A heuristic approach to spatial audio using consumer loudspeaker systems

by

Dimme de Groot

In partial fulfilment of the requirements for the degree of

Master of Science
In Electrical Engineering

Supervisor:

dr. J. Martinez

Graduation Committee:

dr. O. E. Scharenborg
dr. J. Martinez
dr. ir. R. C. Hendriks
dr. ir. A. Noroozi

June 2023

Delft University Of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Electrical Engineering Programme

TU Delft

kien

Preface

The work presented in this thesis investigates the creation of virtual sound sources in a room equipped with a limited number of loudspeakers. This limited number of loudspeakers is typical for consumer loudspeaker systems. Ideally, these systems can provide a listening experience in which localisation cues are completely present. However, in current systems, this is not the case. The limited number of loudspeakers make it impractical to produce a physically accurate sound field. A possible solution is to create a perceptually accurate sound field instead. In this work, a step towards an algorithm which can do so is presented.

The developed algorithm requires knowledge of the listener placement, the room, and the loudspeaker placement. Spatial weighting is used to construct a region in which the acoustic energy should be large and a region in which the acoustic energy should be limited. The loudspeaker playback signals are obtained by maximising the energy ratio between these regions, while at the same time ensuring that the perceptual difference between the received audio and reference audio remains limited. The algorithm employs a convex optimisation problem to facilitate efficiently solving for the playback signals. Concretely, six convex optimisation problems are proposed with somewhat increasing complexity and different weighting matrices.

For each of the optimisation problems, the proposed algorithm is compared against a simple amplitude panning algorithm and a nearest neighbour algorithm. It is found that none of the considered algorithms is clearly preferred over the others in terms of the considered evaluation metrics.

A major limitation of the presented work is that the evaluation metrics do not explicitly test for localisation accuracy. In future work, this should be investigated by including subjective tests.

~

The presented work was carried out in collaboration with KIEN, a company developing and producing loudspeaker systems. I want to thank my colleagues at KIEN, Arash, Nick, Florent, Echo and Filip and my supervisor Jorge for their support and the helpful discussions. In particular, I want to thank Richard Eveleens, with whom I had many discussions (not necessarily about the thesis itself).

Contents

List of notation and definitions

Some of the notation, symbols and definitions used in the thesis are given below.

Notation

$\mathbb{R}$	set of real numbers	$\|\|x\|\|_p$	p-norm
$\mathbb{C}$	set of complex numbers	x^*	complex conjugate of x
$\mathbb{Z}$	set of integers	$\mathbf{x}$	column-vector
$t \in \mathbb{R}$	time	$\mathbf{X}$	matrix
$n \in \mathbb{Z}$	discrete-time	$\mathbf{X}^{\mathrm{T}}$	transpose of $\mathbf{X}$
$f \in \mathbb{R}$	frequency	$\mathbf{X}^{\mathrm{H}}$	Hermitian transpose of $\mathbf{X}$
$k \in \mathbb{Z}$	discrete-frequency	$\mathbf{I}_N$	the $N \times N$ identity matrix
$\omega = 2\pi f$	angular frequency	$\mathrm{diag}(\mathbf{x})$	matrix with $\mathbf{x}$ as diagonal
$x(t)$	continuous-time signal	X	Random variable
$x(n)$	discrete-time signal	$\mathbb{E}(X)$	Expected value of X
$\hat{x}(\omega)$	continuous-frequency signal	δ	delta function
$\hat{x}(k)$	discrete-frequency signal	$x * h$	convolution of x with h
$\mathcal{F}$	Fourier transform	sinc	sinc function
$\mathcal{F}^{-1}$	inverse Fourier transform	$\mathrm{supp}(x)$	support of x

Symbols

$\mathbf{x}_r = (x_r, y_r, z_r)$	receiver coordinate
$\mathbf{x}_i = (x_i, y_i, z_i)$	coordinate of loudspeaker i
$\mathbf{x}_h = (x_h, y_h, z_h)$	expected location of center of listeners head
N_s	number of physical loudspeakers
$i \in \{1, \ldots, N_s\}$	physical loudspeakers
$i = 0$	virtual loudspeaker
$s(\mathbf{x}_i, t),\ i \neq 0$	playback signal of loudspeaker i

$s(\mathbf{x}_0, t)$	acoustic reference signal
$s_r(\mathbf{x}_i, t)$	acoustic signal received at coordinate $\mathbf{x}_r$ due to playing back $s(\mathbf{x}_i, t)$
$s_r(t)$	total received acoustic signal at coordinate $\mathbf{x}_r$
$\mathcal{A}$	the region in which the acoustic energy should be high
$\mathcal{B}$	the region in which the acoustic energy should be low

Definitions

The continuous-time Fourier transform of a signal $x(t)$ is defined as [1]

$$\hat{x}(\omega) = (\mathcal{F}x)(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt, \tag{1}$$

with corresponding inverse Fourier transform [1]

$$x(t) = \left(\mathcal{F}^{-1}\hat{x}\right)(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{x}(\omega)e^{j\omega t}d\omega. \tag{2}$$

The discrete Fourier transform for a signal $x(n)$ with $n \in \{0, 1, \ldots, N-1\}$ is defined as [1]

$$\hat{x}(k) = (\mathcal{F}x)(k) = \sum_{n=0}^{N-1} x(n)e^{-2\pi jkn/N}, \quad k \in \{0, \ldots, N-1\}, \tag{3}$$

with corresponding inverse discrete Fourier transform is defined as [1]

$$x(n) = \left(\mathcal{F}^{-1}\hat{x}\right)(n) = \frac{1}{N}\sum_{k=0}^{N-1} \hat{x}(k)e^{2\pi jkn/N}, \quad n \in \{0, \ldots, N-1\}. \tag{4}$$

The l_p-norm of a discrete-time signal $x(n)$ with $n \in \{0, 1, \ldots, N-1\}$ is defined as [2]

$$||x||_p = \left(\sum_{n=0}^{N-1} |x(n)|^p\right)^{1/p}, \quad p \geq 1. \tag{5}$$

Convolution between two continuous-time signals $x(t)$ and $h(t)$ defined as [1]

$$(h * x)(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau. \tag{6}$$

Correspondingly, discrete-time convolution of two signals $x(n)$ and $h(n)$ is defined as [1]

$$(h * x)(n) = \sum_{m=-\infty}^{\infty} h(m)x(n-m). \tag{7}$$

Convolution is commutative,

$$x * h = h * x, \tag{8}$$

associative,

$$(x * g) * h = x * (g * h), \tag{9}$$

and distributive

$$x * (h + g) = x * h + x * g. \tag{10}$$

A continuous-time signal $x(t)$ is said to be causal if

$$x(t) = 0 \quad \forall\, t \leq 0, \tag{11}$$

a similar definition holds for discrete-time signals.

The support of a discrete-time signal $x(n)$ with $n \in \mathcal{N}$ (where $\mathcal{N}$ is an arbitrary set of integers) is defined as

$$\text{supp}(x) = \{n \in \mathcal{N} | x(n) \neq 0)\}. \tag{12}$$

The Dirac delta function (which is not a function but a distribution) is denoted by $\delta(x)$ and has the following two properties [1, 3]

$$\delta(x) = 0, \quad t \neq 0 \tag{13}$$

and

$$\int_{-\infty}^{\infty} \delta(x')f(x')dx' = f(0). \tag{14}$$

A different δ function is the Kronecker delta, which is also denoted by $\delta(x)$. It is defined as [4]

$$\delta(m - n) = \begin{cases} 0, & \text{if } m \neq n, \\ 1, & \text{if } m = n. \end{cases} \tag{15}$$

Since the notation of the Dirac delta function and Kronecker delta is ambiguous, it is mentioned which one is used.

The sinc function is defined as

$$\text{sinc}(x) = \begin{cases} \frac{\sin(x)}{x}, & \text{if } x \neq 0, \\ 1, & \text{if } x = 0. \end{cases} \tag{16}$$

The diagonal operator diag constructs an $N \times N$ diagonal matrix from an $N \times 1$ vector so that the vector is on the diagonal of the constructed matrix. That is, consider a vector $\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix}^{\text{T}}$, where T denotes the transpose. We have

$$\text{diag}(\mathbf{y}) = \begin{bmatrix} y_1 & 0 & \cdots & 0 \\ 0 & y_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & y_N \end{bmatrix}. \tag{17}$$

The Hermitian transpose of a matrix $\mathbf{A}$ is denoted $\mathbf{A}^{\text{H}}$ and is obtained by transposing $\mathbf{A}$ and taking the complex conjugate of each entry.

The matrix $\mathbf{A}$ is said to be Hermitian if

$$\mathbf{A} = \mathbf{A}^{\text{H}}. \tag{18}$$

A Hermitian matrix $\mathbf{A} \in \mathbb{C}^{N \times N}$ is said to be positive-definite if [5]

$$\mathbf{x}^{\text{H}}\mathbf{A}\mathbf{x} > 0 \quad \forall\, \mathbf{x} \in \mathbb{C}^N,\ \mathbf{x} \neq \mathbf{0}, \tag{19}$$

where $\mathbf{0}$ is the all zero vector of proper size. Similarly, the Hermitian matrix $\mathbf{A}$ is said to be positive-semi-definite if [5]

$$\mathbf{x}^{\text{H}}\mathbf{A}\mathbf{x} \geq 0 \quad \forall\, \mathbf{x} \in \mathbb{C}^N. \tag{20}$$

Consider a continuous random variable X with probability density function (pdf) p_X. The expected value is given by [6]

$$\mathbb{E}(X) = \int_{-\infty}^{\infty} x p_X(x) dx. \tag{21}$$

Similarly, the expected value of $f(X)$ is given by [6]

$$\mathbb{E}(f(X)) = \int_{-\infty}^{\infty} f(x) p_X(x) dx. \tag{22}$$

Of particular interest is the scenario in which we have access to some frequency-domain vector $\mathbf{h}(f)$. In this case, we can define the covariance matrix or power-spectral density matrix $\mathbf{R}$, given by

$$\mathbf{R} = \mathbb{E}(\mathbf{h}\mathbf{h}^{\mathrm{H}}). \tag{23}$$

This matrix is obtained by computing the expected value of the individual matrix entries.

Chapter 1
Introduction

Methods of reproducing spatial audio (i.e. audio containing the proper spatial cues) have received research interest for decades and have found their way to many households [7, 8]. Some well-known examples are stereo and 5.1 surround sound. In stereo, two loudspeakers[1] are employed, while in a 5.1 system five loudspeakers and an optional low-frequency loudspeaker are used [9]. As a rule of thumb, one could say that a larger number of loudspeakers allows for a better spatial reproduction. If this is taken to the limit, one arrives at a collection of techniques known as Sound Field Synthesis (SFS). A specific example of an SFS method is wavefield synthesis (WFS). WFS allows for reproducing a *physically* accurate sound field inside a domain of interest. A disadvantage of WFS is the large amount of loudspeakers required. For example, to obtain an accurate reproduction up to 20 kHz, a loudspeaker spacing of less then 1 cm is necessary [10]. This large number of loudspeakers involved is typical for physically accurate reproduction methods and makes them infeasible for consumers [10].

To still obtain an accurate spatial audio reproduction using systems which are viable for consumers, one can aim for a *perceptually* accurate reproduction instead of a physically accurate reproduction. This introduces some additional freedom, since any physically accurate reproduction must be perceptually accurate, but not necessarily the other way around. Here, "perceptually accurate" relates to the mapping of the received sound to a sound scene. The sound reproduction can thus be considered perceptually accurate if the mapping of sound to points in space is in reasonable agreement to the mapping obtained from a physically correct reproduction.

Throughout this thesis, I consider audio systems which are viable for consumers. Concretely, I consider systems with five full-range loudspeakers (i.e. the entire audible spectrum). Since this is an insufficient number for creating a physically accurate reproduction, the aim is to deliver a perceptually accurate reproduction instead. To achieve this, a heuristic approach is taken which takes into account some properties of the human hearing system.

In the remainder of this chapter, I first discuss some current approaches to spatial audio reproduction and their limitations. From this, the research questions follow. They are given in Section 1.2. Then, in Section 1.3, I introduce the approach to spatial audio taken in this thesis. After this, in Section 1.4, the methodology is given. Lastly, in Section 1.5, the thesis structure is given.

1.1 Current approaches to spatial audio

In this section, some methods that can be used to reproduce spatial audio are mentioned. I do not go into the details, but instead merely state their advantages and limitations. For more information, the reader can refer to the cited sources.

[1] Strictly speaking, I should say channels instead of loudspeakers. One could connect multiple loudspeakers to the same channel.

1.1.1 Stereophony

Perhaps the most well known approach to spatial audio is stereophony. Stereophony systems make use of level differences and time-of-arrival differences (see Section 2.1.2) to create spatial sound [10]. This is known as panning. Examples of stereophony systems are, among others, stereo systems, 5.1 systems and 7.1 systems [8, 10]. All these systems require specific loudspeaker locations, making them sensitive to misplacement.

The stereo and 5.1 layout are defined by the ITU-R BS.775 standard [9]. It is important to note that this standard only specifies the placement of the loudspeakers. The panning algorithms which determine the playback signals of the individual loudspeakers are not defined.

A disadvantage of stereophony systems is that they have a limited "sweet spot". The sweet spot is the region in space where the audio is perceived as intended. Additionally, it should be noted that the 5.1 format is not designed specifically to provide good localisation cues [7], nor is it designed to reconstruct the sound field. Still, stereophony systems have been very successful. This can largely be attributed to properties of the human hearing system [10].

The 5.1 surround setup, depicted in Figure 1.1, is used as reference throughout this thesis. Since the 5.1 system is only corresponds to a loudspeaker placement, a reference algorithm is implemented. This algorithm is a simple amplitude panning algorithm proposed in [11].

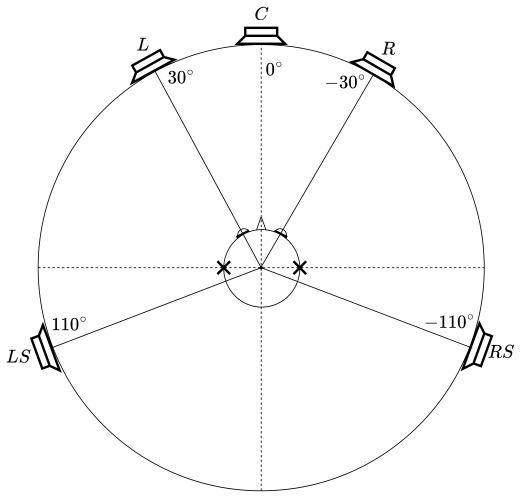


Figure 1.1: The 5.1 surround sound speaker layout according to the ITU-R BS.775 recommendation [9]. The loudspeakers should be placed at approximately the indicated angles. The system consist of three loudspeakers in front of the listener (Left L, right R and center C) and two loudspeakers on the sides of the listener (left side LS and right side RS). Lastly, an additional low-frequency effects loudspeakers may be added (this is the ".1" in the name 5.1). The radius of the circle is not defined.

1.1.2 Crosstalk cancellation

An approach to spatial audio which may be considered as mimicking headphone signals using loudspeakers is crosstalk cancellation. When performing crosstalk cancellation, one attempts to set the desired signal at a number of points in space [12]. A typical example of these points are the ears of the listener. This situation is depicted in Figure 1.2, where the aim is to cancel the

path from the right loudspeaker to the left ear and the other way around. This allows to set the audio received by the left and right ear independently.

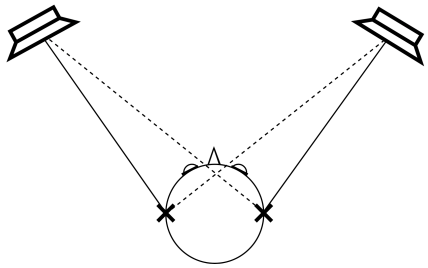


Figure 1.2: A typical example of the setup of the crosstalk cancellation problem. The aim is to deliver the signal of the left loudspeaker only to the left ear, while at the same time delivering the signal of the right loudspeaker only to the right ear. Thus, the dashed paths should be cancelled.

Historically, crosstalk cancellation considered the stereo loudspeaker setup, given by the left and right loudspeaker of Figure 1.1 [13, 14]. Limitations of this setup resulted in different arrangements, such as placing the loudspeakers closer together [14]. Still, systems using only two loudspeakers have a (very) small sweet spot, are sensitive to modelling errors (for example a small loudspeaker displacement) and the signal needs to be boosted massively around frequencies where the sound waves add destructively [12, 15, 16]. Solutions have been sought in regularisation [15, 16] and placing loudspeakers at discrete points in a semicircle around the listener [15]. More recently, line arrays [17–21] and listener-position adaptive crosstalk-cancellation [20, 22] have been considered as well. Additionally, some hybrid methods have been investigated [20, 23]. However, all of the previously mentioned approaches still suffer from a limited sweet spot, require a large number of loudspeakers or require a specific loudspeaker placement. Lastly, to incorporate the spatial cues, the use of Head Related Transfer Functions (HRTFs) or something similar is needed. HRTFs are listener dependent and described further in Section 2.1.2.

1.1.3 Wavefield synthesis

As stated before, wavefield synthesis (WFS) attemps to create a physically accurate soundfield. WFS is based on an integral known as Rayleigh's first integral equation [4] and the constructed soundfield is accurate up to a certain frequency over a large area [10]. The number of channels involved in WFS can easily surpass hundred, making it infeasible for consumer systems [4, 10]. However, it has been shown that even with loudspeaker spacing of 0.41 cm, decent localisation results can be achieved [24]. Typically, WFS assumes anechoic rooms, though it is possible to compensate for reflections [10].

1.1.4 Ambisonics

Just like WFS, ambisonics can be considered as a technique attempting to reconstruct a physically accurate soundfield. Ambisonics may be divided in three types, namely (1) first-order ambisonics (FOA), (2) higher order ambisonics (HOA) and (3) near-field compensated higher-order ambisonics (NFC-HOA) [4, 10, 25]. Here, the word "order" stems from the truncation of some mathematical series, see [4, 10, 26].

FOA can be used well with with only four loudspeakers in the azimuthal plane [25]. However, it suffers from a small sweet spot and limited directional resolution [7, 25]. HOA alleviates these

problems. However, this is at the cost of an increasing number of loudspeakers [26]. Typically, the amount of loudspeakers go beyond what I consider to be consumer viable.

In ambisonics, the radius of the sweet spot is a function of the wavelength and the number of loudspeakers. Namely, the size of the sweet spot for a fixed frequency increases for a larger number of loudspeakers. For a fixed number of loudspeakers, the size of the sweet spot decreases as the frequency increases [10]. The main difference between HOA and NFC-HOA is that, in HOA, the sources are assumed to be at infinite distance. Thus, they radiate plane-waves. On the other hand, in NFC-HOA, the sources are assumed to be at finite distance. Accordingly, they are treated as monopoles [4].

In [27], a localisation experiment using twelve loudspeakers is described. The experiment was performed for first order ambisonics, third order ambisonics and fifth order ambisonics. It was found that the localisation accuracy of subjects increases for higher orders. This is both the case for listeners within and outside of the sweet spot. The localisation performance of listeners inside the sweet spot was found to be better than that of listeners outside the sweet spot.

1.2 Research questions

From Section 1.1, it follows that SFS methods are infeasible for the intended users. Additionally, surround sound systems and crosstalk cancellation systems are sensitive to small system deviations, such as a listener outside the sweet spot and (somewhat) misplaced loudspeakers. Thus, I use a different beamforming approach. Concretely, I propose an algorithm to minimise acoustic energy inside part of a region close to the listener, while maximising the acoustic energy in the remainder of the region. Additionally, it is investigated if the performance of the algorithm can be increased by employing properties of the human hearing system. Thus, the research questions are

Research Question 1 *Can spatial weighting be used in combination with beamforming to synthesise audio containing spatial cues?*

Research Question 2 *Can properties of the human hearing be used to improve the performance of the algorithm?*

Here, performance relates to (1) robustness against system deviations, (2) the inclusion of spatial cues and (3) the quality of the obtained audio.

Additionally, I will limit the research by making the following assumptions

Assumption 1 *The number of loudspeakers which is used is low. In this thesis, I use five loudspeakers.*

Assumption 2 *The used loudspeakers are isotropic and full-range. That is, they radiate equal power in all directions and cover the entire audible spectrum (20 Hz to 20 kHz).*

Assumption 3 *Only a single isotropic virtual source needs to be placed. This virtual source needs to be placed inside the room.*

Assumption 4 *The location of the listener is known up to a certain accuracy. The listener is not moving.*

Assumption 5 *The virtual source, listener and loudspeakers are placed at the same height in an empty room with a fully absorbing floor and ceiling (e.g., no sound is reflected from the floor and ceiling).*

The reason for the above assumptions is to reduce (computational) complexity and to keep the research focused. Some of these assumptions are relaxed when investigating the performance of the algorithm. Similarly, some of the assumptions will only be considered when necessary. Lastly, it should be noted that more assumptions are made throughout the text. The above assumptions are, however, considered to be most relevant for constraining the problem.

As particular usecase, I consider a 5.0 setup (i.e. a 5.1 setup but without the low-frequency effects channel) which synthesises a virtual source on the circle intersecting the individual loudspeakers.

1.3 Outline of idea

As stated before, the approach to spatial audio taken throughout this thesis is that of spatial rejection. Concretely, it is attempted to maximise the acoustic energy inside the region from which the audio should be perceived, while minimising the acoustic energy in the region from which the audio should *not* be perceived. This should happen under the constraint that the received audio resembles the reference audio sufficiently well. The corresponding problem setup is illustrated in Figure 1.3. In the figure, the audio should be perceived as if coming from the gray loudspeaker.

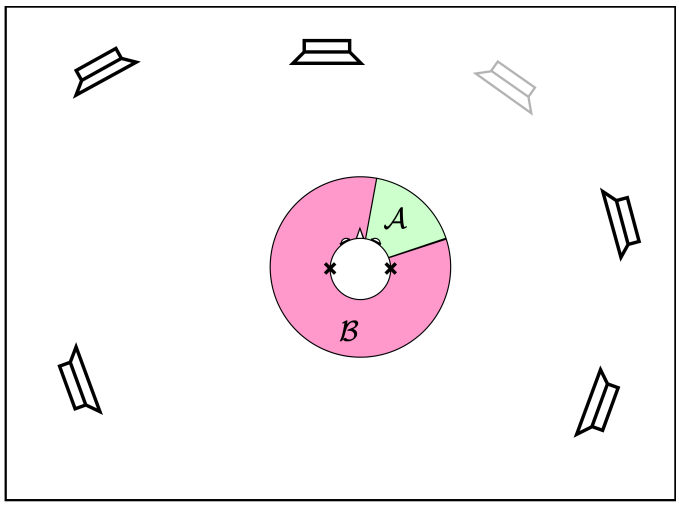


Figure 1.3: An example for the two regions, the virtual source (the loudspeaker drawn in gray) and the physical loudspeakers. The acoustic energy in region $\mathcal{A}$ should be maximised, while the acoustic energy in region $\mathcal{B}$ should be minimised. Given appropriate distance cues, this should make the user perceive the audio as if coming from the virtual loudspeaker.

The reason for using regions instead of, say, points is that (1) regions allow for taking inaccuracies into account, for example loudspeaker displacement, (2) by superposition of sound waves, it is reasonable to assume that the received audio, and thus the estimated location, is given by a weighted sum of the audio at all points on some surface surrounding the listener, and (3) it arguably removes the need of HRTFs (HRTFs are discussed in Section 2.1.2).

To make the idea given above a bit more formal, let us consider the main structure of the algorithm. The algorithm operates on short-time segments. Considering segment number l. The reference audio, (i.e. the audio from the virtual source) is assumed known and given by $s_l(\mathbf{x}_0)$. Here, $\mathbf{x}_0$ equals the location of the virtual source. Similarly, the physical loudspeakers are located

at $\mathbf{x}_i$, with $i \in \{1, \ldots, N_s\}$ and N_s the number of loudspeakers. Their playback signals are given by $s_l(\mathbf{x}_i)$.

Let us now consider two functions d_1 and d_2. Here, d_1 gives the energy in region $\mathcal{B}$ compared to that in region $\mathcal{A}$, and d_2 quantifies the perceptual difference between two sounds. Thus, we can write the optimisation problem as

$$\begin{aligned} \min \quad & d_1(s_l(\mathbf{x}_1), \ldots, s_l(\mathbf{x}_{N_s})) \\ \text{subject to} \quad & d_2\left(\sum_{i=1}^{N_s} s_l(\mathbf{x}_i),\ s_l(\mathbf{x}_0)\right) \leq d_{\max}, \end{aligned} \tag{1.1}$$

where $d_{\max}$ quantifies the maximum allowable deviation from the reference audio.

It should be noted that the functions above are mostly illustrative and as such are not be explicitly considered in the chapters.

1.4 Methodology

The results described in this thesis were obtained using MATLAB R2022b on default settings. Room Impulse Responses (RIRs) were generated using the RIR generator of Habets [28] (the version of Januari 31, 2022). For solving convex optimisation problems, CVX (Version 2.2, Build 1148) was used [29] with MOSEK (version 9.1.9) as solver. Other then this, the settings were kept default.

1.5 Thesis structure

The thesis is divided in two main parts. Firstly, a literature review presenting background theory and secondly the proposed algorithm.

The background theory is given in Chapter 2. Firstly, in Section 2.1, the human auditory system is reviewed briefly. Then, in Section 2.2, a model for the received sound signal inside a room is discussed. After this, in Section 2.3, three objective perceptual models which predict if a tone is audible are given. Lastly, in Section 2.4, a method to perform filtering of long signals is discussed. It should be noted that none of the theory presented in Chapter 2 is my own work.

The proposed algorithm is based on the findings in the literature review and discussed in chapters 3 and 4. In Chapter 3, the regions $\mathcal{A}$ and $\mathcal{B}$ are defined using spatial weighting functions. In combination with the model of the received sound signal, this allows to compute Power Spectral Density (PSD) matrices. These matrices can then be used to minimise the energy ratio in region $\mathcal{B}$ compared to that in region $\mathcal{A}$. Then, in Chapter 4, the complete algorithm to synthesise sound including spatial cues is presented.

The results of the algorithm are discussed in Chapter 5. The conclusions are given in Chapter 6 and a discussion and some recommendations for future work are presented in Chapter 7.

Lastly, I want to highlight that, in Appendix J, a draft of the to-be-submitted paper "Stochastic Model of the Room Impulse Response in Small Rooms" is given.

In Figure 1.4, the structure of the thesis is depicted graphically. The background theory is depicted in green and the contribution of the thesis is depicted in orange. The draft of the paper is given in purple. Note that most appendices and some sections are not shown in the figure.

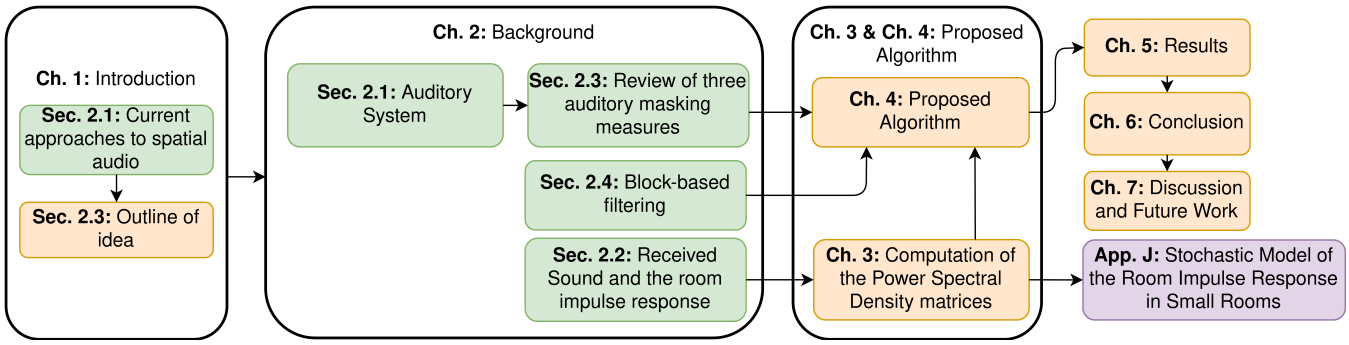


Figure 1.4: The main structure of the thesis. The green coloured sections and chapters are literature reviews. The orange sections are the main contributions of this thesis. The draft of the to-be submitted paper is shown in purple. Note that not all appendices and sections are shown.

Chapter 2
Background

In this chapter, some background theory is discussed. This chapter serves as a tool to understand the various components of the proposed algorithm. None of the theory discussed in this chapter is my own work.

Firstly, in Section 2.1, some basics of the physiology of the ear are introduced and localisation and masking are briefly explained. Then, in Section 2.2, the Green's function solutions to the wave- and Helmholtz equation are presented. These functions allow for calculating the received sound signal when transmitting a sound in the free field. They can also be used in a simple model to qualitatively calculate the room impulse response (RIR). After this, in Section 2.3, some objective masking models are discussed. These models aim to predict the outcome of subjective masking experiments. Lastly, in Section 2.4, a method to perform filtering in short segments is discussed. This is required, since the RIRs are too long to be used directly in real-time applications.

2.1 The auditory system

In this chapter, the human auditory system is discussed briefly. Since this chapter serves as a tool to understand the various aspects of the perceptual measure discussed in Section 2.3, it is definitely not all-encompassing. For more information, the reader can, among others, refer to [30, 31]. A very loose schematic of the ear depicting only the structures which are discussed in this chapter is given in Figure 2.1. For a more complete depiction, the reader can refer to [30].

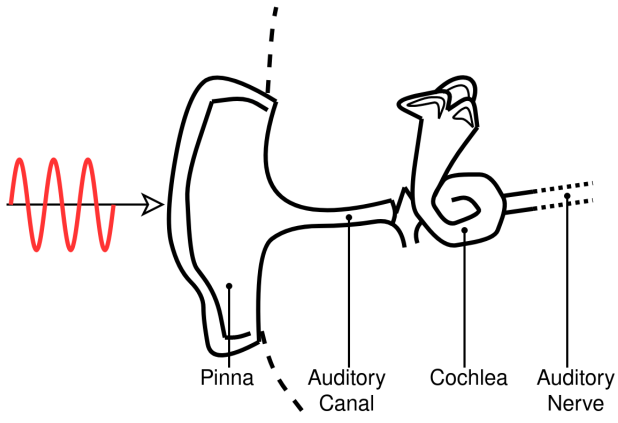


Figure 2.1: A loose schematic view of the human ear depicting only the structures discussed below.

2.1.1 Physiology of the auditory system

When a sound signal is transmitted, it eventually arrives at the listener, it is first modified by the listeners head, torso, and pinna. The latter is the visible part of the ear. For low frequencies, up to about 1500 Hz, the audio is mostly modified due to the listeners head and torso. For high frequencies, the modification can mainly be attributed to the pinna [30, 31]. The exact modification depends on the angle of incidence and is described by the so-called head related transfer function (HRTF). This transfer function is further described in Section 2.1.2.

After passing the pinna, the modified sound traverses the auditory canal. The properties of the auditory canal result in a high sensitivity for frequencies around 4 kHz. This high sensitivity can also be seen in the threshold in quiet described in Section 2.1.3 [31].

As the sound traverses further through the ear, it eventually arrives at the cochlea. While the cochlea is shaped like the shell of a snail, this is not believed to have any function apart from saving space [30]. Along the length of the cochlea lies the basilar membrane. The basilar membrane is about 32 mm long and its mechanical properties vary along the length of the membrane [30]. The basilar membrane is set into motion by incoming sounds, where the pattern of the motion depends on the spectrum of the incoming sound. For sinusoids, the peak of the envelope moves along the membrane for decreasing frequency. Thus the cochlea is often considered to perform a frequency-analysis. It should be noted that the magnitude of the envelope is nonlinear with respect to the magnitude of the input signal. Namely, a wide range of input values is mapped to a smaller range of output values, this is termed "compressive nonlinearity" [30].

The motion of the basilar membrane is translated to electrical signals through the organ of Corti, lying on top of the basilar membrane [31]. Among others, the organ of Corti contains the inner hair cells and the outer hair cells. The inner hair cells are mainly responsible for the translation of the acoustic signal to electrical signal (hereafter referred to as haircell transduction), while the outer hair cells partake in some feedback process modifying the mechanical properties of the cochlea [31, 32]. Among others, the compressive nonlinearity is associated with this feedback process [33].

The electrical signals are transmitted towards the central nervous system through neurons within the auditory nerve. Different neurons are sensitive to different stimuli types (think of, for example, a low-frequency or a high-frequency stimuli). The information is conveyed through a nerve by means of electrical pulses. The firing rate (amount of pulses per unit of time) inside a neuron depends on the level of the input signal. Typically, a neuron starts firing once a certain threshold is reached. After this, the firing rate increases for an increasing stimuli level. This keeps going until the neuron is saturated and cannot fire any faster [30].

For low-frequency sinusoidal input signals, the pulses are synchronised to the input signal: the time between each spike approximately equals some integer multiple of the period of the waveform [30]. This is termed phase locking and is, among others, believed to play a role in localisation and pitch perception [30]. Phase-locking does not only happen to sinusoidal input signals, but also to the envelope of more complex input signals [34]. To the best of my knowledge, it it is not exactly known how the envelope extraction works, though it is believed that distortions due to nonlinearities play an important role [35].

2.1.2 Localisation

In this section, localisation is discussed briefly. Localisation refers to our ability to localise the origin of a sound. This is typically done by comparing the sound received by the left and right ear, though monaural cues also play a role. In the following, I will first consider the ears as two omnidirectional microphones. This gives rise to the Interaural Time Difference (ITD) and the Interaural Level Difference (ILD). Then, the head and body are introduced, giving rise to the

Head Related Impulse Response (HRIR). Lastly, I briefly consider distance estimation and the precedence effect.

2.1.2.1 Time- and level-difference

Let us consider the free-field scenario depicted in Figure 2.2. Since the distances d_L and d_R (to the left ear (L) and right ear (R), respectively) are different, the signal received by each ear has a different delay and attenuation.

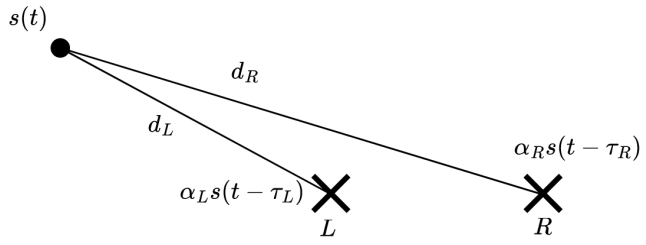
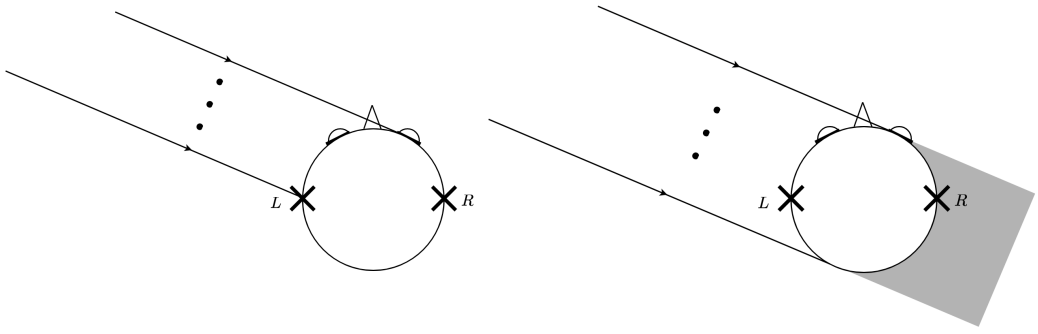


Figure 2.2: A schematic overview of interaural time- and level-differences. The ears (which for now are considered to be omnidirectional microphones) are depicted by $\times$, while the (omnidirectional) loudspeaker is depicted by $\bullet$.

The delay τ and attenuation α depend on the travel distance d. By comparing these values, an estimate of the angle of incidence can be made. The calculation of τ and α is discussed in Section 2.2. The time-difference between the ears is termed the Interaural Time Difference (ITD), while the level-difference is termed the Interaural level Difference (ILD) [30]. For sources which are far away, the level difference introduced in the free-field (so without a listener) is very small.

The presence of a listener influences the ITD and ILD. To explain this, I consider sources which are in the far field. These sources can be approximated as a plane wave. The ITD and ILD are best explained by considering the situation for low frequency signals and high frequency signals separately. These situations are respectively illustrated in Figure 2.3a and Figure 2.3b. For low frequencies, the wavelength is a lot larger then the size of the head (which has a radius



(a) Low frequency (b) High frequency

Figure 2.3: A schematic view of how the ITD and ILD are influenced by the presence of a head. For low frequencies, the sound wave diffracts around the head, resulting in an ITD. For high frequencies, the head "shadows" the ear, resulting in an ILD.

of about 9 cm), resulting in the wave diffracting around the head [30]. Due to this diffraction and the fact that the source is in the far field, the ILD is negligible, and the ITD is the dominant spatial cue. For high frequencies, on the other hand, the ITD is not as useful becuase multiple frequencies might result in the same ITD. The ILD is, however, useful. The reason for this is that the head obstructs the sound signal, thereby resulting in an ILD. This is called shadowing.

While ITDs are not useful for localising high frequency sinusoidal sources, they can provide localisation cues for more complex high frequency signals. This can be done by comparing the time differences across multiple frequencies. For example, suppose that we have a single signal composed of three different frequencies. Each frequency gives rise to a set of possible time differences. By comparing the possible time differences arising from each of these frequencies, an ITD might be found [30]. The auditory system is also able to estimate ITDs from the envelope of complex signals [36, 37].

Let us briefly consider the free-field situation again. The ITD and ILD do not give a unique estimate of the angle of incidence. Consider, for example, the situation depicted in Figure 2.4. Here, both loudspeaker 0 and loudspeaker 1 result in the same time- and level-differences. In a three-dimensional situation, the positions which give rise to the same ILDs and ITDs trace out a cone. This cone is referred to as the "cone of confusion" [30].

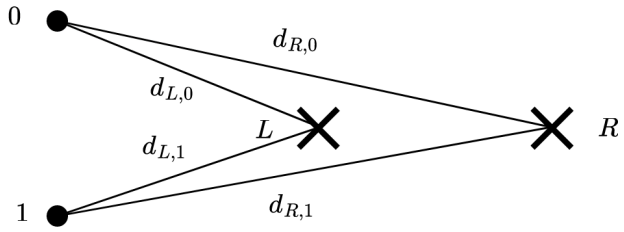


Figure 2.4: A schematic overview of interaural time- and level-differences. The ears (which for now are considered to be omnidirectional microphones) are depicted by $\times$, while the (omnidirectional) loudspeakers are depicted by $\bullet$. In this example, the distances $d_{L,0} = d_{L,1}$ and $d_{R,0} = d_{R,1}$, thereby resulting in ambiguity.

The ambiguity in angle of incidence can be resolved through head movements [30]. However, even without head movements, humans are very good at estimating the angle of incidence. This can be attributed to us having a torso, head, and ears. Together, they act as antennas with an angle, and, to a lesser extent, distance dependent transfer function [30]. This angle dependent transfer function is given by the HRIR. The HRIR is sometimes considered in the frequency domain, where it is termed the Head Related Transfer Function (HRTF).

2.1.2.2 Head related transfer function

For each angle of incidence with which a sound wave arrives at the listener (for example an azimuth θ and an elevation ϕ), there is a unique pair of HRTFs; one for the left ear, and one for the right ear. The HRTF is considered to be distance independent for sources which are further than about 1 m from the head [38–40].

Each pair of HRTFs contains spectral patterns by which the incoming sound is modified, thereby introducing ITDs and ILDs. For frequencies above 6 kHz, this modification can mainly be attributed to the pinna. For lower frequencies, the head and the torso also play a role [30].

Mapping an HRTF to the angle of incidence can be thought of as a simple look-up table [38]. However, the HRTF is not straightforwardly extracted from the received sound signal. The

reason for this is that any source-signal has a spectral pattern by itself. This spectral pattern is then modified by both the HRTF and the environment. As might thus be expected, it has been shown that the localisation works best if we are familiar with the to-be-localised sound and the current environment [30]. However, since the spectral dips and peaks of HRTFs are sharper than typical sound spectra and since the received signal can be compared across the ears, localisation still works to some extent even without prior knowledge [30].

It should be noted that the HRTF is person-dependent. Experiments have shown that subjects are able to estimate the azimuth, and in particular whether the source is on the left or on the right, when using someone else their HRTF. This does not hold for perception of elevation, where large errors occur [38, 41].

In any practical application, the individuality of the HRTF poses a large challenge. While one would ideally measure each listener, this is impractical due to the time-consuming measurement process [42]. Thus, other methods have been developed. For example, one could try to estimate the shape of the users pinna and find the closest fit in an HRTF database [38]. Attempts have also been made to synthesise HRTFs via machine learning [43], or to generate them from pictures of the users ears [44]. Some examples of HRTF databases are [45–48].

2.1.2.3 Distance perception

While the ITD, ILD and HRTFs provide means to estimate the distance of near-field sources, they do not provide reliable cues to estimate the distance of far-field sources. In this section, a number of cues to obtain distance-estimates for far-field sources are given. Before doing so, it should be noted that distance estimation is inaccurate compared to estimates of the angle of incidence. The quality of the estimate also depends on the type of signal [30, 41].

Firstly, if sounds are familiar, the hearing system is able to estimate the distance by comparing the sound level to the reference sound level [30, 41]. Similarly, for moving sources, a cue is provided by the changes in sound level [30]. For sound sources which are sufficiently far away from the listener, the low frequency content is attenuated less with respect to the high frequency content, thereby also providing a distance cue [30]. When listening in rooms, the distance between the listener and source influences the ratio between the direct sound and reflected sound. This ratio is frequency dependent and can be used to estimate the distance [30].

2.1.2.4 The precedence effect

As is discussed in the previous section, reflections provide a cue to estimate the distance of a sound source. However, one might also expect that they influence the estimated angle. To see this, consider the example of Figure 2.5. For the example depicted in the figure, one might expect

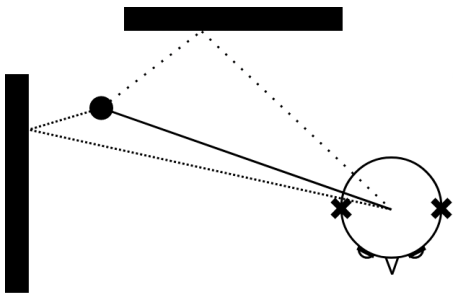


Figure 2.5: A simple example of an environment where the listener not only receives the direct path signal, but also receives some reflections of the same signal.

that three sources are localised. The actual source, and the "virtual" sources localised on the edge of the walls. In practice, however, it is likely that the listener only localises a single source, which is the actual source. This can be attributed to the precedence effect (also known as the Haas effect or the law of the first wavefront) [49].

How the presence of reflections exactly influence the localisation depends on the nature of the reflection. For a reflection which arrives in close succession to the direct path signal (less then 1 ms), the location is some average of the reflection and the direct path signal [30]. A set of reflections is perceived as a single sound event when they arrive within T ms of the direct path. Here, T depends on the type of signal and can be up to 40 ms [30, 49]. It should be noted that this does not imply that the reflections are masked. The precedence effect merely states that the localisation happens based on the first arriving signal mostly [30]. Interestingly, the precedence effect still holds even if the reflection is stronger then the direct path [30, 49]. This breaks down when the reflection becomes a lot stronger (10 to 15 dB) [30]. Lastly, if a sufficiently strong reflection comes in after about 250 ms or more, it is perceived as an echo [49].

In all the previous discussions, the role of the sight on localisation was ignored. It should, however, be noted that sight largely influences localisation. This is known as the ventriloquism effect [49]. Since I am not able to control the visual content throughout this thesis, I do not take it into consideration.

2.1.3 Audibility of sounds

For a sound to be audible, it needs to be sufficiently loud. The term sufficiently loud is quite vague and depends on the listener, the type of sound, the duration for which the sound is present, and on other sounds which are present. In this section, I will first consider the case where no other sounds are present. For a more comprehensive review on audibility of sounds, the reader can refer to [50].

2.1.3.1 Theshold in quiet

When a disturbance is inaudible in the absence of any other sounds, it is below the threshold in quiet (also known as threshold of hearing). In edge-cases, the disturbance may be inaudible at first but becomes audible after it has been playing for a sufficiently long time. The ears are believed to "integrate" the energy of the disturbance, which results in the disturbance becoming audible after a while. This effect is termed temporal integration and it is effective for up to about 300 ms of signal duration [30, 51].

A typical threshold in quiet for a sinusoidal test signal in steady state and listeners with normal hearing is given in Figure 2.6 [31, 52]. Note that the magnitude is given in dB SPL (Sound Pressure Level). This is an often used measure and will be discussed in a bit. Any sinusoidal disturbance which lies below the threshold in quiet can be considered as inaudible, though it should be noted that the exact threshold in quiet varies from individual to individual.

The value dB SPL relates the sound pressure p (in pascal) to a reference sound pressure p_0, also in pascal. This is done according to

$$L_{\text{SPL}} = 20 \log_{10} \left(\frac{p}{p_0} \right) \quad \text{[dB SPL]}, \tag{2.1}$$

where the value $p_0 = 20$ µPa [53].

2.1.3.2 Masking

When two sounds are presented, it sometimes happens that one of the sounds is made inaudible by the presence of the other one. This phenomenon is called masking, and the audible sound

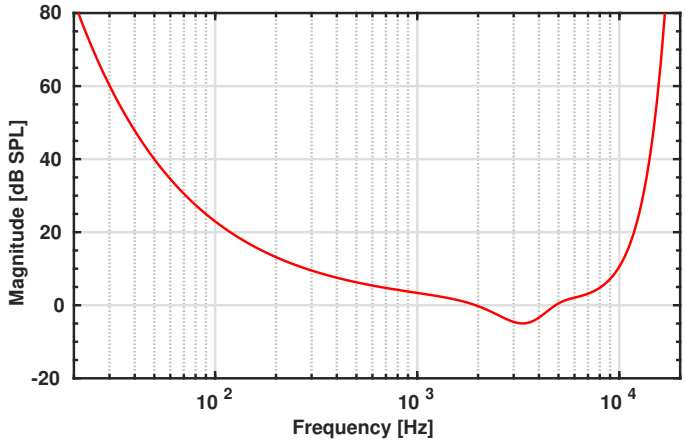


Figure 2.6: The threshold in quiet for listeners with normal hearing. Sinusoidal tones below this curve can be considered inaudible, while those above can be considered audible.

is referred to as the "masker". I will refer to the masked sound as the "disturbance". This is slightly misleading, as it does not have to be an actual disturbance. Three types of masking can be considered: backward masking, forward masking and simultaneous masking. In the former two, the masker respectively precedes and succeeds the disturbance, while in simultaneous masking, the masker is on during the whole period that the disturbance is also on [30].

The amount of backwards masking depends on how practised the subjects are and only lasts a few milliseconds [30]. For forward masking, on the other hand, the effect can last up to 100 to 200 ms after the masker was present. The amount of masking is stronger the closer the disturbance is placed in time to the masker [30, 31]. The effect of masking is stronger for a louder masker and/or a longer duration masker [30]. When the disturbance is on for a longer duration, it becomes easier to detect.

For both simultaneous and forward masking, the disturbance is most easily masked if the frequency components of the masker and the disturbance are close to each other. Thus, it is believed that masking at least partially relates to the frequency-analysis being performed on the basilar membrane, which is considered to act as a filterbank consisting of (infinitely many) filters tuned to different frequencies. These filters are referred to as the "auditory filters" [30]. The shape of these filters is signal-dependent and not necessarily symmetric on a linear frequency scale [30, 54, 55].

An example of a masking curve is given in Figure 2.7. Here, the masking curve is generated for a 50 dB SPL masker at 1 kHz. Note that the masking curve largely coincides with the threshold in quiet, except for frequencies close to the masking curve. The curve is generated using the Par-measure (see Section 2.3.2 or [51])

Another aspect closely related to masking is the just-noticeable level difference (JNDL). It is the amplitude change which listeners can (on average) notice. For example, for a 1 kHz tone at 70 dB SPL, the JNDL is about 0.5 to 1 dB SPL [51]. One can also consider JND's for different types of difference, such as frequency differences.

Finally, I want to explicitly state two aspects which implicitly follow from the above text but are worth to be repeated. Namely, in practice, the signal processing inside the cochlea is nonlinear [56]. Consequently, experimental results for a certain type of signal cannot straightforwardly be extrapolated to other types of signals. Also, models which agree with a wide range of experimental data need to be nonlinear. For a comparison of some cochlear models, the interested reader can

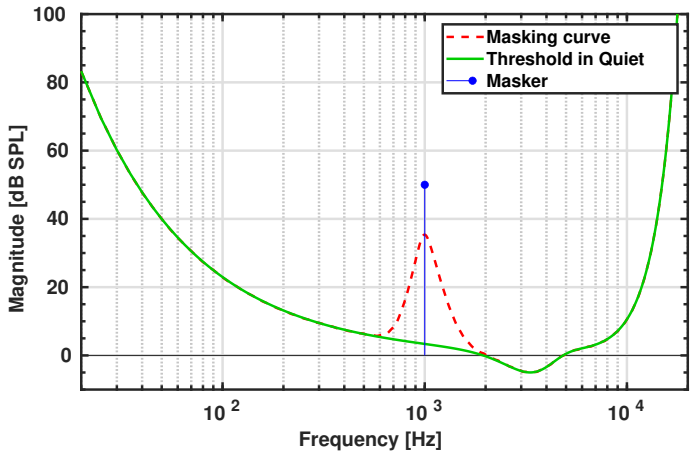


Figure 2.7: An example of a masking curve. The masking curve is given for a 50 dB SPL sinusoidal masker at 1 kHz, and is plotted against the threshold in quiet and the masker. The figure is generated using the Par-measure, see [51] or Section 2.3.2.

refer to [56]. Another note is that the performance of the auditory system varies from listener to listener. Experiments typically mention the "type" of listeners used.

In the upcoming section, a simple model for the received sound signal is discussed. After this, in Section 2.3, the discussion on masking is continued by considering three masking models which aim to predict the outcome of subjective experiments.

2.2 Received sound and the room impulse response

In this section, a model for the received sound signal and the room impulse response (RIR) are discussed briefly. I limit the discussion to two scenarios, namely the free-field and empty shoebox rooms. The latter can be efficiently modelled using the image-source method. For convenience, I consider a continuous-time domain representation throughout the whole section.

2.2.1 Received sound in the free-field

Let us consider a receiver at position $\mathbf{x}_r \in \mathbb{R}^3$ and transmitters at positions $\mathbf{x}_i \in \mathbb{R}^3$, with $i \in \{1, \ldots, N_s\}$ and N_s the number of transmitters. The channel from transmitter $\mathbf{x}_i$ to receiver $\mathbf{x}_r$ is denoted by $h(\mathbf{x}_i, \mathbf{x}_r, t)$. A simple example of this situation in which three loudspeakers are used is depicted in Figure 2.8.

To obtain $h(\mathbf{x}_i, \mathbf{x}_r, t)$, one needs to consider how sound travels through a homogeneous medium. For sound levels up to 160 dB SPL (which is above the threshold of pain for human hearing), this is described by the scalar wave equation [4]. The scalar wave equation is discussed in some more detail in Appendix A.

In the free-field with isotropic receivers and transmitters, the channels equal the Green's function solution $\hat{g}(\mathbf{x}_i, \mathbf{x}_r, \omega)$ to the Helmholtz equation. This equation is the frequency domain equivalent of the wave-equation, described in Appendix A. The Green's function is expressed as [4, 57, 58]

$$\hat{g}(\mathbf{x}_i, \mathbf{x}_r, \omega) = \frac{\exp\{-\frac{j\omega}{c}||\mathbf{x}_i - \mathbf{x}_r||_2\}}{4\pi||\mathbf{x}_i - \mathbf{x}_r||_2}, \tag{2.2}$$

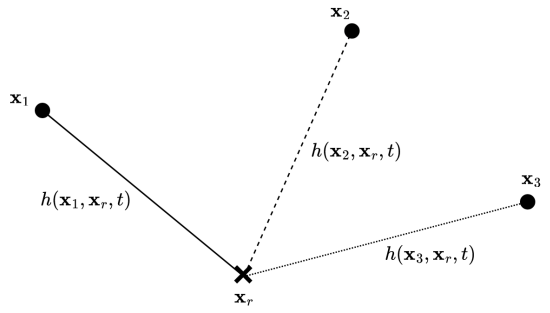


Figure 2.8: An example situation where the sound produced by three loudspeakers (depicted as $\bullet$) in the free field is recorded using an isotropic (depicted as $\times$). The channel from source i to receiver r is denoted as $h(\mathbf{x}_i, \mathbf{x}_r, t)$.

or, in the time domain [57, 59]

$$g(\mathbf{x}_i, \mathbf{x}_r, t) = \frac{1}{4\pi||\mathbf{x}_i - \mathbf{x}_r||_2}\delta\left(t - \frac{||\mathbf{x}_i - \mathbf{x}_r||_2}{c}\right). \tag{2.3}$$

Here, c is the speed of sound and δ the Dirac delta function. At room temperature with air as medium, $c \approx 343$ m/s [4].

Using (2.3) and by the notion that the free-field transfer function equals the Green's function, we may write the received sound $s_r(t)$ as

$$s_r(t) = \sum_{i=1}^{N_s} \left(h(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i)\right)(t) = \sum_{i=1}^{N_s} (g(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i))(t). \tag{2.4}$$

This equation can be extended to allow for directive transmitters and/or receivers. This is explained further in Appendix B.

2.2.2 The room impulse response

In the previous section, the received signal when the transmitter and the receiver are placed in the free-field was discussed. In this scenario, the channels $h(\mathbf{x}_i, \mathbf{x}_r, t)$ are given by the Green's function solution to the scalar wave-equation. In this section, the discussion is extended to incorporate a room. As was done in the previous section, I discuss an isotropic receiver and transmitter. Then, a simple model to estimate RIRs is discussed briefly. This model is straightforwardly extended to incorporate the directivity of the transmitter and receiver. This extension is not discussed here. Instead, the interested reader can refer to [60].

When a transmitter is placed in some environment, the transmitted sound waves will ultimately hit an object. At the boundary of the object, the wave will be partially reflected, partially absorbed (turned into heat or transmitted at the other side of the object), and partially scattered [61]. If one assumes a wall to be uniform and of infinite extent, a part of the incoming wave will be reflected, thereby only undergoing a phase change and an amplitude change. This change can be characterised by the reflection factor R of the wall. Parameter R is a complex valued scalar which depends on the angle of incidence and on the frequency of the incoming wave [61]. The portion of the incident wave which is reflected has the same outgoing angle as the incident angle of the incident wave, but is flipped along the normal to the wall. This is known as specular reflection

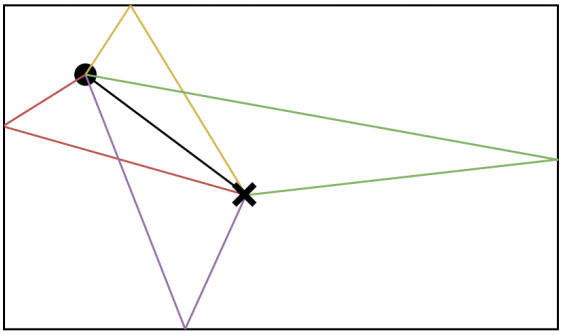


Figure 2.9: A simple example of the direct path and first order reflections in a rectangular room. The sound waves are assumed to behave like rays. The receiver is depicted with a $\times$, and the transmitter by a $\bullet$.

[61, 62]. A simple example of a room and the corresponding reflections is given in Figure 2.9. Here, the direct path and the first order reflections are depicted.

Multiple methods to simulate RIRs exist. These methods vary in how accurate they are and in their computational complexity. Obtaining accurate results over the whole frequency range of interest is typically computationally expensive [62, 63]. A simple approach which gives a qualitative description of the RIR can be obtained through "geometrical acoustics". In geometrical acoustics, the wavelengths are assumed to go to zero. Thus, the sound waves are replaced by sound rays. For empty rooms, the result obtained through geometrical acoustics is valid for wavelengths which are far smaller then the dimensions of the room [61].

A specific model using geometrical acoustics is the mirror image-source method for box-shaped rooms, introduced in [59] and explained in more detail in Appendix C. It was extended to allow for arbitrary polyhedra shaped rooms in [64]. In the mirror image-source method, the source and receiver are assumed to be isotropic and the reflections are considered to be the direct path of a new sound source. These new sound sources are called the image-sources. Since the image-sources are not actual sources, they must transmit the same audio signal as the original source. The image-source method allows to write the channel h as a sum of properly weighted Green's functions. This is briefly explained below.

Recall that the transmitter location is denoted by $\mathbf{x}_i$ and suppose that we consider N_i image-sources corresponding to this transmitter. The locations of the image-sources locations are given by $\mathbf{x}_{i,\xi}$, with $\xi \in \{1, \ldots, N_i\}$. For convenience, $\xi = 0$ is used to include the transmitter, so $\mathbf{x}_i = \mathbf{x}_{i,0}$. Thus, $\xi \in \{0, 1, \ldots, N_i\}$. Since the path from the image-source through the receiver goes through a number of walls, each image-source has a reflection coefficient associated with it. However, as stated before, the reflection factor R associated with each of the walls is complex and depends on the angle of incidence and frequency. The method proposed in [59] ignores this dependence and instead associates a real valued reflection coefficient β with each of the walls. Thus, each mirror-image ξ of transmitter i is attenuated by some factor $\beta_{i,\xi}$. Under this assumption, the channel h is obtained by summing the contribution of each image-source using the proper weight. This gives

$$h(\mathbf{x}_i, \mathbf{x}_r, t) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\delta\left(t - \frac{||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}{c}\right)}{4\pi||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}. \tag{2.5}$$

This equation approximates the exact solution to the wave-equation for box-shaped rooms well for sufficiently high frequencies [61]. The image-source method is described in more detail in

Appendix C. Here, also methods to calculate N_i, $\mathbf{x}_{i,\xi}$ and $\beta_{i,\xi}$ are given, Alternatively, the reader can refer to [28, 59]. Lastly, it should be noted that the length of the RIR in typical living rooms is about 300 ms [61]. This length is known as the T_{60} time and is discussed further in the appendix.

2.3 Review of three auditory masking measures

In this section, three auditory masking models are briefly discussed. These models can be used to take into account the masking of sounds in the proposed algorithm. I name each of the models after their first author. Concretely, I discuss the Dau-model [65–68], the Par-measure [51] and the Taal-measure [69]. The Dau-model is used to predict masking experiments with high accuracy, but lacks mathematical tractability. The remaining two are computationally inexpensive, so that they can be used in algorithms where computation times should be low. All of these models are used to predict the outcome of subjective masking experiments. The Dau-model has been modified to allow for intelligibility prediction of speech [70].

The Taal-measure and Par-measure are only discussed briefly here. For a more detailed discussion, the reader can refer to the corresponding papers ([69] and [51]) or to Appendix D.

2.3.1 The Dau-model

The Dau-model was originally proposed in [65, 66] and a year later extended in [67, 68]. It can accurately predict masking in a variety of conditions [71, 72].

The Dau-model consists of a preprocessing stage, modelling parts of the human ear, and a decision device, estimating whether or not the inputs of the model are perceptually different. The inputs of the model are (1) the masker and (2) the masker plus distortion. For both of these inputs, an "internal" representation is obtained by passing it through the preprocessing stage. In this stage, the input signals are first passed through a gammatone filterbank, modelling the basilar membrane. It should be noted that this is a linear filterbank. Then, half-wave rectification followed by lowpass filtering is performed. This simulates the envelope extraction of the human hearing system. The outputs of the envelope extraction stage are subsequently passed through an "adaptation" stage. In the adaptation stage, five feedback loops are employed to model the adaptive properties of the auditory system. The feedback loops allow for correctly predicting forward masking. The output of the adaptation stage is passed through another set of lowpass filters modelling temporal integration. Lastly, the internal representation is obtained by adding internal noise with fixed variance [67]. To decide whether or not the disturbance is masked, the internal representation of the masker and the internal representation of the masker plus disturbance is compared through a process which resembles matched filtering [65].

The Dau-model has inspired a number of other models. One of these is the CASP (computational auditory signal-processing and perception) model from [71]. Its structure and functionality is very similar to the Dau-model. However, the CASP-model incorporates an outer-and middle-ear transferfunction and the linear gammatone filterbank was replaced by the "dual-resonance nonlinear" (DRNL) filterbank fitted on human data (see [73, 74]). This model is, among others, able to simulate the compressive nonlinearity [71]. The CASP model has been modified to simulate some types of hearing impairment and to predict intelligibility of speech in respectively [75] and [76]. For a thorough and recent comparison of the Dau-model with other auditory models, the reader can refer to [77].

2.3.2 The Par-measure

In contrast to the Dau-model, the Par-measure is a simple model and is suitable for online optimisation. The Par-measure was developed for sinusoidal coding of audio, and thus aims to predict the masking curve for sinusoidal distortions [51]. It has, however, also been successfully applied in sound zones [78] and loudness increase [79]. Since the Par-measure is able to predict the masking curve, it is also able to predict if a listener can notice the difference between two signals x and $y = x + \epsilon$. The former can be considered as the reference signal or masker, while ϵ is the disturbance. The input signals are properly windowed short-time segments (about 20-40 ms) [51, 69].

The Par-measure converts the input signals x and ϵ to an "internal representation" I by passing them through a simple auditory model. The structure of this model is depicted in Figure 2.10.

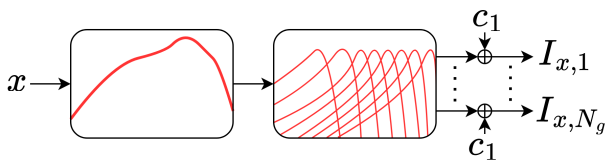


Figure 2.10: A schematic overview of the structure of the auditory model used in the Par-measure. From left to right, it consists of an outer and middle ear filter, a gammatone filterbank, and addition of internal noise. Figure based on [51].

The auditory model consist of an outer- and middle-ear filter h_{om} (modelled as the inverse of the threshold in quiet) and a gammatone filterbank consisting of N_g gammatone filters h_i modelling cochlear signal processing. At the output of the filters, a constant c_1 is added which models internal noise.

The power of the Par-measure lies in the fact that it can be expressed as a weighted l_2-norm, where the weighting g can be calculated independent of ϵ, Namely, it may be expressed as [51]

$$D_{\text{Par}}(x, \epsilon) = ||\hat{g}\hat{\epsilon}||_2^2. \tag{2.6}$$

Here, the weighting $\hat{g}$ is the inverse of the masking curve and given by [69]

$$\hat{g}^2 = c_2 \sum_{i=1}^{N_g} \frac{\hat{h}_{\text{om}}^2 \hat{h}_i^2}{\frac{1}{N_w}||\hat{x}\hat{h}_{\text{om}}\hat{h}_i||_2^2 + N_w c_1}. \tag{2.7}$$

In the equation, c_1 and c_2 are calibration constants. They are discussed further in Appendix D.4.

Since (2.6) is convex in ϵ and for fixed x, it is straightforward to incorporate the Par-measure into an optimisation problem where x is available and ϵ should be found. It should be noted that the Par-measure operates entirely on the frequency domain representation of the signals; the temporal structures are not taken into account.

2.3.3 The Taal-measure

The Taal-measure can be considered as an extended version of the Par-measure or as a simplified version of the Dau-model. Similarly to the Par-measure, it takes two properly windowed short-time segments x and $y = x + \epsilon$. The signals x and y are converted to an internal representation

by passing them through an auditory model [69]. The structure of the auditory model is depicted in Figure 2.11. The main difference with the Par-measure is that the Taal-measure takes into account temporal information through the envelope follower and that it models the compressive nonlinearity through the logarithm.

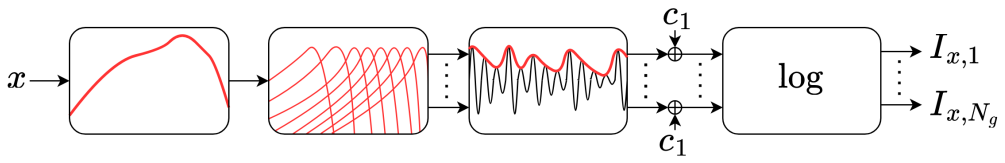


Figure 2.11: A schematic overview of the structure of the auditory model used in the Taal-measure. From left to right, it consists of the outer and middle ear filter, a gammatone filterbank, an envelope follower, addition of internal noise and finally a logarithm to model the compressive nonlinearity. Figure based on [69].

As shown in [69] and discussed in Appendix D.2, the Taal-measure can be simplified such that it is suitable for online optimisation. Similarly to the Par-measure, this simplification allows for computing a number of weighting curves g_i, $i \in \{1, 2, \ldots, N_g\}$ independently of ϵ. The Taal-measure can then be written as [69]

$$D_{\text{Taal}}(x, \epsilon) = \sum_{i=1}^{N_g} ||g_i \epsilon_i||_2^2. \tag{2.8}$$

Here, ϵ_i is the internal representation $I_{\epsilon,i}$ of the simplified measure instead of the complete measure. This is made more specific in Appendix D. The weighting curves g_i are given by [69]

$$g_i^2 = \left(\frac{c_2}{|x_i|^2 \circledast h_s + c_1} \right) \circledast h_s. \tag{2.9}$$

In the equation, $\circledast$ denotes circular convolution, h_s is a low-pass filter and x_i is the internal representation $I_{x,i}$ of the simplified Taal-measure.

Note that (2.8) closely resembles (2.6). In fact, it can be shown that the *simplified* Taal-measure reduces to the Par-measure when the envelope follower is "removed" [69]. This is done in Appendix D.3.

2.3.4 Comparison of Taal, Par and Dau

Taal *et al.* [69] compared the performance of the three measures in three different cases. Namely, (1) evaluating the measure (i.e., is ϵ masked given some masker x), (2) evaluating the measure for fixed x and (3) evaluating the masking curve [69]. Each case is considered for a number of different frame lengths.

It was found that, for short frames, the Par-measure is about three times faster then the simplified Taal-measure. As the framelength increases, the disparity becomes larger, and the Par-measure becomes about 15 times faster than the simplified Taal-measure. The Dau-model is about 10 to 100 times slower than the Taal-measure [69]. In these experiments, the masking curve was not taken into account for the Dau-model, as for this no analytic expression exists and numerically evaluating it would take very long.

From the above, it is clear that in any application where calculations need to be done in real-time, the Par-measure and the Taal-measure are preferred over the Dau-model. The difference between the Par-measure and the Taal-measure is less pronounced, but in terms of computational complexity the Par-model is preferred. A disadvantage of the Par-measure is that it is sensitive to pre-echoes. Pre-echoes are auditory artefacts which occur when the disturbance is turned on before the masker, but where the algorithm predicts that the disturbance is masked [69]. The algorithm proposed later in this thesis uses the Par-measure.

As stated before, the Taal-measure and Par-measure are discussed in more detail in Appendix D. The Taal-measure is still discussed as it is more instructive than the Par-measure and since it can be shown to reduce to the Par-measure under a simplification.

2.4 Block-based filtering

In this section, a method to perform convolution of long signals and filters by splitting the convolution in smaller segments is discussed briefly. In many real-time applications, performing the convolution in smaller steps is desirable as one cannot wait till the full input signal is available or the delay introduced by the filter is too large. An additional motivation for using short-time segments is that the Par-measure operates on segments of about 20 to 40 ms. It follows that the RIR, which reflects the T_{60} time, can easily be multiple segments long. A more complete discussion of block-based filtering is found in Appendix G or in [80].

Throughout this chapter, I consider the convolution between a signal x of infinite length and a causal filter h of length M. This convolution is given by

$$(x * h)(n) = \sum_{m=-\infty}^{\infty} x(m)h(n-m). \tag{2.10}$$

If one wants to evaluate this equation for all n, it is required to have complete knowledge on the signals x and h. In many real-time applications, one only knows $x(n)$ up to some $n = n'$. In this case, segmenting x into short-time segments allows for computing $(x*h)(n)$ up to $n = n'-M+1$. This procedure is described in Section 2.4.1 and given in more detail in Appendix G.1.

An additional challenge occurs for filters h which are so long that the delay introduced by the filtering operation might be too large for any real-time application. In these situations, one might not only wish to segment x, but also h. This is the topic of Section 2.4.2 and is described in more detail in Appendix G.2. Lastly, in Appendix G.3, a frequency-domain version of the derived equation is given. This facilitates an efficient implementation.

Throughout this chapter, the short-time segment length of the input signal is denoted by L_1 and the filter length is denoted by M, such that $\text{supp}(h) \subseteq \{0, 1, \ldots, M-1\}$. The short-time segment length of the filter is denoted by L_2.

2.4.1 Short-time filtering of the input signal

Short-time filtering can be incorporated by means of windowing. Suppose that we have access to some window w_1 of length L_1 for which (2.11) holds,

$$\sum_{l=-\infty}^{\infty} w_1(n - lR_1) = 1 \quad \forall\, n. \tag{2.11}$$

Here $0 \leq R_1 \leq L_1$ is referred to as the hop-rate [80]. Throughout this thesis, the window is considered to have support

$$\text{supp}(w_1) = \{0, \ldots, L_1 - 1\}. \tag{2.12}$$

The window w_1 allows to divide x into blocks of length L_1. Namely, by (2.11), we may write

$$x(n) = x(n) \sum_{l=-\infty}^{\infty} w_1(n - lR_1) = \sum_{l=-\infty}^{\infty} w_1(n - lR_1)x(n) = \sum_{l=-\infty}^{\infty} \tilde{x}_l(n), \tag{2.13}$$

with

$$\tilde{x}_l(n) = w_1(n - lR_1)x(n). \tag{2.14}$$

For notational convenience, the origin of the windowed signal can be shifted to $n = 0$. This yields

$$x_l(n) = \tilde{x}_l(n + lR_1) = w_1(n)x(n + lR_1). \tag{2.15}$$

Substituting (2.13) in (2.10) and simplifying yields

$$(x * h)(n) = \sum_{l=-\infty}^{\infty} \sum_{m=0}^{L_1-1} x_l(m)h(n - m - lR_1). \tag{2.16}$$

The steps used in the simplification are given by (G.7).

Suppose that all blocks with $l \leq l'$ are available. In this case, the convolution can already be calculated partially, and can be updated as new segments $l > l'$ come in. However, in situations where the filter is long, the delay might still be too large. If this is the case, one can choose to segment the filter as well.

2.4.2 Segmenting the filter

In this section, the result of the previous section is extended by also segmenting the filter. I consider filter segments of length L_2. Furthermore, I assume M/L_2 to be a positive integer. In case this is not true, one can simply pad h with zeros until it is true. Note that this also assumes $M \geq L_2$.

Similarly to Section 2.4.1, let us have access to a window w_2 for which

$$\sum_{\iota=\iota_a}^{\iota_b} w_2(n - \iota R_2) = 1 \quad \forall\, n \in \text{supp}(h) \tag{2.17}$$

holds. In this thesis, the window is considered to have support

$$\text{supp}(w_2) = \{0, \ldots, L_2 - 1\}. \tag{2.18}$$

The values ι_a and ι_b are window dependent. We can write

$$h(n) = \sum_{\iota=\iota_a}^{\iota_b} w_2(n - \iota R_2)h(n) = \sum_{\iota=\iota_a}^{\iota_b} \tilde{h}_\iota(n), \tag{2.19}$$

with

$$\tilde{h}_\iota(n) = w_2(n - \iota R_2)h(n). \tag{2.20}$$

For ease of implementation, the support of $\tilde{h}_\iota$ is shifted to start at $n = 0$. To do so, define

$$h_\iota(n) = \tilde{h}_\iota(n + \iota R_2), \tag{2.21}$$

Substituting (2.19) in (2.16) and simplifying yields

$$(x * h)(n) = \sum_{l=-\infty}^{\infty} \sum_{\iota=\iota_1}^{\iota_2} \sum_{m=0}^{L_1-1} x_l(m)h_\iota(n - m - lR_1 - \iota R_2). \tag{2.22}$$

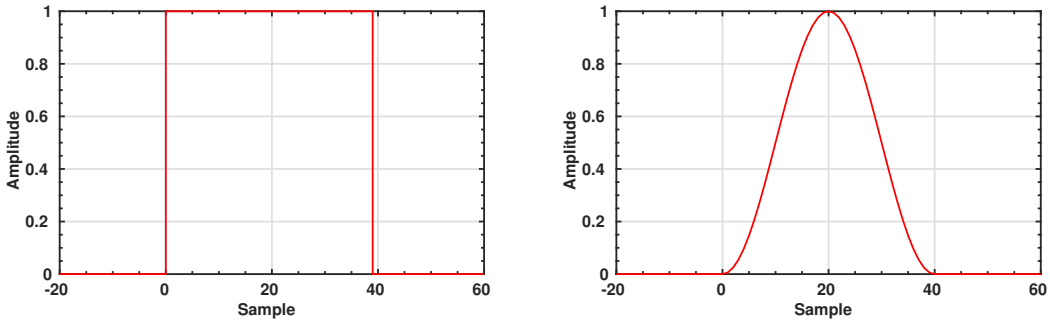
The steps used in the simplification are given by (G.15).

In case of a rectangular window $w_2 = \text{rect}_{L2}$ (see (G.11)), (2.22) reduces to

$$(x * h)(n) = \sum_{l=-\infty}^{\infty} \sum_{\iota=0}^{M/L_2-1} \sum_{m=0}^{L_1-1} x_l(m) h_\iota(n - m - lR_1 - \iota R_2). \tag{2.23}$$

So, for each pair (ι, l), a single convolution is performed. The results of the individual convolutions are added up, resulting in the total output signal.

In Chapter 4, two windows are used. Namely the rectangular window and the Hanning window. They are shown in Figure 2.12 and their properties are given in Appendix G.1.1.



(a) Rectangular window (b) Hanning window

Figure 2.12: An example of the rectangular window and the Hanning window. Both windows are depicted for a length $L = 40$ samples.

Chapter 3
Proposed Algorithm part 1: Power Spectral Density Matrices

In this chapter, the first part of the proposed algorithm is discussed. Namely, the transfer functions to the regions $\mathcal{A}$ and $\mathcal{B}$ and the corresponding power spectral density matrices. Spatial weighting is used to give some regions in space more importance then others. This allows to "trace" out the regions $\mathcal{A}$ and $\mathcal{B}$. Using the spatial weighting functions, the power spectral density matrices are obtained. These matrices are used in Chapter 4 to minimise the energy in region $\mathcal{B}$ with respect to the energy in region $\mathcal{A}$.

In the following, I start from the simple model for the received signal as was described in Section 2.2. By subsequently incorporating the spatial weighting, the model is modified to give the acoustic signal in a region instead of at a point. This allows to construct the PSD matrices. It should be noted that I do most of the derivations for $\mathbb{R}^3$. However, once the spatial weighting functions are chosen, I drop the z-coordinate. This is done to facilitate numerical analysis.

3.1 The spatially weighted playback signals

Using (2.4) and (2.5), the acoustic signal at a point $\mathbf{x}_r$ using isotropic loudspeakers can be written as

$$s_r(t) = \sum_{i=1}^{N_s} \left(h(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i)\right)(t), \tag{3.1}$$

in which h is the acoustic channel from the source to the receiver. Taking the Fourier transform of (3.1) yields

$$\hat{s}_r(\omega) = \sum_{i=1}^{N_s} \hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega)\hat{s}(\mathbf{x}_i, \omega), \tag{3.2}$$

which describes the received signal in the frequency domain. To obtain the signal in a region $\mathcal{V} \subseteq \mathbb{R}^3$ instead of a point $\mathbf{x}_r$, one can integrate (3.2) over the full region. Additionally, spatial weighting over the integration domain can be included by multiplication with some spatial weighting function $p_{\mathcal{V}}$. Doing so gives

$$\hat{s}_{\mathcal{V}}(\omega) = \iiint_{\mathcal{V}\subseteq\mathbb{R}^3} \sum_{i=1}^{N_s} \hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega)\hat{s}(\mathbf{x}_i, \omega)p_{\mathcal{V}}(\mathbf{x}_r)d\mathbf{x}_r. \tag{3.3}$$

Since $\hat{s}$ is independent of $\mathbf{x}_r$ and by linearity of integration and summation, we may write

$$\hat{s}_{\mathcal{V}}(\omega) = \sum_{i=1}^{N_s} \hat{s}(\mathbf{x}_i, \omega) \iiint_{\mathcal{V}\subseteq\mathbb{R}^3} \hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega)p_{\mathcal{V}}(\mathbf{x}_r)d\mathbf{x}_r, \tag{3.4}$$

where it is assumed that the integral converges.

Eq. (3.4) shows that the spatially weighted transfer function is independent of the playback signals. This allows to define the spatially weighted transfer function from i to $\mathcal{V}$ as

$$\hat{v}_{\mathcal{V}}(\mathbf{x}_i, \omega) = \iiint\limits_{\mathcal{V} \subseteq \mathbb{R}^3} \hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega) p_{\mathcal{V}}(\mathbf{x}_r) d\mathbf{x}_r. \tag{3.5}$$

Now assume that $p_{\mathcal{V}}$ is a probability density function (PDF) with support $\mathcal{V}$. In this case, $\hat{v}_{\mathcal{V}}$ can be interpreted as the expected value of the probabilistic transfer function $\hat{V}_{\mathcal{V}}$. This transfer function is obtained by transforming the random vector $\mathbf{X}_r$ using the deterministic transfer function $\hat{h}$. Here, the random vector $\mathbf{X}_r$ has PDF $p_{\mathcal{V}}$. Applying this interpretation to (3.5) gives

$$\hat{V}_{\mathcal{V}}(\mathbf{x}_i, \omega) = \hat{h}(\mathbf{x}_i, \mathbf{X}_r, \omega), \tag{3.6a}$$

$$\hat{v}_{\mathcal{V}}(\mathbf{x}_i, \omega) = \mathbb{E}\left[\hat{V}_{\mathcal{V}}(\mathbf{x}_i, \omega)\right] = \mathbb{E}\left[\hat{h}(\mathbf{x}_i, \mathbf{X}_r, \omega)\right]. \tag{3.6b}$$

In practice, $\hat{h}$ is estimated through the image-source method. This introduces an error which can be incorporated as a noise term. Adding this noise term to (3.6a) gives

$$\hat{V}_{\mathcal{V}}(\mathbf{x}_i, \omega) = \hat{h}(\mathbf{x}_i, \mathbf{X}_r, \omega) + N(\mathbf{x}_i, \omega). \tag{3.7}$$

Let us now briefly reconsider the probabilistic received signal $\hat{S}_{\mathcal{V}}(\omega)$. This signal is obtained by summing the individual probabilistic signals $\hat{S}_{\mathcal{V}}(\mathbf{x}_i, \omega)$. Since the playback signals remain deterministic, the received signal is given by

$$\hat{S}_{\mathcal{V}}(\omega) = \sum_{i=1}^{N_g} \left(\hat{h}(\mathbf{x}_i, \mathbf{X}_r, \omega) + \hat{N}(\mathbf{x}_i, \omega)\right) \hat{s}(\mathbf{x}_i, \omega). \tag{3.8}$$

Eq. (3.8) can be rewritten in a vector representation as

$$\hat{S}_{\mathcal{V}}(\omega) = \hat{\mathbf{s}}^{\mathrm{T}}(\omega) \left(\hat{\mathbf{N}}(\omega) + \hat{\boldsymbol{h}}(\mathbf{X}_r, \omega)\right), \tag{3.9}$$

where

$$\hat{\mathbf{s}}(\omega) = \left[\hat{s}(\mathbf{x}_1, \omega), \quad \ldots, \quad \hat{s}(\mathbf{x}_{N_s}, \omega)\right]^{\mathrm{T}}, \tag{3.10a}$$

$$\hat{\boldsymbol{h}}(\mathbf{X}_r, \omega) = \left[\hat{h}\left(\mathbf{x}_1, \mathbf{X}_r, \omega\right), \quad \ldots, \quad \hat{h}\left(\mathbf{x}_{N_s}, \mathbf{X}_r, \omega\right)\right]^{\mathrm{T}}, \tag{3.10b}$$

$$\hat{\mathbf{N}}(\omega) = \left[\hat{N}\left(\mathbf{x}_1, \omega\right), \quad \ldots, \quad \hat{N}\left(\mathbf{x}_{N_s}, \omega\right)\right]^{\mathrm{T}}. \tag{3.10c}$$

3.2 Estimation of the Power Spectral Densities

To maximise the ratio of the energy in region $\mathcal{A}$ compared to that in region $\mathcal{B}$, it is useful to have access to the power spectral density (PSD) matrices. These matrices are found by computing $\mathbb{E}\left[\hat{S}_{\mathcal{V}}(\omega)\hat{S}^*_{\mathcal{V}}(\omega)\right]$. Doing so gives

$$\mathbb{E}\left[\hat{S}_{\mathcal{V}}(\omega)\hat{S}^*_{\mathcal{V}}(\omega)\right] = \hat{\mathbf{s}}^{\mathrm{T}}(\omega)\mathbb{E}\left[\left(\hat{\mathbf{N}}(\omega) + \hat{\boldsymbol{h}}(\mathbf{X}_r, \omega)\right)\left(\hat{\mathbf{N}}(\omega) + \hat{\boldsymbol{h}}(\mathbf{X}_r, \omega)\right)^{\mathrm{H}}\right]\hat{\mathbf{s}}^*(\omega), \tag{3.11}$$

where it was used that $\hat{\mathbf{s}}$ is deterministic.

While it should be verified in future work, I assume $\hat{\mathbf{N}}(\omega)$ and $\hat{h}(\mathbf{X}_r, \omega)$ to be uncorrelated. This yields

$$\mathbb{E}\left[\hat{S}_{\mathcal{V}}(\omega)\hat{S}^*_{\mathcal{V}}(\omega)\right] = \hat{\mathbf{s}}^{\mathrm{T}}(\omega)\left(\mathbb{E}\left[\hat{\mathbf{N}}(\omega)\hat{\mathbf{N}}^{\mathrm{H}}(\omega)\right] + \mathbb{E}\left[\hat{\boldsymbol{h}}(\mathbf{X}_r, \omega)\hat{\boldsymbol{h}}^{\mathrm{H}}(\mathbf{X}_r, \omega)\right]\right)\hat{\mathbf{s}}^*(\omega). \tag{3.12}$$

The PSD matrices can now be defined as

$$\mathbf{R}_n(\omega) = \mathbb{E}\left[\hat{\mathbf{N}}(\omega)\hat{\mathbf{N}}^{\mathrm{H}}(\omega)\right] \in \mathbb{C}^{N_s \times N_s}, \tag{3.13a}$$

$$\mathbf{R}_{\mathcal{V}}(\omega) = \mathbb{E}\left[\hat{\boldsymbol{h}}(\mathbf{X}_r, \omega)\hat{\boldsymbol{h}}^{\mathrm{H}}(\mathbf{X}_r, \omega)\right] \in \mathbb{C}^{N_s \times N_s}. \tag{3.13b}$$

First consider the matrix $\mathbf{R}_{\mathcal{V}}$. Let the ith element of the jth column be denoted as $\{\mathbf{R}_{\mathcal{V}}\}_{ij}$. Here, the index j should not be confused with the complex value j. This element is given by

$$\begin{aligned}\{\mathbf{R}_{\mathcal{V}}\}_{ij} &= \mathbb{E}\left[\hat{h}\left(\mathbf{x}_i, \mathbf{X}_r, \omega\right)\hat{h}^*\left(\mathbf{x}_j, \mathbf{X}_r, \omega\right)\right] \\ &= \iiint\limits_{\mathcal{V}\subseteq\mathbb{R}^3} \hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega)\hat{h}^*(\mathbf{x}_j, \mathbf{x}_r, \omega)p_{\mathcal{V}}(\mathbf{x}_r)d\mathbf{x}_r.\end{aligned} \tag{3.14}$$

In sections 3.3 and 3.4, we return to solving this equation. Before doing so, let us consider $\mathbf{R}_n(\omega)$.

The matrix $\mathbf{R}_n(\omega)$ can be decomposed into two independent terms. Here, the first term models numerical inaccuracies and the second term models the late reverberation of the room impulse response[1]. Denoting the reverberation term by $\mathbf{R}_{\mathrm{iso}}$ and the numerical inaccuracy term by $\mathbf{R}_{\mathrm{num}}$ gives

$$\mathbf{R}_n(\omega) = \mathbf{R}_{\mathrm{iso}}(\omega) + \mathbf{R}_{\mathrm{num}}(\omega). \tag{3.15}$$

It is assumed that the numerical inaccuracies are uncorrelated from loudspeaker to loudspeaker. Thus, $\mathbf{R}_{\mathrm{num}}$ is a diagonal matrix. While it should be verified in future work, it is reasonable to assume that the standard deviation of the numerical inaccuracies is about equal for each of the loudspeakers. Hence,

$$\mathbf{R}_{\mathrm{num}}(\omega) = \sigma^2_{\mathrm{num}}(\omega)\mathbf{I}_{N_s}, \tag{3.16}$$

with $\sigma^2_{\mathrm{num}}(\omega)$ a to-be determined parameter and $\mathbf{I}_{N_s}$ the $N_s \times N_s$ identity matrix.

The reverberation term can be modelled by considering reflections coming from all possible directions. These reflections can be treated as plane-waves. The corresponding PSD matrix is known and given by [81]

$$\{\mathbf{R}_{\mathrm{iso}}\}_{ij}(\omega) = \sigma^2_{\mathrm{iso}}(\omega)\mathrm{sinc}\left(\frac{\omega}{c}||\mathbf{x}_i - \mathbf{x}_j||_2\right), \tag{3.17}$$

where $\sigma^2_{\mathrm{iso}}(\omega)$ is the PSD of the reverberation and needs to be estimated.

Applying the previously found results to the regions $\mathcal{A}$ and $\mathcal{B}$ gives PSDs $\mathbb{E}\left[\hat{S}_{\mathcal{A}}(\omega)\hat{S}^*_{\mathcal{A}}(\omega)\right]$ and $\mathbb{E}\left[\hat{S}_{\mathcal{B}}(\omega)\hat{S}^*_{\mathcal{B}}(\omega)\right]$. They are given by

$$\mathbb{E}\left[\hat{S}_{\mathcal{A}}(\omega)\hat{S}^*_{\mathcal{A}}(\omega)\right] = \hat{\mathbf{s}}^{\mathrm{T}}\left(\mathbf{R}_{\mathcal{A}}(\omega) + \mathbf{R}_{\mathrm{iso}}(\omega) + \mathbf{R}_{\mathrm{num}}(\omega)\right)\hat{\mathbf{s}}^*, \tag{3.18a}$$

[1]It should be noted that the latter term could also be included by considering a sufficient number of image-sources. However, for reasons concerning the implementation, it is chosen to consider the closest few image-sources only.

$$\mathbb{E}\left[\hat{S}_{\mathcal{B}}(\omega)\hat{S}^*_{\mathcal{B}}(\omega)\right] = \hat{\mathbf{s}}^{\mathrm{T}}\left(\mathbf{R}_{\mathcal{B}}(\omega) + \mathbf{R}_{\mathrm{iso}}(\omega) + \mathbf{R}_{\mathrm{num}}(\omega)\right)\hat{\mathbf{s}}^*. \tag{3.18b}$$

Here, $\mathbf{R}_{\mathcal{V}}$ with $\mathcal{V} \in \{\mathcal{A}, \mathcal{B}\}$, is calculated using (3.14), $\mathbf{R}_{\mathrm{iso}}$ is calculated using (3.17) and $\mathbf{R}_{\mathrm{num}}$ is calculated using (3.16). To do so, it is required to know or estimate the functions $\hat{h}$, $p_{\mathcal{V}}$, $\sigma_{\mathrm{iso}}(\omega)$ and $\sigma_{\mathrm{num}}(\omega)$. The former two are the topic of Section 3.3, while the latter two are considered in Section 3.4.

3.3 The PSD matrices $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$

In this section, the transfer function $\hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega)$ and spatial weighting functions $p_{\mathcal{A}}$ and $p_{\mathcal{B}}$ are given. The transfer functions are rewritten in a form which allows for choosing the spatial weighting functions.

Note that, to facilitate numerically solving integral (3.14), the problem is simplified to a two-dimensional instead of a three-dimensional scenario.

3.3.1 The room transfer function

Recall, from Section 2.2, that the image-source method gives the RIR as

$$h(\mathbf{x}_i, \mathbf{x}_r, t) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\delta\left(t - \frac{||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}{c}\right)}{4\pi||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}. \tag{3.19}$$

Here, N_i is the number of image sources considered for loudspeaker i, $\beta_{i,\xi}$ is the reflection coefficient corresponding to image-source ξ, and $\mathbf{x}_{i,\xi}$ is the location of the ξth image-source. It should be noted that these parameters are independent of $\mathbf{x}_r$. The corresponding frequency domain transfer function is

$$\hat{h}(\mathbf{x}_i, \mathbf{x}_r, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\exp\left\{-\frac{j\omega}{c}||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2\right\}}{4\pi||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}. \tag{3.20}$$

When choosing the spatial weighting functions, it is convenient to have a coordinate system centered around the expected location of the head $\mathbf{x}_h$. Thus, define $\mathbf{x} = \mathbf{x}_r - \mathbf{x}_h$. Substituting in (3.20) and flipping the signs in the l_2-norm yields

$$\hat{h}(\mathbf{x}_i, \mathbf{x} + \mathbf{x}_h, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\exp\left\{-\frac{j\omega}{c}||\mathbf{x} + \mathbf{x}_h - \mathbf{x}_{i,\xi}||_2\right\}}{4\pi||\mathbf{x} + \mathbf{x}_h - \mathbf{x}_{i,\xi}||_2}. \tag{3.21}$$

Eq. (3.21) can now be transformed to a coordinate system which is suitable for choosing the spatial weighting functions. As mentioned in Assumption 5 of Chapter 1, I consider the loudspeakers and listener to be placed at the same height $z = z_h$ and I assume the floor and ceiling to be fully absorbing. This allows for treating the problem as two-dimensional relative to the plane $z = z_h$. The reason for doing this is that the integrals will ultimately be solved numerically. Thus, from here, I consider $\mathbf{x} = (x, y)$, which is transformed to a polar coordinate system shortly. In Appendix H, the same derivations are given for the three-dimensional problem using a spherical and a cylindrical coordinate system.

Expanding the l_2-norm of (3.21) yields

$$\hat{h}(\mathbf{x}_i, \mathbf{x} + \mathbf{x}_h, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{e^{-\frac{j\omega}{c}\sqrt{(x+x_h-x_{i,\xi})^2+(y+y_h-y_{i,\xi})^2}}}{4\pi\sqrt{(x + x_h - x_{i,\xi})^2 + (y + y_h - y_{i,\xi})^2}} \tag{3.22}$$

Now consider the transformation to polar coordinates $x = r\cos(\theta)$ and $y = r\sin(\theta)$, with $\theta \in [0, 2\pi)$ and $r \in [0, \infty)$. Transforming (3.22) yields

$$\hat{h}(\mathbf{x}_i, r, \theta, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{e^{-\frac{j\omega}{c}\sqrt{(r\cos(\theta)+x_h-x_{i,\xi})^2+(r\sin(\theta)+y_h-y_{i,\xi})^2}}}{4\pi\sqrt{(r\cos(\theta)+x_h-x_{i,\xi})^2+(r\sin(\theta)+y_h-y_{i,\xi})^2}}. \tag{3.23}$$

3.3.2 The spatial weighting functions for regions $\mathcal{A}$ and $\mathcal{B}$

In this section, the spatial weighting functions $p_{\mathcal{A}}$ and $p_{\mathcal{B}}$ are chosen. These functions should be chosen such that the weighting is high inside the corresponding region, while being low outside. Ideally, the regions can be considered as a slice of a donut centered at $\mathbf{x} = (0, 0)$. The "thickest" (in terms of weighting) part of the donut should coincide with the circumference of the head and the donut should be large enough to allow for some listener displacement.

To facilitate choosing the weighting functions separately for r and θ, the distributions are chosen as $p_{\mathcal{V}}(\mathbf{x} + \mathbf{x}_h) = p_{\mathcal{V},1}(r)p_{\mathcal{V},2}(\theta)$. Thus, the support of $p_{\mathcal{V},1}(r)$ should be a subset of $[0, \infty)$ and the support of $p_{\mathcal{V},2}(\theta)$ should be a subset of $[0, 2\pi)$. Next to this, for $p_{\mathcal{V},1}(r)$, it is handy if $p_{\mathcal{V},1}(0) = 0$ to avoid a hard edge. Lastly, for both distributions, it is convenient to be able to set the mean and to be able to control how heavy the tails of the distribution are. The last property allows for incorporating a certain amount of uncertainty in, for example, head location and loudspeaker location.

While it is likely that multiple distributions are valid, the following two distributions are chosen:

- In the r-coordinate, a normal distribution is used. The mean μ_r is set to be just outside the head. The standard deviation σ_r is used to control the width of the region. Note that we ideally require $p_{\mathcal{V},1}(0) = 0$, which does not hold for the normal distribution. By ensuring that σ_r is sufficiently small compared to μ_r, we can adhere to the requirement for all practical purpose.
- In the θ-coordinate, the Von Mises distribution is used. The Von Mises distribution somewhat resembles a normal distribution, but is periodic and has a support of length 2π. The mean is set through μ_θ and the width of the region is set though κ_θ.

The distributions are described in more detail below.

3.3.2.1 Spatial weighting of the radius: the normal distribution

The normal distribution is parameterised by the mean μ_r and the standard deviation σ_r. Since $r \in [0, \infty)$, μ_r must be larger then zero. The distribution is given by [6]

$$p_{\mathcal{N}}(r; \mu_r, \sigma_r) = \frac{1}{\sqrt{2\pi\sigma_r^2}} \exp\left\{-\frac{(r-\mu_r)^2}{2\sigma_r^2}\right\}, \quad \sigma_r, \mu_r > 0. \tag{3.24}$$

The ";" in $p_{\mathcal{N}}(r; \mu_r, \sigma_r)$ indicates that the distribution is parameterised by μ_r and σ_r.

An important note here is that ideally we require a distribution which is zero for $r \leq 0$ and smoothly increases from zero for $r > 0$. This is necessary in order to avoid hard edges. Strictly speaking, this can not be achieved using the normal distribution. However, we can get sufficiently close. This is done by considering that the weighting is used to weigh a RIR. Thus, in analogy with the T_{60} time, I consider the region where the value has dropped by 60 dB as irrelevant.

Concretely, if at $r = 0$ the value $p(r; \mu_r, \sigma_r)$ has dropped at least 60 dB below its maximum value, I consider the normal distribution to be usable. We may write

$$\begin{aligned} -60 &= 20 \log_{10} \left(e^{-\frac{(r-\mu_r)^2}{2\sigma_r^2}} \right) \\ &= 20 \log_{10}(e) \ln \left(e^{-\frac{(r-\mu_r)^2}{2\sigma_r^2}} \right) \\ &= -\frac{10 \log_{10}(e)}{\sigma_r^2} (r - \mu_r)^2 \Rightarrow \end{aligned} \tag{3.25a}$$

$$r = \pm \sqrt{\frac{6\sigma_r^2}{\log_{10}(e)}} + \mu_r \approx \pm 3.7\sigma_r + \mu_r. \tag{3.25b}$$

So, if $\mu_r \geq 3.7\sigma_r$, the normal distribution may be used in r while adhering to the soft boundaries requirement.

It is chosen to set $p_{\mathcal{A},1}(r) = p_{\mathcal{B},1}(r) = p_{\mathcal{N}}(r; \mu_r, \sigma_r)$.

3.3.2.2 Spatial weighting of the azimuth: the Von Mises distribution

For the angle θ, the Von Mises distribution is used. This distribution has support of length 2π. The concentration around the mean μ_θ can be set through the parameter κ_θ. The Von Mises distribution $p_{\mathcal{M}}(\theta; \mu_\theta, \kappa_\theta)$ is given by [82]

$$p_{\mathcal{M}}(\theta; \mu_\theta, \kappa_\theta) = \frac{\exp(\kappa_\theta \cos(\theta - \mu_\theta))}{2\pi I_0(\kappa_\theta)}, \tag{3.26}$$

where the semicolon indicates "parameterised by" and I_0 is the modified Bessel function of the first kind with order zero.

It is chosen to set $p_{\mathcal{A},2}(\theta) = p_{\mathcal{M}}(\theta; \mu_{\mathcal{A}}, \kappa_{\mathcal{A}})$ and $p_{\mathcal{B},2}(\theta) = p_{\mathcal{M}}(\theta; \mu_{\mathcal{B}}, \kappa_{\mathcal{B}})$. The point of maximum weight of each of the regions is set opposite by choosing $\mu_{\mathcal{B}} = \mu_{\mathcal{A}} + \pi$. Combining the probability density functions for the radius and azimuth gives

$$p_{\mathcal{A}}(r, \theta) = \frac{1}{I_0(\kappa_{\mathcal{A}}) \sqrt{8\pi^3 \sigma_r^2}} \exp\left\{ -\frac{(r - \mu_r)^2}{2\sigma_r^2} + \kappa_{\mathcal{A}} \cos(\theta - \mu_{\mathcal{A}}) \right\}, \tag{3.27a}$$

$$p_{\mathcal{B}}(r, \theta) = \frac{1}{I_0(\kappa_{\mathcal{B}}) \sqrt{8\pi^3 \sigma_r^2}} \exp\left\{ -\frac{(r - \mu_r)^2}{2\sigma_r^2} + \kappa_{\mathcal{B}} \cos(\theta - \mu_{\mathcal{B}}) \right\}. \tag{3.27b}$$

Substituting (3.27) and (3.23) in (3.14) and transforming the integration measure $dxdy \rightarrow rdrd\theta$ gives

$$\{\mathbf{R}_{\mathcal{V}}\}_{ij}(\omega) = \int_0^\infty \int_0^{2\pi} \hat{h}(\mathbf{x}_i, r, \theta, \omega) \hat{h}^*(\mathbf{x}_j, r, \theta, \omega) p_{\mathcal{V}}(r, \theta) r dr d\theta, \quad \mathcal{V} \in \{\mathcal{A}, \mathcal{B}\}. \tag{3.28}$$

In the implementation, this equation is solved numerically.

3.4 Implementation considerations

In this section, some considerations concerning the implementation are given.

First consider h. As we will see in the next section, the length of the segmentation window is L (in samples) and all signals are zero padded to a length $2L$. Thus, all image-sources $\mathbf{x}_{i,\xi}$ which arrive within L/f_s samples are considered in the calculation of h. Here, f_s is the sample frequency. For a window length of about 20 ms, this translates to image-sources up to about 6.8 m away from the center of the head.

As stated earlier, $\mathbf{R}_{\mathcal{V}}$, as given by (3.28), is calculated numerically. This is done by sampling ω for a discrete number of frequencies and subsequently solving (3.28) using the MATLAB function `integral2`. The frequency bins are given by

$$\omega = \frac{\kappa f_s}{2L}, \quad \kappa \in \{-L, -L+1, \ldots, L-1\}, \tag{3.29}$$

and the integration limits of r are set to $\max\{\mu_r - 4\sigma_r, 0\}$ to $\mu_r + 4\sigma_r$. This results in the following integral being estimated

$$\{\mathbf{R}_{\mathcal{V}}\}_{ij}(\omega) \approx \int_{\max\{\mu_r-4\sigma_r,0\}}^{\mu_r+4\sigma_r} \int_{0}^{2\pi} \hat{h}(\mathbf{x}_i, r, \theta, \omega)\hat{h}^*(\mathbf{x}_j, r, \theta, \omega) p_{\mathcal{V}}(r, \theta) r \, dr \, d\theta, \quad \mathcal{V} \in \{\mathcal{A}, \mathcal{B}\}. \tag{3.30}$$

One note here is that $\mathbf{R}_{\mathcal{V}}$ should be positive-semidefinite by construction. However, during the computation, small errors might occur which make the (almost) zero-valued eigenvalues slightly negative. This is fixed by including the term $\mathbf{R}_{\text{num}}(\omega)$. This term is considered to be independent of ω so that

$$\mathbf{R}_{\text{num}}(\omega) = \sigma^2_{\text{num}} \mathbf{I}_{N_s}, \quad \forall \omega. \tag{3.31}$$

The value of σ^2_{num} is empirically determined. In practice, it was found that $\sigma^2_{\text{num}} = 10^{-12}$ works sufficiently well.

The value of $\sigma^2_{\text{iso}}(\omega)$ can be determined through synthesising a number of rooms using the RIR generator of [28] and subsequently determining the PSD. In practical scenarios, it needs to be estimated. An overview of methods to do so is given by [83]. Due to time-constraints, I consider this as future work and use $\sigma^2_{\text{iso}}(\omega) = 0$ for all ω. This is believed not to be problematic, since the precedence effect states that the estimation of angle of incidence is mainly based on the direct path. However, this needs to be verified in future work.

In the next chapter, I use $\mathbf{R}_A$ and $\mathbf{R}_B$ to denote the total PSD matrices. Additionally, I consider only positive frequency bins k. These are obtained from κ according to

$$k = \begin{cases} \kappa, & \text{for } \kappa \geq 0 \\ \kappa + 2L, & \text{otherwise.} \end{cases} \tag{3.32}$$

Thus, the PSD matrices are

$$\mathbf{R}_A(k) = \mathbf{R}_{\mathcal{A}}\left(\frac{\kappa f_s}{2L}\right) + \mathbf{R}_{\text{num}}\left(\frac{\kappa f_s}{2L}\right) + \mathbf{R}_{\text{iso}}\left(\frac{\kappa f_s}{2L}\right), \tag{3.33a}$$

$$\mathbf{R}_B(k) = \mathbf{R}_{\mathcal{B}}\left(\frac{\kappa f_s}{2L}\right) + \mathbf{R}_{\text{num}}\left(\frac{\kappa f_s}{2L}\right) + \mathbf{R}_{\text{iso}}\left(\frac{\kappa f_s}{2L}\right). \tag{3.33b}$$

Lastly, note that integral (3.30) is oscillatory for sufficiently large ω. This poses problems for "typical" numerical solvers. While not implemented, in Appendix I a method is outlined which is suitable for oscillatory integrals of certain types. This method is likely to improve the

computation times for large ω. It should, however, not be used for small ω. Alternatively, it is desirable to avoid the computation of integrals. This computation can be avoided by considering a sufficiently accurate probabilistic model of the RIR. A first step towards this probabilistic model is taken and described in Appendix J.

Chapter 4
Proposed Algorithm part 2: Proposed Algorithm

In this chapter, the spatial sound algorithm based on a perceptual measure is discussed. In the algorithm, I assume the listener and RIR to be stationary, so the listener is not moving and the room is not varying. Thus, the PSD matrices $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$ as described in Chapter 3 can be precomputed. It should be noted that the extension to slowly time-varying conditions can be incorporated relatively straightforwardly through the block-based filtering process described in Section 2.4.

Before discussing the details of the algorithm, let me repeat some notation and give the outline of the algorithm. We have N_s physical loudspeakers $i \in \{1, \ldots, N_s\}$ and one virtual loudspeaker $i = 0$. Their playback signals are denoted by $s(\mathbf{x}_i, n)$. The corresponding audio received in region $\mathcal{V} \in \{\mathcal{A}, \mathcal{B}\}$ due to loudspeaker i is denoted by $s_{\mathcal{V}}(\mathbf{x}_i, n)$. The PSD matrices are given by $\mathbf{R}_{\text{num}}$, $\mathbf{R}_{\text{iso}}$ and $\mathbf{R}_{\mathcal{V}}$. I also consider the RIR from $\mathbf{x}_i$ to $\mathbf{x}_a$, where $\mathbf{x}_a$ is the point of maximum weight in region $\mathcal{A}$. These RIRs are denoted as $a(\mathbf{x}_i, n)$ and have a length $M > L$. Here, L is the length of the segmentation window.

The algorithm consists of a small number of steps. These are outlined below and visualised in Figure 4.1. The steps are described in more detail in the following sections. Firstly, assume that

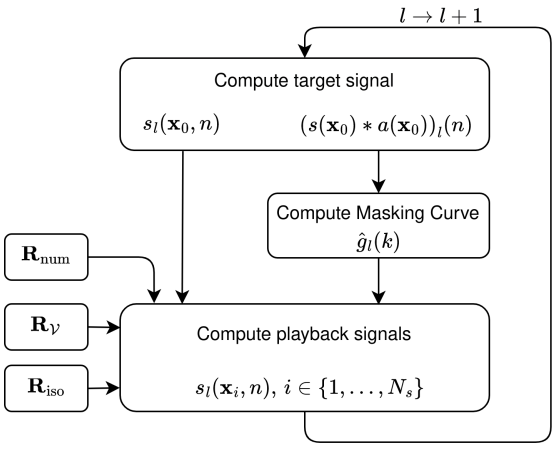


Figure 4.1: A simple outline of the algorithm. The signals which are obtained in each step are also indicated.

we are calculating the playback signals for segment $l = l'$. In this scenario, we already have access to all the playback signals with $l < l'$. The first step is now to compute the reference playback signal $s_{l'}(\mathbf{x}_0, n)$ and the masker $(s(\mathbf{x}_0) * a(\mathbf{x}_0, \mathbf{x}_a))_l(n)$. The masking curve $\hat{g}_{l'}$ can be calculated from the masker using the Par-measure. By plugging the masking curve and the reference signal in

the optimisation problem, the playback signals of loudspeakers $i = 1, 2, \ldots, N_s$ can be determined. When doing so, one should take care that, due to the long RIRs, the contribution of the playback signals with $l < l'$ should also be taken into account.

Firstly, in Section 4.1, the computation of the reference signal, the masking signal and the masking curve is discussed. Then, in Section 4.2, six optimisation problems are presented. These optimisation problems vary slightly in complexity and form and can be solved to obtain the playback signals.

4.1 Computation of the reference signal and masking curve

In this section, the computation of the reference signal and the masking curve is discussed briefly. The reference signal should be interpreted as the desired audio in region $\mathcal{A}$. Since the reference playback signal $s(\mathbf{x}_0, n)$ is assumed to be known, the full-length reference signal in region $\mathcal{A}$ is straightforwardly obtained as

$$s_a(\mathbf{x}_0, n) = (s(\mathbf{x}_0) * a(\mathbf{x}_0))(n). \tag{4.1}$$

4.1.1 The masking curve

Under the assumption that the actual (physical) signal will closely match the reference signal, one may compute the masking curve based on windowed segments of the reference signal. Recall, from Section 2.3.2, that the Par-measure operates on short-time segments. Let us use a segment of even length L, obtained through a window with repetition rate $R_1 = {}^{L}/_{2}$. Using (2.15), the windowed segment is given by

$$s_{a,l}(\mathbf{x}_0, n) = w_1(n) s_a(\mathbf{x}_0, n + lR_1). \tag{4.2}$$

In order to obtain the masking curve using the Par-measure, the frequency domain representation is required. To avoid aliasing, the $2L$-point DFT of (4.2) is taken. Plugging (4.2) into the definition of the Discrete Fourier transform (see (3)) yields,

$$\hat{s}_{a,l}(\mathbf{x}_0, k) = \frac{1}{\sqrt{2L}} \sum_{n=0}^{2L-1} s_{a,l}(\mathbf{x}_0, n) e^{-\pi j k n / L}, \quad k \in \{0, \ldots, 2L-1\}. \tag{4.3}$$

The division by $\sqrt{2L}$ is done for power conservation. This could, alternatively, also be included in the calibration constants.

Using (2.7) and (4.3), the masking curve for the lth segment, $\hat{g}_l$, is obtained. It is given by

$$\hat{g}_l = \sqrt{c_2 \sum_{i=1}^{N_g} \frac{\hat{h}_{\text{om}}^2 \hat{h}_i^2}{\frac{1}{2L} ||\hat{s}_{a,l}(\mathbf{x}_0) \hat{h}_{\text{om}} \hat{h}_i||_2^2 + 2Lc_1}}, \tag{4.4}$$

where all frequency-domain signals are of length $2L$.

4.1.2 The short-time reference signal

Although the full-length reference signal is obtained straightforwardly from (4.1), the short-length reference signal takes some more thought. Namely, I am attempting to obtain the short-time segments of the *playback* signals. These playback signals will still be filtered by the room.

However, convolution is not linear for a pointwise multiplication with a (nonconstant) window $w(n)$. That is

$$\sum_m w(n-m)x(n-m)h(m) \neq w(n)\sum_m x(n-m)h(m). \tag{4.5}$$

Hence, we are not allowed to use $s_{\mathcal{A},l}(\mathbf{x}_0, n)$ as a reference signal. If we would do so anyway, the windowing applied on the target signal, and thus that applied on the obtained playback signals, is not tractable anymore. Instead, we should take a segment $s_l(\mathbf{x}_0, n)$. This segment is given by

$$s_l(\mathbf{x}_0, n) = w_1(n)s(\mathbf{x}_0, n + lR_1). \tag{4.6}$$

The window w_1 should satisfy the constant overlap-add condition or, alternatively, a second window can be used at the synthesising stage (where the segments are combined again). Let the synthesis window be denoted by $\bar{w}_1(n)$. This window is valid if the total window $w_1(n)\bar{w}_1(n)$ satisfies the constant overlap-add condition. Among others, a possible choice is to use w_1 a Hanning window and $\bar{w}_1$ a rectangular window. Alternatively, it can be chosen to use a square root Hanning window for both the analysis and synthesis stage. The square root Hanning window is obtained by taking the elementwise square root of (G.12). Doing so gives

$$\mathrm{sqrthann}_L(n) = \mathrm{rect}_L(n)\sqrt{\left(\frac{1}{2} + \frac{1}{2}\cos\left(\frac{2\pi}{L}\left(n - \frac{L}{2}\right)\right)\right)}. \tag{4.7}$$

The playback signals are synthesised from the individual playback segments according to

$$s(\mathbf{x}_i, n) = \sum_{l=-\infty}^{\infty} \bar{w}_1(n)s_l(\mathbf{x}_i, n - lR_1). \tag{4.8}$$

4.1.3 Segmentation of the Room Impulse Response

Now let us consider the RIRs briefly. Let them be windowed using a rectangular window $w_2 = \mathrm{rect}_L$ (see (G.11)) of length L and repetition rate $R_2 = L$. The filter segments are given by

$$a_\iota(\mathbf{x}_i, n) = w_2(n)a(\mathbf{x}_i, n + \iota R_2), \tag{4.9}$$

with $\iota \in \{0, 1, \ldots, {}^{M}/_{L} - 1\}$. Recall that M is the length of the filter, possibly including zero padding to ensure ${}^{M}/_{L}$ integer.

The filters introduce two problems. First of all, due to their large length, we can not optimise for all blocks simultaneously. Thus, it is chosen to optimise only for $\iota = 0$ and consider the contribution of the other blocks as an error to be corrected, or to simply ignore this contribution.

If it is chosen to correct for the error, an additional problem is introduced. Namely, if we would simply calculate the error and window it so that it has valid support, the advantage of the windowing would be lost. Additionally, as is explained in Section 4.1.2, the window would not translate properly to a window on the input signals. Instead, a solution is to consider only the error term

$$\epsilon_l(\mathbf{x}_i, n) = \sum_{\iota=1}^{M/L-1} (s_{(l-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i))(n). \tag{4.10}$$

When l is odd, this error term takes only the contribution of all odd segments with $l' \leq l$ into account. If l is even, only the contributions of the even segments are taken into account. This ensures that the window remains correctly placed and that no additional windowing is needed to ensure the error signal to have the proper support. This is explained further in Appendix K.

4.2 Computation of the playback signals

In this section, the computation of the playback signals is described. The playback signals are computed by solving an optimisation problem consisting of a cost function and a set of constraints. Concretely, I propose six different but highly related optimisation problems. In the following, I first give a general form. Then, each of the optimisation problems is described briefly. Before doing so, let me define a few vectors and matrices.

4.2.1 Notation used in the optimisation problem

In this section, some notation used in the optimisation problems is defined. For each signal, both the discrete-time and discrete-frequency domain vector are given. Additionally, Fourier transform matrices are provided which allow to convert from discrete-time domain representation to discrete-frequency domain representation. Note that each frequency domain signal is assumed to be obtained from a discrete-time domain signal zero-padded to $2L$ samples.

Firstly, let $\mathbf{s}_l(n) \in \mathbb{R}^{N_s}$ denote per-sample signal vector

$$\mathbf{s}_l(n) = \big[s_l(\mathbf{x}_1, n), \quad \ldots, s_l(\mathbf{x}_{N_s}, n)\big]^{\mathrm{T}}, \quad n \in \{0, \ldots, L-1\}, \tag{4.11}$$

with discrete-frequency domain equivalent $\hat{\mathbf{s}}_l(k) \in \mathbb{C}^{N_s}$.

$$\hat{\mathbf{s}}_l(k) = \big[\hat{s}_l(\mathbf{x}_1, k), \quad \ldots, \hat{s}_l(\mathbf{x}_{N_s}, k)\big]^{\mathrm{T}}, \quad k \in \{0, \ldots, 2L-1\}. \tag{4.12}$$

Define the signal vector $\mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^L$

$$\mathbf{s}_l(\mathbf{x}_i) = \big[s_l(\mathbf{x}_i, 0), \quad \ldots, \quad s_l(\mathbf{x}_i, L-1)\big]^T, \tag{4.13}$$

with discrete-frequency domain equivalent $\hat{\mathbf{s}}_l(\mathbf{x}_i) \in \mathbb{C}^{2L}$

$$\hat{\mathbf{s}}_l(\mathbf{x}_i) = \big[\hat{s}_l(\mathbf{x}_i, 0), \quad \ldots, \quad \hat{s}_l(\mathbf{x}_i, 2L-1)\big]^T. \tag{4.14}$$

Define the filter vector $\mathbf{a}_0(\mathbf{x}_i) \in \mathbb{R}^L$

$$\mathbf{a}_0(\mathbf{x}_i) = \big[a_0(\mathbf{x}_i, 0), \quad \ldots, \quad a_0(\mathbf{x}_i, L-1)\big]^T, \tag{4.15}$$

with discrete-frequency domain equivalent $\hat{\mathbf{a}}_0(\mathbf{x}_i) \in \mathbb{C}^{2L}$

$$\hat{\mathbf{a}}_0(\mathbf{x}_i) = \big[\hat{a}_0(\mathbf{x}_i, 0), \quad \ldots, \quad \hat{a}_0(\mathbf{x}_i, 2L-1)\big]^T, \tag{4.16}$$

it is handy to have this vector in matrix form. This matrix is given by

$$\hat{\mathbf{A}}_0(\mathbf{x}_i) = \mathrm{diag}\left(\hat{\mathbf{a}}_0(\mathbf{x}_i)\right) \in \mathbb{C}^{2L \times 2L}. \tag{4.17}$$

Define the error vector $\boldsymbol{\epsilon}_l(\mathbf{x}_i) \in \mathbb{R}^L$

$$\boldsymbol{\epsilon}_l(\mathbf{x}_i) = \big[\epsilon_l(\mathbf{x}_i, 0), \quad \ldots, \quad \epsilon_l(\mathbf{x}_i, L-1)\big]^T, \tag{4.18}$$

with discrete-frequency domain equivalent $\hat{\mathbf{e}}_0(\mathbf{x}_i) \in \mathbb{C}^{2L}$

$$\hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_i) = \big[\hat{\epsilon}_l(\mathbf{x}_i, 0), \quad \ldots, \quad \hat{\epsilon}_l(\mathbf{x}_i, 2L-1)\big]^T. \tag{4.19}$$

Using (4.4), the masking matrix $\mathbf{G}_l \in \mathbb{R}^{2L \times 2L}$ is defined as

$$\mathbf{G}_l = \mathrm{diag}\left(\big[\hat{g}_l(0), \quad \ldots, \quad \hat{g}_l(2L-1)\big]\right). \tag{4.20}$$

The vectors (4.13), (4.15) and (4.18) are related to their frequency-domain equivalent through the zero padded Fourier transform. The zero padding is included through a zero-padding matrix $\mathbf{Z} \in \mathbb{R}^{2L\times L}$, which can be expressed as an $L \times L$ identity matrix $\mathbf{I}_L$ and an all-zero matrix $\mathbf{0} \in \mathbb{R}^{L\times L}$. That is,

$$\mathbf{Z} = \begin{bmatrix} \mathbf{I}_L \\ \mathbf{0} \end{bmatrix}. \tag{4.21}$$

The Fourier transform is obtained through the $2L\times 2L$ DFT matrix $\mathbf{W}$. Define $w = \exp\{-j\pi/L\}$. The DFT matrix is then given by [84]

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & w & w^2 & \cdots & w^{2L-1} \\ 1 & w^2 & w^4 & \cdots & w^{2(2L-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{2L-1} & w^{2(2L-1)} & \cdots & w^{(2L-1)(2L-1)} \end{bmatrix} \in \mathbb{C}^{2L\times 2L}. \tag{4.22}$$

The zero-padded DFT matrix can be calculated in advance and is given by

$$\mathbf{W}_Z = \mathbf{W}\mathbf{Z} \in \mathbb{C}^{2L\times L}. \tag{4.23}$$

So that, for example,

$$\hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_i) = \mathbf{W}_Z \boldsymbol{\epsilon}_l(\mathbf{x}_i). \tag{4.24}$$

Obtaining vector (4.12) is somewhat more involved. Let $\mathbf{W}_{Z,k}$ be kth row of matrix $\mathbf{W}_Z$. This allows to write

$$\hat{\mathbf{s}}_l(k) = \begin{bmatrix} \mathbf{W}_{Z,k} & \mathbf{0}_L & \cdots & \mathbf{0}_L \\ \mathbf{0}_L & \mathbf{W}_{Z,k} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0}_L \\ \mathbf{0}_L & \cdots & \mathbf{0}_L & \mathbf{W}_{Z,k} \end{bmatrix} \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix}. \tag{4.25}$$

Here, $\mathbf{0}_L$ is the $1 \times L$ all zero vector.

The components of (4.25) vectors are used repeatedly. Hence, define

$$\mathbf{s}_l = \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix} \in \mathbb{R}^{LN_s\times 1}, \tag{4.26}$$

and

$$\mathbf{K}_k = \begin{bmatrix} \mathbf{W}_{Z,k} & \mathbf{0}_L & \cdots & \mathbf{0}_L \\ \mathbf{0}_L & \mathbf{W}_{Z,k} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0}_L \\ \mathbf{0}_L & \cdots & \mathbf{0}_L & \mathbf{W}_{Z,k} \end{bmatrix} \in \mathbb{C}^{N_s\times LN_s}. \tag{4.27}$$

Now that all the vector- and matrix-definitions are in place, let us return to the general form of the optimisation problem.

4.2.2 General form optimisation problem

Recall that the aim is to minimise the power in region $\mathcal{B}$ while maximising that in $\mathcal{A}$. At the same time, it should be ensured that the audio in region $\mathcal{A}$ resembles the reference audio sufficiently well. This is done using the Par-measure as described in Section D.3. Using (3.18), the solution can be summarised by the following optimisation problem

$$
\begin{aligned}
\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_k w(k) \frac{\hat{\mathbf{s}}_l^{\mathrm{T}}(k)\mathbf{R}_B(k)\hat{\mathbf{s}}_l^*(k)}{\hat{\mathbf{s}}_l^{\mathrm{T}}(k)\mathbf{R}_A(k)\hat{\mathbf{s}}_l^*(k)} \\
\text{subject to} \quad & \left\| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} (\mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) + \hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_i)) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) - \hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_0) \right) \right\|_2^2 \leq d_{\max}, \\
& \mathbf{s}_l = \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix}, \\
& \hat{\mathbf{s}}_l(k) = \mathbf{K}_k \mathbf{s}_l, \qquad \forall\, k, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z \mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\dots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\dots,N_s\}.
\end{aligned}
\tag{4.28}
$$

Here, $w(k)$ is added as an optional term to introduce frequency dependent weighting and the division by $\sqrt{2L}$ is for power conservation through the DFT.

Eq. (4.28) is non-convex due too the quadratic division in the cost function. In each of the six proposed optimisation problems, I modify the cost function to be a sum of quadratics (or related) so that convexity is guaranteed.

Each even-numbered optimisation problem is equal to the preceding odd-numbered optimisation problem, except for a perceptual weighting term in the cost-function. Optimisation problems 1 and 2 are relatively simple and only consider the diagonal of the PSD matrices. Additionally, only the early reflections of the reference signal are considered. Optimisation problems 3 and 4 also only consider the early reflections, but take the non-diagonal terms of the PSD matrices into account as well. Lastly, optimisation problems 5 and 6 are equal to 3 and 4, but take the later reflections into account as well.

4.2.3 Optimisation problem 1

The first optimisation problem is very basic and considers the power delivered by the loudspeakers independently. To achieve this, all non-diagonal terms of the PSD matrices are set to zero. Additionally, the error terms are ignored and the weighting term $w(k)$ is equal for all k.

Recall that the PSD matrices are positive-definite[1]. Thus, the diagonal is real and positive. This allows to define the matrix

$$
\hat{\mathbf{R}}(\mathbf{x}_i) = \operatorname{diag}\left(\left[\frac{\{\mathbf{R}_B\}_{ii}(0)}{\{\mathbf{R}_A\}_{ii}(0)}, \quad \dots, \quad \frac{\{\mathbf{R}_B\}_{ii}(2L-1)}{\{\mathbf{R}_A\}_{ii}(2L-1)} \right] \right). \tag{4.29}
$$

[1] Note that positive-definiteness is not a property of PSD matrices. However, positive-semi-definiteness is. Due to the numerical-inaccuracy term $\mathbf{R}_{\mathrm{num}}$, the PSD matrices can be considered positive-definite.

The optimisation problem becomes

$$\begin{aligned}
\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_{i=1}^{N_s} \hat{\mathbf{s}}_l^{\mathrm{T}}(\mathbf{x}_i)\mathbf{R}(\mathbf{x}_i)\hat{\mathbf{s}}_l^{*}(\mathbf{x}_i) \\
\text{subject to} \quad & \left|\left| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} \mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) \right) \right|\right|_2^2 \leq d_{\max}, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z \mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\dots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\dots,N_s\}.
\end{aligned} \tag{4.30}$$

While not particularly relevant in this specific case, the quadratic form in the cost function has some numerical issues. Namely, due to finite accuracy, the cost value can get small but nonzero complex values. Next to this, in scenarios where the eigenvalues would ideally be slightly positive (or zero), numerical inaccuracies might cause them to become slightly negative. In this case, the PSD matrices are not positive-(semi)definite anymore and the problem becomes non-convex [85]. This can be resolved by rewriting the cost function as a sum of norms. Define the square root of a diagonal matrix as the square root of its individual elements. A related but non-equivalent optimisation problem can be written as

$$\begin{aligned}
\min_{\mathbf{s}_l} \quad & \sum_{i=1}^{N_s} \left|\left| \mathbf{R}^{\frac{1}{2}}(\mathbf{x}_i)\hat{\mathbf{s}}_l^{*}(\mathbf{x}_i) \right|\right|_2 \\
\text{subject to} \quad & \left|\left| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} \mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) \right) \right|\right|_2^2 \leq d_{\max}, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z \mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\dots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\dots,N_s\}.
\end{aligned} \tag{4.31}$$

The reason for using the norm instead of the square norm in the cost function is that it was found to give better results. I am not entirely sure why that is the case. A possibility is that the errors get distributed more evenly, thus giving a better result. Using the norm also "decouples" the loudspeaker in some sense. Namely, when using the squared norm, the error would have been equal to the sum of all bins and their respective weighting squared. However, using the norm, the error is the sum of the within-loudspeaker errors instead. Eq. (4.31) is the equation considered to define optimisation problem 1.

4.2.4 Optimisation problem 2

Optimisation problem 2 is equal to optimisation problem 1, but with the addition of a weighting term. Note that the weighting term gives a way to control the relative importance of the errors. A logical choice is to simply use the inverse of the masking curve. Namely, this curve gives (1) additional weighting according to the frequencies present in the playback signal, (2) gives more importance to those frequencies which can be considered perceptually important from a masking perspective and (3) it is already available. Other viable options are likely to exist as well. For example, investigating the use of localisation based weightings is likely worth it. Here, I will limit

myself to the masking curve. Thus, optimisation problem 2 is given by

$$\begin{aligned}\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_{i=1}^{N_s} \left|\left|\mathbf{G}_l\mathbf{R}^{\frac{1}{2}}(\mathbf{x}_i)\hat{\mathbf{s}}_l^*(\mathbf{x}_i)\right|\right|_2 \\ \text{subject to} \quad & \left|\left|\frac{\mathbf{G}_l}{\sqrt{2L}}\left(\sum_{i=1}^{N_s}\mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0)\right)\right|\right|_2^2 \leq d_{\max}, \\ & \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z\mathbf{s}_l(\mathbf{x}_i), \qquad i\in\{1,\dots,N_s\}, \\ & \mathbf{s}_l(\mathbf{x}_i)\in\mathbb{R}^{L\times 1}, \qquad i\in\{1,\dots,N_s\}.\end{aligned} \tag{4.32}$$

4.2.5 Optimisation problem 3

In optimisation problem 3, I again consider a constant weighting term w. optimisation problem 3 is thus similar to optimisation problem 1. However, the full PSD matrices are taken into account. To do this, a relaxation of the template optimisation problem (4.28) is required. Let there exist a reference solution to $\hat{\mathbf{s}}_l(k)$. Say, $\hat{\boldsymbol{\varsigma}}_l(k)$. This reference could, for example, be obtained through running one of the previous optimisation problems. A convex relaxation of the quadratic over quadratic term is then

$$\hat{\mathbf{s}}_l^{\mathrm{T}}(k)\mathbf{R}_B(k)\hat{\mathbf{s}}_l^*(k) + \left(\hat{\mathbf{s}}_l^{\mathrm{T}}(k) - \hat{\boldsymbol{\varsigma}}_l^{\mathrm{T}}(k)\right)\mathbf{R}_A(k)\left(\hat{\mathbf{s}}_l^*(k) - \hat{\boldsymbol{\varsigma}}_l^*(k)\right). \tag{4.33}$$

Minimising this function should be interpreted as calculating the power in region $\mathcal{B}$, while punishing a large "mistake" in $\mathcal{A}$.

As is done in optimisation problem 1 and 2, I will not use this function directly. Instead, I simply consider the norm of the "square root". This "square root" is obtained through the Cholesky decomposition. The Cholesky decomposition allows to decompose a Hermitian positive-definite matrix $\mathbf{A}$ as [5]

$$\mathbf{A} = \mathbf{Q}^{\mathrm{H}}\mathbf{Q}. \tag{4.34}$$

The PSD matrices are Hermitian positive-definite, and thus this decomposition exists. Let the matrix corresponding to $\mathbf{R}_A(k)$ as $\mathbf{Q}_A(k)$. Similarly, the matrix corresponding to $\mathbf{R}_B(k)$ is denoted $\mathbf{Q}_B(k)$. Optimisation problem 3 is then given by

$$\begin{aligned}\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_{k=0}^{2L-1} ||\mathbf{Q}_B(k)\hat{\mathbf{s}}_l^*(k)||_2 + ||\mathbf{Q}_A(k)\left(\hat{\mathbf{s}}_l^*(k) - \hat{\boldsymbol{\varsigma}}_l^*(k)\right)||_2 \\ \text{subject to} \quad & \left|\left|\frac{\mathbf{G}_l}{\sqrt{2L}}\left(\sum_{i=1}^{N_s}\mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0)\right)\right|\right|_2^2 \leq d_{\max}, \\ & \mathbf{s}_l = \begin{bmatrix}\mathbf{s}_l(\mathbf{x}_1)\\ \mathbf{s}_l(\mathbf{x}_2)\\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s})\end{bmatrix}, \\ & \hat{\mathbf{s}}_l(k) = \mathbf{K}_k\mathbf{s}_l, \qquad \forall\, k, \\ & \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z\mathbf{s}_l(\mathbf{x}_i), \qquad i\in\{1,\dots,N_s\}, \\ & \mathbf{s}_l(\mathbf{x}_i)\in\mathbb{R}^{L\times 1}, \qquad i\in\{1,\dots,N_s\}.\end{aligned} \tag{4.35}$$

4.2.6 Optimisation problem 4

Optimisation problem 4 is equal to optimisation problem 3, but includes the inverse masking curve as additional weighting term. Thus, it is given by

$$\begin{aligned}
\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_{k=0}^{2L-1} ||\hat{g}_l(k)\mathbf{Q}_B(k)\hat{\mathbf{s}}_l^*(k)||_2 + ||\hat{g}_l(k)\mathbf{Q}_A(k)\left(\hat{\mathbf{s}}_l^*(k) - \hat{\boldsymbol{\zeta}}_l^*(k)\right)||_2 \\
\text{subject to} \quad & \left|\left| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} \mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) \right) \right|\right|_2^2 \leq d_{\max}, \\
& \mathbf{s}_l = \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix}, \\
& \hat{\mathbf{s}}_l(k) = \mathbf{K}_k\mathbf{s}_l, \qquad \forall\, k, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z\mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\dots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\dots,N_s\}.
\end{aligned} \tag{4.36}$$

4.2.7 Optimisation problem 5

Optimisation problem 5 is a straightforward extension of problem 3. However, while earlier the filtering of the reference signal by blocks with $\iota > 0$ was ignored, it is included in this algorithm. The additional filtering of the playback signals is left ignored, since it is included through the term $\mathbf{R}_{\text{iso}}$. The optimisation problem is given by

$$\begin{aligned}
\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\dots,N_s\}} \quad & \sum_{k=0}^{2L-1} ||\mathbf{Q}_B(k)\hat{\mathbf{s}}_l^*(k)||_2 + ||\mathbf{Q}_A(k)\left(\hat{\mathbf{s}}_l^*(k) - \hat{\boldsymbol{\zeta}}_l^*(k)\right)||_2 \\
\text{subject to} \quad & \left|\left| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} \mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) - \hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_0) \right) \right|\right|_2^2 \leq d_{\max}, \\
& \mathbf{s}_l = \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix}, \\
& \hat{\mathbf{s}}_l(k) = \mathbf{K}_k\mathbf{s}_l, \qquad \forall\, k, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z\mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\dots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\dots,N_s\}.
\end{aligned} \tag{4.37}$$

4.2.8 Optimisation problem 6

As probably expected by now, optimisation problem 6 is a straightforward extension of problem 5 and is obtained by including the inverse masking term. Thus, it is given by

$$\begin{aligned}
\min_{\mathbf{s}_l(\mathbf{x}_i),\, i\in\{1,\ldots,N_s\}} \quad & \sum_{k=0}^{2L-1} ||\hat{g}_l(k)\mathbf{Q}_B(k)\hat{\mathbf{s}}_l^*(k)||_2 + ||\hat{g}_l(k)\mathbf{Q}_A(k)\left(\hat{\mathbf{s}}_l^*(k) - \hat{\boldsymbol{\varsigma}}_l^*(k)\right)||_2 \\
\text{subject to} \quad & \left|\left| \frac{\mathbf{G}_l}{\sqrt{2L}} \left(\sum_{i=1}^{N_s} \mathbf{A}_0(\mathbf{x}_i)\hat{\mathbf{s}}_l(\mathbf{x}_i) - \mathbf{A}_0(\mathbf{x}_0)\hat{\mathbf{s}}_l(\mathbf{x}_0) - \hat{\boldsymbol{\epsilon}}_l(\mathbf{x}_0) \right) \right|\right|_2^2 \leq d_{\max}, \\
& \mathbf{s}_l = \begin{bmatrix} \mathbf{s}_l(\mathbf{x}_1) \\ \mathbf{s}_l(\mathbf{x}_2) \\ \vdots \\ \mathbf{s}_l(\mathbf{x}_{N_s}) \end{bmatrix}, \\
& \hat{\mathbf{s}}_l(k) = \mathbf{K}_k\mathbf{s}_l, \qquad \forall\, k, \\
& \hat{\mathbf{s}}_l(\mathbf{x}_i) = \mathbf{W}_Z\mathbf{s}_l(\mathbf{x}_i), \qquad i \in \{1,\ldots,N_s\}, \\
& \mathbf{s}_l(\mathbf{x}_i) \in \mathbb{R}^{L\times 1}, \qquad i \in \{1,\ldots,N_s\}.
\end{aligned} \tag{4.38}$$

The six optimisation problems allow to define six algorithms for obtaining the playback signals. The results are presented in the next chapter, where each of the algorithms is referred to as "algorithm" and the number of the used optimisation problem. For example, the algorithm where the playback signals are determined using optimisation problem 3 is referred to as "algorithm 3".

Chapter 5
Results

In this chapter, the results are presented. Firstly, in Section 5.1, the simulation and evaluation methodology is discussed. Then, in Section 5.2 the results are given for a number of different scenarios.

5.1 Evaluation and simulation methodology

The performance of the proposed algorithms is evaluated by means of simulation.

In line with Assumption 5, the considered room has dimensions $L_x \times L_y \times L_z = 6 \times 5.4 \times 3$ m^3, reflection coefficients $\boldsymbol{\beta} = \{-0.3, 0.4, -0.4, 0.3, 0, 0\}$ and the height of the listener and loudspeakers is set to 1.5 m. The length of the simulated RIR is set to 200 ms. Since the floor and ceiling are non-reflective, this is sufficiently long to incorporate the full T_{60} time for the considered room. The virtual source is placed at $\mathbf{x}_0 = (0.821, 1.82, 1.50)$ and the listener at $\mathbf{x}_h = (2.70, 2.50, 1.50)$ m. The physical loudspeakers are placed at $\mathbf{x}_1 \approx (4.70, 2.50, 1.50)$ m, $\mathbf{x}_2 \approx (4.43, 3.50, 1.50)$ m, $\mathbf{x}_3 \approx (4.43, 1.50, 1.50)$ m, $\mathbf{x}_4 \approx (2.02, 4.38, 1.50)$ m and $\mathbf{x}_5 = (2.02, 0.62, 1.50)$ m. These loudspeakers are respectively the Center (C), Left (L), Right (R), Left Side (LS) and Right Side (RS) loudspeakers of the 5.0 system. Note that these loudspeakers are placed on a circle in the xy-plane with a radius of 2 m and centred at $\mathbf{x}_h$. The size of the room and the positions of the loudspeakers and receiver are given in Figure 5.1.

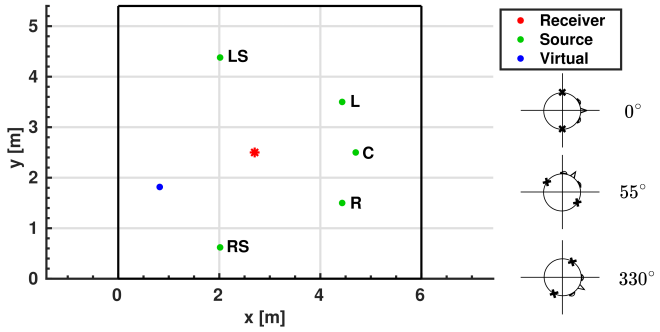
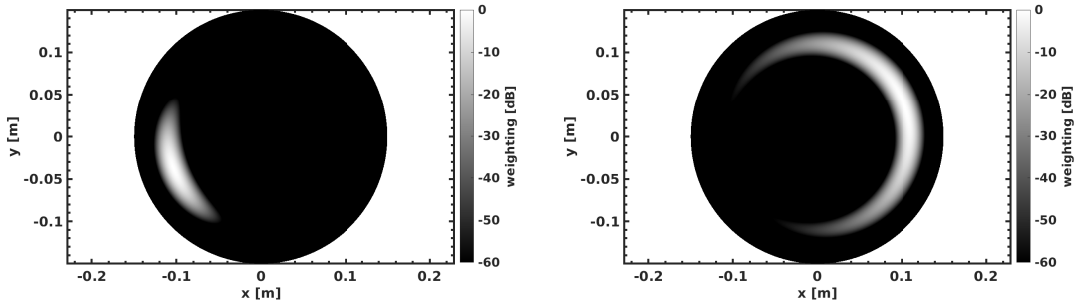


Figure 5.1: The consider problem setup. The receiver (listener), the sources (loudspeakers), the virtual source and some example head orientations are depicted. The physical loudspeakers are indicated as left (L), center (C), right (R), left side (LS) and right side (RS).

For the windowing, square root Hanning windows of 20 ms are used with 50% overlap at both the analysis and synthesis side. The RIR is segmented using a rectangular window of 20 ms length and 0% overlap.

In the calculation of the PSD matrices $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$, image-sources with time-of-arrival up to 20 ms are considered. The spatial weighting function in r is parameterised with $\mu_r = 0.11$ m and $\sigma_r = {}^{0.03}/_{6}$ m. Note that this particular choice of μ_r gives a point of maximum weight which lies just outside the head (a typical head has a radius of about 9 cm) [30]. The von Mises distributions are parameterised with $\kappa_{\mathcal{A}} = {}^{15\pi}/_{2}$, $\kappa_{\mathcal{B}} = {}^{5\pi}/_{4}$, $\mu_{\mathcal{A}} \approx 3.49$ and $\mu_{\mathcal{B}} \approx 3.49 + \pi$. The resulting total distributions (centred around zero) are depicted in Figure 5.2. Note that the values were normalised to a maximum of one. In Appendix L, the individual distributions are plotted without normalisation. Here, also the image-sources considered in the calculation of $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$ are given.



(a) $p_{\mathcal{A}}(\mathbf{x} + \mathbf{x}_h)$, normalised.

(b) $p_{\mathcal{B}}(\mathbf{x} + \mathbf{x}_h)$, normalised.

Figure 5.2: The normalised spatial weighting functions. In (a), $p_{\mathcal{A}}$ is depicted. In (b), $p_{\mathcal{B}}$ is depicted. Note that the distributions are centred $\mathbf{x} = (x, y) = (0, 0)$. Thus, in the normal coordinate system, the spatial weighting functions are centred around the listener.

The RIRs are simulated using a modified version of Habets RIR generator [28]. This version returns a list of reflections, their angle of incidence with respect to the listener, and their time of arrival. This allows to incorporate HRIRs in the room impulse responses. The used HRIRs are those measured by Braren and Fels on the KEMAR head and torso, see [45]. The KEMAR head is a "typical" head and the measurements have a 1 degree resolution [45]. For implementation convenience, nearest neighbour interpolation is used to obtain interpolated HRIRs. This allows to construct the audio signals at the left and right ear, respectively denoted as s_L and s_R.

The audio segments used in the evaluation are MATLABs default signal *gong*, sampled at 8192 Hz and a female-voiced speech segment from the TIMIT database downsampled to 8192 Hz [86]. This downsampling is required, since otherwise the computation times become too long[1].

In all optimisation problems, the maximum allowable distortion $d_{\max} = 25$. The speed of sound is set to $c = 342$ m/s.

Lastly, the result of algorithm 1 is used as initial guess for algorithms 3 and 5. Similarly, the result of algorithm 2 is used as initial guess for algorithms 4 and 6.

5.1.1 Evaluation measures

To quantify the performance of the algorithms, a number of different metrics are considered. Firstly, I consider the difference in received energy between the left and the right ear, measured as the 2-norm. If the algorithm works properly, it is expected that the ear closest to the loudspeaker receives the most energy. A second measure is the time-argument for which the cross-correlation between the signals received at the entrance of the ear canals attains its maximum value. The

[1]It was attempted to perform algorithm 4 on a speech segment sampled at 16 kHz. However, each frame took about two minutes. For algorithm 1 and 2, the time-per-frame was about 8 seconds.

reason that the ITD and ILD of the received signals is not computed directly is because their computation is relatively complex and listener- and frequency-dependent [87, 88]. Additionally, the ITD and ILD can be traded to some extent, so specifically comparing them to some reference value does not give a complete picture [89].

Speech quality is quantified using Perceptual Evaluation Speech Quality (PESQ), defined by ITU-T recommendation P.862 [90], and speech intelligibility is quantified using Speech Intelligibility in Bits (SIIB) [91]. Both PESQ and SIIB take two input signals, namely the degraded audio signal and a reference audio signal. SIIB returns an output with unit bits/s. This estimate is based on the information shared between the degraded and the reference signal [91]. The used implementation is publicly available, see [92]. PESQ returns a value between -0.5 and 4.5, where higher is better [90]. For PESQ, the freely available Python package [93] is used in wideband mode. To do so, the computed playback signals are upsampled to 16 kHz. For both PESQ and SIIB, the reference signal is the signal received at $\mathbf{x}_h$ when playing back from the ideal source location. The degraded signal is the signal obtained at $\mathbf{x}_h$ computed using one of the algorithms.

Recall that s_R is the audio at the right ear and s_L is the audio at the left ear. The cross-correlation is obtained as

$$r(n) = s_L(n) * s_R(-n). \tag{5.1}$$

To obtain the time-difference of arrival, I consider the argument n where this is maximum. However, I only consider arguments within -1 ms and 1 ms, since this is the perceptually relevant range including some margin [94]. I.e.

$$n_{\max} = \arg\max_n r(n), \text{ for } -10^{-3} \leq \frac{n}{f_s} \leq 10^{-3}, \tag{5.2}$$

or, in seconds,

$$t_{\max} = \frac{n_{\max}}{f_s}. \tag{5.3}$$

The difference in energy between the left and the right ear is considered using normalised two-norms. Let s_h be the audio received at the center of the head. The normalised energy E at the ears is given by

$$E_L = \frac{||s_L||_2}{||s_h||_2}, \quad E_R = \frac{||s_R||_2}{||s_h||_2}. \tag{5.4}$$

The performance of the algorithm is compared against three simple reference algorithms. The first algorithm is the ideal solution and is obtained by playing back the reference signal from a loudspeaker located at the location of the virtual source. The second reference algorithm only uses the loudspeaker which is nearest to the virtual source. The third algorithm is a simple amplitude panning algorithm proposed in [11]. This algorithm constrains the placement of loudspeakers to a circle of constant radius around the listener. The reference algorithms are respectively referred to as “Ideal”, “NN” (nearest neighbour) and as “Sadek2004”.

5.2 Results

In this section, the results are presented. Firstly, in Section 5.2.1, I consider the listener placed exactly at $\mathbf{x}_h$. These results are considered for listeners oriented at angles in $\{0°, 1°, \ldots, 359°\}$. Then, in Section 5.2.2, the listener is slightly misplaced from the expected value. These results are obtained for listeners oriented at angles 0°, 55° and 330°. Note that these are the same orientations as those shown in Figure 5.1. All results are presented for both the gong signal and the female-voiced speech signal.

5.2.1 Results for exact listener placement

In this section, the results are given for the scenario in which the listener is exactly placed at $\mathbf{x}_h$.

Firstly, let us consider the time $t_{\max}$ at which the cross-correlation attains its maximum value. For the female-voiced speech signal, $t_{\max}$ is given in Figure 5.3. Since the results for the gong signal are similar, they are left out of the main text and given in Appendix L.

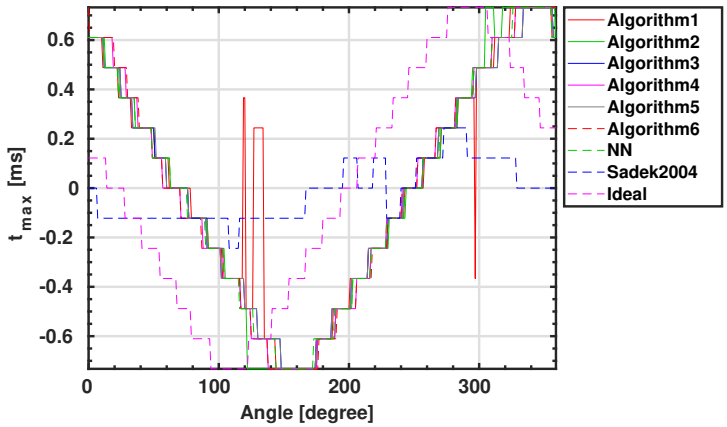


Figure 5.3: The argument $t_{\max}$ (in ms) for which the cross-correlation between the audio received at the left and at the right ear attains its maximum value. The considered signal is the female-voiced speech signal. The staircase shape is due to the limited sample rate.

As can be seen, all algorithms except Sadek2004 have a clear shape with peak values of about +0.7 ms and -0.7 ms. This is the maximum attainable ITD and relates to the size of the head [30]. It is expected that Sadek2004 does not show the peaks as clearly as the other algorithms, since it only uses amplitude panning. The $t_{\max}$ curve of Algorithm 1 up to and including Algorithm 6 coincide with that of the NN algorithm. Thus, it seems that the proposed algorithms favour the nearest neighbour. This can be verified by considering what fraction of the total transmitted energy each of the loudspeakers transmits. For the female-voiced speech signal, this is depicted in Table 5.1. For the gong, it is depicted in Table 5.2.

Table 5.1: The fraction of the energy transmitted by each loudspeaker with respect to the total transmitted energy. The signal considered is female-voiced speech. Note that each row sums to one up to round-off errors. Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	C	L	R	LS	RS
Alg. 1	0.049	0.025	0.267	0.191	0.468
Alg. 2	0.004	0.008	0.189	0.230	0.570
Alg. 3	0.182	0.108	0.218	0.161	0.331
Alg. 4	0.151	0.098	0.185	0.179	0.388
Alg. 5	0.182	0.111	0.213	0.162	0.332
Alg. 6	0.157	0.102	0.178	0.177	0.389
NN	0.000	0.000	0.000	0.000	1.000
Sad.	0.013	0.003	0.078	0.343	0.563

Table 5.2: The fraction of the energy transmitted by each loudspeaker with respect to the total transmitted energy. The signal considered is gong.Note that each row sums to one up to round-off errors. Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	C	L	R	LS	RS
Alg. 1	0.011	0.011	0.068	0.214	0.697
Alg. 2	0.001	0.000	0.030	0.203	0.766
Alg. 3	0.162	0.097	0.138	0.162	0.442
Alg. 4	0.139	0.098	0.130	0.162	0.471
Alg. 5	0.165	0.098	0.140	0.160	0.437
Alg. 6	0.144	0.098	0.132	0.161	0.464
NN	0.000	0.000	0.000	0.000	1.000
Sad.	0.013	0.003	0.078	0.343	0.563

As expected, the loudspeaker closest to the virtual source transmits most of the energy. For the algorithms in which the full correlation matrices are taken into account, this difference is less pronounced.

Now let us consider the results for the energy difference $E_L - E_R$. The results of the female-voiced speech signal are given in Figure 5.4. The individual energies E_L and E_R are given in Appendix L. Since the results for the gong signal are comparable, they are given in Appendix L as well.

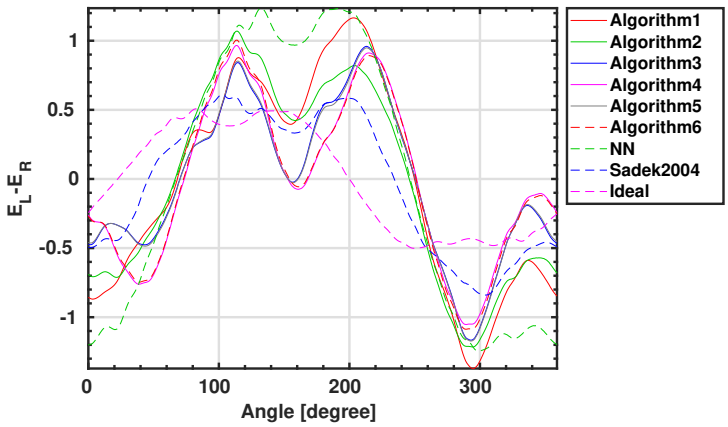


Figure 5.4: The difference in energy received at the left and right ear $E_L - E_R$. The considered signal is the female-voiced speech signal.

As is the case for the cross-correlation, the curves have the expected shape. The energy at the left ear w.r.t that at the right ear increases as the left ear moves towards the virtual source and vice versa. However, as also is the case for the cross-correlation, the curves tend to be shifted towards the nearest-neighbour.

The results of Scenario 1 are summarised in Table 5.3. Here, the left side of the table gives the results for the female-voiced speech signal and the right side gives the results for the gong signal. For both signals, the mean error with respect to the ideal result is given and the corresponding standard deviation is indicated. The error is calculated by considering the difference of the

result of the algorithm with the ideal signal. The mean error μ_t and the corresponding standard deviation σ_t are given for $t_{\max}$. The mean error μ_E and the corresponding standard deviation σ_E are given for $E_L - E_R$. These errors are calculated over the angles. For the speech signal, the PESQ and SIIB values are given as well.

Table 5.3: The results of Scenario 1. The mean error of with respect to the ideal result for all of the algorithms. The value μ_t and σ_t indicate the mean and standard deviation for $t_{\max}$. The value μ_E and σ_E indicate the mean and standard deviation for $E_L - E_R$. The left side of the table gives the results for the female-voiced speech signal. The right side gives the results for the gong. For the speech signal, the PESQ and SIIB value is also given. The mean error and the standard deviation are calculated over the angles $\{0°, 1°, \ldots, 359°\}$. The SIIB of the ideal speech signal w.r.t. itself is 1335.8 b/s (bits/second). Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	Speech				Gong	
	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$	PESQ	SIIB (b/s)	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$
Alg. 1	$+19.0 \pm 432.1$	-0.0211 ± 0.6195	1.547	496.6	$+3.4 \pm 411.3$	-0.0102 ± 0.5929
Alg. 2	-4.1 ± 407.8	-0.0176 ± 0.5323	3.773	758.8	-0.7 ± 407.6	-0.0048 ± 0.6339
Alg. 3	-3.1 ± 397.9	-0.0094 ± 0.5104	1.417	453.2	$+5.8 \pm 397.0$	-0.0002 ± 0.5163
Alg. 4	-3.1 ± 401.5	-0.0132 ± 0.5633	3.316	686.7	-0.7 ± 398.2	-0.0026 ± 0.5639
Alg. 5	-2.7 ± 398.3	-0.0113 ± 0.5108	1.411	454.3	$+5.1 \pm 397.0$	-0.0007 ± 0.5163
Alg. 6	-3.4 ± 404.3	-0.0129 ± 0.5601	3.255	681.1	-0.7 ± 397.4	-0.0035 ± 0.5581
NN	-3.1 ± 399.3	-0.0031 ± 0.7621	4.222	780.0	$+1.4 \pm 413.8$	-0.0041 ± 0.7233
Sad.	-24.4 ± 378.8	-0.0186 ± 0.2941	4.302	818.3	$+40.4 \pm 391.1$	-0.0061 ± 0.3464

From the table, it is found that all algorithms except Sadek2004 and Algorithm 1 have a comparable error in $t_{\max}$. Sadek2004 has a larger mean error but a comparable standard deviation and Algorithm 1 has a larger mean error and standard deviation in the speech signal only. This is likely due to the outliers as observed in Figure 5.3. For $E_L - E_R$, all algorithms except Sadek2004 have a similar mean error. However, the standard deviation of Sadek2004 is smaller then that of the others. For PESQ and SIIB, the newly proposed algorithms perform less then the reference algorithms. Algorithms 1, 3 and 5 perform less then Algorithms 2, 4 and 6. This can be attributed to the even-numbered algorithms taking into account the masking curve in the cost function.

5.2.2 Results for varying listener placement

All the previous results were obtained for an ideal room and for the listener in the ideal location. Let us now consider what happens when small deviations are introduced in the placement of the listener. The results are obtained by sampling 100 locations $X_h = x_h + X$ and $Y_h = y_h + Y$. Here, X and Y are independent normal distributed random variables with mean 0 and a standard deviation of 5 cm. The z coordinate is kept unchanged. In tables 5.4, 5.6 and 5.5, the results are respectively given for a head orientations of 0°, 55° and 330°. Note that these are the orientations shown in Figure 5.1. It should also be noted that, for each of the orientations and for both the speech and the gong signal, the same set of head locations was used.

Table 5.4: The results for a head orientation of 0° and a standard deviation in listener location of 5 cm. The results are calculated over 100 runs. The mean error of with respect to the ideal result is indicated for all of the algorithms. The value μ_t and σ_t indicate the mean and standard deviation for $t_{\max}$. The value μ_E and σ_E indicate the mean and standard deviation for $E_L - E_R$. The left side of the table gives the results for the female-voiced speech signal. The right side gives the results for the gong. The mean error and the standard deviation are calculated over the different listener location samples. Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	Speech		Gong	
	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$
Alg. 1	$+441.9 \pm 84.6$	-0.277 ± 0.099	$+487.1 \pm 53.5$	-0.380 ± 0.117
Alg. 2	$+454.1 \pm 71.7$	-0.222 ± 0.112	$+504.2 \pm 59.2$	-0.520 ± 0.160
Alg. 3	$+213.6 \pm 456.5$	$+0.042 \pm 0.206$	$+479.7 \pm 189.5$	-0.142 ± 0.143
Alg. 4	$+487.1 \pm 122.1$	$+0.058 \pm 0.254$	$+424.8 \pm 302.3$	-0.254 ± 0.208
Alg. 5	$+146.5 \pm 487.4$	$+0.046 \pm 0.211$	$+472.4 \pm 196.8$	-0.140 ± 0.144
Alg. 6	$+491.9 \pm 128.0$	$+0.051 \pm 0.255$	$+415.0 \pm 306.5$	-0.239 ± 0.206
NN	$+444.3 \pm 58.9$	-0.632 ± 0.040	$+474.9 \pm 42.1$	-0.679 ± 0.057
Sad.	-216.1 ± 387.9	$+0.028 \pm 0.215$	-146.5 ± 354.3	-0.085 ± 0.212

As can be seen in Table 5.4, for a head orientation of zero degrees, the mean error μ_t is typically somewhere around 400 to 500 us. This is to be expected, since this resembles the error which is present in the ideal case, see Figure 5.2. Interestingly, for the speech signal, the mean error μ_t is lower for algorithm 3, algorithm 5 and Sadek2004. However, they have a larger standard deviation σ_t, indicating that they are less robust to listener displacement. This larger standard deviation is also observed in the results with gong. Algorithm 1, algorithm 2 and NN show the smallest values of σ_t. This is to be expected, since, for these algorithms, the majority of the energy is transmitted by the nearest-neighbouring loudspeaker (see tables 5.1 and 5.2).

For the mean error in energy difference μ_E, the NN algorithm has the largest (absolute) mean error, followed by algorithm 1 and algorithm 2. Similarly, NN has the smallest standard deviation, followed by algorithm 1 and 2. Sadek2004 has the smallest (absolute) mean error μ_E for both the gong and the speech signal. Comparing Sadek2004 with algorithm 3 up to and including algorithm 6, it is found that, for the speech signal, the error μ_E of algorithms 3, 4, 5, and 6 is about twice as large. Additionally, the standard deviation is comparable. For the gong, the mean error is two to three times as large, while the standard deviation of algorithm 4 and algorithm 6 is comparable. For algorithms 3 and 5, the standard deviation is smaller.

Let us now consider the results for a head orientation of 55°, shown in Table 5.5. The performance for the gong signal is similar for all algorithms except Sadek2004. Sadek2004 has

Table 5.5: The results for an head orientation of 55° and a standard deviation in listener location of 5 cm. The results are calculated over 100 runs. The mean error of with respect to the ideal result is indicated for all of the algorithms. The value μ_t and σ_t indicate the mean and standard deviation for $t_{\max}$. The value μ_E and σ_E indicate the mean and standard deviation for $E_L - E_R$. The left side of the table gives the results for the female-voiced speech signal. The right side gives the results for the gong. The mean error and the standard deviation are calculated over the different runs. Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	Speech		Gong	
	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$
Alg. 1	$+438.2 \pm 60.3$	-1.019 ± 0.090	$+416.3 \pm 60.3$	-1.016 ± 0.153
Alg. 2	$+433.3 \pm 111.6$	-1.026 ± 0.095	$+416.3 \pm 60.3$	-1.137 ± 0.148
Alg. 3	$+430.9 \pm 242.6$	-1.040 ± 0.162	$+410.2 \pm 66.1$	-1.097 ± 0.183
Alg. 4	$+437.0 \pm 214.0$	-1.149 ± 0.275	$+416.3 \pm 63.0$	-1.187 ± 0.192
Alg. 5	$+443.1 \pm 236.4$	-1.048 ± 0.162	$+410.2 \pm 66.2$	-1.103 ± 0.182
Alg. 6	$+433.3 \pm 177.3$	-1.145 ± 0.273	$+417.5 \pm 63.0$	-1.183 ± 0.189
NN	$+438.2 \pm 60.3$	-1.204 ± 0.057	$+416.3 \pm 60.3$	-1.227 ± 0.086
Sad.	$+151.4 \pm 343.1$	-0.811 ± 0.463	$+242.9 \pm 324.8$	-0.902 ± 0.272

a smaller mean error μ_t but a larger standard deviation σ_t, indicating that it is less robust with respect to listener displacement. Furthermore, still for the gong signal, Sadek2004 has the smallest mean error μ_E, but the largest standard deviation. The mean error μ_E and the standard deviation σ_E is similar for all other algorithms.

For the speech signal, the same story holds. However, there is an exception. Namely, the standard deviations of NN, algorithm 1 and algorithm 2 are lower than that of the other algorithms, while the mean error is comparable.

Let us now consider the results for a head orientation of 330°, shown in Table 5.6. Here, the NN algorithm performs very well in μ_E and σ_E, for both the speech and gong signal. This is likely because the nearest neighbouring loudspeaker (RS) is placed at an angle of 250°, the virtual source at an angle of 200° and the right ear at an angle of 240°. Thus, the ear receiving most energy is placed with only 30° difference between the nearest neighbour and the virtual source, making the received signals similar.

Lastly, the average run-time per frame of the algorithm is given in Table 5.7. Note that Sadek2004 and NN do not operate on a frame-by-frame basis. Thus, they are left out. However, it should be noted that these algorithms could be run in real-time due to their simplicity. It should be noted that these times are merely to give an indication. Namely, at times, the laptop was in use while calculating the runtimes, and the runtimes are only calculated over a single run. All experiments are performed on a laptop with 24 GB of RAM and an Intel i5-8250U CPU clocked at 1.60 GHz. As can be seen, algorithm 1 and algorithm 2 are significantly faster than algorithms 3 to 6. Additionally, it seems that the odd-numbered algorithms take slightly longer then the even-numbered algorithms, though this is not observed in algorithm 6 for the speech signal.

Table 5.6: The results for an head orientation of 330° and a standard deviation in listener location of 5 cm. The results are calculated over 100 runs. The mean error of with respect to the ideal result is indicated for all of the algorithms. The value μ_t and σ_t indicate the mean and standard deviation for $t_{\max}$. The value μ_E and σ_E indicate the mean and standard deviation for $E_L - E_R$. The left side of the table gives the results for the female-voiced speech signal. The right side gives the results for the gong. The mean error and the standard deviation are calculated over the different runs. Algorithm is abbreviated as Alg. Sadek2004 is abbreviated as Sad.

	Speech		Gong	
	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$	$\mu_t \pm \sigma_t$ (us)	$\mu_E \pm \sigma_E$
Alg. 1	$+242.9 \pm 12.2$	$+0.235 \pm 0.125$	$+334.5 \pm 53.8$	$+0.182 \pm 0.127$
Alg. 2	$+241.7 \pm 17.2$	$+0.259 \pm 0.123$	$+295.4 \pm 79.8$	$+0.158 \pm 0.187$
Alg. 3	$+90.3 \pm 362.5$	$+0.501 \pm 0.200$	$+296.6 \pm 78.0$	$+0.432 \pm 0.173$
Alg. 4	$+216.1 \pm 54.5$	$+0.526 \pm 0.213$	$+279.5 \pm 78.2$	$+0.409 \pm 0.259$
Alg. 5	$+79.3 \pm 377.9$	$+0.506 \pm 0.201$	$+296.6 \pm 78.0$	$+0.443 \pm 0.173$
Alg. 6	$+223.4 \pm 52.2$	$+0.515 \pm 0.208$	$+278.3 \pm 77.7$	$+0.427 \pm 0.255$
NN	$+244.1 \pm 0.00$	-0.068 ± 0.019	$+334.5 \pm 56.5$	$+0.071 \pm 0.049$
Sad.	-212.4 ± 410.8	$+0.445 \pm 0.198$	$+3.7 \pm 393.5$	$+0.543 \pm 0.264$

Table 5.7: The average per-frame runtime in seconds of the six proposed algorithms. Recall that each frame has a length of 20 ms. The gong signal and the speech signal respectively have 512 and 396 frames.

	Alg. 1	Alg. 2	Alg. 3	Alg. 4	Alg. 5	Alg. 6
Speech	2.09	1.83	18.35	17.30	18.11	18.75
Gong	2.05	1.83	18.42	17.35	17.84	17.38

Chapter 6
Conclusion

In this thesis, an algorithm to reproduce or synthesise audio including spatial cues in empty rooms using a limited number of loudspeakers was presented. The algorithm assumes knowledge of the listener location, knowledge of the room and knowledge of the loudspeaker locations. By computing the power spectral density matrices in two regions $\mathcal{A}$ and $\mathcal{B}$ partially surrounding the listener, the energy in region $\mathcal{B}$ can be minimised with respect to the energy in region $\mathcal{A}$. In order to control how perceptually similar the received signal and the target signal are, an auditory masking measure was incorporated. Additionally, this masking measure was also used to introduce frequency-dependent weighting into the energy-ratio minimisation. The performance of the algorithms was compared against two reference algorithms. Namely, the nearest-neighbour algorithm and a simple amplitude panning algorithm proposed in [11]. The results do not indicate a clear preference for one of the algorithms in terms of the considered evaluation metrics.

The results, presented in Chapter 5, allow to partially answer the research questions. First, recall Research Question 1,

Research Question 1 *Can spatial weighting be used in combination with beamforming to synthesise audio containing spatial cues?*

While the results do not explicitly test for the spatial cues, they indicate that Research Question 1 can be answered by a partial yes. Namely, the difference in energy received at each of the ears indicates an ILD cue and the time for which the cross-correlation is largest indicates an ITD difference. For the considered situation, however, the quality of the results is dependent on the orientation of the head and the results resemble the results obtained from a nearest neighbour algorithm. Further research, including subjective tests, is required to give a definite answer to this research question.

Now recall Research Question 2,

Research Question 2 *Can properties of the human hearing be used to improve the performance of the algorithm?*

The results for Perceptual Evaluation Speech Quality (PESQ) and Speech Intelligibility in Bits (SIIB) indicate that this research question can be answered positively. Namely, when including the inverse masking curve as a weighting term, the SIIB and PESQ improve significantly, while the other metrics remain similar.

Chapter 7
Discussion and Future Work

In this chapter, the limitations of the presented work and some possibilities of future work are given.

Firstly, due to time-constraints, the results were only investigated for a perfectly known room and fixed loudspeaker locations. The robustness of the algorithm should be investigated for deviations from the ideal room and deviations from the expected loudspeaker locations. A related topic of future work is the choice of spatial weighting functions and corresponding parameters. Namely, it is of interest to see how different choices affect the accuracy and/or robustness of the algorithm. A specific example which is of interest is how the azimuthal "width" of the regions influences the performance. For example, the performance might differ if region $\mathcal{A}$ is chosen smaller so that it does not have significant weight on the direct path from a physical loudspeaker to a listener. Additionally, a larger width in r might improve the robustness to system deviations. Furthermore, it is interesting to research how reducing or increasing the number of loudspeakers could improve the performance and how the algorithm performs in real-life scenarios.

A second limitation of the described work is that the assumptions which were used to limit the study are not representative for real world scenarios. Namely, (1) the floor and ceiling were assumed to be fully absorbing, (2) the virtual source, loudspeakers and listener were assumed to be at the same height and (3) the loudspeakers were assumed isotropic. In practical scenarios, these assumptions are invalid. Thus, it should be investigated how a deviation from these assumptions influence the performance of the algorithm. In particular, spatial weighting functions for the z-coordinate (or possibly ϕ-coordinate) should be chosen and it should be investigated how the reflections on the floor and ceiling influence the algorithms performance. Additionally, loudspeaker directivity should be incorporated. Assuming that the directivity pattern is available, this can straightforwardly be done through the use of frequency dependent reflection coefficients $\beta_{i,\xi}(\omega)$ (see (3.19)).

Thirdly, improvements might be made in the computation of the power spectral density matrices. Namely, the parameters σ^2_{iso} and σ^2_{num} of the power spectral density matrices were set to zero and to 10^{-12}, respectively. These values can likely be improved. For the former, possible approaches to do so in real-life scenarios are given in [83]. Furthermore, due to the oscillatory nature of the integral solved in the calculation of $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$, the computation becomes time-expensive for high ω. A possible solution which allows for solving oscillatory integrals is given by [95]. This approach is worked-out for a three-dimensional integral in Appendix I. Alternatively, it is worth investigating if the integrals can (partially) be solved analytically or if integration can be avoided entirely. The latter may be done by employing a stochastic characterisation of the RIR. A step towards this stochastic characterisation of the room impulse response in small rooms is taken and given in Appendix J.

A fourth, and perhaps the most important, limitation is that the current approach to characterising the performance of the algorithms is very crude. The only way to obtain an entirely proper characterisation of the results would be through performing subjective tests. Due to time-constraints, it was chosen to focus on the development of the algorithm instead. However, it is important to perform subjective tests in possible future work. These subjective tests should also

indicate if the use of the Par-measure was a proper choice or if the possible pre-echoes are an annoying artefact.

Furthermore, currently only a single virtual source is synthesised. While it seems straightforward that this can be extended to multiple sources, it should be investigated if this is true.

Next, currently only the performance for sound segments sampled at 8192 Hz is investigated. The performance of the algorithms should also be investigated for higher sample-rates as used in, for example, music.

Another recommendation is to investigate possible approaches to speed up the computation times corresponding to the optimisation problem. A possible option is to optimise per (weighted) set of frequency bins instead of per frequency bin. Ideally, this should be done in a way which makes sense perceptually. A different method o speed-up the computation times and which can be done relatively straightforwardly is to explicitly write the problem as a second-order conical program. This has some advantages associated with it, see [85]. Lastly, the complexity could be reduced if a method is found to avoid the conversion from discrete-time domain to discrete-frequency domain in the optimisation problems.

Additionally, currently the algorithm does not directly relate to the spatial cues. Instead, the spatial cues are assumed to happen implicitly through the beamformer. A major topic of future research is to develop a cost-function which directly relates to the spatial cues and can be optimised for. To the best of my knowledge, this does not yet exist.

Appendix A
The Wave Equation

The three-dimensional wave equation considers the deviation $p(\mathbf{x}, t)$ (with $\mathbf{x} = (x, y, z)$) of some pressure around its equilibrium value. For a source-less domain of interest, this pressure can be described by the homogeneous scalar wave equation,

$$\nabla^2 p - \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} = 0, \tag{A.1}$$

with ∇^2 the Laplacian operator, given by

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}. \tag{A.2}$$

From a physical perspective, this equation is valid for a constant valued c and inside a homogeneous, lossless and linear domain [4, 58]. It has been shown that these conditions are met sufficiently well for sound levels below the threshold of pain for humans [4]. In the case that sources are placed inside the domain of interest, the right hand side of (A.1) will take on a non-zero and, generally, time-dependent value describing the source [58].

Now, supposing that it exists, consider the temporal frequency domain representation $\hat{p}(\mathbf{x}, \omega) = \mathcal{F}(p(\mathbf{x}))(\omega)$. Using (2), the inverse Fourier transform is

$$p(\mathbf{x}, t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{p}(\mathbf{x}, \omega) e^{j\omega t} d\omega. \tag{A.3}$$

By noting that the wave-equation is a linear partial differential equation and by substituting (A.3) into (A.1) we obtain

$$\nabla^2 \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{p}(\mathbf{x}, \omega) e^{j\omega t} d\omega - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{p}(\mathbf{x}, \omega) e^{j\omega t} d\omega = 0. \tag{A.4}$$

Simplifying yields

$$\frac{1}{2\pi}\int_{-\infty}^{\infty} \nabla^2 \hat{p}(\mathbf{x}, \omega) e^{j\omega t} d\omega + \frac{\omega^2}{c^2}\frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{p}(\mathbf{x}, \omega) e^{j\omega t} d\omega = 0. \tag{A.5}$$

The above equation can be rewritten as the inverse temporal Fourier transform of the homogeneous Helmholtz equation. Namely,

$$\frac{1}{2\pi}\int_{-\infty}^{\infty} \left(\nabla^2 \hat{p}(\mathbf{x}, \omega) + \frac{\omega^2}{c^2}\hat{p}(\mathbf{x}, \omega)\right) e^{j\omega t} d\omega = 0. \tag{A.6}$$

So, the homogeneous Helmholtz equation is given by

$$\nabla^2 \hat{p}(\mathbf{x}, \omega) + \frac{\omega^2}{c^2}\hat{p}(\mathbf{x}, \omega) = 0. \tag{A.7}$$

Alternatively, one can also take the Fourier transform of the wave equation directly. The value ω/c is known as the wavenumber and typically denoted by k [58]. Thus, the "typical" form of the homogeneous Helmholtz equation is

$$\nabla^2 \hat{p}(\mathbf{x}, \omega) + k^2 \hat{p}(\mathbf{x}, \omega) = 0. \tag{A.8}$$

By adding a source term $s(\mathbf{x}, t)$ to (A.1) one obtains the inhomogeneous wave-equation,

$$\nabla^2 p - \frac{1}{c^2} \frac{\partial^2 p}{\partial t^2} = -s(\mathbf{x}, t). \tag{A.9}$$

Assuming that $s(\mathbf{x}, t)$ has a Fourier transform $\hat{s}(\mathbf{x}, \omega)$, one can obtain the inhomogeneous Helmholtz equation using similar steps as before [96]. This gives

$$\nabla^2 \hat{p}(\mathbf{x}, \omega) + k^2 \hat{p}(\mathbf{x}, \omega) = -\hat{s}(\mathbf{x}, \omega). \tag{A.10}$$

When s is available, one can solve the above equation for p. However, it is inefficient to do so for each new s. A more efficient approach exists and follows from the notion that (A.10) is linear. The idea behind this approach is outlined below.

A.1 The Green's function

Consider a linear differential operator $\mathcal{L}$. This operator acts on a function Ψ, defining a differential equation

$$\mathcal{L}\Psi(\mathbf{x}, t) = f(\mathbf{x}, t). \tag{A.11}$$

For example, in the wave equation, $\mathcal{L} = -\nabla^2 + \frac{1}{c^2}\frac{\partial^2}{\partial t^2}$, $\Psi(\mathbf{x}, \omega) = p(\mathbf{x}, t)$ and $f(\mathbf{x}, \omega) = s(\mathbf{x}, t)$. The idea behind the use of Green's functions is now that, if one can solve

$$\mathcal{L}G(\mathbf{x}, \mathbf{x}', t) = \delta(\mathbf{x} - \mathbf{x}')\delta(t - t'), \tag{A.12}$$

a solution $\Psi(\mathbf{x}, t)$ to (A.11) is straightforwardly obtained through the convolution [97, 98]

$$\Psi(\mathbf{x}, t) = \int_{\mathbf{x}'} \int_{t'} G(\mathbf{x}, \mathbf{x}', t) f(\mathbf{x}', t') d\mathbf{x}' dt'. \tag{A.13}$$

The delta functions in the above are Dirac delta functions.

A.2 The Green's function solution to the wave-equation

Let $\hat{g}(\mathbf{x}, \mathbf{x}', \omega)$ be the Green's function solution to the Helmholtz equation. By considering the temporal Fourier transform, it is required to find a solution which satisfies

$$\nabla^2 \hat{g}(\mathbf{x}, \mathbf{x}', \omega) + k^2 \hat{g}(\mathbf{x}, \mathbf{x}', \omega) = -\delta(\mathbf{x} - \mathbf{x}'). \tag{A.14}$$

Note that the $\delta(t)$ transforms to 1. By requiring causality and by requiring that $\hat{g}(\mathbf{x}, \mathbf{x}', \omega) \to 0$ as $||\mathbf{x} - \mathbf{x}'||_2 \to \infty$, it can be shown that [98]

$$\hat{g}(\mathbf{x}, \mathbf{x}', \omega) = \frac{e^{-jk||\mathbf{x}-\mathbf{x}'||_2}}{4\pi||\mathbf{x} - \mathbf{x}'||_2}. \tag{A.15}$$

For a complete description, the reader can, among others, refer to [98]. Note that [98] uses a different definition of the Fourier transform and thus arrives at a slightly different Green's function.

Appendix B
Directive Transmitter and Receiver

In this appendix, the equation for the received signal in the free field as described in Section 2.2 is extended to incorporate a directive transmitter and a directive receiver.

First, recall (2.4), which states that the received sound $s_r(t)$ for isotropic transmitters i and an isotropic receiver in the free-field is given by

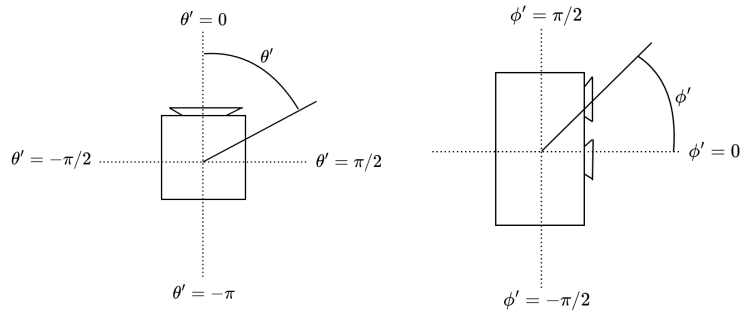
$$s_r(t) = \sum_{i=1}^{N_s} \left(h(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i)\right)(t) = \sum_{i=1}^{N_s} (g(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i))(t). \tag{B.1}$$

Here, N_s is the number of loudspeakers and g is the greens function solution to the scalar wave-equation, given by

$$g(\mathbf{x}_i, \mathbf{x}_r, t) = \frac{1}{4\pi||\mathbf{x}_i - \mathbf{x}_r||_2} \delta\left(t - \frac{||\mathbf{x}_i - \mathbf{x}_r||_2}{c}\right). \tag{B.2}$$

B.1 Directive transmitter

Now suppose that the receiver is isotropic, but the transmitter is directive. Let the directivity impulse response be described by $h_{\text{dir}}(\theta_i', \phi_i', t)$[1], where the angles θ_i' and ϕ_i' are relative to the transmitter. The coordinate system is illustrated in Figure B.1.



(a) Azimuth θ' (b) Elevation ϕ'

Figure B.1: A schematic view of the coordinate system used for the transmitter directivity. The azimuth $\theta' \in [0, 2\pi)$ and elevation $\phi' \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ are calculated relative to the transmitter orientation.

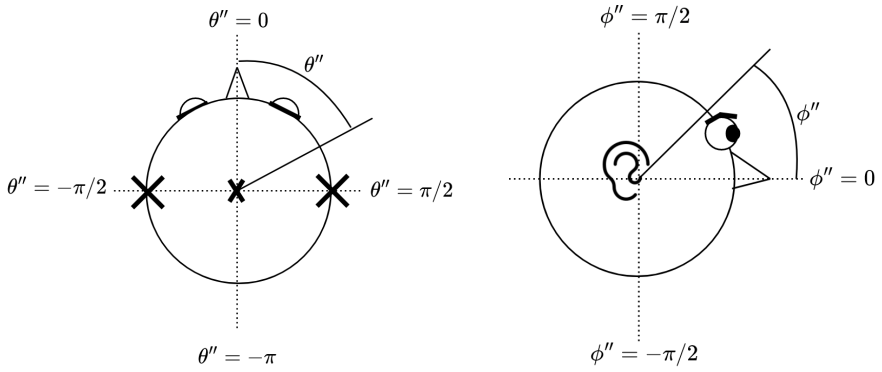
[1] Note that each transmitter is assumed to have the same directivity pattern.

We may associate a pair (θ_i', ϕ_i') to each transmitter i. Adding the directivity to (2.4) yields

$$s_r(t) = \sum_{i=1}^{N_s} \left(h_{\text{dir}}(\theta_i', \phi_i') * g(\mathbf{x}_i, \mathbf{x}_r) * s_i\right)(t). \tag{B.3}$$

B.2 Directive transmitter and receiver

Let us now consider a listener as a receiver. The center of the listeners head is located at $\mathbf{x}_r$ and the directivity of the listener is expressed using the HRIRs $h_{\text{H},L}(\theta'', \phi'', t)$ and $h_{\text{H},R}(\theta'', \phi'', t)$ (where L corresponds to the left ear, and R to the right ear). Note that the HRIRs are considered to be independent of distance. The angles θ'' and ϕ'' are defined relative to the listener. The corresponding coordinate system is illustrated in Figure B.2.



(a) Azimuth θ'' (b) Elevation ϕ''

Figure B.2: A schematic view of the coordinate system used for the receiver (or listener) directivity. The azimuth $\theta'' \in [0, 2\pi)$ and elevation $\phi'' \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ are calculated relative to the reciever orientation.

Incorporating the human receiver and the directive transmitter into the example of Figure 2.8 results in Figure B.3.

The received signal now differs between the left and right ear. For the left ear, it is calculated as

$$s_L(t) = \sum_{i=1}^{N_s} \left(h_{\text{dir}}(\theta_i', \phi_i') * g(\mathbf{x}_i, \mathbf{x}_r) * h_{H,L}(\theta_i'', \phi_i'') * s_i\right)(t), \tag{B.4}$$

a similar expression holds for the right ear.

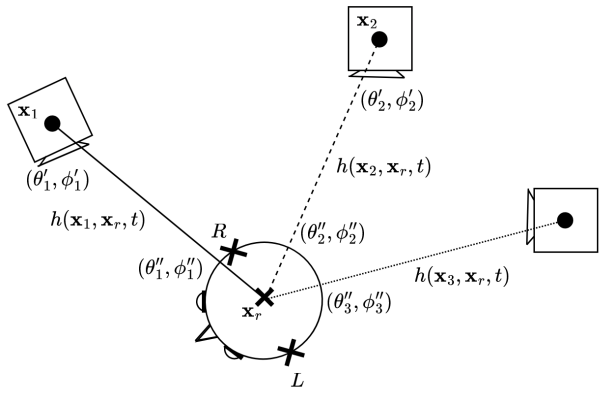


Figure B.3: An example situation where the sound produced by three loudspeakers centred at coordinate $\mathbf{x}_i$ in the free field is received by a human. The channel from source i to receiver r is denoted as $h_r(\mathbf{x}_i, t)$ and the center of the head is denoted as $\mathbf{x}_r$. Each transmitter has an azimuth and elevation (θ', ϕ') associated with it and each receiver has an azimuth and elevation (θ'', ϕ'') associated with it.

Appendix C
The image-source method

In this appendix, the details of the image-source method are discussed in some more detail. The image-source methods was proposed [59] and briefly mentioned in Section 2.2.2. The image-source method allows to estimate the acoustic channel h in box shaped rooms as a sum of weighted Green's functions.

Recall that the image-source method is a form of geometrical acoustics. Thus, the sound-waves are assumed to behave as rays. Due to the specular reflection, the locations of the image-sources are found by mirroring the physical source along the wall [61]. This is illustrated in Figure C.1. Note that this mirroring can be repeated indefinitely. Thus, there are infinitely many image-

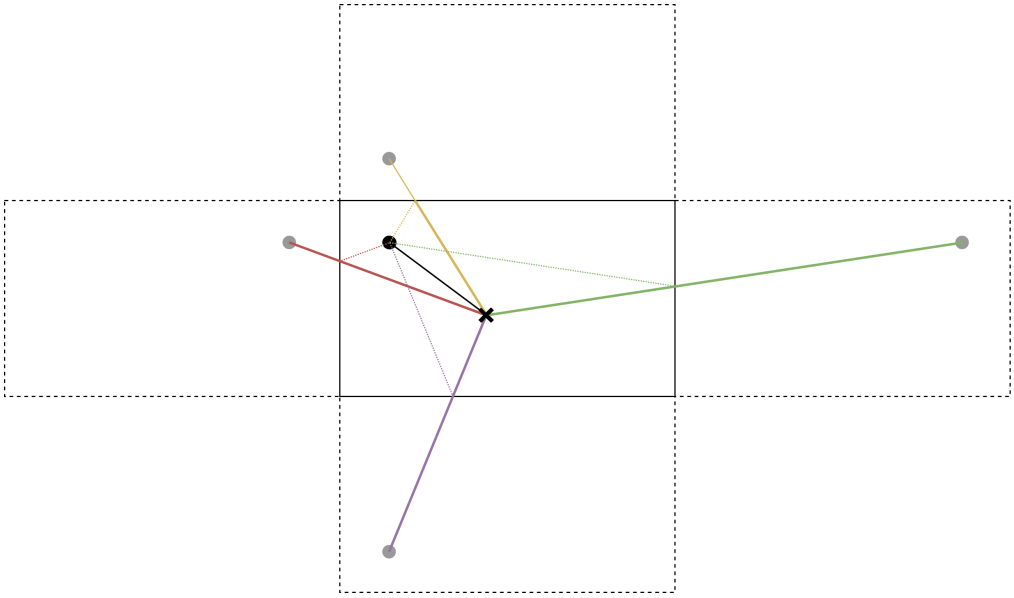


Figure C.1: A simple example of the direct path and first order reflections in a rectangular room. The sound waves are assumed to behave like rays. The receiver is depicted with a $\times$, and the transmitter by a $\bullet$. The image-sources are depicted by a $\bullet$.

sources. In practice, only a finite number is considered [62].

For rigid (fully reflective with $R = 1$) walls, the solution found using the mirror image-source method is exact and the channel $h(\mathbf{x}_i, \mathbf{x}_r, t)$ equals [59]

$$h(\mathbf{x}_i, \mathbf{x}_r, t) = \sum_{\xi=0}^{N_i} \frac{\delta\left(t - \frac{||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}{c}\right)}{4\pi||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}, \tag{C.1}$$

here, $\mathbf{x}_{i,\xi}$ is the location of image-source ξ corresponding to loudspeaker i and N_i is the number of image sources associated with loudspeaker i.

As explained in Section 2.2.2, the case for non-rigid walls is more involved. Namely, in this scenario, R is typically frequency-dependent, complex valued and dependent on the angle of incidence. The image-source method ignores this dependence and instead associates a real valued reflection coefficient β with each of the walls. Thus, each mirror-image ξ of transmitter i is attenuated by some factor $\beta_{i,\xi}$. Under this assumption, the channel h is obtained by summing the contribution of each image-source using the proper weight. This gives

$$h(\mathbf{x}_i, \mathbf{x}_r, t) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\delta\left(t - \frac{||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}{c}\right)}{4\pi||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}, \tag{C.2}$$

reprinted from (2.5). The reflection coefficient corresponding to image-source ξ of loudspeaker i is given by $\beta_{i,\xi}$. This equation approximates the exact solution to the wave-equation for box-shaped rooms well for sufficiently high frequencies [61].

Interchanging the Greens function in (2.4) for the room impulse $h(\mathbf{x}_i, \mathbf{x}_r, t)$ allows to estimate the received signal according to

$$s_r(t) = \sum_{i=1}^{N_s} \left(h(\mathbf{x}_i, \mathbf{x}_r) * s(\mathbf{x}_i)\right)(t). \tag{C.3}$$

The mirror image-source model can be extended to include the directivity of the receiver (listener) and transmitter. When doing so, the change in transmitter orientation due to the mirroring procedure should be kept in mind. In line with Assumption 2, I consider this to be beyond the scope of the thesis. However, it can be implemented through, for example, the use of quaternions. This is explained in [60].

A question which remains is how N_i, $\mathbf{x}_{i,\xi}$ and $\beta_{i,\xi}$ are determined. This is answered below.

C.1 The image-sources

A simple approach to calculating the image-source locations is presented in [28, 59]. Consider a box-shaped room whose origin lies at $(x, y, z) = (0, 0, 0)$. The room has a length L_x, a width L_y and a height L_z. This room is illustrated in Figure C.2.

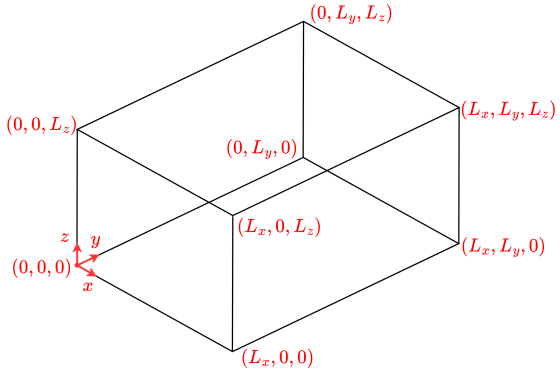


Figure C.2: The coordinate system of the consider rooms. The origin of the room is at $(x, y, z) = (0, 0, 0)$. The length, width, and height of the room are respectively given by L_x, L_y and L_z.

The walls of the room have reflection coefficients β_{x_1}, β_{x_2}, β_{y_1}, β_{y_2}, β_{z_1} and β_{z_2}. Here, β_{x_1} corresponds to the wall which at $x = 0$, while β_{x_2} corresponds to the wall at $x = L_x$. Similarly for the other walls.

The coordinates of the image-sources are found by first constructing the seven image-sources by mirroring along the walls which directly connect to the origin. This yields a total of eight (image)-sources, which can be copied along all spatial dimension. This is described by [28]

$$\mathbf{x}_{i,\xi} = \begin{bmatrix} (1-2q)x_i \\ (1-2j)y_i \\ (1-2k)z_i \end{bmatrix} + \begin{bmatrix} 2m_x L_x \\ 2m_y L_y \\ 2m_z L_z \end{bmatrix}, \quad q, j, k \in \{0, 1\},\ m_x, m_y, m_z \in \mathbb{Z}. \tag{C.4}$$

The values ξ are obtained by assigning each ξ a unique set $\{q, j, k, m_x, m_y, m_z\}$. The value $\xi = 0$ is reserved for the transmitter, so $\xi = 0$ correspond to $\{q, j, k, m_x, m_y, m_z\} = \{0, 0, 0, 0, 0, 0\}$.

The reflection coefficient $\beta_{i,\xi}$ corresponding to the image-source at $\mathbf{x}_{i,\xi}$ is given by [28]

$$\beta_{i,\xi} = \beta_{x_1}^{|m_x - q|} \beta_{x_2}^{|m_x|} \beta_{y_1}^{|m_y - j|} \beta_{y_2}^{|m_y|} \beta_{z_1}^{|m_z - k|} \beta_{z_2}^{|m_z|}. \tag{C.5}$$

In any practical implementation, the values m_x, m_y and m_z will be limited between a finite minimum and maximum value. This value is typically determined through the T_{60}-time. The T_{60} time is the time it takes for the room impulse response to decay by 60 dB. For living rooms, the T_{60}-time equals about 300 ms [61]. The number of (image)-sources N_i corresponding to loudspeaker i is consequently also determined through the T_{60} time. Namely, one can include all images which correspond to a traveltime less then the T_{60}-time and discard the rest. Concretely, keep all sources $\mathbf{x}_{i,\xi}$ for which

$$\frac{||\mathbf{x}_{i,\xi} - \mathbf{x}_r||_2}{c} \leq T_{60} \tag{C.6}$$

holds, and discard the others.

Appendix D
Details of the Taal- and Par-measure

In this appendix, the details of the Taal- and Par-measure are discussed further. These measures are introduced in Section 2.3, where it was found that they are are preferred over the Dau-model due to their computational tractability.

Of these two models, the Taal-measure has better correlation with psychoacoustic listening tests. It is, however, more computationally expensive. While I ultimately chose to use the Par model, I first consider the Taal-measure in detail. This is more instructive, since the Taal-measures employs a more sophisticated auditory model. Furthermore, the Par-measure can be considered as a special case of the Taal-measure [69]. It should be noted that, in the proposed algorithm, the Par-measure is used.

D.1 Structure of the Taal-measure

The Taal-measure takes as input two finite length discrete-time signals x and y having equal length, say N_w. Of these signals, x is considered the original audio signal, while $y = x + \epsilon$ is the degraded audio signal. To obtain the distance measure, both x and y are passed through an auditory model, this results in so-called internal representations I_x and I_y. By comparing the internal representations, a perceptual distance is obtained. The structure of the auditory model used in the Taal-measure is illustrated in Fig. D.1 (reprinted from Section 2.3.3).

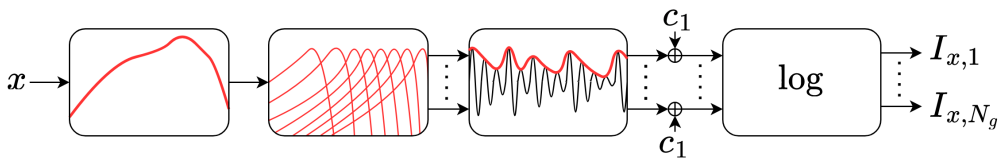


Figure D.1: A schematic overview of the structure of the auditory model used in the Taal-measure. From left to right, it consists of the outer- and middle-ear filter, a gammatone filterbank, an envelope follower, addition of internal noise and finally a logarithm to model the compressive nonlinearity. Figure based on [69].

The measure is applied on short time-frames, so in practice x and y are properly windowed short-time frames of some full length signal [69]. Since the measure is applied on a frame-by-frame basis, I will not introduce notation for the frame index. The signals x and y are indexed by $n \in \{0, \ldots, N_w - 1\}$.

As can be seen in Fig. D.1, the auditory model employs a few stages. The first stage is referred to as the outer-and middle-ear filter h_{om}, and is modelled by taking the inverse of the threshold in quiet.

After this, the basilar membrane is modelled by means of a gammatone filterbank. This is a linear parallel filterbank comprising N_{g} filters, referred to as h_i with $i \in \{1, \ldots, N_g\}$. The output of filter i is referred to as x_i, y_i, or ϵ_i, depending on the signal under consideration. For example, $x_i = x * h_{\text{om}} * h_i$.

The outputs of the gammatone filterbank are passed through an envelope follower used to model the haircell transduction. The envelope follower is implemented by taking a pointwise square, for example $|x_i(n)|^2$, followed by a low pass filter h_s. A constant c_1, modelling internal noise, is added to the output signal. Lastly, the logarithm is taken to model the compressive nonlinearity. Thus, the internal representation $I_{x,i}$ can be expressed using (D.1),

$$I_{x,i}(n) = \log\big(\left(|x_i|^2 * h_s\right)(n) + c_1\big), \tag{D.1}$$

and similarly for $I_{y,i}$. Note that both the logarithm and the square are taken pointwise (sample-by-sample). Taal *et al.* do not mention this explicitly, but other papers of the same author (see [99, 100]) suggest so.

Now that within-channel internal representations $I_{x,i}$ and $I_{y,i}$ are available, it is required to define a within-channel detectability $d_i(x, y)$. In order to arrive at a mathematically tractable measure,[69] proposed to use the l_1 norm. This yields

$$d_i(x, y) = ||I_{y,i} - I_{x,i}||_1. \tag{D.2}$$

The total detectability $d(x, y)$ is now obtained by simply summing the within-channel detectabilities. After some simplification, this yields,

$$d(x, y) = c_2 \sum_{i=1}^{N_g} \left|\left|\log\left(\frac{|y_i|^2 * h_s + c_1}{|x_i|^2 * h_s + c_1}\right)\right|\right|_1, \tag{D.3}$$

where the constant c_2 is added to set the model-sensitivity. Furthermore, the division, squaring and logarithm are done pointwise.

While (D.3) is provides a closed form expression for the detectability, it is not yet suitable for online optimisation. Taal *et al.* showed that, under some assumptions, the model can be simplified further.

D.2 Simplification of the Taal-measure

The first assumption that allows for simplifying the Taal-measure is that x and ϵ are uncorrelated[1]. Combining this assumption with the notion that the lowpass filter performs some kind of averaging yields the following approximation

$$|y_i|^2 * h_s \approx \left(|x_i|^2 + |\epsilon_i|^2\right) * h_s. \tag{D.4}$$

Substituting (D.4) in (D.3) results in

$$d(x, y) \approx c_2 \sum_{i=1}^{N_g} \left|\left|\log\left(1 + \frac{|\epsilon_i|^2 * h_s}{|x_i|^2 * h_s + c_1}\right)\right|\right|_1. \tag{D.5}$$

[1]Note that, in my application, this is not necessarily true. Nevertheless, this is believed not to be a large problem since I do not mind if the difference is detectable. Instead, I mostly use it as a means to limit the difference in a way relating to perception.

As the introduced error ϵ is typically small, the logarithm can be approximated using the first term of the Maclaurin series: $\log(1+z) \approx z$. This yields the simplified measure $D_{\text{Taal}}(x, \epsilon)$, given by

$$d(x,y) \approx D_{\text{Taal}}(x,\epsilon) = c_2 \sum_{i=1}^{N_g} \left|\left| \frac{|\epsilon * h_{\text{om}} * h_i|^2 * h_s}{|x * h_{\text{om}} * h_i|^2 * h_s + c_1} \right|\right|_1, \tag{D.6}$$

where x_i and ϵ_i were expanded.

To efficiently implement (D.6), the filters are considered in the frequency domain [69]. The frequency-domain filters are chosen to be completely real and symmetric, so $\hat{h}(f) = \hat{h}(-f)$. As a result, the convolution operators in (D.6) become circular convolutions, this yields (D.7). To ensure equality between (D.6) and (D.7), zero padding is required.

$$D_{\text{Taal}}(x,\epsilon) = c_2 \sum_{i=1}^{N_g} \left|\left| \frac{|\epsilon \circledast h_{\text{om}} \circledast h_i|^2 \circledast h_s}{|x \circledast h_{\text{om}} \circledast h_i|^2 \circledast h_s + c_1} \right|\right|_1 \tag{D.7}$$

It can be shown that (D.7) can be efficiently calculated by first computing the term g_i^2 [69],

$$g_i^2 = \left(\frac{c_2}{|x_i|^2 \circledast h_s + c_1} \right) \circledast h_s, \tag{D.8}$$

which then allows evaluating the distance measure using [69]

$$D_{\text{Taal}}(x,\epsilon) = \sum_{i=1}^{N_g} \sum_{n=0}^{N_w-1} |\epsilon_i(n) g_i(n)|^2 = \sum_{i=1}^{N_g} ||g_i \epsilon_i||_2^2. \tag{D.9}$$

It should be noted that g_i follows from (D.8) and the fact that $g_i \geq 0$. Furthermore, as previously, the division and squaring in (D.8) happens on a sample-by-sample basis.

In the next section, the equivalence between the Taal-measure and the Par-measure is briefly discussed. Here, also an approach to set the constants c_1 and c_2 is given for the Par-measure.

D.3 Equivalence between the Taal-measure and Par-measure

Interestingly, the Taal-measure can be shown to reduce to the Par-measure. The Par-measure is given by (D.10) [69], although appearing in a slightly different form in the original paper by Par *et al.* [51].

$$D_{\text{Par}}(x,\epsilon) = N_w c_2 \sum_{i=1}^{N_g} \frac{\frac{1}{N_w}||\epsilon_i||_2^2}{\frac{1}{N_w}||x_i||_2^2 + c_1} \tag{D.10}$$

The Taal-measure becomes equivalent to the Par-measure when the cut off frequency of the low pass filter h_s is set to zero [69]. In that case $\hat{h}_s(0) = 1$, and all other values are zero (to verify this, refer to (E.7)). This yields

$$\begin{aligned} (|\epsilon_i|^2 \circledast h_s)(n) &= \frac{1}{N_w} \sum_{k=0}^{N_w-1} \widehat{|\epsilon_i(k)|^2} \hat{h}_s(k) e^{2\pi j k n / N_w} \\ &= \frac{1}{N_w} \widehat{|\epsilon_i(0)|^2} \hat{h}_s(0) \\ &= \frac{1}{N_w} ||\epsilon_i||_2^2, \end{aligned} \tag{D.11}$$

where equivalence between circular convolution in discrete-time domain and multiplication in discrete-frequency domain was used [69]. The last line follows from the definition of the discrete Fourier transform. Note that the result of (D.11) is independent of n and that a similar result holds for x_i. Using this and substituting the result of (D.11) in (D.7) yields the Par-measure (D.10) [69]. Here, it is used that the l_1 norm turns into a multiplication by N_w due to the independence of n.

To allow for an efficient implementation and for easily setting the calibration constants, it is convenient to expand (D.10) into a frequency domain representation. Using Plancherels formula, this yields [101]

$$\begin{aligned} D_{\text{Par}}(x,\epsilon) &= N_w c_2 \sum_{i=1}^{N_g} \frac{\frac{1}{N_w^2}||\hat{\epsilon}\hat{h}_{\text{om}}\hat{h}_i||_2^2}{\frac{1}{N_w^2}||\hat{x}\hat{h}_{\text{om}}\hat{h}_i||_2^2 + c_1} \\ &= c_2 \sum_{i=1}^{N_g} \frac{||\hat{\epsilon}\hat{h}_{\text{om}}\hat{h}_i||_2^2}{\frac{1}{N_w}||\hat{x}\hat{h}_{\text{om}}\hat{h}_i||_2^2 + N_w c_1}. \end{aligned} \tag{D.12}$$

Since the filters were defined as purely real, the Par-measure can also be written as

$$D_{\text{Par}}(x,\epsilon) = \sum_{k=0}^{N_w-1} |\hat{g}(k)\epsilon\hat{(k)}|^2 = ||\hat{g}\hat{\epsilon}||_2^2, \tag{D.13}$$

with

$$\hat{g}^2 = c_2 \sum_{i=1}^{N_g} \frac{\hat{h}_{\text{om}}^2 \hat{h}_i^2}{\frac{1}{N_w}||\hat{x}\hat{h}_{\text{om}}\hat{h}_i||_2^2 + N_w c_1}. \tag{D.14}$$

Note that the constants are larger than zero, thus we can simply take the square root to obtain $\hat{g}$.

On a sidenote, recall that $\hat{h}_{\text{om}}$ is chosen equal to the inverse of the threshold in quiet. When no masker is present, $\hat{g}^2$ reduces to

$$\hat{g}^2 = \frac{c_2}{N_w c_1} \hat{h}_{\text{om}}^2 \sum_{i=1}^{N_g} \hat{h}_i^2. \tag{D.15}$$

As can be seen from this equation, $\hat{g}^2$ reduces to some scaled version of the inverse of the threshold in quiet if $\sum_{i=1}^{N_g} \hat{h}_i^2$ is constant for all frequencies. This is approximately the case for gammatone filters [51]. In fact, even for nonzero $\hat{x}$, it can be shown that $\hat{g}^2$ traces the masking curve for sinusoidal distortions [51].

D.4 Implementation and the calibration constants c_1 and c_2

As mentioned before, the filters are implemented in the frequency domain. To facilitate ease of implementation, a single sided spectrum is considered. Expressions for $\hat{h}_{\text{om}}$, $\hat{h}_s$ and $\hat{h}_i$ are given in Appendix E.

The constants c_1 and c_2 are set such that (1) the threshold in quiet at $f_1 = 1000$ Hz and (2) the 1 dB JND for a 70 dB SPL sinusoid at a frequency of $f_2 = 1000$ Hz are correctly predicted. Furthermore, the value of the distortion measure should be $D = 1$ when the distortion is just not detectable [51, 69].

For requirement (1), the masker is set to $\hat{x}_1 = 0$ and the disturbance is set to equal the threshold in quiet at a frequency f_1. Recalling that the outer- and middle-ear filter was taken equal to the inverse of the threshold in quiet gives

$$\hat{\epsilon}_1(f) = \left(\hat{h}_{\text{om}}(f)\right)^{-1} \delta(f - f_1), \tag{D.16}$$

where δ is the Kronecker delta function.

Using (D.12) and setting $D_{\text{par}}(x_1, \epsilon_1) = 1$ yields

$$1 = c_2 \sum_{i=1}^{N_g} \frac{|\hat{h}_i(f_1)|^2}{N_w c_1} \Leftrightarrow c_1 = \frac{c_2}{N_w} \sum_{i=1}^{N_g} |\hat{h}_i(f_1)|^2. \tag{D.17}$$

To incorporate requirement (2), the 70 dB SPL sinusoidal masker can be modelled as $\hat{x}_2(f) = A_{70}\delta(f - f_2)$, with δ the Kronecker delta function and A_{70} the amplitude corresponding to 70 dB SPL. To get a signal $y_2 = x_2 + \epsilon_2$ with an amplitude of at most 71 dB SPL, so that the distortion is just noticeable, the distortion needs to be at least 18 dB SPL below the masker. Thus, $\hat{\epsilon}_2(f) = A_{52}\delta(f - f_2)$. Note that this assumes that the masker is added in-phase.

To facilitate finding c_2, a function $f(c_2)$ can be defined. This is done by substituting the calibration signals and (D.17) in (D.12). We obtain

$$f(c_2) = c_2 \sum_{i=1}^{N_g} \frac{|A_{52}\hat{h}_{\text{om}}(f_2)\hat{h}_i(f_2)|^2}{\frac{1}{N_w}|A_{70}\hat{h}_{\text{om}}(f_2)\hat{h}_i(f_2)|^2 + c_2 \sum_{j=1}^{N_g} |\hat{h}_j(f_1)|^2} - 1, \tag{D.18}$$

where the -1 originates from $D_{\text{par}}(x_2, \epsilon_2) = 1$. A value $c_2 = c_2^\star > 0$ should now be found such that $f(c_2^\star) = 0$.

In the following, it is shown when $c_2^\star$ exists. Firstly, by taking the derivative of $f(c_2)$ with respect to c_2, one can show that $f(c_2)$ is monotonically increasing for $c_2 > 0$ [102]. Furthermore, the following two limits hold

$$\lim_{c_2 \to 0} f(c_2) = -1 \tag{D.19a}$$

$$\lim_{c_2 \to \infty} f(c_2) = |A_{52}\hat{h}_{\text{om}}(f_2)|^2 \frac{\sum_{i=1}^{N_g} |\hat{h}_i(f_2)|^2}{\sum_{j=1}^{N_g} |\hat{h}_j(f_1)|^2} - 1. \tag{D.19b}$$

Recall that $f_1 = f_2 = 1000$ Hz. Hence, (D.19b) can be approximated by

$$\lim_{c_2 \to \infty} f(c_2) = |A_{52}\hat{h}_{\text{om}}(f_2)|^2 - 1. \tag{D.20}$$

Note that, even if $f_1 \neq f_2$, the gammatone filters approximately sum to a constant value for all f [51]. As such, the simplified limit will still hold approximately.

Since $f(c_2)$ is monotonically increasing for $c_2 > 0$ and since $f(0) < 0$, an optimal solution $c_2^\star$ exists if the limit of (D.19b) is larger then zero. This is the case for $f_1 = f_2 = 1000$ Hz, since $\hat{h}_{\text{om}}(1000) > {}^1/_{A_{52}}$.

To solve (D.18), it is first needed to find numerical values for the amplitudes A_{52} and A_{70}. A challenge here is to convert the digital audio representation into a sound pressure level (which is a physical quantity). Once these constants are determined, a solution for c_2 can be determined using numerical methods such as the bisection method [102, 103]. Lastly, c_1 is straightforwardly calculated using (D.17). Converting a digital representation to a sound pressure level is discussed in Appendix F.

Appendix E
Filters of the Par- and Taal-measure

In this chapter, some details of the implementation of the Taal-measure and (by extension) Par-measure which were left out of Appendix D are discussed. I start by discussing the frequency domain filters $\hat{h}_{\text{om}}$, $\hat{h}_s$ and $\hat{h}_i$. Of these, h_i and h_{om} are used in the Par-measure as well. Then, in Section F, the Sound Pressure Level (SPL) is discussed.

E.1 The filters used in the Taal-measure

In this section, the implementation of the frequency domain filters $\hat{h}_{\text{om}}$, $\hat{h}_s$ and $\hat{h}_i$ is discussed. Before doing so, note that, since a single-sided spectrum is considered, only the positive half of the spectrum is taken into account.

Let us consider input signals x and ϵ of length N_w, where N_w is even. The filters are indexed by $k \in \{0, 1, \ldots, \frac{N_w}{2}\}$. The frequency f corresponding to k equals kf_s/N_w, with f_s the sampling frequency.

E.1.1 Outer- and middle-ear filter

The first filter in the auditory model is termed the outer- and middle-ear filter. In practice, it is taken to equal the inverse of the threshold in quiet [51]. The threshold in quiet can be estimated as [52]

$$T_q(f) = 3.64\left(\frac{f}{1000}\right)^{-0.8} - 6.5\exp\left\{-0.6\left(\frac{f}{1000} - 3.3\right)^2\right\} + 10^{-3}\left(\frac{f}{1000}\right)^4, \tag{E.1}$$

where the frequency unit equals Hz, and the unit of T_q is dB SPL. Removing the transform to dB SPL results in the expression for $\hat{h}_{\text{om}}$ [51],

$$\hat{h}_{\text{om}}(f) = \left(\frac{\alpha}{p_0}\right)^{-1} 10^{-T_q(f)/20}. \tag{E.2}$$

The term $\frac{\alpha}{p_0}$ originates from the dB SPL mapping and is explained in Appendix F.

E.1.2 Gammatone filter

The Taal-measure employs $N_g = 64$ gammatone filters. A good approximation to the magnitude spectrum of a gammatone filter centered at frequency f_c (in Hz) is given by [51],

$$\hat{h}_g(f) = \left(1 + \left(\frac{f - f_c}{\kappa \text{ERB}(f_c)}\right)^2\right)^{-\eta/2} \tag{E.3}$$

In the equation, κ is a normalising constant, η is the order of the filter and ERB(f_c)[1]. is the equivalent rectangular bandwidth of the gammatone filter centered at f_c (see (E.5)). Typically, the filter order is assumed to be $\eta = 4$ [51]. The corresponding normalising constant is given by

$$\kappa = \frac{2^{\eta-1}(\eta-1)!}{\pi(2\eta-3)!!} \tag{E.4}$$

with ! the factorial and !! the double factorial. The double factorial $n!!$ equals $2 \cdot 4 \cdot \ldots \cdot n$ for even positive numbers n, and $1 \cdot 3 \cdot 5 \cdot \ldots \cdot n$ for odd positive numbers n. Lastly, the value ERB(f_c) can be approximated using [104]

$$\text{ERB}(f_c) = 24.7\left(\frac{4.37 f_c}{1000} + 1\right). \tag{E.5}$$

Par *et al.* and Taal *et al.* chose the N_g center frequencies such that they linearly divide the ERB-rate scale on the interval $[E(0), E(f_s/2)]$ in N_g steps. Here, f_s is the sampling frequency. The ERB-rate scale is approximated as [104]

$$E(f_c) = 21.4 \log_{10}\left(\frac{4.37 f_c}{1000} + 1\right). \tag{E.6}$$

E.1.3 The lowpass filter

A simple first order lowpass filter with a cutoff frequency $f_\tau = 1000$ Hz is used. The filter $\hat{h}_s$ is obtained as [69]

$$\hat{h}_s(f) = \frac{1+\alpha}{\sqrt{1+\alpha^2+2\alpha\cos(2\pi f/f_s)}}, \tag{E.7}$$

where α equals

$$\alpha = -\exp\left\{-\frac{2\pi f_\tau}{f_s}\right\}. \tag{E.8}$$

[1]The Equivalent Rectangular Bandwidth (ERB) of some reference filter is the bandwidth of a rectangular filter which has the same maximum magnitude as the reference filter and transmits the same power when white noise is given as input [30]. Even though they are not rectangular, auditory filters are often characterised using their ERB.

Appendix F
The Sound Pressure Level

The Sound Pressure Level (SPL) gives the intensity of an acoustic stimuli with respect to a reference value. It is given by

$$L_{\mathrm{SPL}} = 20\log_{10}\left(\frac{p}{p_0}\right) \quad \text{[dB SPL]}, \tag{F.1}$$

with p the absolute sound pressure in Pascals and p_0 a reference value equal to 20 µPa [53]. As an example, normal speech is about 60 to 70 dB SPL [30, 49].

To determine L_{SPL}, one needs to have access to the sound pressure p. In my application, this value is not straightforwardly known since only a digital representation x is available. However, throughout this thesis, I assume the sound pressure level to be a linear function of the input signal. Thus, we can write (F.1) as

$$L_{\mathrm{SPL}} = 20\log_{10}\left(\frac{\alpha|x|}{p_0}\right) = 20\log_{10}(|x|) + 20\log_{10}\left(\frac{\alpha}{p_0}\right) \quad \text{[dB SPL]} \tag{F.2}$$

where α depends on environmental factors such as the loudspeaker, cables, amplifier, and room.

To find α, prior information on the sound level needs to be available (or a conservative estimate needs to be made). To be specific, let us have access to a value $|x| = x_{\mathrm{ref}}$ for which the received sound pressure level is $x_{\mathrm{dB,\ ref}}$ (in dB SPL). Substituting this into (F.2) yields

$$20\log_{10}\left(\frac{\alpha}{p_0}\right) = x_{\mathrm{dB,ref}} - 20\log_{10}\left(x_{\mathrm{ref}}\right), \tag{F.3}$$

which can straightforwardly be solved for α (or $\frac{\alpha}{p_0}$).

As an example, consider a normalised digital representation, so $\max(|x|) = 1$. Suppose that this corresponds to 70 dB SPL. Solving (F.3) results in $\alpha = 0.0632$, or, equivalently, $\alpha/p_0 \approx 3162$. Once α is available, it is straightforward to find the amplitude corresponding to a different sound pressure level and vice versa using

$$L_{\mathrm{SPL}} = 20\log_{10}(|x|) + 20\log_{10}\left(\frac{\alpha}{p_0}\right) \Leftrightarrow |x| = \left(\frac{\alpha}{p_0}\right)^{-1} 10^{\frac{L_{\mathrm{SPL}}}{20}} \tag{F.4}$$

For this example, the value $A_{70} = 1$ and $A_{52} \approx 0.1259$. The value $A_{\mathrm{Tq}}(f_m)$ is slightly more involved. For $f_m = 1000$ Hz, $T_q(f_m) \approx 3.37$ dB SPL. Thus, $A_{\mathrm{Tq}}(f_m) \approx 0.466 \times 10^{-3}$.

Even though I assume α to be constant. It should be noted that this is a false assumption in any practical scenario. This can easily be seen by noting that a loudspeaker will already introduce a frequency dependency. Similarly, α might very well dependent on the amplitude of x due to amplifier nonlinearities.

Appendix G
Details Block-based filtering

In this appendix, the method to perform convolution of long signals and filters by splitting the convolution in smaller segments as introduced in Section 2.4 is discussed in more detail.

Recall that I consider the convolution between a signal x of infinite length and a causal filter h of length M. This convolution is given by

$$(x * h)(n) = \sum_{m=-\infty}^{\infty} x(m)h(n-m). \tag{G.1}$$

In order to perform this convolution while ensuring that delays remain acceptable, one can choose to segment only the input signal or to segment both the input signal and the filter. The former is described in Section G.1 and the latter is described in Section G.2. Note that the discussions repeat some equations of Sections 2.4.1 and 2.4.2, respectively.

G.1 Short-time filtering of the input signal

Recall (2.11) and (2.12), which stated that we have access to some window w_1 and repetition rate R_1 for which

$$\sum_{l=-\infty}^{\infty} w_1(n - lR_1) = 1 \,\forall n, \tag{G.2}$$

$$\text{supp}(w_1) = \{0, \ldots, L_1 - 1\}, \tag{G.3}$$

holds. This allows to write (reprint from (2.13))

$$x(n) = x(n) \sum_{l=-\infty}^{\infty} w_1(n - lR_1) = \sum_{l=-\infty}^{\infty} w_1(n - lR_1)x(n) = \sum_{l=-\infty}^{\infty} \tilde{x}_l(n), \tag{G.4}$$

with

$$\tilde{x}_l(n) = w_1(n - lR_1)x(n), \tag{G.5a}$$

$$\text{supp}(\tilde{x}_l) \subseteq \{lR_1, \ldots, L_1 - 1 + lR_1\}. \tag{G.5b}$$

or, shifting the origin to $n = 0$,

$$x_l(n) = \tilde{x}_l(n + lR_1) = w_1(n)x(n + lR_1), \tag{G.6a}$$

$$\text{supp}(x_l) \subseteq \{0, \ldots, L_1 - 1\}. \tag{G.6b}$$

In Section 2.4, (2.16) was presented without proof. Working out the simplification yields

$$\begin{aligned}(x*h)(n) &= \sum_{l=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} w_1(m-lR_1)x(m)h(n-m)\\ &= \sum_{l=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \tilde{x}_l(m)h(n-m)\\ &= \sum_{l=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} x_l(m)h(n-m-lR_1)\\ &= \sum_{l=-\infty}^{\infty}\sum_{m=0}^{L_1-1} x_l(m)h(n-m-lR_1).\end{aligned} \tag{G.7}$$

Where the second to third line follows from the substitution $m \to m + lR_1$, and the third to fourth line from the support of $x_l(m)$ (see (G.6b)).

Due to the finite support of the window, at any time $n = n'$ there exists some $l = l'$ for which we have full knowledge of all the blocks with $l \leq l'$. Hence, we may decompose the convolution of (G.7) into a "known" part and an "unknown" part. This yields

$$(x*h)(n) = \sum_{l=-\infty}^{l'}\sum_{m=0}^{L_1-1} x_l(m)h(n-m-lR_1) + \sum_{l=l'+1}^{\infty}\sum_{m=0}^{L_1-1} x_l(m)h(n-m-lR_1). \tag{G.8}$$

It follows that, in real-time applications, one can simply compute the left double sum. Each time a new block comes in, the output can be updated to incorporate the new block.

The approach outlined above is only valid if the right hand double sum does not influence the output up to $n = n' - \epsilon$, where ϵ is a finite non-negative delay. It can be shown that this is true by considering the support of the output signal, this is done below.

Let us consider the filtering of a single block. Define $y_l(n)$ as

$$y_l(n) = \sum_{m=0}^{L_1-1} x_l(m)h(n-m-lR_1). \tag{G.9}$$

It can be shown that y_l has support

$$\text{supp}(y_l) \subseteq \{lR_1, \ldots, lR_1 + L_1 + M - 2\}. \tag{G.10}$$

It follows that the left double sum of (G.8) has support $\{-\infty, \ldots, l'R_1 + L_1 + M - 2\}$, while the right double sum has a support $\{(l'+1)R_1, \ldots, \infty\}$. These supports partially overlap when $L_1 + M - 2 \geq R_1$. As $L_1 \geq R_1$, this will be the case in any practical scenario. However, since the overlap is limited, the solution to (G.8) can be calculated up to a time instant $n = n' - \epsilon$ for any n' and finite ϵ. The actual value of ϵ depends on the lengths L_1, M and on n' itself.

In some scenarios, the filter length M might be too large to produce usable results in real-time applications. In those cases, one can choose to segment the filter as well. This is a straightforward extension of this section and is discussed in Section G.2. Before doing so, let us briefly consider two windows.

G.1.1 The rectangular window and the Hanning window

A possible choice for the window w_1 is the rectangular window $\text{rect}_L(n)$ with $|\text{supp}\{\text{rect}_{L1}\}| = L_1$,

$$\text{rect}_{L1}(n) = \begin{cases} 1, & 0 \leq n < L_1 \\ 0, & \text{otherwise.}\end{cases} \tag{G.11}$$

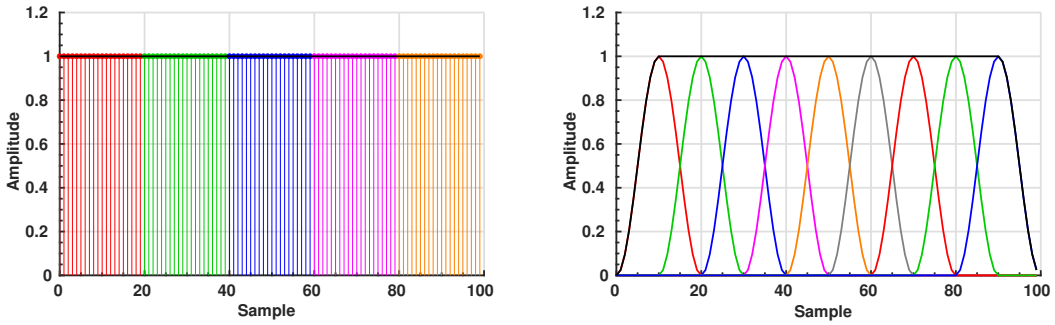
The rectangular window is an example where the supports of the windows do not overlap, e.g. $R_1 = L_1$. However, in some cases, one might prefer to use overlapping filters. This is typically the case when nonlinear or time-varying processing is performed [80]. A possible choice is the Hanning window with $R_1 = L_1/2$ for even length L_1. The Hanning window of length L_1 is described by [80]

$$\text{hann}_{L1}(n) = \text{rect}_{L1}(n)\left(\frac{1}{2} + \frac{1}{2}\cos\left(\frac{2\pi}{L_1}\left(n - \frac{L_1}{2}\right)\right)\right), \tag{G.12}$$

where the shift in the argument of the cosine shifts the peak of the window from 0 to $L_1/2$.

Note that, strictly speaking, the Hanning window has support a support $\{1, \ldots, L_1 - 1)$. Thus, if w_1 is chosen to be the Hanning window, the equality sign of (G.3) changes into a subset sign. This does not influence the validity of the results.

The rectangular window and Hanning window are shown in Figure G.1.



(a) $\text{rect}_L(n)$ (b) $\text{hann}_L(n)$

Figure G.1: Example of the rectangular window and the Hanning window. Both are depicted for $L = 20$ and the corresponding repetition rates. The sum of the windows is indicated by the black line. Note that the windows sum to one over the range with overlap.

G.2 Segmenting the filter

Recall, from Section 2.4.2, that the filter can be segmented as well. This is done using a window w_2 with length L_2 and repetition rate R_2. It is furthermore assumed that M/L_2 is integer. The window is given by (repeated from (2.17) and (2.18))

$$\sum_{\iota=\iota_a}^{\iota_b} w_2(n - \iota R_2) = 1 \, \forall n \in \text{supp}(h), \tag{G.13a}$$

$$\text{supp}(w_2) = \{0, \ldots, L_2 - 1\}, \tag{G.13b}$$

for some ι_a and ι_b. So that

$$h(n) = \sum_{\iota=\iota_a}^{\iota_b} w_2(n - \iota R_2)h(n) = \sum_{\iota=\iota_a}^{\iota_b} \tilde{h}_\iota(n). \tag{G.14}$$

Here, $\tilde{h}_\iota$ is defined analogously to $\tilde{x}_\iota$ (see (G.5) or (2.20)). Similarly, we may define h_ι analogously to x_l (see (G.6) or (2.21)).

In Section 2.4, (2.22) was given without proof. Working out the simplification yields

$$\begin{aligned}(x * h)(n) &= \sum_{l=-\infty}^{\infty} \sum_{m=0}^{L_1-1} x_l(m) h(n-m-lR_1) \\ &= \sum_{l=-\infty}^{\infty} \sum_{m=0}^{L_1-1} x_l(m) \sum_{\iota=\iota_a}^{\iota_b} \tilde{h}_\iota(n-m-lR_1) \\ &= \sum_{l=-\infty}^{\infty} \sum_{\iota=\iota_a}^{\iota_b} \sum_{m=0}^{L_1-1} x_l(m) h_\iota(n-m-lR_1-\iota R_2),\end{aligned} \tag{G.15}$$

where the last line follows from the definition of h_ι, see (2.21).

G.3 Frequency Domain

To facilitate an efficient implementation, the convolutions are performed in the frequency domain. This is discussed below. I consider a rectangular window w_2, so that $\iota_a = 0$ and $\iota_b = {}^{M}/_{L_2} - 1$.

Consider a single block (l, ι). Define

$$y_{(l,\iota)}(n) = \sum_{m=0}^{L_1-1} x_l(m) h_\iota(n-m-lR_1-\iota R_2), \tag{G.16}$$

so that

$$\operatorname{supp}(y_{(l,\iota)}) \subseteq \{lR_1 + \iota R_2, \ldots, lR_1 + \iota R_2 + L_1 + L_2 - 2\}. \tag{G.17}$$

It will be convenient to define a shift operator shift_L, which shifts some signal $x(n)$ by L samples, so

$$(\text{shift}_L\{x\})(n) = x(n+L). \tag{G.18}$$

Using the shift operator, we may write (G.16) as

$$y_{(l,\iota)}(n) = (x_l * \text{shift}_{-lR_1-\iota R_2}\{h_\iota\})(n). \tag{G.19}$$

Or, equivalently

$$y_{(l,\iota)}(n) = (\text{shift}_{-lR_1-\iota R_2}\{x_l * h_\iota\})(n), \tag{G.20}$$

with

$$\operatorname{supp}(x_l * h_\iota) \subseteq \{0, \ldots, L_1 + L_2 - 2\}. \tag{G.21}$$

Transforming the convolution of (G.20) to a frequency domain equivalent yields

$$y_{(l,\iota)}(n) = (\text{shift}_{-lR_1-\iota R_2}\{\mathcal{F}^{-1}\left[(\mathcal{F}x_l)(\mathcal{F}h_\iota)\right]\})(n). \tag{G.22}$$

Combining (G.15), (G.16) and (G.22) gives

$$\begin{aligned}(x * h)(n) &= \sum_{l=-\infty}^{\infty} \sum_{\iota=0}^{M/L_2-1} y_{(l,\iota)}(n) \\ &= \sum_{l=-\infty}^{\infty} \sum_{\iota=0}^{M/L_2-1} \sum_{m=0}^{L_1-1} x_l(m) h_\iota(n-m-lR_1-\iota R_2) \\ &= \sum_{l=-\infty}^{\infty} \sum_{\iota=0}^{M/L_2-1} (\text{shift}_{-lR_1-\iota R_2}\{\mathcal{F}^{-1}\left[(\mathcal{F}x_l)(\mathcal{F}h_\iota)\right]\})(n),\end{aligned} \tag{G.23}$$

which provides a method to efficiently implement both block-based filtering and input signal segmentation.

Note that, in an implementation, the infinite length discrete signals x_l and h_ι will be represented by finite length vectors $\mathbf{x}_l \in \mathbb{R}^{L_1}$ and $\mathbf{h}_\iota \in \mathbb{R}^{L_2}$ (or similar). However, frequency domain multiplication corresponds to *circular* convolution. Thus, to ensure that the circular convolution is equivalent to a "normal" convolution, one needs to zeropad the vectors $\mathbf{x}_l$ and $\mathbf{h}_\iota$ to a length $L_3 \geq L_1 + L_2 - 1$. In practice, L_3 is chosen as an integer power of two [80].

Appendix H
Three-dimensional approach

In this appendix, the equation for the room transfer function in Cartesian coordinates is rewritten to a cylindrical coordinate system and spherical coordinate system. This is done in Section H.1 and H.2 respectively. The considered equation is (reprinted from (3.21))

$$\hat{\hbar}(\mathbf{x}_i, \mathbf{x}+\mathbf{x}_h, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \frac{\exp\{-\frac{j\omega}{c}||\mathbf{x}+\mathbf{x}_h-\mathbf{x}_{i,\xi}||_2\}}{4\pi||\mathbf{x}+\mathbf{x}_h-\mathbf{x}_{i,\xi}||_2}. \tag{H.1}$$

Note that this is simply a sum of weighted Greens functions (see (2.2))

$$\hat{\hbar}(\mathbf{x}_i, \mathbf{x}+\mathbf{x}_h, \omega) = \sum_{\xi=0}^{N_i} \beta_{i,\xi} \hat{g}(\mathbf{x}_{i,\xi}, \mathbf{x}+\mathbf{x}_h, \omega), \tag{H.2}$$

with

$$\hat{g}(\mathbf{x}_{i,\xi}, \mathbf{x}+\mathbf{x}_h, \omega) = \frac{\exp\{-\frac{j\omega}{c}||\mathbf{x}+\mathbf{x}_h-\mathbf{x}_{i,\xi}||_2\}}{4\pi||\mathbf{x}+\mathbf{x}_h-\mathbf{x}_{i,\xi}||_2}. \tag{H.3}$$

For notational convenience, I only consider the inner term $\hat{g}(\mathbf{x}_{i,\xi}, \mathbf{x}+\mathbf{x}_h, \omega)$. Lastly, the coordinate transform of the spatial weighting $p_(\mathbf{x}+\mathbf{x}_h)$ is assumed to equal $p_4(r)p_5(\theta)p_6(z)$ in cylindrical coordinates and $p_7(r)p_8(\theta)p_9(\phi)$ in spherical coordinates. Likely, $p_4(r)$ and $p_7(r)$ can be set to equal $p_1(r)$, while $p_5(\theta)$ and $p_8(\theta)$ can be set equal to $p_2(\theta)$ as given in Section 3.3.

H.1 Cylindrical coordinates

The Greens function given by (H.3) can be changed to a cylindrical coordinate system. This facilitates an easy approach to choosing the spatial weighting functions.

Using (H.3) and expanding the l_2-norm yields

$$\hat{g}(\mathbf{x}_{i,\xi}, \mathbf{x}+\mathbf{x}_h, \omega) = \frac{e^{-\frac{j\omega}{c}\sqrt{(x+x_h-x_{i,\xi})^2+(y+y_h-y_{i,\xi})^2+(z+z_h-z_{i,\xi})^2}}}{4\pi\sqrt{(x+x_h-x_{i,\xi})^2+(y+y_h-y_{i,\xi})^2+(z+z_h-z_{i,\xi})^2}} \tag{H.4}$$

Now consider the transformation to cylindrical coordinates $x = r\cos(\theta)$ and $y = r\sin(\theta)$, with $\theta \in [0, 2\pi)$ and $r \in [0, \infty)$. Transforming (H.4) yields

$$\hat{g}(\mathbf{x}_{i,\xi}, r, \theta, z, \omega) = \frac{e^{-\frac{j\omega}{c}\sqrt{(r\cos(\theta)+x_h-x_{i,\xi})^2+(r\sin(\theta)+y_h-y_{i,\xi})^2+(z+z_h-z_{i,\xi})^2}}}{4\pi\sqrt{(r\cos(\theta)+z_h-x_{i,\xi})^2+(r\sin(\theta)+y_h-y_{i,\xi})^2+(z+z_h-z_{i,\xi})^2}} \tag{H.5}$$

Note that, when integration is required, the integration measure becomes $dxdydz \rightarrow rdrd\theta dz$ [105].

Note that, the coordinate system was shifted such that the expected location of the head $\mathbf{x}_h$ corresponds to $\mathbf{x} = (0, 0, 0)$. It follows that the value $(r, z) = (0, 0)$ also corresponds to the

expected location of the head. This should be kept in mind when choosing the spatial weighting functions. Furthermore, note that the sine and cosine are periodic with 2π. Hence, if the spatial weighting function $p_5(\theta)$ is chosen to be periodic with period 2π, we are free to integrate θ over $[C, 2\pi + C)$ for any $C \in \mathbb{R}$.

H.2 Spherical coordinates

Consider the transformation to spherical coordinates $x = \rho\cos(\theta)\sin(\phi)$, $y = \rho\sin(\theta)\sin(\phi)$ and $z = \rho\cos(\phi)$, with $\rho \in [0, \infty)$, $\theta \in [0,\, 2\pi)$ and $\phi \in [0,\, \pi/2]$ [105]. Substituting in (H.4) gives

$$\hat{g}(\mathbf{x}_{i,\xi}, r, \theta, \phi, \omega) = \frac{e^{-\frac{j\omega}{c}\sqrt{\alpha}}}{4\pi\sqrt{\alpha}}, \tag{H.6}$$

with

$$\alpha = (\rho\cos(\theta)\sin(\phi) + x_h - x_{i,\xi})^2 + (\rho\sin(\theta)\sin(\phi) + y_h - y_{i,\xi})^2 + (\rho\cos(\phi) + z_h - z_{i,\xi})^2 \tag{H.7}$$

Note that the coordinate system was shifted such that the expected location of the head $\mathbf{x}_h$ corresponds to $\mathbf{x} = (0, 0, 0)$. It follows that the value $r = 0$ also corresponds to the expected location of the head. This should be kept in mind when choosing the spatial weighting functions. Lastly, when integrating, the integration measure becomes $dxdydz \to \rho^2\sin(\phi)drd\theta d\phi$.

When calculating the covariance matrices $\mathbf{R}_{\mathcal{V}}$, it is required to solve integrals containing products of Greens functions. I.e., integrals of the form

$$I = \iiint \hat{g}(\mathbf{x}_{i,\xi}, \rho, \theta, \phi, \omega)\hat{g}^*(\mathbf{x}_{j,\xi}, \rho, \theta, \phi, \omega)p(\rho, \theta, \phi)\rho^2\sin\phi d\rho d\theta d\phi, \tag{H.8}$$

or similar. Because of the complex exponent, these integrals are oscillatory for large ω. For "normal" numerical solvers, this oscillatory behaviour is challenging. In Appendix I, an alternative numerical approach is outlined which can be used to solve the integrals. This method is, however, not implemented.

Appendix I
L-eRPIM

In this chapter, we aim to develop a numerical scheme for solving integrals of the form (H.8). I first give a brief explanation of the origin and idea behind the used numerical method below. Then, in Section I.1, the equations needed to solve a three-dimensional integral are derived.

The numerical integration scheme is based on the theory presented in [95] and referred to as L-eRPIM (Levin - Enriched Point Interpolation Method). It might not come as a surprise that this approach is (partially) based on Levins collocation method [106]. This method approximates integrals of the form

$$I = \int_a^b f(x) e^{j\omega g(x)} dx, \tag{I.1}$$

with real-valued g, both f and g slowly varying, and $|\frac{d}{dx} g(x)| \gg (b-a)^{-1}$. The function $f(x)\exp\{j\omega g(x)\}$ is referred to as the integrand. Levins method start by noting that, if $f(x)$ is of the form

$$f(x) = jg'(x)p(x) + p'(x), \tag{I.2}$$

the integral equals

$$I = \int_a^b (jg'(x)p(x) + p'(x)) e^{jg(x)} dx = \int_a^b \frac{d}{dx} p(x) e^{jg(x)} dx = p(b)e^{jg(b)} - p(a)e^{jg(a)}. \tag{I.3}$$

The trick is now to find a suitable function $p(x)$. It is important to note that $p(x)$ generally is as oscillatory as the integrand [106]. However, Levin showed that for f and g' slowly oscillatory, there exists a particular solution to the differential equation which is slowly oscillatory. For a short and intuitive overview of what is meant with oscillatory, the reader can refer to [107]. It should be stated that, for small ω, other numerical schemes might be more suitable.

I.1 Derivation for three-dimensional integral

Since the theory behind the approach is quite involved, I will not go into the details. However, the approach presented in [95] is only fully explained for a two dimensional integral. Since, in our application, the integral is three-dimensional, the corresponding equations need to be derived. This is mostly bookkeeping and not relevant from a theoretical perspective.

We start by defining the three dimensional integral and the corresponding bounds. The integral which needs to be solved is

$$I = \int_a^b \int_d^e \int_s^t f(x,y,z) e^{j\omega g(x,y,z)} dz dy dx \tag{I.4}$$

Analogous to Eq. (I.3), we need to find the function $p(x, y, z)$ such that

$$\frac{\partial^3}{\partial x \partial y \partial z}[p(x,y,z)e^{j\omega g(x,y,z)}] = f(x,y,z)e^{j\omega g(x,y,z)} \tag{I.5}$$

Substituting this in Eq. (I.4) yields

$$\begin{aligned}
I =& \int_a^b \int_d^e \int_s^t \frac{\partial^3}{\partial x \partial y \partial z}[p(x,y,z)e^{j\omega g(x,y,z)}]dzdydx \\
=& \int_a^b \int_d^e \frac{\partial^2}{\partial x \partial y}\left(p(x,y,t)e^{j\omega g(x,y,t)} - p(x,y,s)e^{j\omega g(x,y,s)}\right)dydx \\
=& \int_a^b \frac{\partial}{\partial x}\left(p(x,e,t)e^{j\omega g(x,e,t)} - p(x,e,s)e^{j\omega g(x,e,s)}\right)dx \\
&+ \int_a^b \frac{\partial}{\partial x}\left(-p(x,d,t)e^{j\omega g(x,d,t)} + p(x,d,s)e^{j\omega g(x,d,s)}\right)dx \\
=& p(b,e,t)e^{j\omega g(b,e,t)} - p(b,e,s)e^{j\omega g(b,e,s)} - p(b,d,t)e^{j\omega g(b,d,t)} + p(b,d,s)e^{j\omega g(x,d,s)} \\
&- p(a,e,t)e^{j\omega g(a,e,t)} + p(a,e,s)e^{j\omega g(a,e,s)} + p(a,d,t)e^{j\omega g(a,d,t)} - p(a,d,s)e^{j\omega g(a,d,s)}
\end{aligned} \tag{I.6}$$

Hence, if we can find a suitable function $p(x, y, z)$, it is straightforward to evaluate integral (I.4) by means of Eq. (I.6). Finding this function is the topic of the following section.

I.1.1 Finding a suitable function

Working out the left side of Eq. (I.5) and denoting the derivative with respect to some variable by a subscript yields

$$\begin{aligned}
\frac{\partial^3}{\partial x \partial y \partial z}(pe^{j\omega g}) =& \frac{\partial^2}{\partial x \partial y}(p_z e^{j\omega g} + j\omega p g_z e^{j\omega g}) \\
=& \frac{\partial}{\partial x}[(p_{yz} + j\omega p_z g_y + j\omega p_y g_z + j\omega p g_{yz} - \omega^2 p g_z g_y)e^{j\omega g}] \\
=& (p_{xyz} + j\omega p_{yz} g_x + j\omega p_{xz} g_y + j\omega p_z g_{xy} - \omega^2 p_z g_x g_y + j\omega p_{xy} g_z \\
&+ j\omega p_y g_{xz} - \omega^2 p_y g_x g_z + j\omega p_x g_{yz} + j\omega p g_{xyz} - \omega^2 p g_x g_{yz} - \omega^2 p_x g_z g_y \\
&- \omega^2 p g_y g_{xz} - \omega^2 p g_z g_{xy} - j\omega^3 p g_z g_y g_x)e^{j\omega g} \\
=& [p_{xyz} + j\omega(p g_{xyz} + p_x g_{yz} + p_y g_{xz} + p_z g_{xy} + p_{xy} g_z + p_{xz} g_y + p_{yz} g_x) \\
&- \omega^2(p g_x g_{yz} + p g_y g_{xz} + p g_z g_{xy} + p_x g_y g_z + p_y g_x g_z + p_z g_x g_y) \\
&- j\omega^3 p g_x g_y g_z]e^{j\omega g}.
\end{aligned} \tag{I.7}$$

Comparing Eq. (I.5) and Eq. (I.7) shows that

$$\begin{aligned}
f =& p_{xyz} + j\omega(p g_{xyz} + p_x g_{yz} + p_y g_{xz} + p_z g_{xy} + p_{xy} g_z + p_{xz} g_y + p_{yz} g_x) \\
&- \omega^2(p g_x g_{yz} + p g_y g_{xz} + p g_z g_{xy} + p_x g_y g_z + p_y g_x g_z + p_z g_x g_y) \\
&- j\omega^3 p g_x g_y g_z.
\end{aligned} \tag{I.8}$$

We now have all the necessary background needed to start searching for a function $p(x, y, z)$. In line with the method proposed in [95], we choose $p(x, y, z)$ as the sum of a number of weighted basis functions. The weightings are found by fitting Eq. (I.8) for a number of different coordinates, referred to as nodes. As such, we do not obtain an exact solution $p(x, y, z)$, but an estimate $\hat{p}(x, y, z)$. This estimate can be written as

$$\hat{p}(x, y, z) = \sum_{i=1}^{N} a_i R_i(x, y, z) + \sum_{\eta=1}^{m} b_\eta P_\eta(x, y, z) + \sum_{\xi=1}^{l} c_\xi T_\xi(x, y, z), \tag{I.9}$$

where a_i, b_η and c_ξ are the weights, $R_i(x, y, z)$ is a so-called Radial Basis Function (RBF), P_η a monomial basis function and T_ξ a trigonometric basis function. Lastly, N is the number of nodes which are fitted, m is the number of monomial basis functions which are used, and l is the number of trigonometric basis functions which are used.

There exist a number of different RBFs. Among others, common choices are the Multi Quadrics (MQ) function and the Gaussian function [108]. The MQ function equals

$$R_i(x, y, z) = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 + c^2}, \tag{I.10}$$

where (x, y, z) is the point of evaluation, (x_i, y_i, z_i) is the point around which the MQ RBF is centered, and c is the shape parameter. The shape parameter influences the performance and is typically determined by experimentation.

The monomial basis functions P_η are used to be able to fit a polynomial of certain degree. For a 1-dimensional case, the basis equals $\{x^0, x^1, x^2, \ldots\}$. For the 3-dimensional case, a basis is

$$\{1, x, y, z, x^2, y^2, z^2, xy, xz, yz, \ldots\}, \tag{I.11}$$

in principle, this can be extended indefinitely. The function P_η corresponds to the η^{th} function in the basis.

Similarly, a trigonometric basis is given by

$$\{1, \cos(x), \sin(y), \sin(z), \cos(2x), \sin(2y), \sin(2z), \cos(x)\sin(y), \cos(x)\sin(z), \sin(z)\sin(y), \ldots\}, \tag{I.12}$$

in principle, this can be extended indefinitely. The function T_ξ corresponds to the ξ^{th} function in the basis.

I.1.2 Constructing the weights

We need to find the weights a_i, b_η and c_ξ. This is done by substituting $\hat{p}(x, y, z)$ in Eq. (I.8). By subsequently choosing N reference nodes inside the integration domain, a matrix equation is constructed which can be solved for the weights. This does, however, result in a system with $N + m + l$ variables and N equations. We will add the additional $m + l$ equations later.

Substituting Eq. (I.9) into Eq. (I.8) and making use of linearity yields

$$
\begin{aligned}
\hat{f} = & \sum_{i=1}^{N} a_i \partial_{xyz} R_i + \sum_{\eta=1}^{m} b_\eta \partial_{xyz} P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_{xyz} T_\xi \\
& + j\omega \left(g_{xyz} \left(\sum_{i=1}^{N} a_i R_i + \sum_{\eta=1}^{m} b_\eta P_\eta + \sum_{\xi=1}^{l} c_\xi T_\xi \right) \right) \\
& + j\omega \left(g_{yz} \left(\sum_{i=1}^{N} a_i \partial_x R_i + \sum_{\eta=1}^{m} b_\eta \partial_x P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_x T_\xi \right) \right) \\
& + j\omega \left(g_{xz} \left(\sum_{i=1}^{N} a_i \partial_y R_i + \sum_{\eta=1}^{m} b_\eta \partial_y P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_y T_\xi \right) \right) \\
& + j\omega \left(g_{xy} \left(\sum_{i=1}^{N} a_i \partial_z R_i + \sum_{\eta=1}^{m} b_\eta \partial_z P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_z T_\xi \right) \right) \\
& + j\omega \left(g_{z} \left(\sum_{i=1}^{N} a_i \partial_{xy} R_i + \sum_{\eta=1}^{m} b_\eta \partial_{xy} P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_{xy} T_\xi \right) \right) \\
& + j\omega \left(g_{y} \left(\sum_{i=1}^{N} a_i \partial_{xz} R_i + \sum_{\eta=1}^{m} b_\eta \partial_{xz} P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_{xz} T_\xi \right) \right) \\
& + j\omega \left(g_{x} \left(\sum_{i=1}^{N} a_i \partial_{yz} R_i + \sum_{\eta=1}^{m} b_\eta \partial_{yz} P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_{yz} T_\xi \right) \right) \\
& - \omega^2 \left(g_x g_{yz} \left(\sum_{i=1}^{N} a_i R_i + \sum_{\eta=1}^{m} b_\eta P_\eta + \sum_{\xi=1}^{l} c_\xi T_\xi \right) \right) \\
& - \omega^2 \left(g_y g_{xz} \left(\sum_{i=1}^{N} a_i R_i + \sum_{\eta=1}^{m} b_\eta P_\eta + \sum_{\xi=1}^{l} c_\xi T_\xi \right) \right) \\
& - \omega^2 \left(g_z g_{xy} \left(\sum_{i=1}^{N} a_i R_i + \sum_{\eta=1}^{m} b_\eta P_\eta + \sum_{\xi=1}^{l} c_\xi T_\xi \right) \right) \\
& - \omega^2 \left(g_y g_z \left(\sum_{i=1}^{N} a_i \partial_x R_i + \sum_{\eta=1}^{m} b_\eta \partial_x P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_x T_\xi \right) \right) \\
& - \omega^2 \left(g_x g_z \left(\sum_{i=1}^{N} a_i \partial_y R_i + \sum_{\eta=1}^{m} b_\eta \partial_y P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_y T_\xi \right) \right) \\
& - \omega^2 \left(g_x g_y \left(\sum_{i=1}^{N} a_i \partial_z R_i + \sum_{\eta=1}^{m} b_\eta \partial_z P_\eta + \sum_{\xi=1}^{l} c_\xi \partial_z T_\xi \right) \right) \\
& - j\omega^3 \left(g_x g_y g_z \left(\sum_{i=1}^{N} a_i R_i + \sum_{\eta=1}^{m} b_\eta P_\eta + \sum_{\xi=1}^{l} c_\xi T_\xi \right) \right),
\end{aligned}
\tag{I.13}
$$

where ∂_x denotes the partial derivative with respect to x, similarly for the other cases.

The next step is to rewrite Eq. (I.13) such that it is ordered per type of basis function. This yields

$$\begin{aligned}
\hat{f} =& \sum_{i=1}^{N} a_i \left(\partial_{xyz} R_i + j\omega \left(g_{xyz} R_i + g_{yz}\partial_x R_i + g_{xz}\partial_y R_i + g_{xy}\partial_z R_i + g_z \partial_{xy} R_i + g_y \partial_{xz} R_i + g_x \partial_{yz} R_i\right)\right) \\
&+ \sum_{i=1}^{N} a_i \left(-\omega^2 \left(g_x g_{yz} R_i + g_y g_{xz} R_i + g_z g_{xy} R_i + g_y g_z \partial_x R_i + g_x g_z \partial_y R_i + g_x g_y \partial_z R_i\right) - j\omega^3 g_x g_y g_z R_i\right) \\
&+ \sum_{\eta=1}^{m} b_\eta \left(\partial_{xyz} P_\eta + j\omega \left(g_{xyz} P_\eta + g_{yz}\partial_x P_\eta + g_{xz}\partial_y P_\eta + g_{xy}\partial_z P_\eta + g_z \partial_{xy} P_\eta + g_y \partial_{xz} P_\eta + g_x \partial_{yz} P_\eta\right)\right) \\
&+ \sum_{\eta=1}^{m} b_\eta \left(-\omega^2 \left(g_x g_{yz} P_\eta + g_y g_{xz} P_\eta + g_z g_{xy} P_\eta + g_y g_z \partial_x P_\eta + g_x g_z \partial_y P_\eta + g_x g_y \partial_z P_\eta\right) - j\omega^3 g_x g_y g_z P_\eta\right) \\
&+ \sum_{\xi=1}^{l} c_\xi \left(\partial_{xyz} T_\xi + j\omega \left(g_{xyz} T_\xi + g_{yz}\partial_x T_\xi + g_{xz}\partial_y T_\xi + g_{xy}\partial_z T_\xi + g_z \partial_{xy} T_\xi + g_y \partial_{xz} T_\xi + g_x \partial_{yz} T_\xi\right)\right) \\
&+ \sum_{\xi=1}^{l} c_\xi \left(-\omega^2 \left(g_x g_{yz} T_\xi + g_y g_{xz} T_\xi + g_z g_{xy} T_\xi + g_y g_z \partial_x T_\xi + g_x g_z \partial_y T_\xi + g_x g_y \partial_z T_\xi\right) - j\omega^3 g_x g_y g_z T_\xi\right)
\end{aligned} \tag{I.14}$$

We now define the functions $\gamma(x,y,z)$, $\kappa(x,y,z)$ and $\psi(x,y,z)$ such that

$$\hat{f} = \sum_{i=1}^{N} a_i \gamma_i + \sum_{\eta=1}^{m} b_\eta \kappa_\eta + \sum_{\xi=1}^{l} c_\xi \psi_\xi. \tag{I.15}$$

By comparing Eq. (I.14) and Eq. (I.15), we find

$$\begin{aligned}
\gamma_i =& j\omega \left(g_{xyz} R_i + g_{yz}\partial_x R_i + g_{xz}\partial_y R_i + g_{xy}\partial_z R_i + g_z \partial_{xy} R_i + g_y \partial_{xz} R_i + g_x \partial_{yz} R_i\right) \\
&- \omega^2 \left(g_x g_{yz} R_i + g_y g_{xz} R_i + g_z g_{xy} R_i + g_y g_z \partial_x R_i + g_x g_z \partial_y R_i + g_x g_y \partial_z R_i\right) \\
&+ \partial_{xyz} R_i - j\omega^3 g_x g_y g_z R_i
\end{aligned} \tag{I.16a}$$

$$\begin{aligned}
\kappa_\eta =& j\omega \left(g_{xyz} P_\eta + g_{yz}\partial_x P_\eta + g_{xz}\partial_y P_\eta + g_{xy}\partial_z P_\eta + g_z \partial_{xy} P_\eta + g_y \partial_{xz} P_\eta + g_x \partial_{yz} P_\eta\right) \\
&- \omega^2 \left(g_x g_{yz} P_\eta + g_y g_{xz} P_\eta + g_z g_{xy} P_\eta + g_y g_z \partial_x P_\eta + g_x g_z \partial_y P_\eta + g_x g_y \partial_z P_\eta\right) \\
&+ \partial_{xyz} P_\eta - j\omega^3 g_x g_y g_z P_\eta
\end{aligned} \tag{I.16b}$$

$$\begin{aligned}
\psi_\xi =& j\omega \left(g_{xyz} T_\xi + g_{yz}\partial_x T_\xi + g_{xz}\partial_y T_\xi + g_{xy}\partial_z T_\xi + g_z \partial_{xy} T_\xi + g_y \partial_{xz} T_\xi + g_x \partial_{yz} T_\xi\right) \\
&- \omega^2 \left(g_x g_{yz} T_\xi + g_y g_{xz} T_\xi + g_z g_{xy} T_\xi + g_y g_z \partial_x T_\xi + g_x g_z \partial_y T_\xi + g_x g_y \partial_z T_\xi\right) \\
&+ \partial_{xyz} T_\xi - j\omega^3 g_x g_y g_z T_\xi
\end{aligned} \tag{I.16c}$$

The previous set of equations allows for constructing the matrix equation which is used to obtain the weights. As stated before, we start by choosing a set of N reference nodes to which Eq. (I.8) is fitted. These coordinates are denoted as $\mathbf{x}_1 = (x_1, y_1, z_1), \ldots, \mathbf{x}_N = (x_N, y_N, z_N)$. Furthermore, define the matrices $\Gamma \in \mathbb{C}^{N\times N}$, $K \in \mathbb{C}^{N\times m}$ and $\Psi \in \mathbb{C}^{N\times l}$. The elements Γ_{ki}, $K_{k\eta}$ and $\Psi_{k\xi}$ denote $\gamma_i(\mathbf{x}_k)$, $\kappa_\eta(\mathbf{x}_k)$ and $\psi_\xi(\mathbf{x}_k)$ respectively (for $k \in \{1, 2, \ldots, N\}$).

By enforcing $\hat{f}(\mathbf{x}_k) = f(\mathbf{x}_k)$ for $k \in \{1, \ldots, N\}$ and by making use of Eq. (I.15) we can write

$$\begin{bmatrix} f(\mathbf{x_1}) \\ \vdots \\ f(\mathbf{x_N}) \\ \mathbf{0}_{(m+l)\times 1} \end{bmatrix} = \begin{bmatrix} \Gamma & K & \Psi \\ K^T & \mathbf{0}_{m\times m} & \mathbf{0}_{m\times l} \\ \Psi^T & \mathbf{0}_{l\times m} & \mathbf{0}_{l\times l} \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_N \\ b_1 \\ \vdots \\ b_m \\ c_1 \\ \vdots \\ c_l \end{bmatrix}, \tag{I.17}$$

where $\mathbf{0}_{a\times b}$ is the $a \times b$ all-zero matrix of indicated size. The additional zeros are added to enforce a unique solution.

Eq. (I.17) can be solved for the weights, though it can be ill-conditioned. In that case, a regularised solution should be taken [95].

I.2 Discussion

I want to finalise this discussion by mentioning two papers which might provide more accurate an/or efficient means to solve the integrals, though it should be noted that they do not solve the exact same integral. One of these papers, namely [109], explicitly considers solving the azimuthal Fourier component by evaluating the Fourier integral over the Greens functions in cylindrical coordinates and can be considered state-of-the-art [110]. The second paper, [110], considers an adaptive Levin method. It is shown that the accuracy of the obtained results is comparable to those of [109], albeit taking more time [110]. The larger flexibility of the second paper might, however, make this a more promising candidate for the considered use-case.

Appendix J
Stochastic Model of the Room Impulse Response in Small Rooms

On the following few pages, a draft of the paper "Stochastic model of the room impulse response in small rooms" is given. Such a model could, among others, be useful to avoid computing the RIR through the image-source method and to avoid computing the PSD matrices $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$ through the procedure described in Chapter 3.

Stochastic model of the room impulse response in small rooms

Dimme de Groot, Richard Eveleens, Arash Noroozi, and Jorge Martinez

***Abstract*—The room and its contents have a great impact on the behaviour of the sound field in a room. The reverberation time in a small and densely furnitured room can be roughly 200 ms, while the reverberation time in concert halls can be multiple seconds. Modelling the sound behaviour of a room is generally done using the Room Impulse Response (RIR). The RIR has many appliances in the field of audio processing. Think of, for example, concert hall design, commercial entertainment and localization. The estimation and modelling of the RIR is generally performed in a deterministic manner with techniques ranging from the idealistic image source method to highly complex room models. In this paper, a zeroth order stochastic characterisation of the RIR is proposed for rooms that can be considered average living rooms. By means of randomly generated rooms, a large set of simulations is performed. From this, time dependent distributions are derived. Additionally, it is shown that the directivity and transfer function of a loudspeaker has a great impact on the RIR and can thus not be ignored when modeling the RIR in a practical scenario.**

***Index Terms*—Room Impulse Response (RIR), Speaker directivity, Probability Density Function (PDF)**

I. Introduction

THE behaviour of a sound field in a room greatly depends on the shape and contents of the room. One way to characterize this influence of the room is the acoustic channel from a source to a receiver inside a room. This response is known as the Room Impulse Response (RIR), and it is of interest in many algorithms [1], [2], [3]. Examples include spatial sound [4], [5], [6], acoustic echo cancellation [3], [7], [8], [9], blind source separation [3], [9], speech dereverberation [3], [10], [11], [12] and beamforming [13].

Modelling the RIR has proven to be a computationally expensive task. Approaches include numerically solving the wave- or Helmholtz-equation with proper boundary conditions [14], [15] and geometrical acoustics [1], [16]. The former provides accurate RIRs, but is computationally too expensive to be used in real-time algorithms [2], [16]. Geometrical acoustic based approaches can be used in some real-time algorithms, but lack accuracy most pronounced in the low-frequency range [2], [16]. Additionally, properly modelling a (furnitured) room is difficult. Namely, the behaviour of each reflections depends on the type of material, angle of incidence, and signal frequency [1]. On top of these problems, additional challenges are introduced by the loudspeaker and receiver directivity pattern that need to be taken into account [17], [18].

Due to the above mentioned challenges, algorithms may profit from a stochastic characterisation of the RIR. The RIR can be decomposed in three parts, the direct path, the early reflections and the (late) reverberation [1], [2]. The stochastic characterisation of the reverberation is well known and may be modelled using plane-waves which arrive from all directions [13], [19]. The resulting distribution can be approximated using a Gaussian or logistic probability density function (pdf) [20]. Literature on the stochastic characterisation of the early reflections is limited.

In this paper, we aim to stochastically characterise the impulse response for an isotropic receiver located in a small room. The stochastic characterisation is limited to a zeroth order Markov process. The RIRs are simulated using the mirror-image source method [21], [22] and a number of different sources of variation are considered.

In the following, we first describe the simulation setup in Section II. This gives rise to three different scenarios with increasing source of variation. The results of these simulations are presented in Section III-A and further analysed in Section III-B. We finalize with the conclusion in Section IV.

II. Simulation setup

The simulation setup described in the following is designed to limit any biases in the obtained data. This is done by identifying parameters of interest and randomising these over some range of interest. We consider box-shaped rooms with length L_x, width L_y and height L_z. The room has origin $(0, 0, 0)$ and its corners are given by non-negative coordinates. The six walls have reflection coefficients specified by $\beta_i \in \mathbb{R}$, $i \in \{1, \ldots, 6\}$. For each of the three simulations described below, an isotropic receiver located at $(x_l, y_l, z_l) = (2, 1, 1)$ m is considered.

A. Simulation 1

In the first simulation, only the reflection coefficients are randomised. The room dimensions are $(L_x, L_y, L_z) = (5, 5, 2.5)$ (m) and the loudspeakerhas a fixed coordinates which is specified by a spherical coordinate system centered around $\mathbf{x}_l$. Using the coordinate convention of [23], the loudspeaker location is given by $(\theta, \phi, r) = (0°, 60°, 1\text{ m})$.

Paper submitted on:... This work was performed in collaboration with the research team Kien

Dimme de Groot is with the Delft University of Technology, Delft 2628 CD The Netherlands (e-mail: dccjdegroot@student.tudelft.nl)

Richard Eveleens is with the Delft University of Technology, Delft 2628 CD The Netherlands (e-mail: reveleens@student.tudelft.nl)

Arash Noroozi is with Kien, Rotterdam 3013 AK The Netherlands (e-mail: arash@kien.io)

Jorge Martinez is with the Delft University of Technology, Delft 2628 CD The Netherlands (e-mail: J.A.MartinezCastaneda@tudelft.nl)

In Cartesian coordinates, this corresponds to $(x, y, z) \approx (2.87, 1.00, 1.50)$ m.

Three reflection coefficients are drawn independently from a uniform distribution $U(\cdot)$ according to

$$\begin{aligned} \beta_c &\sim U[0.5, 0.7], \\ \beta_{w_1}, \beta_{w_2} &\sim U[0.05, 0.5]. \end{aligned} \tag{1}$$

The remaining three reflection coefficients are obtained by changing the sign of the three drawn reflection coefficients. The reflection coefficients are subsequently assigned randomly to the walls while ensuring that the ceiling has either β_c or $-\beta_c$.

B. Simulation 2

Simulation 2 adds a few randomizing factors to Simulation 1 by additionally varying the loudspeaker location, the size of the room and the origin of the room. Namely, a random loudspeaker coordinate is drawn adhering to

$$\begin{aligned} \theta &\in [0, 2)\,(^\circ), \\ \phi &\in [60, 65)\,(^\circ), \\ r &\in [1, 1.4)\,(\mathrm{m}). \end{aligned} \tag{2}$$

The coordinate is drawn so that the distribution is uniform over the corresponding volume in a Cartesian coordinate system. The size of the region is based on findings in psychoacoustic literature. Namely, Humans have approximately 2° accuracy in localising azimuthal direction and 5° accuracy in localising elevation. The accuracy in distance r depends on the scenario and prior knowledge. It should be noted that the exact localisation accuracy varies and depends on, among others, the type of signal [24], [25].

The room-dimensions are varied as well and drawn according to

$$\begin{aligned} L_x &\sim U[5.0, 7.0] \quad (\mathrm{m}), \\ L_y &\sim U[5.0, 7.0] \quad (\mathrm{m}), \\ L_z &\sim U[2.5, 3.0] \quad (\mathrm{m}). \end{aligned} \tag{3}$$

To randomize the room placement, the origin of the room is shifted from $(0, 0, 0)$ to $(x_0, y_0, 0)$. The value (x_0, y_0) is drawn from

$$(x_0, y_0) \sim (U[-L_x + 5, 0],\, U[-L_y + 5, 0]) \quad (\mathrm{m}). \tag{4}$$

The origin in the z-direction is not varied since the receiver is assumed to remain at equal height.

C. Simulation 3

Simulation 3 is equal to Simulation 2, but with the addition of a directive loudspeaker. The normal (point of maximum gain) of the loudspeaker is set such that it points towards the receiver location. Note that this implies that it varies per per drawn loudspeaker location. The loudspeaker considered in our simulation is the KEF LS50. The directivity patterns are obtained through the implementation provided by [26], where a spherical harmonics representation is fitted on sparse measurement data provided by [27] to form a complete directivity pattern.

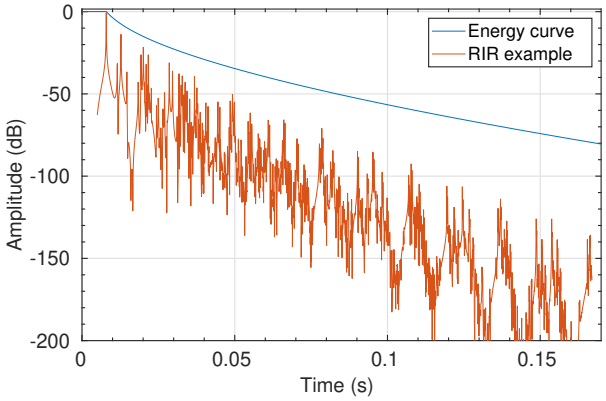


Fig. 1. The energy curve e on which the range of the histograms is based. An example RIR is added for reference

D. Implementation Details

The simulations were performed on MATLAB R2021b with default settings. For each of the simulations, a total of 60000 runs are considered. Since the simulations were collected over multiple runs, the random number generator was set to "shuffle". The RIRs were simulated through a modified version of [22]. The modified code returns individual reflections and the incidence and outgoing angles which are combined to form a RIR. No high pass filter is applied. As further explained in [22], the sampling occurs through a Hanning-windowed ideal low-pass filter with a length of 8 ms and a cutoff-frequency $f_s/2$ with $f_s = 16$ kHz the sampling frequency. The speed of sound was set to $c = 342$ m/s. The loudspeaker directivity patterns is based on linear interpolation with 5° resolution was created based on the directivity pattern. Downsampling was done using the resample function. The length of the considered impulse responses is 170 ms, which captures the majority of the possible rooms their reverberation time. The calculated RIR is normalized by shifting and scaling the response such that the reflection corresponding to the direct-path starts at the same time-sample and has its magnitude multiplied by $4\pi r_{l,s}$, with $r_{l,s}$ the distance between the loudspeaker and receiver.

For each simulation and speaker-receiver pair, each time-sample n of the computed RIRs corresponds to one histogram. The histogram limits are given by a time-sample dependent range which is derived based on the RIR energy curves e presented in [28]. The resulting curve is shown in Fig. 1. The centers of the 301 histogram bins are linearly spaced between $[-e(n), e(n)]$, so that bin 151 serves as the zero amplitude bin.

III. Simulation Results

The results of the three simulations are shown in Fig. 2 and Fig. 3. In both figures, each column is a different simulation.

In Fig. 2 the simulation results are presented by time-dependent histograms, where the x-axis denotes time and the y-axis denotes bin number. Note that, for each time-sample, the amplitude corresponding to a given histogram bin may vary. This happens in accordance with Fig. 1.

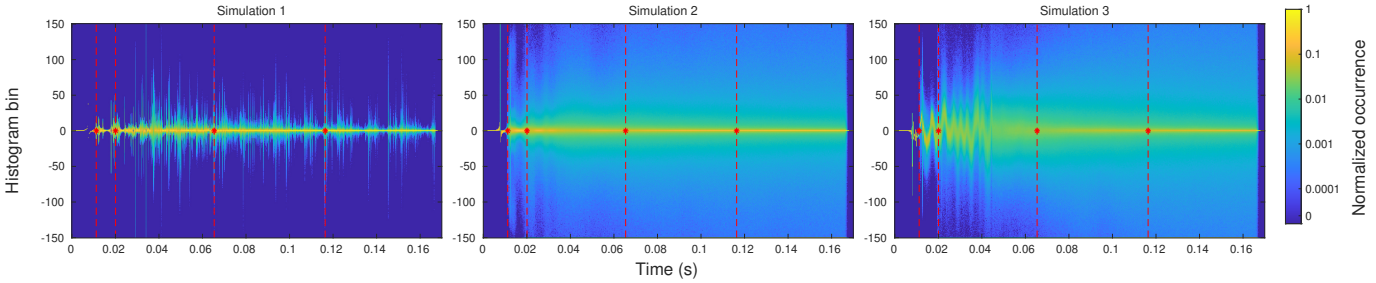


Fig. 2. The histograms resulting from the simulation. The colorbar shows a logarithmically scaled normalized occurrence of a certain histogram bin. Note that the y-axis represents the histogram bin number, this bin number should be translated to a time dependent amplitude by means of the energy curve found in Fig. 1. This is done for a some selected time samples in Fig. 3.

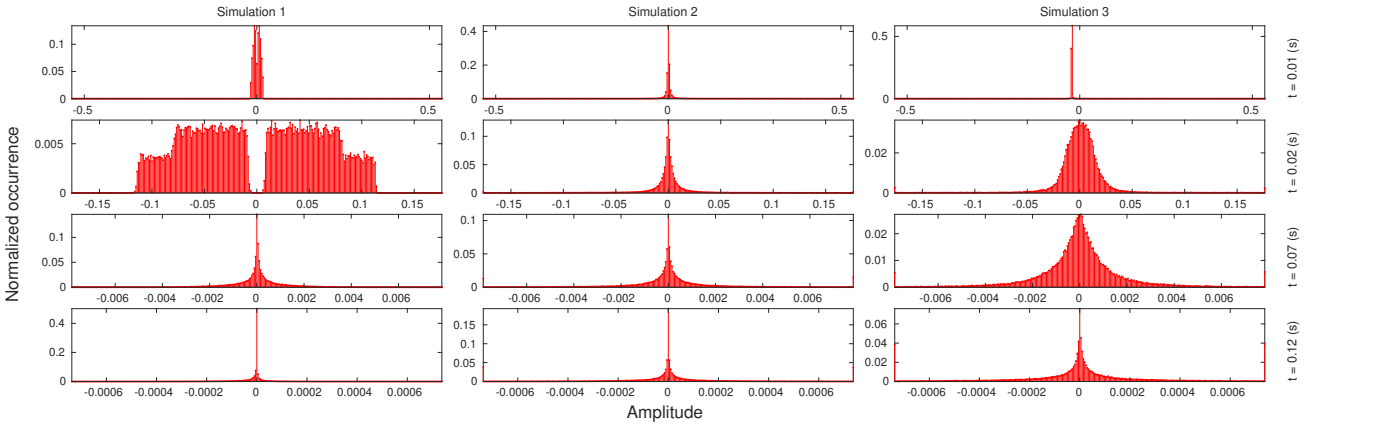


Fig. 3. The amplitude - normalized occurrence plots of a few time samples from Fig. 2 (indicated by the red-dotted line). The amplitude distribution of each time sample of the RIR for each simulation is estimated using these amplitude occurrences. Note that, per column, the amplitude axis remains equal. This is not true for the normalised occurrence, which differs per figure.

In Fig. 3 we zoom in on a few selected histograms (denoted by the red dotted lines in Fig. 2). In here, the x-axis corresponds to the amplitude of the RIR and the y-axis to the normalised number of occurrence.

A. Interpreting simulation results

As is expected, the results of Simulation 1 show clear peaks. The reason for this is that the only source of variation are the reflection coefficients. Thus, the time-of-arrival of each reflections remains equal, but the intensity differs. This is illustrated well by first figure in the second row of Fig. 3. Here, an early reflection is the sole reflection arriving at the receiver. Hence, it is not possible for the amplitude to be zero. The two step shape of this plot also shows the two reflection coefficients sizes, where a lower coefficient (β_ω) occurs more often than a higher coefficient (β_c), and that their signs flip. During the late reverberation, $t = 0.07$ s and $t = 0.12$ s, the distribution becomes more random as is predicted in (bron). (TOO: Ik zie dit niet echt tbh D̃imme.)

Both the histogram and the highlighted distributions of Simulation 2 show an increase in randomness. Apart from the direct path, the histograms show that mast samples fall into center bin. This is especially visible when comparing the results for $t = 0.02$ s, where the clear pattern that is observed in Simulation 1 is not visible anymore. This can be attributed to the time-of-arrival of the reflection becoming room dependent, thereby "smearing" the energy over multiple subsequent histograms. It is hypothesised that the histograms obtained from Simulation 2 serve as the fundamental distributions for the RIR of rooms fitting the room type considered.

Simulation 3 serves as a small but interesting sidestep to Simulation 2. In Simulation 3, the directivity and the directive transfer functions from a loudspeaker, the KEF LS50, are added to the equation. The histogram in Fig. 2 of Simulation 3 shows the great impact of the characteristics of a speaker on the RIR. Although concise conclusions can not be drawn on the exact influence of speakers on the RIR, it is clear that the speakers directivity and transfer function can not be ignored when modelling the RIR in a practical scenario.

B. Estimating RIR distributions

The time dependent histograms, as depicted in Fig. 3, can be used to derive Probability Density Functions (PDFs) for each time sample. The PDFs are be derived for the data from Simulation 2. Three possible distributions are selected to be fit on the data: the Laplace distribution, the normal distribution and the logistic distribution. The three distributions are fitted on the histograms and the best fitting distribution is selected by comparing the L_2-norms of the difference between PDF and the data. Doing this for all the obtained data shows that the direct path response (response up until $t \approx 0.01$ s) is best

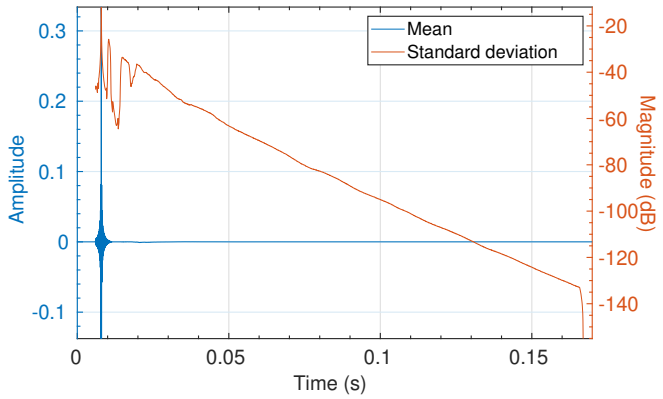


Fig. 4. An example of the time-dependent mean and standard deviation of one speaker-receiver pair. As expected, the mean is zero apart from the direct path. The standard deviation shows a clear pattern in the early reflections and shows a logarithmic delay afterwards.

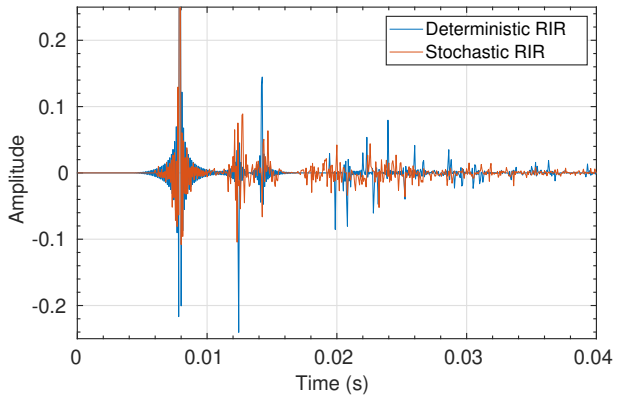


Fig. 5. A zoomed in example of a generated RIR with the proposed stochastic model and a single deterministic RIR from the same region. The major difference between the proposed solution and the deterministic model is that the deterministic model results in a few distinct peaks in the response while the proposed stochastic model has a more smoothed out response which should represent the generalized RIR for different but similar speaker-receiver pairs.

described by a normal distribution and the remainder is best described by a Laplace distribution.

An example of the mean and standard deviation of a speaker-receiver pair is given in Fig. 4. The figure shows that, apart from the direct path, the mean of the distributions is (approximately) zero. This is expected and validates that the definition of the reflection coefficients results in a zero mean RIR on average. The standard deviation shows a pattern in the early reflections (response up until $t \approx 0.025$ s) after which a logarithmic decay is observed.

An example of RIR generated based on the proposed model is shown in Fig. 5. The figure shows that the major difference between the proposed model and the deterministic model is that the deterministic response consists of a few distinct peaks while the proposed model shows a more smoothed out response. The behaviour of the standard deviation before $t \approx 0.025$ s as presented in Fig. 4 can be found in Fig. 5.

IV. Conclusion

In this paper, a stochastic Room Impulse Response (RIR) is introduced that is applicable for any furnitured and decorated room that can be characterised as standard shoebox living room. Classic approaches to derive a RIR are based on deterministic simulations or models that require precise prior knowledge on the room properties. In practical room scenarios, this prior knowledge is generally not available and expensive to obtain. A more general and widely applicable model of the RIR is desirable in this case. Deriving the stochastic RIR is performed by simulating a large amount of deterministic RIR's and fitting a probability density function on the set of simulated RIR's. The simulated RIR's show a direct path response best described by a normal distribution and the response caused by the reflections are best described by a Laplace distribution. Additionaly, it is shown that the influence of the directivity and response of the speaker on the RIR can not be ignored due to its significant impact.

REFERENCES

[1] H. Kuttruff, *Room Acoustics*, 5th ed, Taylor & Francis, 2009.

[2] J. Martinez, "Low-complexity computer simulation of multichannel room impulse responses," Ph.D. dissertation, Delft Univ. Technol., Delft, The Netherlands, 2013.

[3] Y. Huang, J. Benesty and J. Chen, *Acoustic MIMO Signal Processing*, 1st ed, New York, NY, USA: Springer, 2006.

[4] F. Melchior, "Investigations on spatial sound design based on measured room impulse responses," Ph.D. dissertation, Delft Univ. Technol., Delft, The Netherlands, 2011.

[5] M. Kolundzija, C. Faller, M. Vetterli. (2009, May). Sound field reconstruction: an improved approach for wave field synthesis. presented at AES Conv. 126.

[6] E. C. Hamdan, F. M. Fazi, "A modal analysis of multichannel crosstalk cancellation systems and their relationship to amplitude panning," *Journal of Sound and Vibr.*, vol. 490, pp. 115473, 2021, DOI. https://doi.org/10.1016/j.jsv.2020.115743.

[7] H. Buchner, J. Benesty, W. Kellermann, "Generalized multichannel frequency-domain adaptive filtering: efficient realization and application to hands-free speech communication," *Signal Processing*, vol. 85, pp. 549-570, 2005. DOI. https://doi.org/10.1016/j.sigpro.2004.07.029.

[8] K. Nathwani, "Joint acoustic echo and noise cancellation using spectral domain Kalman filtering in double-talk scenario" in IWAENC2018, Tokyo, Japan, 2018, pp. 326–330.

[9] M. Souden, Z. Liu, "Optimal joint linear acoustic echo cancelation and blind source separation in the presence of loudspeaker nonlinearity" in ICME, New York City, NY, USA, 2009, pp. 117–120.

[10] E.A.P. Habets, S. Gannot, "Dual-microphone speech dereverberation using a reference signal" in ICASSP '07, Honolulu, HI, USA, 2007, pp. 901–904.

[11] J. Zhang, M. D. Plumbley, W. Wang, "Weighted magnitude-phase loss for speech dereverberation" in ICASSP '21, Toronto, ON, Canada, 2021, pp. 5794–5798.

[12] P. A. Naylor, N. D. Gaubitch, *Speech Dereverberation*, 1st ed, Springer, 2010.

[13] J. Martinez, N. Gaubitch, W. B. Kleijn, "A robust region-based near-field beamformer" in ICASSP '15, South Brisbane, QLD, Australia, 2015, pp. 2494–2498.

[14] L. Thompson, "A review of finite-element methods for time-harmonics acoustics" *The Journal of the Acoustical Society of America*, vol. 119, pp. 1315-1330, 2006. DOI. https://doi.org/10.1121/1.2164987.

[15] R. Mehra, N. Raghuvanshi, L. Savioja, M. C. Lin, D. Manocha, "An efficient GPU-based time domain solver for the acoustic wave equation" *Applied Acoustics*, vol. 73, pp. 83-94, 2012. DOI. https://doi.org/10.1016/j.apacoust.2011.05.012.

[16] L. Savioja, U. P. Svensson, "Overview of geometrical room acoustic modeling techniques" *The Journal of the Acoustical Society of America*, vol. 138, pp. 708-730, 2015. DOI. https://doi.org/10.1121/1.4926438.

[17] F. Zotter, M. Frank, "Investigation of auditory objects caused by directional sound sources in rooms" *Acoustical Engineering*, vol. 128, pp. A5-A10, 2015. DOI. https://doi.org/10.12693/APhysPolA.128.A-5.

[18] H. Steffens, S. van der Par, S. D. Ewert "Perceptual relevance of speaker directivity modelling in virtual rooms" in ICA2019, Aachen, Germany, 2019, pp. 2651–2658.

[19] T. D. Abhayapala, R. A. Kennedy, R. C. Williamson "Isotropic noise modelling for nearfield array processing" in WASPAA'99, New Paltz, NY, USA, 1999, pp. 11-14.

[20] E. A. Lehmann, A. M. Johansson, "Diffuse Reverberation Model for Efficient Image-Source Simulation of Room Impulse Responses" *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, pp. 1429-1439, 2010. DOI. https://doi.org/10.1109/TASL.2009.2035038.

[21] J. B. Allen, D. A. Berkley, "Image method for efficiently simulating small-room acoustics" *The Journal of the Acoustical Society of America*, vol. 65, pp. 943-950, 1979. DOI. https://doi.org/10.1121/1.382599.

[22] E. A. P. Habets, "Room Impulse Response Generator", Internal Report, pp. 1-17, 2006.

[23] J. Stewart, D. Clegg, S. Watson, *Calculus: Early Transcendentals*, 9th ed. Metric Version, Cengage, 2021

[24] T. R. Letowski, S. T. Letowski, "Auditory Spatial Perception: Auditory Localization", ARL, Aberdeen P.G., MD, Scotland, Tech. Rep. ARL-TR-6016, May. 2012

[25] J. C. Middlebrooks, D. M. Green, "sound localization by human listeners" *Annual review of psychology*, vol. 42, pp. 135-159, 1991. DOI. https://doi.org/10.1146/annurev.ps.42.020191.001031.

[26] J. Ahrens, S. Bilbao, "Computation of Spherical Harmonics Based Sound Source Directivity Models from Sparse Measurement Data" *Forum Acusticum*, pp. 2019-2026, 2020. DOI. https://doi.org/10.48465/fa.2020.0042.

[27] J. G. Tylka, R. Sridhar, E. Y. Choueiri "A database of loudspeaker polar radiation measurements" in AES convention 139, New York, NY, USA, 2015, pp. 1-4.

[28] E. A. Lehmann, A. M. Johansson, "Prediction of energy decay in room impulse response simulated with an image-source model" *The Journal of the Acoustical Society of America*, vol. 124, pp. 269-277, 2008. DOI. https://doi.org/10.1121/1.2936367.

Appendix K
Including the Tail of the Room Impulse Response

This appendix described a possible solution to the problem mentioned in Section 4.1.3. This problem was that, due to the large number of filter segments ι, it is infeasible to optimise over all filter blocks simultaneously. Instead, it was chosen to optimise only for $\iota = 0$. The contribution of the remaining blocks is then treated as an error to be corrected, or they are simply ignored. In this appendix, a possible approach to correct for them is discussed. Thus, we aim to find an expression for the to-be corrected errors $\epsilon_l(\mathbf{x}_i)$, $i \in \{0, \dots, N_s\}$.

Consider the situation in which we want to find the playback signals corresponding to segment $l = l'$. Recall, from sections 2.4.1 and G.3, that the convolution between the signal $s(\mathbf{x}_i)$ and the filter $h(\mathbf{x}_i)$ can be written as

$$(s(\mathbf{x}_i) * h(\mathbf{x}_i))(n) = \sum_{l=-\infty}^{l'} \sum_{\iota=0}^{M/L-1} (\text{shift}_{-lR_1-\iota R_2}\{s_l(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) + \epsilon_{l>l'}(\mathbf{x}_i)(n), \tag{K.1}$$

where $\epsilon_{l>l'}$ is an error term depending only on the RIR and the "future" signal segments $l > l'$.

It should be noted that this equation assumes segmentation windows adhering to the constant overlap-add condition. In particular, in this appendix, I consider segmenting the signal with a Hanning window w_1 of even length L and repetition rate $R_1 = L/2$. The filter is segmented using a rectangular window w_2 of length L and repetition rate $R_2 = L$.

Since the playback segments are only calculated with respect to the tail of the previous playback segments and the filter block $\iota = 0$, we may define a signal $\bar{y}$ consisting of only those terms. This gives

$$\bar{y}_{l'}(\mathbf{x}_i, n) = \sum_{l=-\infty}^{l'-1} \sum_{\iota=0}^{M/L-1} (\text{shift}_{-lR_1-\iota R_2}\{s_l(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) + (\text{shift}_{-l'R_1}\{s_{l'}(\mathbf{x}_i) * h_0(\mathbf{x}_i)\})(n), \tag{K.2}$$

Now notice that, in accordance with the problem statement, we can only influence the time-samples for which the rightmost term is nonzero. Concretely, these samples are given by

$$\text{supp}(\text{shift}_{-l'R_1}\{s_{l'}(\mathbf{x}_i) * h_0(\mathbf{x}_i)\}) \subseteq \{l'R_1, \dots, l'R_1 + 2L - 2\}. \tag{K.3}$$

In order to include only these samples in the current optimisation step, one could chose to window $\bar{y}_{l'}$ using some window of proper length and repetition rate. However, this implies a window on the output of the convolution which, as we have seen in (4.5), does not translate well to a window on the reference signal. Instead, an easier approach can be found by considering that the length of the windows w_1 and w_2 are equal and that the repetition rates are R_1 and $R_2 = 2R_1$. This allows to only consider the contribution of *half* of the previous reference playbacks segments per optimisation step. Namely, if l' is even, all even valued segments with $l \leq l'$ are considered,

and similarly for odd l'. The reason why this works is probably best illustrated by an example, this is done below.

Consider a length $L = 10$, such that $R_1 = 5$ and $R_2 = 10$. Furthermore, let $M = 30$, such that $M/L = 3$. Lastly, consider to be at a segment $l = 8$. Using (K.1), the support of playback segment l filtered using filterblock ι is given by

$$\text{supp}(\text{shift}_{-lR_1-\iota R_2}\{s_l(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\}) \subseteq \{lR_1 + \iota R_2, \ldots, lR_1 + \iota R_2 + 2L - 2\}. \tag{K.4}$$

For the example, the right hand side evaluates to

$$\{lR_1 + \iota R_2, \ldots, lR_1 + \iota R_2 + 2L - 2\} = \{5l + 10\iota, \ldots, 5l + 10\iota + 18\}. \tag{K.5}$$

Evaluating (K.5) for some values of l and ι yields Table K.1.

Table K.1: The result of evaluating (K.5) for $l \in \{1, 2, 3, 4, 5, 6, 7, 8\}$ and $\iota \in \{0, 1, 2\}$. The bold font indicates an example of pairs (l, ι) where the support is the same.

l, ι	0	1	2
1	$\{5, ..., 23\}$	$\{15, ..., 33\}$	$\{25, ..., 43\}$
2	$\{10, ..., 28\}$	$\{20, ..., 38\}$	$\{30, ..., 48\}$
3	$\{15, ..., 33\}$	$\{25, ..., 43\}$	$\{35, ..., 53\}$
4	$\{20, ..., 38\}$	$\{30, ..., 48\}$	$\{\mathbf{40}, ..., \mathbf{58}\}$
5	$\{25, ..., 43\}$	$\{35, ..., 53\}$	$\{45, ..., 63\}$
6	$\{30, ..., 48\}$	$\{\mathbf{40}, ..., \mathbf{58}\}$	$\{50, ..., 68\}$
7	$\{35, ..., 53\}$	$\{45, ..., 63\}$	$\{55, ..., 73\}$
8	$\{\mathbf{40}, ..., \mathbf{58}\}$	$\{50, ..., 68\}$	$\{60, ..., 78\}$

As can be seen in the table, the support of the pair $(l, \iota) = (8, 0)$ is equal to that of $(6, 1)$ and that of $(4, 2)$. More generally, the support of the pair $(l, \iota) = (l', 0)$ equals that of $(l' - 2, 1)$, that of $(l' - 4, 2)$ etc. up to and including $\left(l' - \frac{2M}{L} + 2, \frac{M}{L} - 1\right)$.

This pattern allows to construct a properly windowed target signal $\tilde{y}_{l'}(\mathbf{x}_i, n)$. It is given by

$$\tilde{y}_{l'}(\mathbf{x}_i, n) = \sum_{\iota=0}^{M/L-1} (\text{shift}_{-(l'-2\iota)R_1-\iota R_2}\{s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n). \tag{K.6}$$

Recall that the windows were chosen such that $R_1 = L_1/2$, $R_2 = L$ and $R_2 = 2R_1$. Thus, as would be expected from Table K.1,

$$-(l' - 2\iota)R_1 - \iota R_2 = -(l' - 2\iota)R_1 - 2\iota R_1 = -l'R_1. \tag{K.7}$$

It follows that (K.6) reduces to

$$\tilde{y}_{l'}(\mathbf{x}_i, n) = \sum_{\iota=0}^{M/L-1} (\text{shift}_{-l'R_1}\{s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n). \tag{K.8}$$

In a few paragraphs, it will be convenient to have access to a version of this signal with support $\{0, \ldots, 2L - 2\}$. This is version is straightforwardly obtained by defining

$$y_{l'}(\mathbf{x}_i, n) = \tilde{y}_{l'}(\mathbf{x}_i, n + l'R_1) = \sum_{\iota=0}^{M/L-1} (s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i))(n). \tag{K.9}$$

Lastly, note that the approach to computing the target signal outlined above is only valid if

$$\sum_{l=-\infty}^{\infty} \bar{y}_l(\mathbf{x}_i, n) = (s(\mathbf{x}_i) * h(\mathbf{x}_i))(n). \tag{K.10}$$

To prove this, one should consider summing (K.6) over all l' and see if it can be rewritten to equal (K.1). Doing so results in

$$\begin{aligned}
&\sum_{l'=-\infty}^{\infty} \sum_{\iota=0}^{M/L-1} (\text{shift}_{-(l'-2\iota)R_1-\iota R_2}\{s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) = \\
&\sum_{\iota=0}^{M/L-1} \sum_{l'=-\infty}^{\infty} (\text{shift}_{-(l'-2\iota)R_1-\iota R_2}\{s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) = \\
&\sum_{\iota=0}^{M/L-1} \sum_{l=-\infty}^{\infty} (\text{shift}_{-lR_1-\iota R_2}\{s_l(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) = \\
&\sum_{l=-\infty}^{\infty} \sum_{\iota=0}^{M/L-1} (\text{shift}_{-lR_1-\iota R_2}\{s_l(\mathbf{x}_i) * h_\iota(\mathbf{x}_i)\})(n) = (s(\mathbf{x}_i) * h(\mathbf{x}_i))(n).
\end{aligned} \tag{K.11}$$

Where the substitution $l = l' - 2\iota$ was used. Thus, indeed, one can use $y_{l'}(\mathbf{x}_i, n)$ as the target signal for segment l'.

The signal $y_{l'}(\mathbf{x}_i, n)$ can be split in the error term $\epsilon_{l'}(\mathbf{x}_i)(n)$ and the reference signal. Doing so gives

$$\begin{aligned}
y_{l'}(\mathbf{x}_i, n) &= (s_{l'}(\mathbf{x}_i) * h_0(\mathbf{x}_i))(n) + \sum_{\iota=1}^{M/L-1} (s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i))(n) \\
&= (s_{l'}(\mathbf{x}_i) * h_0(\mathbf{x}_i))(n) + \epsilon_{l'}(\mathbf{x}_i, n).
\end{aligned} \tag{K.12}$$

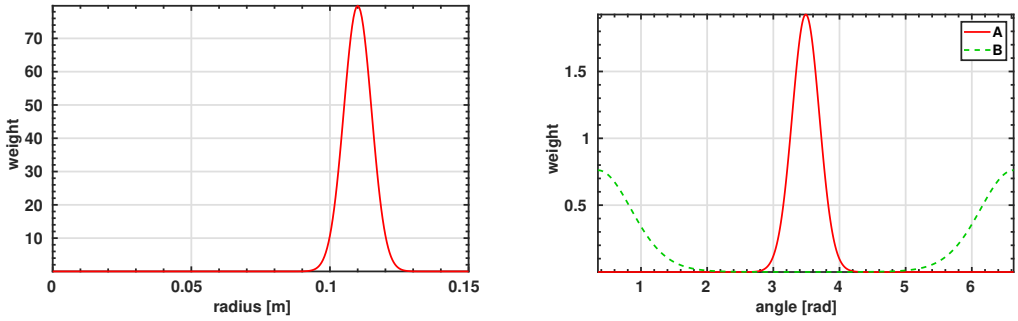
so that

$$\epsilon_{l'}(\mathbf{x}_i, n) = \sum_{\iota=1}^{M/L-1} (s_{(l'-2\iota)}(\mathbf{x}_i) * h_\iota(\mathbf{x}_i))(n). \tag{K.13}$$

Appendix L
Additional figures concerning the results

In this appendix, some additional figures concerning the results are given.

The weighting functions in r and θ are given in Figure L.1. The image sources considered in the calculation of $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$ are shown in Figure L.2.



(a) $p(r)$ (b) $p(\theta)$

Figure L.1: The spatial weighting functions used in r (normal distribution) and θ (von Mises distribution). The mean $\mu_r = 0.11$ and the standard deviation, $\sigma_r = {}^{0.03}/_{6}$. In regions $\mathcal{A}$ and $\mathcal{B}$, the Von Mises distribution are respectively parameterised by $(\mu_{\mathcal{A}}, \kappa_{\mathcal{A}}) \approx (3.49, {}^{15\pi}/_{2})$ and $(\mu_{\mathcal{B}}, \kappa_{\mathcal{B}}) \approx (3.49 + \pi, {}^{5\pi}/_{4})$.

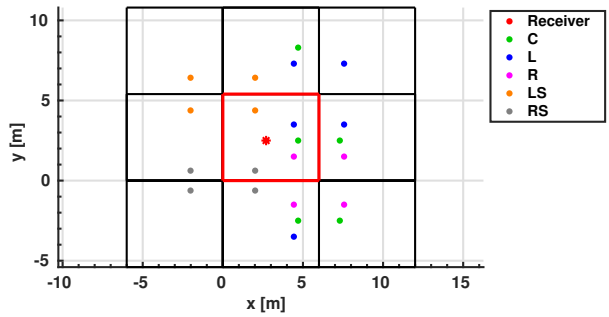


Figure L.2: The image sources considered in the calculation of $\mathbf{R}_{\mathcal{A}}$ and $\mathbf{R}_{\mathcal{B}}$. The physical room is highlighted.

L.1 Additional results speech signal

The normalised energy received at the left ear and at the right ear are given in Figure L.3 and Figure L.4, respectively.

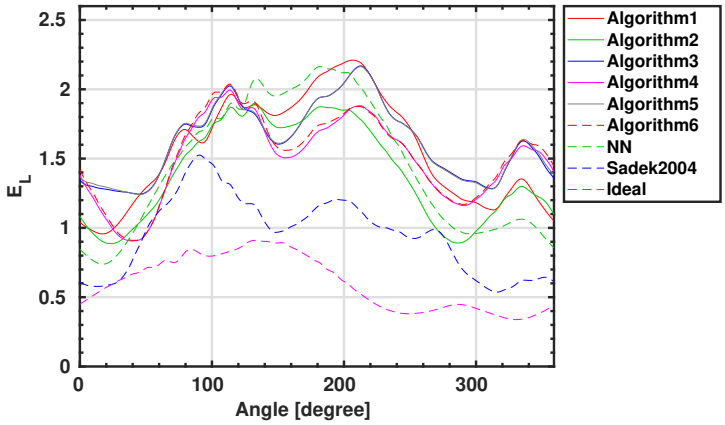


Figure L.3: The energy E_L received at the left ear. The considered signal is the female-voiced speech signal.

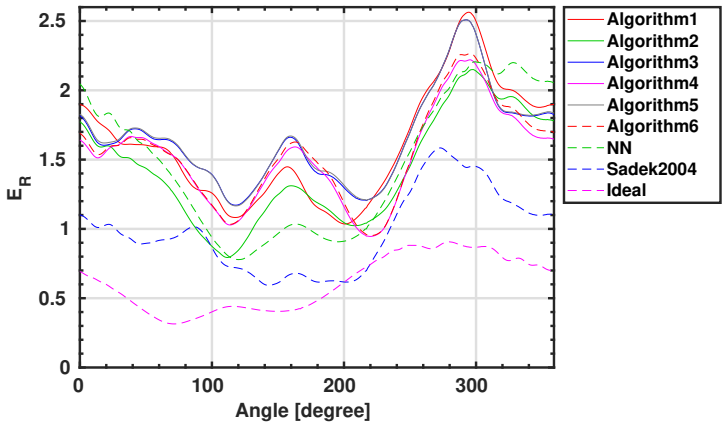


Figure L.4: The energy E_R received at the right ear. The considered signal is the female-voiced speech signal.

L.2 Additional results gong signal

For the gong signal, the argument at which the cross-correlation attains its maximum value (within -1 to 1 ms) is given in Figure L.5. The difference in received energy at the left and right ear is given in Figure L.6.

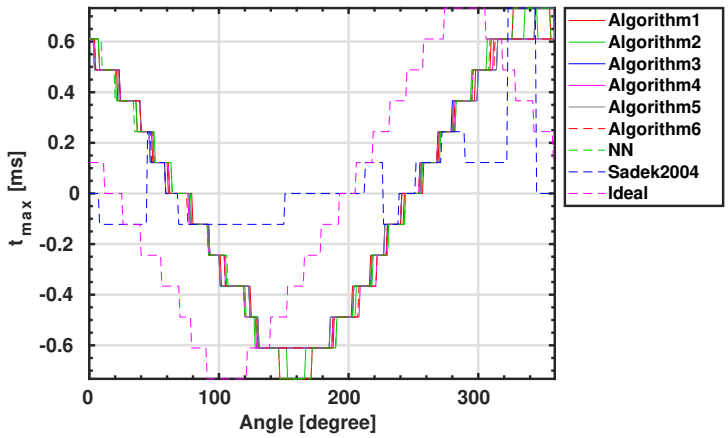


Figure L.5: The argument $t_{\max}$ (in ms) for which the cross-correlation between the audio received at the left and at the right ear attains its maximum value. The considered signal is the gong signal. The staircase shape is due to the limited sample rate.

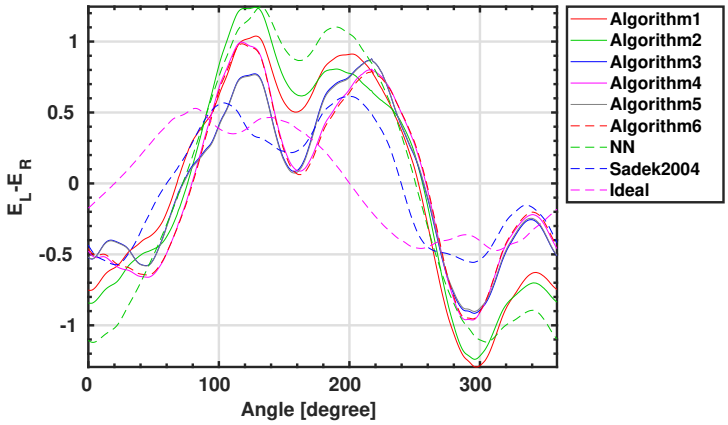


Figure L.6: The difference in energy received at the left and right ear $E_L - E_R$. The considered signal is the gong signal.

The normalised energy received at the left ear and at the right ear are given in Figure L.7 and Figure L.8, respectively.

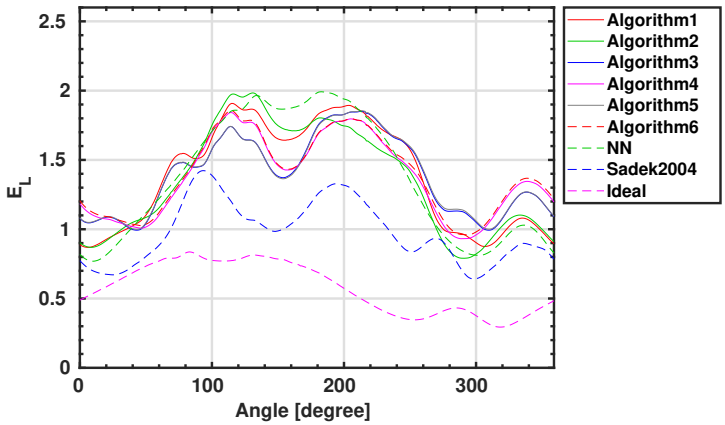


Figure L.7: The energy E_L received at the left ear. The considered signal is the gong signal.

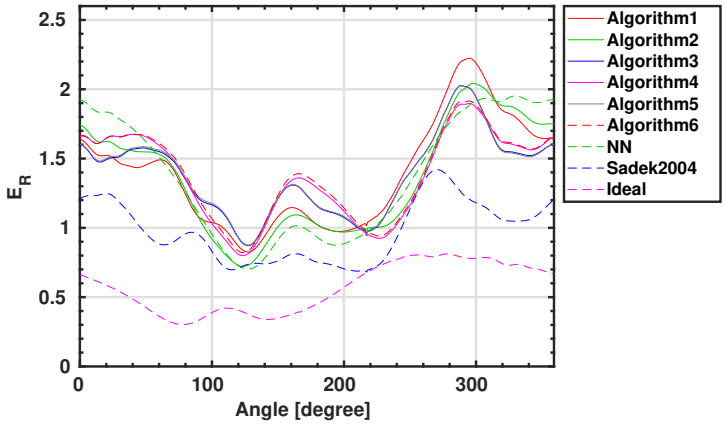


Figure L.8: The energy E_R received at the right ear. The considered signal is the gong signal.

Bibliography

[1] L. Chaparro and A. Akan, *Signals and Systems Using MATLAB*. Academic Press, 2019.

[2] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2009.

[3] M. Stéphane, *A Wavelet Tour of Signal Processing (Third Edition)*, 3rd ed. Academic Press, 2009.

[4] J. Ahrens, *Analytic Methods of Sound Field Synthesis*. Springer-Verlag, 2012.

[5] C. R. J. Roger A. Horn, *Matrix Analysis*. Cambridge University Press, 1990.

[6] R. Yates and D. Goodman, *Probability and Stochastic Processes, Third Ed., International student version*. Wiley, 2015.

[7] F. Rumsey, *Spatial Audio*. Focal press, 2001.

[8] M. Schoeffler, A. Silzle, and J. Herre, "Evaluation of spatial/3d audio: Basic audio quality versus quality of experience," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 1, pp. 75–88, 2017.

[9] International Telecommunication Union, "Recommendation ITU-R BS.775-4: Multichannel stereophonic sound system with and without accompanying picture," Geneva, Switzerland, Dec. 2022.

[10] S. Spors, H. Wierstorf, A. Raake, F. Melchior, M. Frank, and F. Zotter, "Spatial sound with loudspeakers and its perception: A review of the current state," *Proceedings of the IEEE*, vol. 101, no. 9, pp. 1920–1938, 2013.

[11] C. Kyriakakis and R. Sadek, "A novel multichannel panning method for standard and arbitrary loudspeaker configurations," *journal of the audio engineering society*, october 2004.

[12] E. C. Hamdan and F. M. Fazi, "A modal analysis of multichannel crosstalk cancellation systems and their relationship to amplitude panning," *Journal of Sound and Vibration*, vol. 490, p. 115743, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022460X20305733

[13] P. Damaske, "Head-related two-channel stereophony with loudspeaker reproduction," *The Journal of the Acoustical Society of America*, vol. 50, no. 4B, pp. 1109–1115, 1971.

[14] O. Kirkeby, P. A. Nelson, and H. Hamada, "Local sound field reproduction using two closely spaced loudspeakers," *The Journal of the Acoustical Society of America*, vol. 104, no. 4, pp. 1973–1981, 1998. [Online]. Available: https://doi.org/10.1121/1.423763

[15] T. Takeuchi and P. A. Nelson, "Optimal source distribution for binaural synthesis over loudspeakers," *The Journal of the Acoustical Society of America*, vol. 112, no. 6, pp. 2786–2797, Dec. 2002.

[16] E. Choueiri, *Optimal Crosstalk Cancellation for Binaural Audio with Two Loudspeakers*. Princeton University, 2010.

[17] M. R. Bai, C.-W. Tung, and C.-C. Lee, "Optimal design of loudspeaker arrays for robust cross-talk cancellation using the taguchi method and the genetic algorithm," *The Journal of the Acoustical Society of America*, vol. 117, no. 5, pp. 2802–2813, May 2005.

[18] M. Simón Gálvez and F. Fazi, "Loudspeaker arrays for transaural reproduction," in *Proceedings of the 22nd International Congress on Sound and Vibration, Florence, Italy*, July 2015.

[19] C. Hohnerlein and J. Ahrens, "Perceptual evaluation of a multiband acoustic crosstalk canceler using a linear loudspeaker array," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 96–100.

[20] X. Ma, C. Hohnerlein, and J. Ahrens, "Listener-position adaptive crosstalk cancelation using a parameterized superdirective beamformer," in *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, 2018, pp. 114–118.

[21] m. f. s. gálvez, d. menzies, and f. m. fazi, "dynamic audio reproduction with linear loudspeaker arrays," *journal of the audio engineering society*, vol. 67, no. 4, pp. 190–200, april 2019.

[22] M. Simón Gálvez, T. Takeuchi, and F. Fazi, "Low-complexity, listener's position-adaptive binaural reproduction over a loudspeaker array," *Acta Acustica united with Acustica*, vol. 103, pp. 847–857, 09 2017.

[23] J. Ahrens, M. R. P. Thomas, and I. Tashev, "Gentle acoustic crosstalk cancelation using the spectral division method and ambiophonics," in *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2013, pp. 1–4.

[24] H. Wierstorf, A. Raake, and S. Spors, "Localization of a virtual point source within the listening area for wave field synthesis," in *133rd Audio Engineering Society Convention*, San Francisco, CA, October 2012, p. Paper 8743. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=16485

[25] F. Zotter and M. Frank, *Ambisonics*. Springer, 2019.

[26] J. Ahrens and S. Spors, "An analytical approach to sound field reproduction using circular and spherical loudspeaker distributions," *Acta Acustica united with Acustica*, vol. 94, pp. 988–999, 12 2008.

[27] M. Frank, F. Zotter, and A. Sontacchi, "Localization experiments using different 2d ambisonics decoders (lokalisationsversuche mit verschiedenen 2d ambisonics dekodern)," in *Proc. 25. Tonmeistertagung*, 2008, pp. 696–704.

[28] E. Habets, "Room impulse response generator," *Internal Report*, pp. 1–17, 01 2006.

[29] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.2," http://cvxr.com/cvx, Mar. 2020.

[30] B. Moore, *An Introduction to the Psychology of Hearing: Sixth Edition*. Leiden, The Netherlands: Brill, 2013.

[31] E. Z. Hugo Fastl, *Psychoacoustics*. Springer Berlin, Heidelberg, 2013.

[32] R. Fettiplace, "Hair cell transduction, tuning, and synaptic transmission in the mammalian cochlea," *Compr Physiol*, vol. 7, no. 4, pp. 1197–1227, Sep. 2017.

[33] O. N. Milekhina, D. I. Nechaev, V. V. Popov, and A. Y. Supin, "Compressive nonlinearity in the auditory system: Manifestation in the action of complex sound signals," *Biology Bulletin*, vol. 44, no. 6, pp. 603–609, Nov. 2017.

[34] P. Heil and A. J. Peterson, "Basic response properties of auditory nerve fibers: a review," *Cell and Tissue Research*, vol. 361, no. 1, pp. 129–158, Jul. 2015.

[35] A. L. Nuttall, A. J. Ricci, G. Burwood, J. M. Harte, S. Stenfelt, P. Cayé-Thomasen, T. Ren, S. Ramamoorthy, Y. Zhang, T. Wilson, T. Lunner, B. C. J. Moore, and A. Fridberger, "A mechanoelectrical mechanism for detection of sound envelopes in the hearing organ," *Nat Commun*, vol. 9, no. 1, p. 4175, Oct. 2018.

[36] L. R. Bernstein, "Auditory processing of interaural timing information: new insights." *Journal of neuroscience research*, vol. 66, no. 6, pp. 1035–1046, Dec 2001.

[37] L. R. Bernstein and C. Trahiotis, "How sensitivity to ongoing interaural temporal disparities is affected by manipulations of temporal features of the envelopes of high-frequency stimuli," *The Journal of the Acoustical Society of America*, vol. 125, no. 5, pp. 3234–3242, 2009.

[38] K. Iida, *Head-Related Transfer Function and Acoustic Virtual Reality*. Springer, 2017.

[39] G. Yu, R. Wu, Y. Liu, and B. Xie, "Near-field head-related transfer-function measurement and database of human subjects," *The Journal of the Acoustical Society of America*, vol. 143, no. 3, pp. EL194–EL198, 2018.

[40] D. S. Brungart and W. M. Rabinowitz, "Auditory localization of nearby sources. head-related transfer functions," *The Journal of the Acoustical Society of America*, vol. 106, no. 3, pp. 1465–1479, 1999.

[41] J. C. Middlebrooks, "Chapter 6 - sound localization," in *The Human Auditory System*, ser. Handbook of Clinical Neurology, M. J. Aminoff, F. Boller, and D. F. Swaab, Eds. Elsevier, 2015, vol. 129, pp. 99–116.

[42] S. Li and J. Peissig, "Measurement of head-related transfer functions: A review," *Applied Sciences*, vol. 10, no. 14, 2020.

[43] R. Fernandez Martinez, P. Jimbert, E. M. Sumner, M. Riedel, and R. Unnthorsson, "Prediction of head related transfer functions using machine learning approaches," *Acoustics*, vol. 5, no. 1, pp. 254–267, 2023.

[44] R. Sridhar and E. Choueiri, "A method for efficiently calculating head-related transfer functions directly from head scan point clouds," in *Audio Engineering Society Convention 143*, Oct 2017. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=19289

[45] H. S. Braren and J. Fels, "A High-Resolution Head-Related Transfer Function Data Set and 3D-Scan of KEMAR," 2020.

[46] V. Algazi, R. Duda, D. Thompson, and C. Avendano, "The cipic hrtf database," in *Proceedings of the 2001 IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics (Cat. No.01TH8575)*, 2001, pp. 99–102.

[47] Institut für Schallforschung, "HRTF-Database," https://www.oeaw.ac.at/en/isf/das-institut/software/hrtf-database, 2020, accessed on 29-06-2022.

[48] I. Kazuhiro, "The CIT HRTF database ver. 1.3," http://www.iida-lab.it-chiba.ac.jp/HRTF/index-j.html, 2014, accessed on 29-06-2022.

[49] F. Toole, *Sound Reproduction: The Acoustics and Psychoacoustics of Loudspeakers and Rooms*, ser. Audio Engineering Society Presents. Taylor and Francis, 2018.

[50] A. J. Oxenham and M. Wojtczak, "5 Frequency selectivity and masking," in *Oxford Handbook of Auditory Science: Hearing*. Oxford University Press, 01 2010.

[51] S. van de Par, A. Kohlrausch, R. Heusdens, J. Jensen, and S. H. Jensen, "A Perceptual Model for Sinusoidal Audio Coding Based on Spectral Integration," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 9, p. 317529, Jun. 2005.

[52] E. Terhardt, "Calculating virtual pitch," *Hear Res*, vol. 1, no. 2, pp. 155–182, Mar. 1979.

[53] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 451–515, 2000.

[54] R. J. Baker and S. Rosen, "Auditory filter nonlinearity across frequency using simultaneous notched-noise masking," *The Journal of the Acoustical Society of America*, vol. 119, no. 1, pp. 454–462, 2006.

[55] R. D. Patterson and I. Nimmo-Smith, "Off-frequency listening and auditory-filter asymmetry," *The Journal of the Acoustical Society of America*, vol. 67, no. 1, pp. 229–245, 1980.

[56] A. Saremi, R. Beutelmann, M. Dietz, G. Ashida, J. Kretzberg, and S. Verhulst, "A comparative study of seven human cochlear filter models," *The Journal of the Acoustical Society of America*, vol. 140, no. 3, pp. 1618–1634, 2016.

[57] E. Skudrzyk, *The Foundations of Acoustics*. Springer-verlag, 1971.

[58] G. Williams, *Fourier Acoustics: Sound Radiation and Nearfield Acoustical Holography*. Elsevier Science, 1999. [Online]. Available: https://books.google.nl/books?id=vjfKLFBgMeIC

[59] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[60] R. Eveleens, "Low complexity crosstalk cancellation algorithm for consumer audio systems," 2023, unpublished thesis.

[61] H. Kuttruff, *Room Acoustics, Fifth Edition*. Taylor & Francis, 2009. [Online]. Available: https://books.google.nl/books?id=X4BJ9ImKYOsC

[62] J. Martínez Castañeda, "Low-complexity computer simulation of multichannel room impulse responses," Ph.D. dissertation, Delft University of Technology, 2013.

[63] L. Savioja and U. P. Svensson, "Overview of geometrical room acoustic modeling techniques," *The Journal of the Acoustical Society of America*, vol. 138, no. 2, pp. 708–730, 2015.

[64] J. Borish, "Extension of the image model to arbitrary polyhedra," *The Journal of the Acoustical Society of America*, vol. 75, no. 6, pp. 1827–1836, 1984.

[65] T. Dau, D. Püschel, and A. Kohlrausch, "A quantitative model of the "effective" signal processing in the auditory system. i. model structure," *J Acoust Soc Am*, vol. 99, no. 6, pp. 3615–3622, Jun. 1996.

[66] ——, "A quantitative model of the "effective" signal processing in the auditory system. II. simulations and measurements," *J Acoust Soc Am*, vol. 99, no. 6, pp. 3623–3631, Jun. 1996.

[67] T. Dau, B. Kollmeier, and A. Kohlrausch, "Modeling auditory processing of amplitude modulation. i. detection and masking with narrow-band carriers," *The Journal of the Acoustical Society of America*, vol. 102, no. 5, pp. 2892–2905, 1997.

[68] ——, "Modeling auditory processing of amplitude modulation. ii. spectral and temporal integration," *The Journal of the Acoustical Society of America*, vol. 102, no. 5, pp. 2892–2905, 1997.

[69] C. H. Taal, R. C. Hendriks, and R. Heusdens, "A low-complexity spectro-temporal distortion measure for audio processing applications," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 5, pp. 1553–1564, 2012.

[70] C. Christiansen, M. S. Pedersen, and T. Dau, "Prediction of speech intelligibility based on an auditory preprocessing model," *Speech Communication*, vol. 52, no. 7, pp. 678–692, 2010.

[71] M. L. Jepsen, S. D. Ewert, and T. Dau, "A computational model of human auditory signal processing and perception," *The Journal of the Acoustical Society of America*, vol. 124, no. 1, pp. 422–438, 2008.

[72] C. Taal, "Prediction and optimization of speech intelligibility in adverse conditions," Ph.D. dissertation, Technische Universiteit Delft, 2013.

[73] R. Meddis, L. P. O'Mard, and E. A. Lopez-Poveda, "A computational algorithm for computing nonlinear auditory frequency selectivity," *The Journal of the Acoustical Society of America*, vol. 109, no. 6, pp. 2852–2861, 2001.

[74] E. A. Lopez-Poveda and R. Meddis, "A human nonlinear cochlear filterbank," *The Journal of the Acoustical Society of America*, vol. 110, no. 6, pp. 3107–3118, 2001.

[75] B. Warnaar, "Modeling and clinical diagnosis of dead regions in the cochlea," Doctoral Thesis, University of Amsterdam, 2013.

[76] H. Relaño-Iborra, J. Zaar, and T. Dau, "A speech-based computational auditory signal processing and perception model," *The Journal of the Acoustical Society of America*, vol. 146, no. 5, pp. 3306–3317, 2019.

[77] A. O. Vecchi, L. Varnet, L. H. Carney, T. Dau, I. C. Bruce, S. Verhulst, and P. Majdak, "A comparative study of eight human auditory models of monaural processing," *Acta Acustica*, vol. 6, p. 17, 2022.

[78] N. de Koeijer, "Sound zones with a cost function based on human hearing," Delft University of Technology, Tech. Rep., 2021.

[79] A. Jeannerot, N. de Koeijer, P. Martínez-Nuevo, M. B. Møller, J. Dyreby, and P. Prandoni, "Increasing loudness in audio signals: A perceptually motivated approach to preserve audio quality," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 1001–1005.

[80] J. O. Smith, *Spectral Audio Signal Processing*. http://ccrma.stanford.edu/~jos/sasp/, accessed 03-03-2023, online book, 2011 edition.

[81] J. Martinez, N. Gaubitch, and W. B. Kleijn, "A robust region-based near-field beamformer," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 2494–2498.

[82] L. Devroye, *Non-Uniform Random Variate Generation*. Springer-verlag, 1986.

[83] S. Braun, A. Kuklasiński, O. Schwartz, O. Thiergart, E. A. P. Habets, S. Gannot, S. Doclo, and J. Jensen, "Evaluation and comparison of late reverberation power spectral density estimators," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1056–1071, 2018.

[84] J. Proakis and D. Manolakis, *Digital Signal Processing*, ser. Prentice Hall international editions. Pearson Prentice Hall, 2007.

[85] E. D. Andersen, "On formuling quadratic functions in optimization models," MOSEK, Tech. Rep., 2013.

[86] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, N. L. Dahlgren, and V. Zue, "Timit acoustic-phonetic continuous speech corpus," 1993, lDC93S1. Web Download. Philadelphia: Linguistic Data Consortium.

[87] J. Breebaart, S. van de Par, A. Kohlrausch, and E. Schuijers, "Parametric coding of stereo audio," *EURASIP Journal on Advances in Signal Processing*, vol. 2005, no. 9, p. 561917, Jun 2005. [Online]. Available: https://doi.org/10.1155/ASP.2005.1305

[88] M. Raspaud, H. Viste, and G. Evangelista, "Binaural source localization by joint estimation of ild and itd," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, pp. 68–77, 2010.

[89] C. F. Altmann, R. Ueda, B. Bucher, S. Furukawa, K. Ono, M. Kashino, T. Mima, and H. Fukuyama, "Trading of dynamic interaural time and level difference cues and its effect on the auditory motion-onset response measured with electroencephalography," *NeuroImage*, vol. 159, pp. 185–194, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811917306237

[90] International Telecommunication Union, "Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs," Geneva, Switzerland, Feb. 2001.

[91] S. Van Kuyk, W. B. Kleijn, and R. C. Hendriks, "An instrumental intelligibility metric based on information theory," *IEEE Signal Processing Letters*, vol. 25, no. 1, pp. 115–119, 2018.

[92] Steven Van Kuyk, "Matlab code," 3-10-2017, accessed May 15, 2023. https://stevenvankuyk.com/matlab_code/.

[93] M. Wang, C. Boeddeker, R. G. Dantas, and ananda seelan, "ludlows/python-pesq: supporting for multiprocessing features," May 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6549559

[94] P. Damaske and Y. Ando, "Interaural crosscorrelation for multichannel loudspeaker reproduction," *Acta Acustica united with Acustica*, vol. 27, no. 4, pp. 232–238, 1972.

[95] S. M. Hosseini and S. Smaeili, "Numerical integration of multi-dimensional highly oscillatory integrals, based on eRPIM," *Numerical Algorithms*, vol. 68, no. 2, pp. 423–442, Feb. 2015.

[96] J. D. Jackson, *Classical Electrodynamics, third edition.* John Wiley & Sons, Inc., 1999.

[97] G. Barton, *Elements of Green's Functions and Propagation.* Oxford University Press, 1989.

[98] R. Fitzpatrick, "Classical electromagnetism," https://farside.ph.utexas.edu/teaching/jk1/Electromagnetism/index.html, accessed: 2023-24-04.

[99] C. Taal and R. Heusdens, "A low-complexity spectro-temporal based perceptual model," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp. 153–156.

[100] C. H. Taal, R. C. Hendriks, and R. Heusdens, "Speech energy redistribution for intelligibility improvement in noise based on a perceptual distortion measure," *Computer Speech & Language*, vol. 28, no. 4, pp. 858–872, 2014.

[101] M. Stéphane, "Chapter 3 - discrete revolution," in *A Wavelet Tour of Signal Processing (Third Edition)*, 3rd ed. Boston: Academic Press, 2009, pp. 33–57.

[102] G. Charestan, R. Heusdens, and S. van de Par, "A gammatone-based psychoacoustical modeling approach for speech and audio coding," in *SAFE - ProRISC - SeSens 2001: proceedings. Semiconductor Advances for Future Electronics - Program for Research on Integrated Systems and Circuits - Semiconductor Sensor and Actuator Technology.* STW Technology Foundation, 2001, pp. 321–326.

[103] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization.* Philadelphia, PA: Society for Industrial and Applied Mathematics, 2019.

[104] B. R. Glasberg and B. C. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, no. 1, pp. 103–138, 1990.

[105] J. Stewart, *Calculus: Early Transcendentals.* Cengage Learning, 2021.

[106] D. Levin, "Procedures for computing one- and two-dimensional integrals of functions with rapid irregular oscillations," *Mathematics of Computation*, vol. 38, no. 158, pp. 531–538, Apr. 1982.

[107] R. Fateman, *DRAFT: When is a function oscillatory*, University of California, August 2009.

[108] G. R. Liu and Y. T. Gu, *Meshfree Shape Function Construction.* Dordrecht: Springer Netherlands, 2005, pp. 54–144. [Online]. Available: https://doi.org/10.1007/1-4020-3468-7_3

[109] J. Garritano, Y. Kluger, V. Rokhlin, and K. Serkh, "On the efficient evaluation of the azimuthal fourier components of the green's function for helmholtz's equation in cylindrical coordinates," *Journal of Computational Physics*, vol. 471, p. 111585, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999122006477

[110] S. Chen, K. Serkh, and J. Bremer, "The adaptive levin method," 2023.