). The algorithm, then, calls itself for the reflected rays, recursively.

After the whole scan is computed, bin values are normalized to  $[0, \dots, 255]$  (again according to Trittech's Micron configuration ). Upon these values, an additional Weibull's distributed noise is applied [53].

## 2.4.2 Results

For both environments discribed on section 2.3.1, several sonar positions were simulated. The absolute position and orientation were chosen, but the bearing w.r.t. the environment was a random value.

Polar plots displayed here is the expected visualization, without noise filtering, when the sonar uses 500 bins of resolution with a 12 meters range. Each polar pixel has a  $3^\circ$  arc length, but, as the bearing step is  $1.8^\circ$ , they overlap while being rendered.

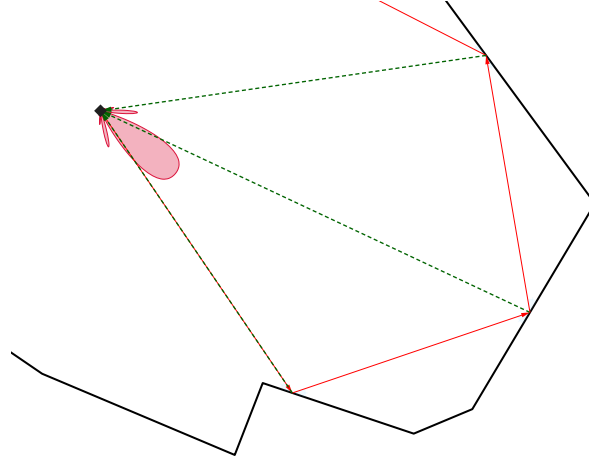


Figure 2.14: Ray Tracing: Red lines are specular reflections, green lines are diffuse backscattering.

### Box-like Environment

The half-infinity box-like structure is depicted on figure 2.15. Axis aligned sonar orientation on figures 2.15c and 2.15d make clear its rectangular cross section, while its half-infinity characteristic is visible on perpendicular oriented scans, figures 2.15a and 2.15b.

### Complex Environment

Figure 2.16 shows the more complex structure from four view points. Images 2.16a and 2.16b are scans from between walls of the indent. Figure 2.16d is cross sectional view of the indent and figure 2.16c is a scan from the same position, but with different orientation, making the hemisphere at the bottom more noticeable.

Both environments reveal interesting features of a sonar scan, but they tend to be more pronounced on the complex environment. Figures 2.16b, 2.16d and 2.15b present clear signals of multipath. Another interesting feature is the halo on the background of figures 2.16a and 2.16b caused by a trade-off between the sonar directional gain and the low backscattering at shallow angles.



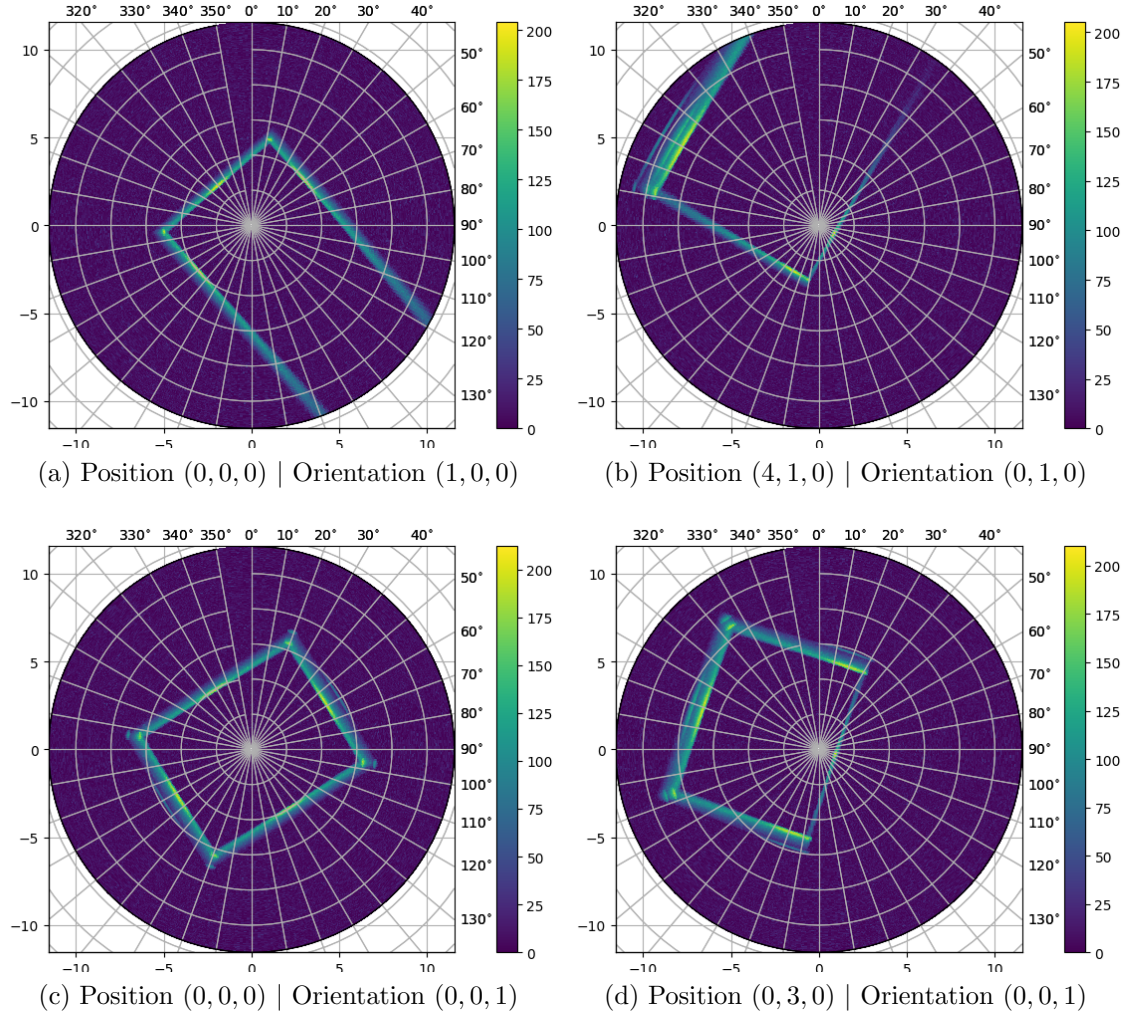


Figure 2.15: Sonar simulation for the box-like scene.

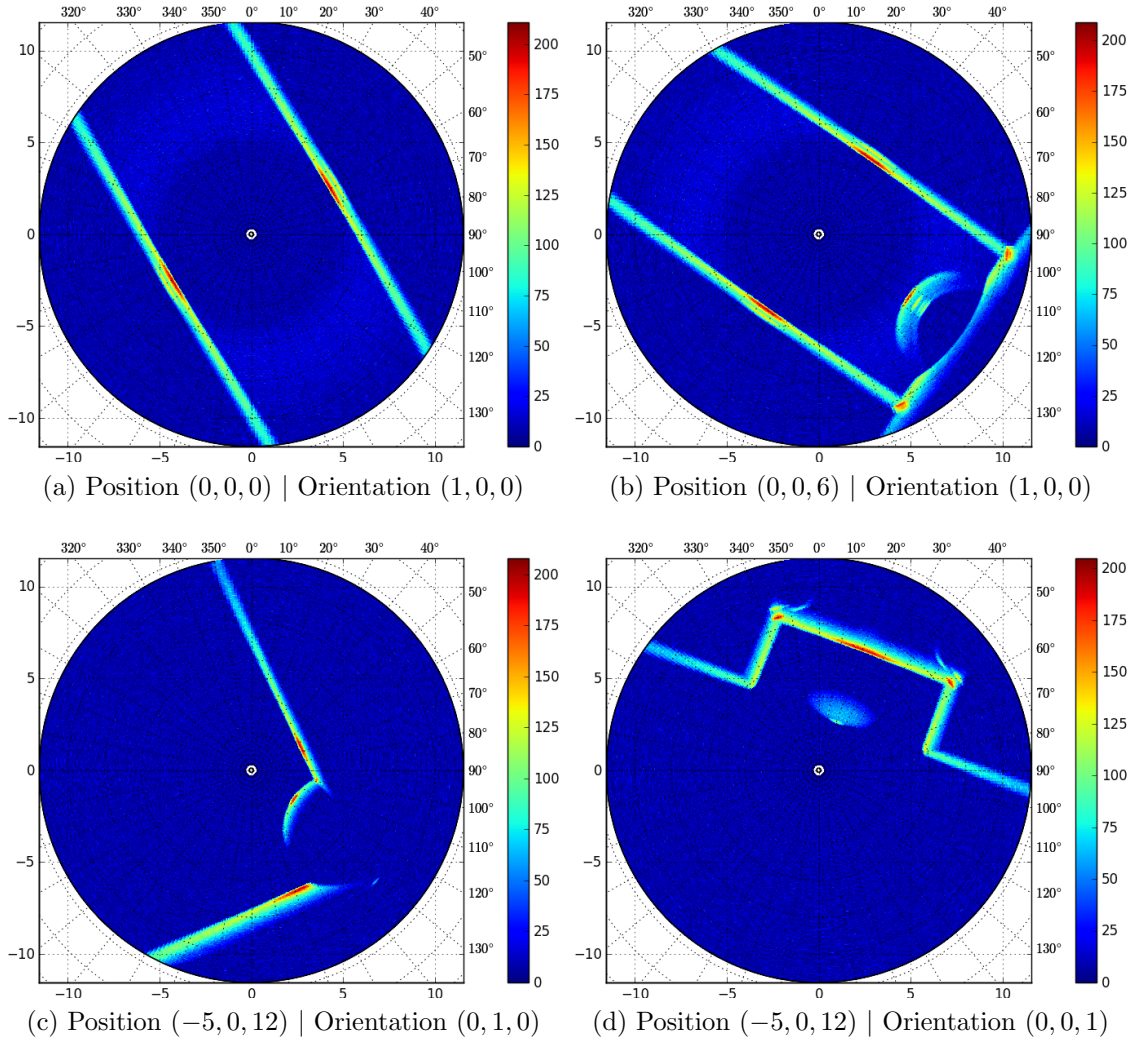


Figure 2.16: Sonar simulation for the complex scene.

# Chapter 3

## Mathematical Preliminaries

”Obvious” is the most dangerous word in mathematics.

---

*Eric Temple Bell, 1938*

Some of the more advanced mathematical tools<sup>1</sup> used on the development of this thesis are presented here. Other knowledges, as basic linear algebra, statistics and analysis, are take as granted.

### 3.1 Hilbert Space

A Hilbert Space is a complete inner product space [58]. It is a complete metric space with respect to the metric induced by its inner product (which in turn can be thought indirectly by it’s induced norm). A nice picture is as a generalization of the Euclidean space. Which means intuition works well, in contrast with the broader concept of Banach Spaces, a complete normed space, where the infinite dimensional case can be quite different from what one would expect<sup>2</sup>.

An inner product space[59] is a (possibly infinite dimensional) vector space  $V$  over  $\mathbb{C}$  (or  $\mathbb{R}$  by restriction ), together with a map (called the **inner product**):

$$\langle \cdot, \cdot \rangle_V : V \times V \rightarrow \mathbb{C}$$

Satisfing the following properties, for all  $x, y, z \in V$  and all  $\mu, \lambda \in \mathbb{C}$ :

I  $\langle x, \lambda y + \mu z \rangle_V = \lambda \langle x, y \rangle_V + \mu \langle x, z \rangle_V$  (linear in the second argument)

II  $\langle x, y \rangle_V = \overline{\langle y, x \rangle_V}$  (Hermitian symmetric)

---

<sup>1</sup>As judged by the autor from the perspective of a graduated student.

<sup>2</sup>Banach Space are complete metric spaces where the metric does not come necessarily from a inner product. (See HUNTER and NACHTERGAELE [59])

III  $\langle x, x \rangle_V \geq 0$  and  $\langle x, x \rangle_V = 0 \Leftrightarrow x = 0$  (positive definite)

A classical example of a inner product is the euclidean dot product.

Another important exemple is the inner product defined on the space  $C[a, b]$  of complex (or real) valued continuous functions on the interval  $[a, b]$ , defined, for every  $f$  and  $g$  in  $C[a, b]$  as:

$$\langle f, g \rangle_{C[a, b]} = \int_a^b f(x) \overline{g(x)} dx \quad (3.1)$$

On any Hibert Space  $\mathcal{H}$  the norm induced by the inner product is:

$$\|x\|_{\mathcal{H}} = \sqrt{\langle x, x \rangle_{\mathcal{H}}} \quad (3.2)$$

where  $x \in \mathcal{H}$ . And the subsequent metric is defined as:

$$d_{\mathcal{H}}(x, y) = \|x - y\|_{\mathcal{H}} \quad (3.3)$$

for any  $x, y \in \mathcal{H}$ .

A vector space endowed with a inner product is a *inner product space* (aka. pre-Hilbert space). For it to be a Hibert Space it also has to be complete with respect to the above metric. Completeness means that any Cauchy sequence converges in this space (which provides a suitable framework to apply the tools of calculus). A Cauchy sequence is a sequence where every term becomes arbitrarily close to each other as the sequence progress (not only to term right next to it). It can be formalised as the sequence  $x_1, x_2, x_3, \dots$  on a metric space (with a metric  $d(\cdot, \cdot)$ ) where:

$$\forall \epsilon \in \mathbb{R}^+, \exists N \in \mathbb{Z}^+, \forall n, m > N \implies d(x_n, x_m) < \epsilon$$

On a pre-Hilbert space, the metric is given by equation 3.3. If a metric space  $M$  is complete, then every Cauchy sequence  $(x_1, x_2, x_3, \dots)$  converges in that space:

$$\exists x \in M, \forall \epsilon \in \mathbb{R}^+, \exists N \in \mathbb{Z}^+, \forall n > N \implies d(x_n, x) < \epsilon$$

So,

$$x = \lim_{n \rightarrow \infty} x_n$$

A complete metric space can be obtained from a pre-Hilbert space, by completion, in the same way that  $\mathbb{Q}$  is “completed” to make  $\mathbb{R}$ . Although completeness is a technicality, it is easy to find exemples of pre-Hibert spaces that lacks this property. The space of continuous functions  $C[a, b]$  with the inner product defined on 3.1 gives an exemple of pre-Hibert space that is not complete. For it to be a Hilbert space, the space have to be extend to include some discontinous functions, as in the larger

set of Lebesgue measurable<sup>3</sup> functions that are square integrable (with the Lebesgue integral).

Some examples of Hilbert space are:

- Any finite dimensional vector space over the field  $\mathbb{R}$  or  $\mathbb{C}$  with the standard dot product.
- The space  $\ell^2$  of square-summable sequences of complex numbers, i.e.  $(c_1, c_2, c_3, \dots)$  with  $c_i \in \mathbb{C}$  and  $\sum_{i=1}^{\infty} |c_i|^2 < \infty$ , is a Hilbert space with the inner product defined as: Given two sequences  $x = (x_1, x_2, x_3, \dots)$  and  $y = (y_1, y_2, y_3, \dots)$ , define  $\langle x, y \rangle_{\ell^2} = \sum_{i=1}^{\infty} x_i \bar{y}_i$ .
- Fourier series can be seen as the representation of a square-integrable function on the interval  $[0, 1]$  (member of  $L^2[0, 1]$ ) on the orthogonal basis  $\{e^{2\pi i n \theta} : n \in \mathbb{Z}\}$  with the inner product given by 3.1.

## 3.2 RKHS - Reproducing Kernel Hilbert Space

### 3.2.1 The Evaluation Functional

A *Reproducing Kernel Hilbert Space*, RKHS for short, is a special kind of Hilbert Space of functions. In a RKHS, closeness in the sense of the metric is actual pointwise proximity. That is to say, if two real-valued functions  $f$  and  $g$  of a (non-empty) set  $\mathcal{X}$  belong to a RKHS  $\mathcal{H}$  ( $f, g \in \mathcal{H} \subset \mathbb{R}^{\mathcal{X}}$ ), then whenever  $\|f - g\|_{\mathcal{H}}$  is small so is  $|f(x) - g(x)|$  for all  $x \in \mathcal{X}$ [60].

A more formal and useful characterization of a RKHS is consequence of studying linear operators on Hilbert Spaces. The evaluation functional  $\delta_x : \mathcal{H} \rightarrow \mathbb{R}$ ,  $\delta_x : f \rightarrow f(x)$  is easily seen as such: given  $f, g \in \mathcal{H}$  and  $a, b \in \mathbb{R}$ ,  $\delta_x(af + bg) = (af + bg)(x) = af(x) + bg(x) = a\delta_x(f) + b\delta_x(g)$ . When the evaluation functional is continuous on  $\mathcal{H}$ ,  $\mathcal{H}$  is said to be a RKHS.

Although  $L^2[a, b]$  is not a RKHS (it is not even a proper space of functions, but instead a space of classes of equivalences of functions), its bandlimited ( $L^2 \cap L^1$ ) version  $PW_{\pi} := \{f \in L^2(\mathbb{R}) \mid \text{supp } \mathcal{F}(f) \subseteq [-\pi, \pi]\}$ , for example, has a continuous evaluation functional. Here  $\mathcal{F} : L^2(\mathbb{R}) \rightarrow L^2[-\pi, \pi]$  is the Fourier transform[61]:

$$\mathcal{F}(f) = \frac{1}{\sqrt{2\pi}} \int f(t) e^{-i\omega t} dt \quad f \in L^2(\mathbb{R}) \quad (3.4a)$$

---

<sup>3</sup>The Lebesgue measurability of a, bounded with compact support, function is a highly technical exigence and the existence of a bounded non-Lebesgue measurable set (which allow the construction of such a function) is dependent on the axiomatic choice of the underlying set theory - it can only be proven with the addition of the *choice axiom* to the ZF (Zermelo-Fraenkel) set of axioms (ZFC).

$$\mathcal{F}^{-1}(\hat{f}) = \frac{1}{\sqrt{2\pi}} \int_{-\pi}^{\pi} \hat{f}(\omega) e^{i\omega t} d\omega \quad \hat{f} \in L^2[-\pi, \pi] \quad (3.4b)$$

The proof of continuity for the evaluation functional on  $PW_{\pi}$  relies on the inverse Fourier transform 3.4b, Cauchy-Schwarz's and Parseval's theorems, used on equations 3.5c, 3.5d and 3.5e respectively. For  $f, g \in PW_{\pi}$  it goes as follows:

$$|\delta_x f - \delta_x g| = |f(x) - g(x)| \quad (3.5a)$$

$$= |\mathcal{F}^{-1}(\mathcal{F}(f - g))| \quad (3.5b)$$

$$= \left| \langle \mathcal{F}(f - g), \frac{e^{-i\omega x}}{\sqrt{2\pi}} \rangle_{L^2[-\pi, \pi]} \right| \quad (3.5c)$$

$$\leq \|f - g\|_{L^2[-\pi, \pi]} \left\| \frac{e^{-i\omega x}}{\sqrt{2\pi}} \right\|_{L^2[-\pi, \pi]} \quad (3.5d)$$

$$= \|f - g\|_{L^2(\mathbb{R})} \quad (3.5e)$$

Other examples of RKHSs will be further explore as its relation to kernels are developed.

### 3.2.2 Reproducing Kernels

Riesz representation theorem is an extension, for Hilbert Spaces, of the classical isomorphism between a finite vector space  $\mathcal{V}$  and its dual  $\mathcal{V}^*$ , the space of linear functions on  $\mathcal{V}$ . It states that for every element  $\phi \in \mathcal{H}^*$ , where  $\mathcal{H}^*$  is the space *continuous* linear functionals from  $\mathcal{H}$  into  $\mathbb{R}$  (dual space), there exist a unique  $f_{\phi} \in \mathcal{H}$ , defined by:

$$\phi(g) = \langle g, f_{\phi} \rangle_{\mathcal{H}} \quad \forall g \in \mathcal{H}$$

As consequence, the evaluation functional  $\delta_x$  has a representation on  $\mathcal{H}$  as  $k_x$ , the reproducing property:

$$f(x) = \delta_x(f) = \langle f, k_x \rangle_{\mathcal{H}} \quad \forall g \in \mathcal{H}$$

The important idea of pointwise convergence can be recovered:

$$|f(x) - g(x)| = |\delta_x(f - g)| \quad (3.6a)$$

$$= |\langle f - g, k_x \rangle_{\mathcal{H}}| \quad (3.6b)$$

$$\leq \|f - g\|_{\mathcal{H}} \|k_x\|_{\mathcal{H}} \quad (3.6c)$$

Where Cauchy-Schwarz inequality was used on line 3.6c and  $\|k_x\|_{\mathcal{H}}$  acts as a scaling factor the closeness at each specific  $x$ .



The evaluation functional represented in  $\mathcal{H}$  as  $k_x$  can be seen as a function itself. As such, its evaluation at every point  $y$  of  $\mathcal{X}$  point construct a two-variable function, the kernel:

$$k_x(y) = K(x, y) = \langle k_x, k_y \rangle_{\mathcal{H}} \quad (3.7)$$

The kernel function  $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  is symmetric (because this is the real case) and positive definite as direct consequence of inner product definition. The converse, however, is a result of the Moore-Aronszajn theorem which says that for every symmetric positive definite function  $K(\bullet, \bullet)$  (kernel) on  $\mathcal{X} \times \mathcal{X}$  there is a unique Hilbert space  $\mathcal{H}$  of functions (RKHS) on  $\mathcal{X}$  for which  $K$  is a reproducing kernel. The reproducing property of the kernel is:

$$f(x) = \langle K(x, \bullet), f \rangle_{\mathcal{H}} \quad f \in \mathcal{H} \quad (3.8)$$

Examples of common used kernels:

- Gaussian Kernel/Radial Basis Function Kernel (RBF)

$$K(x, y) = e^{-\gamma \|x-y\|^2} \quad \gamma \in \mathbb{R}^+ \quad (3.9)$$

- Laplacian Kernel

$$K(x, y) = e^{-\lambda \|x-y\|} \quad \lambda \in \mathbb{R}^+$$

- $PW_{\pi}$  Kernel (the bandlimited  $L^2(\mathbb{R})$  space, see 3.2.1)

$$K(x, y) = \frac{\sin \pi(x-y)}{\pi(x-y)}$$

- Linear Kernel

$$K(x, y) = \langle x, y \rangle$$

- Polynomial Kernel

$$K(x, y) = (\gamma \langle x, y \rangle + 1)^n \quad \gamma \in \mathbb{R}, n \in \mathbb{N}^+$$

Its possible to operate with kernels and generate new valid kernels. If  $K_1, K_2$  are kernels for  $\mathcal{H}_1$  and  $\mathcal{H}_2$ , respectively, then for any  $\alpha, \beta \in \mathbb{R}_{\geq 0}$  it is possible to construct  $K = \alpha K_1 + \beta K_2$  as a kernel for the RKHS  $\mathcal{H} = \alpha \mathcal{H}_1 + \beta \mathcal{H}_2 = \{\alpha f_1 + \beta f_2 | f_1 \in \mathcal{H}_1, f_2 \in \mathcal{H}_2\}$ . Kernel products are valid even for functions acting on different sets, that is,  $K_1 : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$  and  $K_2 : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$  define  $K : (\mathcal{X} \times \mathcal{Y}) \times (\mathcal{X} \times \mathcal{Y}) \rightarrow \mathbb{R}$  as

$K((x, y), (x', y')) = K_1(x, x')K_2(y, y')$  with the RKHS  $\mathcal{H} \cong \mathcal{H}_1 \otimes \mathcal{H}_2$ , having  $\mathcal{X} = \mathcal{Y}$  as a special case.

### 3.2.3 Feature Maps

A feature map is a map  $\varphi$  from a set  $\mathcal{X} \neq \emptyset$  to a Hilbert space  $\mathcal{H}$ , the feature space. Any feature map can define a RKHS through the kernel:

$$K(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}} \quad x, y \in \mathcal{X} \quad (3.10)$$

For example, if  $\mathcal{X}$  is already a RKHS by itself, then feature map  $\varphi(x) = x$  reconstruct the Linear Kernel. Also, a sequence of functions  $f_i \in \mathbb{R}^{\mathcal{X}}, \forall i \in \mathbb{N}$  that  $\{f_i(x)\} \in \ell^2, \forall x \in \mathcal{X}$  are themselves a feature map  $\varphi(x) = \{f_i(x)\}$  with kernel:

$$K(x, y) = \sum_{i=1}^{\infty} f_i(x)f_i(y)$$

The converse, however, is not unique. Given a kernel there are multiple feature maps that can generate it. A simple example is  $K(x, y) = 2xy$  with  $\mathcal{X} = \mathbb{R}$ , this kernel can also be generated by:

$$K(x, y) = \begin{bmatrix} x \\ x \end{bmatrix} \cdot \begin{bmatrix} y \\ y \end{bmatrix}$$

In the first case  $\varphi_1(x) = x\sqrt{2} \in \mathbb{R}$ , while in the second  $\varphi_2(x) = \begin{bmatrix} x \\ x \end{bmatrix} \in \mathbb{R}^2$ . Standard ways of finding and approximating feature maps will be discussed in the next sections.

## 3.3 Probabilistic Regression

Probabilistic regression is similar to classification, both infer properties of a sample based on previous information. However, instead of giving a definite answer for which class an element belongs, probabilistic regression gives the probability for such a classification [62]. More formally, a training set is a sequence of  $n$  pairs  $\{(x_i, y_i) \mid i = 1, \dots, n\}$ , where  $x_i$  and  $y_i$  are samples drawn from random variables  $X$  and  $Y$ , respectively, with joint probability distribution  $\Pr(X, Y)$  [63]. From this training set, a conditioning probability  $\Pr(Y|X = x)$  has to be estimated.

The special case where  $Y$  is a Bernoulli random variable, i.e. a binary variable, is called binary regression. It is the single most important regression for mapping, as such, no other kind is explored in this thesis.



### 3.3.1 Binary Logistic Regression

A binary regression the dependent variable  $Y \in \{-1, 1\}$  (or some set of equal cardinality like  $\{0, 1\}$ ) has two possible estimations that are related by  $\Pr(Y = 1|X = x) + \Pr(Y = -1|X = x) = 1$ . As such, the conditional probability can be denoted

$$p(x) = \Pr(Y = 1|X = x)$$

The probability for  $Y = -1$  can be recovered from  $P(Y = -1|X = x) = 1 - p(x)$ . The linear logistic model  $p(x; \mathbf{w})$  for  $x \in \mathbb{R}^d$ , with  $\mathbf{w} \in \mathbb{R}^d$  as explicit parameter:

$$p(x; \mathbf{w}) = \frac{1}{1 + \exp(-\mathbf{w} \cdot x)} \quad (3.11)$$

The rationale behind the model is that the function  $\text{expit}(\alpha) = (1 + \exp(-\alpha))^{-1}$  is a bijection  $\mathbb{R} \rightarrow (0, 1)$  [63]. Ensuring that the model is a probabilistic distribution.

The classical regression theory requires a loss function  $\mathcal{C}$  and minimizes an empirical risk over some space of functions  $R[f] = \mathbb{E}(\mathcal{C}(X, Y, f(X)))$  [62]. The estimation of  $f$  from samples uses a regularized version:

$$R_{\text{reg}}[f] = \frac{1}{n} \sum_{i=1}^n \mathcal{C}(x_i, y_i, f(x_i)) + \lambda \mathcal{S}[f] \quad (3.12)$$

Where  $\lambda > 0$  and  $\mathcal{S}[\bullet]$  stabilization (regularization) term, as the minimization problem is typically ill-posed [62]. On normed spaces usually  $\mathcal{S}[f] = g(\|f\|)$ , where  $g(\bullet)$  is a monotonically increasing function. The loss function  $\mathcal{C}$  is problem dependent and for binary regression a typical negative log likelihood (NLL) is used

$$\mathcal{C}(x, y, p(x)) = -\log \Pr(Y = y|X = x) = \begin{cases} -\log p(x) & y = 1 \\ -\log(1 - p(x)) & y = -1 \end{cases} \quad (3.13)$$

For a linear logistic model, the regularized negative log likelihood empirical risk simplifies to a  $d$  dimensional minimization:

$$\text{NLL}_{\text{reg}}(\mathbf{w}) = \sum_{i=1}^n \log(1 + \exp(-y_i \mathbf{w} \cdot x_i)) + \lambda \mathcal{S}(\mathbf{w}) \quad (3.14)$$

Reasonable choices for  $\mathcal{S}(\bullet)$  are the  $\ell_1$  norm (LASSO)  $\mathcal{S}(\mathbf{w}) = \|\mathbf{w}\|_1$  or elastic net, a combination of  $\ell_1$  (LASSO) and squared  $\ell_2$  norms (ridge),  $\mathcal{S}(\mathbf{w}) = \alpha_1 \|\mathbf{w}\|_1 + \alpha_2 \|\mathbf{w}\|_2^2$ , where  $\alpha_1 + \alpha_2 = 1$ ,  $\alpha_1, \alpha_2 \in [0, 1]$  [64].

Although it is an applicable setting for simple situations, it is not expected to perform well for classification/regression of point on a 3D environment as  $x \in \mathbb{R}^3$  and  $\mathbf{w} \in \mathbb{R}^3$ . A 3 real parameter  $\mathbf{w}$  is not enough to capture all the complexities

of the environment. To keep its linear simplicity, the alternative is to increase dimensionality using Hilbert Spaces.

### 3.3.2 Regression on Hilbert Spaces

Samples defined on a low dimensional space  $\mathcal{X}$ , e.g.  $\mathbb{R}^3$ , can be raised to a high (possible infinite) dimensional space using a feature map  $\varphi : \mathcal{X} \rightarrow \mathcal{H}$ , where  $\mathcal{H}$  is hilbert space. This allow linear models to express more generic functions  $f(x) = \langle \mathbf{w}, \varphi(x) \rangle_{\mathcal{H}}$  with  $\mathbf{w} \in \mathcal{H}$ , such a space of functions  $f(x)$  is the actual RKHS with kernel define as in equation 3.10.

The linear logistic model from equation 3.11 lifted to the Hilbert Space  $\mathcal{H}$  is

$$p(x; \mathbf{w}) = \frac{1}{1 + \exp(-\langle \mathbf{w}, \varphi(x) \rangle_{\mathcal{H}})} \quad (3.15)$$

Thus equation 3.14 for log likelihood empirical risk becomes<sup>4</sup>

$$\text{NLL}_{\text{reg}}(\mathbf{w}) = \sum_{i=1}^n \log(1 + \exp(-y_i \langle \mathbf{w}, \varphi(x_i) \rangle_{\mathcal{H}})) + \lambda \mathcal{S}(\mathbf{w}) \quad (3.16)$$

In practice, one choose a kernel  $K(x, y)$  with desired properties and find finite dimensional approximate features  $\hat{\varphi}(x)$ , such that  $K(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}} \approx \hat{\varphi}(x) \cdot \hat{\varphi}(y)$ . Non kernel specific methods for finding approximate features include sampling the Fourier transform of shift invariant kernels [65], i.e.  $K(x, y) = k(x - y) = k(\delta)$ , and Nyström method that projects the Gram matrix  $\mathbf{G}_{ij} = K(x_i, x_j)$  of the sample points  $\{x_i\}$  into some subset of these points and use this projection to approximate the feature maps [66].

---

<sup>4</sup>With appropriate ajustment of  $\lambda$ 's value, i.e.  $\lambda = n\lambda_0$ , the usual average  $1/n$  for the summation can be dropped, as it keeps the same minimizer value of  $\mathbf{w}$ .

# Chapter 4

## Mapping

‘Would you tell me, please, which way  
I ought to go from here?’  
‘That depends a good deal on where  
you want to get to,’ said the Cat.  
‘I don’t much care where—’ said Alice.  
‘Then it doesn’t matter which way you  
go,’ said the Cat.

---

Alice’s Adventures in Wonderland  
by Lewis Carroll

Mapping is not a pre-defined concept, there are different ways to think about mapping, all depends on one’s need. The unifying idea is a method to fuse and represent geometric information about a given environment. Nevertheless, the meaning of how to represent which information is a consequence of the application. Here, in this thesis, the objective is to have human-understandable map yet with a probabilistic interpretation of a 3D environment.

### 4.1 Map Representation

Classically maps are binary functions over the space. The binary choices of function’s range stand for fullness or emptiness of a point. Mapping is to construct a probability distribution over the set of all maps [4], i.e.  $\mathcal{M}_d = \{\mathbf{m} \mid \mathbf{m} : \mathbb{R}^d \rightarrow \{0, 1\}\}$ . The cardinality of such a set  $|\mathcal{M}_d| = \beth_2$  is too big<sup>1</sup> to be tractable even considering computational discretization (finite precision).

Three approaches to this problem are presented here. One option is to discretize the space even further than the computational limit. Another is to renounce the idea of binary maps and consider maps to be a collection of predefined objects. And

---

<sup>1</sup> $\beth_2 = 2^{\mathfrak{c}}$  the cardinality the power set of the continuum, where  $\mathfrak{c}$  is the cardinality of the continuum

lastly, keep the concept of binary maps, yet to consider some restriction on the space of functions from which the map  $\mathbf{m}$  is drawn.

Despite their differences, all approaches describe marginal probabilities instead of the probability for an actual map  $\mathbf{m}$ . For binary maps, that means  $\Pr(\mathbf{m}(x_i) = 1)$  and not  $\Pr(\mathbf{m} = m_i)$  and, when considering maps as collection of objects, it is  $\Pr(\mathfrak{s}(\mathbf{l}_n) = \mathfrak{s}_i)$ , where  $\mathfrak{s}(\bullet)$  is the defining properties of a object  $\mathbf{l}_n$ , for example point and angle in the case of a line.

#### 4.1.1 Discrete Map

Discrete maps are also denominated *grid maps* because when discretizing each axis of a  $\mathbb{R}^d$  space the result is necessarily a grid. Originally developed for 2D maps [4], their where later extended to 3D maps in different ways.

#### 3D grids

The first obvious extension was a 3D grid of cubes by discretizing a range of each direction on  $N$  elements. Reasoning that each cube still full or empty, the set of possible maps on a  $N_1 \times N_2 \times N_3$  grid  $\mathbf{D}$  is  $\bar{\mathcal{M}}_d = \{\mathbf{m} \mid \mathbf{m} : \mathbf{D} \rightarrow \{0, 1\}\}$ . The cardinality of  $|\bar{\mathcal{M}}_d| = 2^{N_1 \cdot N_2 \cdot N_3}$  is too big to store the probability of every element.

The simplifying assumption for 3D grid is the conditional independence of grid elements  $\mathbf{m}(d_i)$  and  $\mathbf{m}(d_j)$  for  $d_i \neq d_j \in \mathbf{D}$  on the sensors measurements  $\mathbf{z}_n$ . Therefore, the probability of a map become the product of the marginals:

$$\Pr(\mathbf{m} = m_i \mid \mathbf{z}_n) = \prod_{d \in \mathbf{D}} \Pr(\mathbf{m}(d) = m_i(d) \mid \mathbf{z}_n)$$

Writing marginals as  $p_n(d) = \Pr(\mathbf{m}(d) = 1 \mid \mathbf{z}_n)$  keeps same information because  $\Pr(\mathbf{m}(d) = m_i(d) \mid \mathbf{z}_n)$  equals  $p_n(d)$  if  $m_i(d) = 1$  and  $1 - p_n(d)$  otherwise. The advantage is that it makes clearer that they can be stored and updated independently, and also that the number of stored elements is  $|\mathbf{D}| = N_1 \cdot N_2 \cdot N_3$ <sup>2</sup>. That might still be a lot, but with clever memory implementations like Octomaps [67], it can be manageable.

Marginal probability computation on each cube is a direct application of Bayes rule<sup>3</sup> (a Bayes filter) with a log-odds representation for better faster computaion, known as occupancy in this context [4]:

$$l_n(d) = l_{n-1}(d) + \text{inverse\_sensor}(d, z_n) - l_0(d) \quad (4.1)$$

---

<sup>2</sup>e.g. if  $N_1 = N_2 = N_3 = 200$  for a 5cm resolution on cube with 10m edge,  $|\mathbf{D}| = 8,000,000$  already.

<sup>3</sup> $\Pr(A \mid B) = \frac{\Pr(B \mid A) \Pr(A)}{\Pr(B)}$

Here  $l_n(d)$  is the log-odds representation of  $p_n(d)$ , the  $n$ th estimate after all previous  $n$  sensor measurements  $\mathbf{z}_n$ , including the last one  $z_n$ .

$$l_n(d) = \log \frac{p_n(d)}{1 - p_n(d)}$$

The *prior* of the occupancy is  $l_0(d)$ , log-odds of the *prior* probability  $p_0(d)$ . Defining **inverse\_sensor** $(d, z_n) = \Pr(\mathbf{m}(d) = 1 \mid z_n)$ , it is the probability of fullness for a grid element  $d$  given **only** the last measurement  $z_n$ , it can be interpreted as inference from the sensor response, justifying the name.

THRUN *et al.* [4] suggests to abandon independency between grid elements. To archive that, he employs a forward sensor model, instead of a **inverse\_sensor**, and uses optimization algorithm Expectation Maximization (EM) on the marginal to find a best fit. It is successful on solving “conflicts” between sonar responses when, because of a wide beamwidth, the same region appears to be full and empty depending on viewpoint.

Another, not so well explored, approach for calculating the marginal probability for grid elements comes from Evidential Theory. Evidential Theory, a.k.a. Dempster-Shafer theory (DST), is a mathematical theory of evidence, assigning “probabilities” (belief mass) to all non-empty elements of the power set of events. On the binary  $\{0, 1\}$  case, the three non-empty subsets are 0,1,0, 1 standing for evidence of emptiness, fullness or both, which “probabilities” add to one. Consequently yielding to two maps, one for fullness other for emptiness. In DST the actual probability (in the classical sense) appears as lower and upper bounds (Plausability and Belief), allowing ignorance to be modeled adequately. The 2D case was explored by PAGAC *et al.* [68], their article also further describes DST.

Grid based algorithms on 3D environments suffer from their high number of grid elements, the next model try to avoid this problem.

## Elevation Maps

In an attempt to keep the grid to a reasonable size, elevation maps, or 2.5D maps, keep the discretization only on the 2 horizontal dimensions. The third dimension is represented as a height value assigned to each 2D discretization.

Some work on seabed reconstruction using sonars has been done by COIRAS and GROEN [2], COIRAS *et al.* [69]. They attempt to map by reconstructing a 2.5D surface through optimization on the height value of each grid element. That leads to information gain on local surface’s reflectivity, an indication of its composition. However, the expectation-maximization method does not leave a direct probabilistic interpretation for the values.

Although elevation maps reduce memory requirements by not discretizing on the

vertical axis, its elevation value is unique for each grid element. As such, it is not able to represent objects above the floor level, e.g. ceil, trees, caves, etc. That is addressed in the next option.

### Multi Layer Surface - MLS

As a compromise between the last two solution, grid and elevation maps, Multi Layer Surface (MLS) was developed. It originates as elevation maps, but instead of having only one height per grid element, it splits into many layers of varying height, called *surface patches*.

When MLS was first proposed by TRIEBEL *et al.* [70], each surface patch of each grid element stored statistics as mean height and standard deviation. It was soon realised that a flat horizontal plane for a grid and a preferential vertical direction could be an issue for well describing statistical knowledge of the environment. Some extensions have already been suggested to address this matter, e.g. works of RIVADENEYRA and CAMPBELL [71] and SCHWENDNER [72].

#### 4.1.2 Map of Features

A map can, in certain situations, be approximated by a collection of geometric objects, viz. line segments, circles, etc. This is specially true for structured environments [73] where walls and flat surfaces are easily found.

The 2D simplification, only considering the horizontal plan, was explored for underwater SLAM with Imagins Sonars by RIBAS *et al.* [74]. They used lines to represent a environment, with endpoints only for display purpose. Line extraction relied on polar the parametric space of lines (angle and distance to the origin) as on Hough Voting and extensions.

Besides only applied to 2D, a particular downside of the approach is its little generality. Generic, unstructured or complex environments are unhandleable.

#### 4.1.3 Continuous Map

There is no need for discretization on space if some restrictions are applied to the space of function of maps. Formally that means  $\tilde{\mathcal{M}}_d \subseteq \mathcal{V} \cap \mathcal{M}_d$ , where  $\mathcal{V}$  is some restricted space of fuctions, e.g. continuous compact supported, and  $\mathcal{M}_d$  as defined on section 4.1. In practice the restriction is not directly applied to the space of functions, instead it is enforced on the probability distribution, such that functions outside  $\tilde{\mathcal{M}}_d$  have zero probability.

A marginal probability for continuous map evaluates at every point in  $\mathbb{R}^d$ , not in some discretized space as for 3D grids presented in section 4.1.1. The marginals

shall then be written as  $p(x) = \Pr(\mathbf{m}(x) = 1)$  for  $x \in \mathbb{R}^d$ .

## Gaussian Process Occupancy Maps - GPOM

Possibly the first succesfull attemp to have a continous map was Gaussian Process Occupancy Maps (GPOM) by O'CALLAGHAN and RAMOS [75], as noted by the author there were previous attemps, but they lacked computability or did not truly represented occupancy. GPOM's method apply learning techniques, Gaussian Process, to estimate the best marginal form a family of function  $p(x) = \Phi(\bullet)$ ,  $\Phi$  is the cumulative unit Gaussian. Details of the learning process are numerous and complex, for the interested reader it is suggested to check the original paper. GPOM was already applied to 3D environments for path planning of a 6DOF Rotary Unmanned Aerial Vehicle (RUAV) using laser sensors [76].<sup>4</sup>

Following ideas from GPOM, Hilbert Maps were proposed as less computationally expensive alternative for continuous maps. It is also able to naturally account for spatial correlations and handle noise, but while GPOM scales cubically with data size, Hibert Maps can be updated in linear time. Hibert Maps is yet much simpler to implement and it was the chosen altertive for sonar reconstruction on this work. It suits well the task as most of imaging sonar drawbacks as a sensor are its noise and wide beamwidth.

## Hibert Maps

Hibert Maps is a recent development by RAMOS and OTT [77], from 2016. It was applied to the 2D case, on a laser sensor setting. This thesis apply the method to 3D environment with sonar, imaging sonar especially. Most of the mathematical machinery necessary was described in chapter 3, however some details still to be presented.

Marginal probabilities are assumed to be linear logistic as in equation 3.15:

$$p(x; \mathbf{w}) = \frac{1}{1 + \exp(-\langle \mathbf{w}, \varphi(x) \rangle_{\mathcal{H}})}$$

Where  $\mathbf{w}, \varphi(x) \in \mathcal{H}$ ,  $\mathcal{H}$  a Hilbert space that generates a RKHS with kernel  $K(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}}$ . As long range correlations are not expected, a suitable kernel choice is the RBF defined on equation 3.9:

$$K(x, y) = e^{-\gamma \|x-y\|^2} \quad \gamma \in \mathbb{R}^+$$

The features  $\varphi(x)$  are approximated as Nyström features, other methods did not

---


$$^4 \Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-t^2/2} dt$$

showed results as good [77], approximations were discussed in section 3.3.2. Nyström features are sample dependent, given set of  $n$  “inducing” points  $\nu_i \in \mathbb{R}^d$ ,  $i = 1 \dots n$  chosen (possibly random) on the region being mapped, it is calculated as:

$$\hat{\varphi}(x) = \sqrt{\Lambda} \mathbf{Q}^T \hat{K}(x) \quad (4.2)$$

Where  $\mathbf{G} = \mathbf{Q} \Lambda \mathbf{Q}^T$  is the spectral decomposition of the Gram matrix of the “inducing” points,  $\mathbf{G}_{ij} = K(\nu_i, \nu_j)$ , and  $\hat{K}_i(x) = K(x, \nu_i)$  is a column vector. After this finite approximations,  $\mathbf{w} \in \mathbb{R}^n$  and equation 3.15 becomes:

$$p(x; \mathbf{w}) = \frac{1}{1 + \exp(-\mathbf{w} \cdot \hat{\varphi}(x))} \quad (4.3)$$

And the negative log likelihood - NLL - regression, from equation 3.16 becomes:

$$\text{NLL}_{\text{reg}}(\mathbf{w}) = \sum_{i=1}^n \log(1 + \exp(-y_i \mathbf{w} \cdot \hat{\varphi}(x_i))) + \lambda \mathcal{S}(\mathbf{w}) \quad (4.4)$$

Where  $(y_i, x_i)$  are the sample points and  $\mathcal{S}(\bullet)$  an elastic-net regulariser, described on the context of logistic regression of section 3.3.1:

$$\mathcal{S}(\mathbf{w}) = \alpha_1 \|\mathbf{w}\|_1 + \alpha_2 \|\mathbf{w}\|_2^2 \quad (4.5)$$

With  $\alpha_1 + \alpha_2 = 1$ ,  $\alpha_1, \alpha_2 \in [0, 1]$ .

Minimization of  $\text{NLL}_{\text{reg}}(\mathbf{w})$  gives a representation of the map as a  $n$  parameter vector  $\mathbf{w}$ . It can then be evaluated at any point with equation 4.3. The minimization step is solved iteratively as a learning processed, discussed further ahead, which is naturally an online process.

## 4.2 Inverse Sonar Model

Methods that do not use Expectation-Maximization (EM), as some of those described in sections 4.1.1, usually require some form of inversion of the sensor model. This is a way to characterize the environment from a sensor response.

In the context of grid maps, it appear as a conditional probability on the measurement, i.e.  $\text{inverse\_sensor}(d, z_n) = \Pr(\mathbf{m}(d) = 1 \mid z_n)$  in equation 4.1. For sonars, it is generally considered as a constant value for grid elements inside an occupied bin (see figure 2.12) and another for those outside[4]. THRUN *et al.* [4] also creates an inverse model by training a machine learning algorithm with generated responses.

The difference between an occupied bin and an empty one can be a simplistic



bin-value threshold [4, 74, 78] or some more complex procedure, e.g. histograms.

### 4.2.1 Sonar on Feature Maps

For Hilbert Maps, only procedures for laser are already discribed, thus here a strategy is proposed for sonar responses. After defining the empty and full bins, there are three main differences between sonars and lasers: Sonars bins are volumetric; Lasers have a definite hit point, a full bin only means that there is something somewhere inside the bin; Sonars may have mutiple full bins, on a beam, while lasers have only on hit.

Given the sonar/laser differences, it is suggested:

- I Sample the empty beam volume between the sonar and the first full bin.
- II Sample all full bin volumes, with a possible different sample density.

A beam volume is a frustrum of a sphere, a region between the simple concept of apperture angles (see section 2.1.2) and within a range of distances given by the bin. The sample density used was uniform on the volume, because it is a guide for the expected echo point within the bin and there is no obvious privileged point.

The rationale behind ignoring all empty bins after the first full bin is to avoid considering shadow zones as empty regions, i.e. acoustic shadows of the first echo. As empty regions means no echo anywhere inside the bin, they are actually the most important information provided by the sonar. Each sample from the empty beam volume is evaluated to the feature map and passed to the learning algorithm. However, full bins indicate a echo somewhere within the bin. Thus a whole colection of samples from a full bin are embedded to the Hilbert Space as a distribution [77, 79]:

$$\hat{\varphi}(\mathbb{P}) = \frac{1}{n} \sum_{i=1}^n \hat{\varphi}(x_i) \quad (4.6)$$

Where  $\mathbb{P}$  is the original distribution from where  $n$  samples  $x_i$  are drawn. In the learning setting, this  $\hat{\varphi}(\mathbb{P})$  can be considered more than once, if the certainty is higher, as it will be commented in the next section.

## 4.3 Map Learning

Map learning is an iterative optimization process to find the  $\mathbf{w}$  that minimizes  $\text{NLL}_{\text{reg}}(\mathbf{w})$ , equation 4.4. As it is a convex function, gradient descent method would find it global minimum. The gradient of the objective function is:

$$\nabla \text{NLL}_{\text{reg}}(\mathbf{w}) = \sum_{i=1}^n -y_i \hat{\varphi}(x_i) (1 + \exp(y_i \mathbf{w} \cdot \hat{\varphi}(x_i)))^{-1} + \lambda \nabla \mathcal{S}(\mathbf{w}) \quad (4.7)$$

The gradient  $\nabla \mathcal{S}(\mathbf{w})$  is calculated using sub-differentials, as the  $\ell_1$  part of a elastic-net  $\mathcal{S}(\mathbf{w})$ , equation 4.5, is non-differentiable at  $\mathbf{w} = \mathbf{0}$ .

The gradient descent method generates a sequence of approximated values for  $\mathbf{w}$  by descenting on the gradient direction:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t) \quad (4.8)$$

Where  $\eta \in \mathbb{R}^+$  is a step value and  $\mathbf{w}_t$  is a sequence of approximations. The issue with this approach is the cost of computing  $\nabla \text{NLL}_{\text{reg}}(\mathbf{w})$  for a whole map, the summation of equation 4.7 ranges over all sampled points from all beams from all measurements, with sampling as in section 4.2, and it is computed at every step of  $\mathbf{w}_t$ .

### 4.3.1 Stochastic Gradient Descent - SGD

To overcome the sampling size issue of gradient descent, stochastic gradient descent proposes the use of a single, or small batch, of samples at a time. One first shuffle the training samples [80] then directly update  $\mathbf{w}_t$  as:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; (y_t, x_t)) \quad (4.9a)$$

$$\nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; (y_t, x_t)) = -y_t \hat{\varphi}(x_t) (1 + \exp(y_t \mathbf{w} \cdot \hat{\varphi}(x_t)))^{-1} + \lambda \nabla \mathcal{S}(\mathbf{w}) \quad (4.9b)$$

Where  $(y_t, x_t)$  are the shuffled version of  $(y_i, x_i)$ . The mini-batch variation [81] of this method takes partition the set of samples  $\sqcup_k I_k = \{(y_i, x_i) | i = 1 \dots n\}$  and shuffle, then the update equation becomes:

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; I_t) \quad (4.10a)$$

$$\nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; I_t) = \sum_{(y_i, x_i) \in I_t} -y_i \hat{\varphi}(x_i) (1 + \exp(y_i \mathbf{w} \cdot \hat{\varphi}(x_i)))^{-1} + \lambda \nabla \mathcal{S}(\mathbf{w}) \quad (4.10b)$$

The algorithm is guaranteed to converge (under mild conditions BOTTOU [80]), given that  $\sum_t \eta_t^2 < \infty$  and  $\sum_t \eta_t = \infty$ . A classic choice for  $\eta_t$  is

$$\eta_t = \frac{\eta_0}{1 + t/n} \quad (4.11)$$

Where  $\eta_0$  is a initial step determined from a small sample and  $n$  is the number of samples. Variations of this form also are commum, RAMOS and OTT [77] provides another choice:

$$\eta_t = \frac{1}{\lambda \alpha_2(t_0 + t)}$$

Where  $\lambda$  is the regulator gain, equation 4.10a,  $\alpha_2$  is the  $\ell_2$  elastic-net gain, equation 4.5, and  $t_0$  is chosen from a small sample test. However, TSURUOKA *et al.* [82] adopts an exponential decay for  $\eta_t$ , which is not compliant with theoretical requirements, and they had a better result than using equation 4.11. The reason provided was that a harmonic progression decays too fast at the beginning and too slowly at the end. As a trade-off, this work employs a theoretically valid step that does not suffer from the aforementioned limitations:

$$\eta_t = \frac{\eta_0}{2} \left( \frac{1}{(1 + t/n) \log_n(n + t)} + k_1 e^{-t/n} \right) \quad (4.12)$$

Where  $\eta_0$ , similarly to the classical case, is an initial step. The rationale is to accommodate a slower decay at the beginning, dictated by the exponential component, and faster at the end without losing the divergence, as  $\sum_x x \log x$  is still divergent. Faster decaying end rates can always be found, if needed.

## 4.4 Implementation

### 4.4.1 Algorithm

The implemented algorithm receives as input a sequence of sonar positions  $\mathbf{P}_k$  and its respective sonar responses, that is a sequence of beams  $\mathbf{b}_j^{(k)}$  containing bearing and bins values. The procedure goes as illustrated in Algorithm 3.

In the description of the algorithm  $\nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; \text{feats})$  has a slightly different meaning:

$$\nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; \text{feats}) = \sum_{\hat{\varphi} \in \text{feats}} -y_t \hat{\varphi} (1 + \exp(y_t \mathbf{w} \cdot \hat{\varphi}))^{-1} + \lambda \nabla \mathcal{S}(\mathbf{w})$$

### 4.4.2 Results

Although the algorithm generates a 3D representation of an environment, results displayed here are plane cuts of this reconstruction only for better appreciation. The environment considered here is the 4x5 semi-infinity box of section 2.3.1. There were used 3000 inducing points (dimension of the feature map approximation - section 4.1.3) and 21 measurements from 7 different positions in 3 orthogonal orientations each from the sonar simulation. Different fractions of the number of beams were explored on figures 4.1, 4.2 and 4.3. Colors represent the value of the occupancy  $p(x; \mathbf{w})$ .

---

**Algorithm 3** Mapping
 

---

```

procedure MAPPING( $\mathbf{P}_k, \mathbf{b}_j^{(k)}$ )
   $I_k = \{\mathbf{b}_j^{(k)} | j \in \mu\mathbb{N}\}$  ▷ Partitioning at every  $\mu$  beam
   $\mathbf{w}_0 = \mathbf{0}$ 
  for all  $\mathbf{P}_k$  do
    for all  $\mathbf{b} \in I_k$  do
       $\mathbf{b} = \text{threshold}(\mathbf{b})$  ▷ Classify empty/full, section 4.2
       $e_i = \text{empty\_samples}(\mathbf{b})$  ▷ Sample first empty bins, section 4.2.1
       $f_i^{(z)} = \text{full\_samples}(\mathbf{b})$  ▷ Sample from every  $z$  full bin, section 4.2.1
       $\text{feats} = \{\}$ 
      for all  $e_i$  do
         $\text{feats} = \text{feats} \cup \{\hat{\varphi}(e_i)\}$  ▷ equation 4.2
      end for
      for all  $f_i^{(z)}$  do
         $\text{feats} = \text{feats} \cup \{\mathbb{E}_i[\hat{\varphi}(f_i^{(z)})]\}$  ▷ equation 4.6
      end for
    end for
     $\mathbf{w}^- = \eta_k \nabla \text{NLL}_{\text{reg}}(\mathbf{w}_t; \text{feats})$  ▷ equation 4.12 on a variant of 4.10a
  end for
  return  $\mathbf{w}$ 
end procedure

```

---

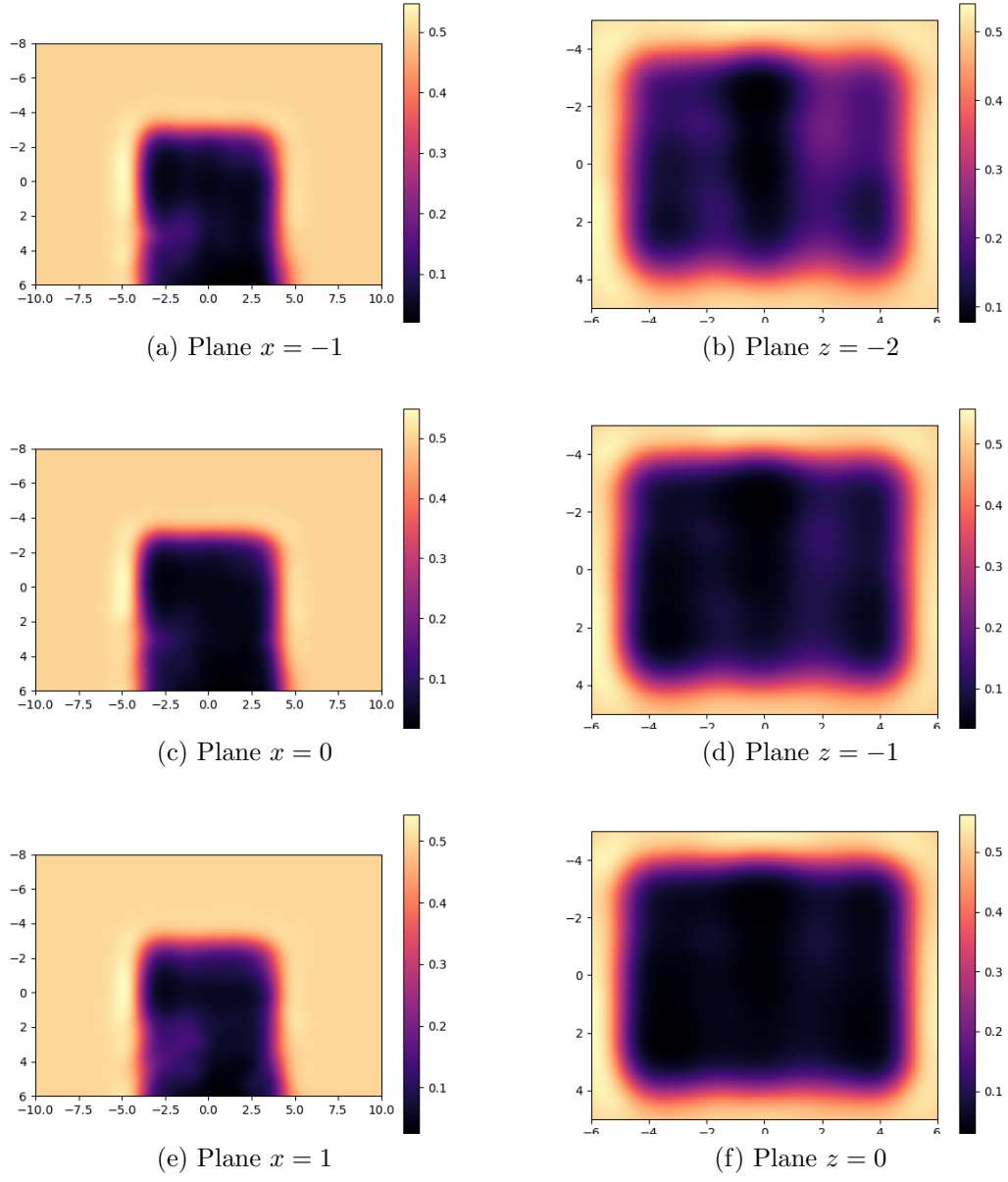


Figure 4.1: Mapping with 10% of available beams.

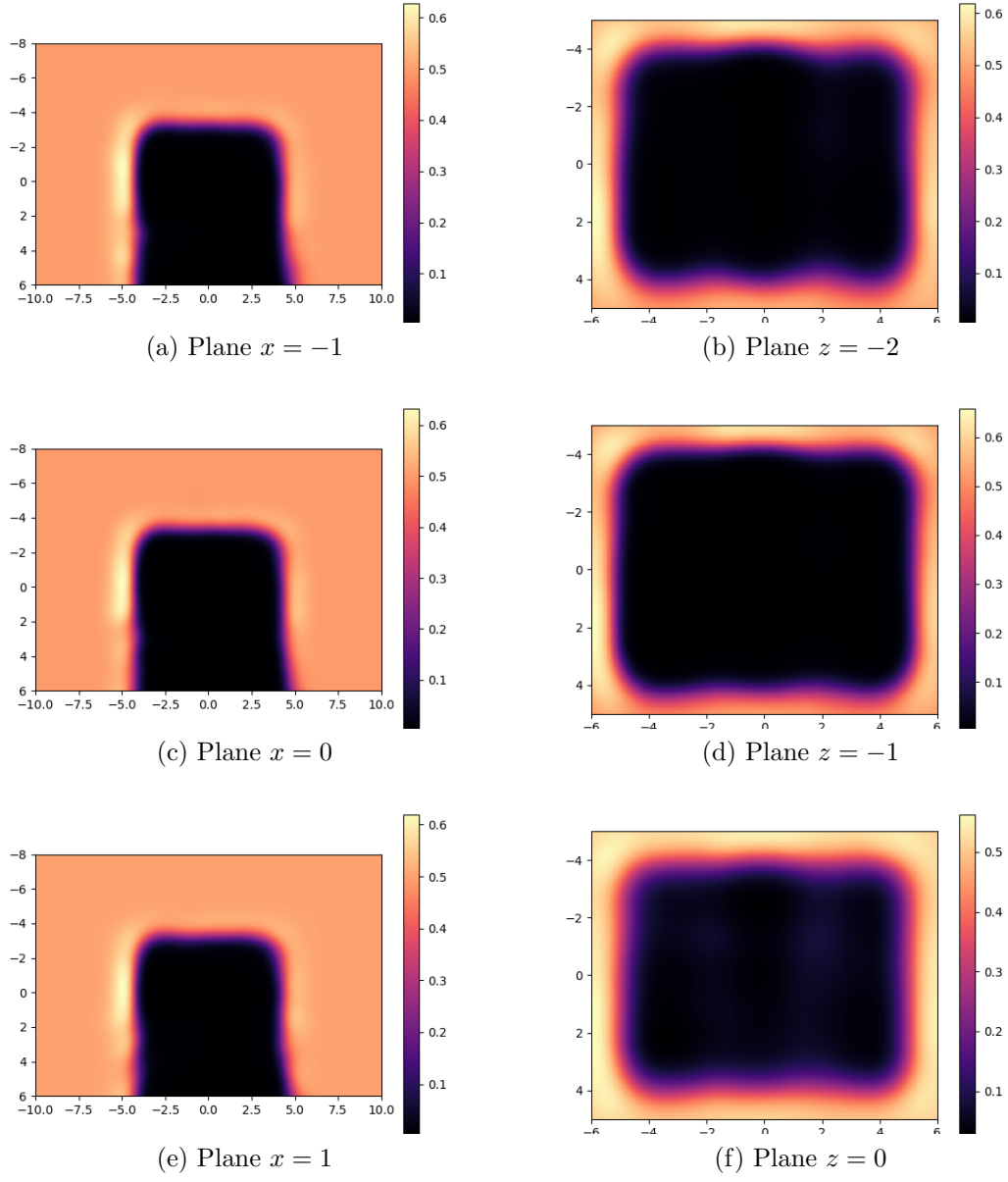


Figure 4.2: Mapping with 30% of available beams.

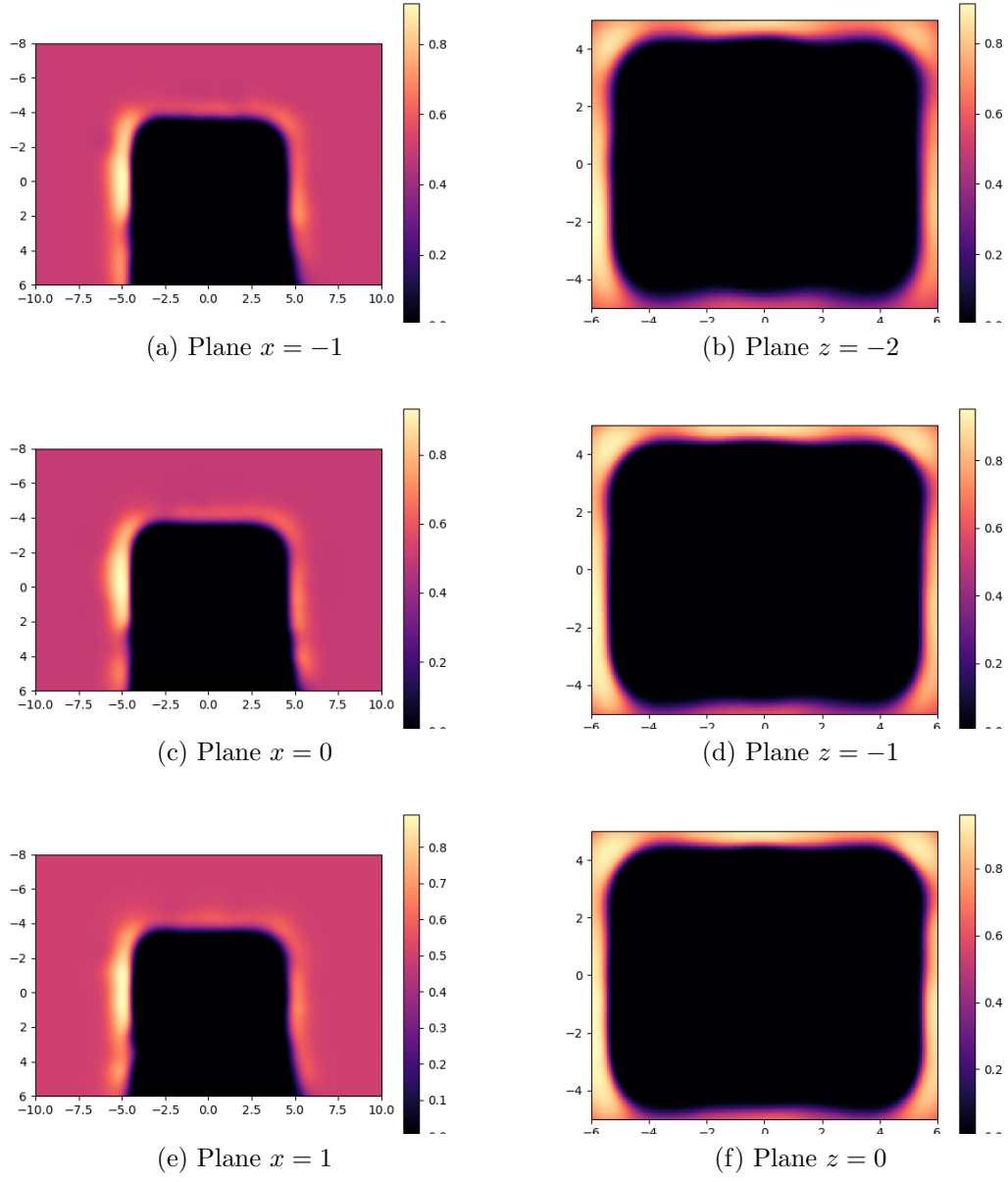


Figure 4.3: Mapping with 100% of available beams.

# Chapter 5

## Conclusion

All models are wrong, but some are useful.

---

George E. P. Box



# Bibliography

- [1] MILLER, T. E. *Real time bottom reverberation simulation in deep and shallow ocean environments*. Tese de Doutorado, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, 2015.
- [2] COIRAS, E., GROEN, J. “Simulation and 3d reconstruction of sidelooking sonar images”, *Advances in sonar technology. IN-TECH*, , n. February, pp. 1–15, 2009.
- [3] COIRAS, E., RAMIREZ-MONTESINOS, A., GROEN, J. “GPU-based simulation of side-looking sonar images”. In: *OCEANS 2009-EUROPE*, pp. 1–6. IEEE, 2009.
- [4] THRUN, S., BURGARD, W., FOX, D. *Probabilistic Robotics*. Intelligent robotics and autonomous agents. MIT Press, 2005. ISBN: 9780262201629. Disponível em: <<https://books.google.com.br/books?id=2Zn6AQAAQBAJ>>.
- [5] ETTER, P. C. *Underwater Acoustic Modeling and Simulation, Fourth Edition*. CRC Press, feb 2013. ISBN: 978-1-4665-6493-0. doi: 10.1201/b13906. Disponível em: <<http://www.crcnetbase.com/doi/book/10.1201/b13906>>.
- [6] FEYNMAN, LEIGHTON, SANDS. *The Feynman Lectures on Physics*, v. 1. Basic Book, 2010.
- [7] BRUNEAU, M. *Fundamentals of Acoustics*. ISTE, 2006.
- [8] FILIPPI, P., BERGASSOLI, A., HABAULT, D., et al. *Acoustics: basic physics, theory, and methods*. Academic Press, 1998.
- [9] LURTON, X. *An Introduction to Underwater Acoustics: Principles and Applications*. Springer Praxis Publishing, London, UK, 2010.
- [10] FAHY, F., WALKER, J. *Fundamentals of Noise and Vibration*. Taylor & Francis, 1998. ISBN: 9780419241805. Disponível em: <<https://books.google.com.br/books?id=lo4wUkHm9j0C>>.

- [11] D'AMICO, A., PITTENGER, R. *A brief history of active sonar*. Relatório técnico, DTIC Document, 2009.
- [12] CHAUDHARI, M. “Chirp Sonar and Electrical Resistivity Imaging survey for integrity of concrete lining in a Hydel Channel”, *JOURNAL OF INDIAN GEOPHYSICAL UNION*, v. 19, n. 2, pp. 167–174, 2015.
- [13] B.I.LEMBRIKOV, D. *Chirped Gaussian Pulses*. Relatório técnico, Holon Academic Institute of Technology, 2005.
- [14] “Sidescan Sonar Beamwidth”. 2005. Application Note.
- [15] CLAPP, M. A., ETIENNE-CUMMINGS, R. “Single ping-multiple measurements: Sonar bearing angle estimation using spatiotemporal frequency filters”, *IEEE Transactions on Circuits and Systems I: Regular Papers*, v. 53, n. 4, pp. 769–783, 2006.
- [16] KNEIPFER, R. R. *Sonar Beamforming-An Overview Of Its History and Status*. Relatório técnico, DTIC Document, 1992.
- [17] CHU, D., HUFNAGLE, L. C. “Time varying gain (TVG) measurements of a multibeam echo sounder for applications to quantitative acoustics”. In: *OCEANS 2006*, pp. 1–5. IEEE, 2006.
- [18] CHEW, J. L., CHITRE, M. “Object detection with sector scanning sonar”. In: *Oceans-San Diego, 2013*, pp. 1–8. IEEE, 2013.
- [19] DOBSON, C. “Introducing Sonar Technology as a Tool for Underwater Cave Surveying”, *BCRA Cave Radio & Eletronics Group*, v. 94, pp. 20–23, 2016.
- [20] JENSEN, F., KUPERMAN, W., PORTER, M., et al. *Computational Ocean Acoustics*. Modern Acoustics and Signal Processing. Springer New York, 2011. ISBN: 9781441986788.
- [21] FUNKHOUSER, T., TSINGOS, N., JOT, J.-M. “Survey of methods for modeling sound propagation in interactive virtual environment systems”. 2003.
- [22] URICK, R. J. *Sound propagation in the sea*. Relatório técnico, DARPA, 1979.
- [23] DEINES, E., BERTRAM, M., MOHRING, J., et al. “Comparative visualization for wave-based and geometric acoustics”, *IEEE Transactions on Visualization and Computer Graphics*, v. 12, n. 5, pp. 1173–1180, 2006.
- [24] BUCKINGHAM, M. J. *Ocean-acoustic propagation models*. EUR-OP, 1992.

- [25] TORRES, J. C. *Modeling of high-frequency acoustic propagation in shallow water*. Tese de Doutorado, Monterey California. Naval Postgraduate School, 2007.
- [26] STATES, N. D. R. C. U. *Physics of Sound in the Sea*. Department of the Navy, Headquarters Naval Material Command, 1969.
- [27] DESANTO, J. *Scalar Wave Theory: Green's Functions and Applications*. Springer Series on Wave Phenomena. Springer Berlin Heidelberg, 2012. ISBN: 9783642847387.
- [28] ABRAMOWITZ, M., STEGUN, I. A. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, v. 55. Courier Corporation, 1964.
- [29] BAKER, M., SUTLIEF, S. “Green’s Functions in Physics Version”, 2003.
- [30] WORZEL, J., EWING, M., PEKERIS, C. *Propagation of Sound in the Ocean*. Geological Society of America Memoir. Geological Society of America, 1948. ISBN: 9780813710273. Disponível em: <<https://books.google.com.br/books?id=UWaGs8720DcC>>.
- [31] BELL, J. M. “Application of optical ray tracing techniques to the simulation of sonar images”, *Optical Engineering*, v. 36, n. 6, pp. 1806–1813, 1997.
- [32] BARTELL, F. O., DERENIAK, E., WOLFE, W. “The theory and measurement of bidirectional reflectance distribution function (BRDF) and bidirectional transmittance distribution function (BTDF)”. In: *1980 Huntsville Technical Symposium*, pp. 154–160. International Society for Optics and Photonics, 1981.
- [33] RÖBER, N., KAMINSKI, U., MASUCH, M. “Ray acoustics using computer graphics technology”. In: *10th International Conference on Digital Audio Effects (DAFx-07)*, S, pp. 117–124. Citeseer, 2007.
- [34] BLAKE, V. S. “Remote sensing in underwater archaeology: Simulation of side scan sonar images using ray tracing techniques”, *BAR INTERNATIONAL SERIES*, v. 598, pp. 39–39, 1995.
- [35] DURANY, J., MATEOS, T., GARRIGA, A. “Analytical Computation of Acoustic Bidirectional Reflectance Distribution Functions”, *Open Journal of Acoustics*, v. 5, n. 04, pp. 207, 2015.

- [36] OF STANDARDS, U. S. N. B., NICODEMUS, F. E. *Geometrical considerations and nomenclature for reflectance*, v. 160. US Department of Commerce, National Bureau of Standards, 1977.
- [37] COX, T., D'ANTONIO, P. *Acoustic Absorbers and Diffusers: Theory, Design and Application*. Taylor & Francis, 2004. ISBN: 9780203492994. Disponível em: <<https://books.google.com.br/books?id=IDAuR3dn5ZsC>>.
- [38] COX, T. J., DALENBACK, B.-I., D'ANTONIO, P., et al. "A tutorial on scattering and diffusion coefficients for room acoustic surfaces", *Acta Acustica united with ACOUSTICA*, v. 92, n. 1, pp. 1–15, 2006.
- [39] JONES, A. D., SENDT, J., DUNCAN, A. J., et al. "Modelling the acoustic reflection loss at the rough ocean surface". In: *Proceedings of ACOUSTICS*, pp. 23–25, 2009.
- [40] DUNN, F., HARTMANN, W., CAMPBELL, D., et al. *Springer handbook of acoustics*. Springer, 2015.
- [41] VORLÄNDER, M., MOMMERTZ, E. "Definition and measurement of random-incidence scattering coefficients", *Applied Acoustics*, v. 60, n. 2, pp. 187–199, 2000.
- [42] RINDEL, J. H. "Scattering in room acoustics and related activities in ISO and AES". In: *Proceedings of 17th International Congress on Acoustics, Rome, Italy, 2-7 September*, 2001.
- [43] SILTANEN, S., LOKKI, T., KIMINKI, S., et al. "The room acoustic rendering equation", *The Journal of the Acoustical Society of America*, v. 122, n. 3, pp. 1624–1635, 2007.
- [44] PHONG, B. T. "Illumination for computer generated pictures", *Communications of the ACM*, v. 18, n. 6, pp. 311–317, 1975.
- [45] TRACER, A. R. C., BEAM, G., MOA, Q., et al. "Outdoor Sound Propagation with". .
- [46] CHANDAK, A., ANTANI, L., TAYLOR, M., et al. "Fast and accurate geometric sound propagation using visibility computations", *Building Acoustics*, v. 18, n. 1-2, pp. 123–144, 2011.
- [47] MUNJAL, M., MECHEL, F., VORLÄNDER, M., et al. *Formulas of Acoustics*. Springer Berlin Heidelberg, 2013. ISBN: 9783662072967. Disponível em: <<https://books.google.com.br/books?id=At95BgAAQBAJ>>.

- [48] REIF, J. H., TYGAR, J., YOSHIDA, A. “Computability and complexity of ray tracing”, *Discrete & Computational Geometry*, v. 11, n. 3, pp. 265–288, 1994.
- [49] BLAKEY, E. “Ray tracing—computing the incomputable?” *arXiv preprint arXiv:1404.0075*, 2014.
- [50] VEACH, E., GUIBAS, L. J. “Metropolis light transport”. In: *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pp. 65–76. ACM Press/Addison-Wesley Publishing Co., 1997.
- [51] LESLIE, J., CHEESMAN, W. “An ultrasonic method of studying deterioration and cracking in concrete structures”, *Journal of the American Concrete Institute*, v. 21, n. 1, pp. 17–36, 1949.
- [52] BELL, J. M., LINNETT, L. “Simulation and analysis of synthetic sidescan sonar images”, *IEE Proceedings-radar, sonar and navigation*, v. 144, n. 4, pp. 219–226, 1997.
- [53] MAUSSANG, F., CHANUSSOT, J., HÉTET, A., et al. “Mean–standard deviation representation of sonar images for echo detection: Application to SAS images”, *IEEE Journal of Oceanic Engineering*, v. 32, n. 4, pp. 956–970, 2007.
- [54] BELL, J., DARLINGTON, D., ELSTON, G. “Techniques for the physical modeling of the sonar image generation process”. In: *Proc. 1st Int. Symp. Physics Signal Image Processing (PSIP’99)*, pp. 66–72. Citeseer, 1999.
- [55] BORAWSKI, M., FORCZMAŃSKI, P. “Sonar image simulation by means of ray tracing and image processing”. In: *Enhanced methods in computer security, biometric and artificial intelligence systems*, Springer, pp. 209–214, 2005.
- [56] *Micron Sonar Product Manual*, 02 ed. Trittech International Ltd, . 0650-SOM-00003.
- [57] *Micron Modem Product Manual*, 02 ed. Trittech International Ltd, . 0630-SOM-00001.
- [58] YOUNG, N. *An introduction to Hilbert space*. 14th ed. The Edinburgh Building, Cambridge CB2 8RU, UK, Cambridge University Press, 2011.
- [59] HUNTER, J. K., NACHTERGAELE, B. *Applied Analysis*. Department of Mathematics, University of California, 2000.

- [60] BERLINET, A., THOMAS-AGNAN, C. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.
- [61] TREFETHEN, L. N. *Finite difference and spectral methods for ordinary and partial differential equations*. Cornell University-Department of Computer Science and Center for Applied Mathematics, 1996.
- [62] JAAKKOLA, T. S., HAUSSLER, D. “Probabilistic kernel regression models.” In: *AISTATS*, 1999.
- [63] FRIEDMAN, J., HASTIE, T., TIBSHIRANI, R. *The elements of statistical learning*, v. 1. Springer series in statistics Springer, Berlin, 2001.
- [64] HASTIE, T., TIBSHIRANI, R., WAINWRIGHT, M. *Statistical learning with sparsity*. CRC press, 2015.
- [65] RAHIMI, A., RECHT, B., OTHERS. “Random Features for Large-Scale Kernel Machines.” In: *NIPS*, v. 3, p. 5, 2007.
- [66] WILLIAMS, C. K., SEEGER, M. “Using the Nyström method to speed up kernel machines”. In: *Proceedings of the 13th International Conference on Neural Information Processing Systems*, pp. 661–667. MIT press, 2000.
- [67] HORNUNG, A., WURM, K. M., BENNEWITZ, M., et al. “OctoMap: An efficient probabilistic 3D mapping framework based on octrees”, *Autonomous Robots*, v. 34, n. 3, pp. 189–206, 2013.
- [68] PAGAC, D., NEBOT, E. M., DURRANT-WHYTE, H. “An Evidential Approach to Map-Bulding for Autonomous Vehicles”, *IEEE Transactions on Robotics and Automation*, v. 14, No 4, n. 4, pp. 623–629, 1998.
- [69] COIRAS, E., PETILLOT, Y., LANE, D. M. “Multiresolution 3-D reconstruction from side-scan sonar images.” *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, v. 16, n. 2, pp. 382–90, feb 2007. ISSN: 1057-7149.
- [70] TRIEBEL, R., PFAFF, P., BURGARD, W. “Multi-level surface maps for outdoor terrain mapping and loop closing”. In: *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pp. 2276–2282. IEEE, 2006.
- [71] RIVADENEYRA, C., CAMPBELL, M. “Probabilistic multi-level maps from LIDAR data”, *The International Journal of Robotics Research*, v. 30, n. 12, pp. 1508–1526, 2011.

- [72] SCHWENDNER, J. *Embodied Localisation and Mapping*. Tese de Doutorado, University of Bremen, 2013.
- [73] RIBAS, D., RIDAO, P., NEIRA, J., et al. “SLAM using an imaging sonar for partially structured underwater environments”, *IEEE International Conference on Intelligent Robots and Systems*, pp. 5040–5045, 2006. doi: 10.1109/IROS.2006.282532.
- [74] RIBAS, D., RIDAO, P., NEIRA, J. *Underwater SLAM for structured environments using an imaging sonar*, v. 65. Springer, 2010.
- [75] O’CALLAGHAN, S. T., RAMOS, F. T. “Gaussian process occupancy maps”, *The International Journal of Robotics Research*, v. 31, n. 1, pp. 42–62, 2012.
- [76] GAN, S. K., YANG, K., SUKKARIEH, S. “3d path planning for a rotary wing uav using a gaussian process occupancy map”. In: *Australasian Conference on Robotics and Automation (ACRA)*, 2009.
- [77] RAMOS, F., OTT, L. “Hilbert maps: scalable continuous occupancy mapping with stochastic gradient descent”, *The International Journal of Robotics Research*, v. 35, n. 14, pp. 1717–1730, 2016.
- [78] MORAVEC, H., ELFES, A. “High resolution maps from wide angle sonar”, *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, 1985. doi: 10.1109/ROBOT.1985.1087316. Disponível em: <[http://ieeexplore.ieee.org/xpls/abs/\\_all.jsp?arnumber=1087316](http://ieeexplore.ieee.org/xpls/abs/_all.jsp?arnumber=1087316)>.
- [79] SONG, L., FUKUMIZU, K., GRETTON, A. “Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models”, *IEEE Signal Processing Magazine*, v. 30, n. 4, pp. 98–111, 2013.
- [80] BOTTOU, L. “Stochastic gradient descent tricks”. In: *Neural networks: Tricks of the trade*, Springer, pp. 421–436, 2012.
- [81] LI, M., ZHANG, T., CHEN, Y., et al. “Efficient mini-batch training for stochastic optimization”. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 661–670. ACM, 2014.
- [82] TSURUOKA, Y., TSUJII, J., ANANIADO, S. “Stochastic gradient descent training for l1-regularized log-linear models with cumulative penalty”. In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the*

*ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pp. 477–485. Association for Computational Linguistics, 2009.