Efficient Dynamic Rendering of Room Acoustics

The Development of a Hybrid Real-Time Model

Simon Rostami Mosen
Sound & Music Computing, 20176458, 2022-05

Master's Project

Studyboard of Media Technology
Aalborg University Copenhagen
http://smc.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Efficient Dynamic Rendering of Room Acoustics

Theme:
Sound & Music Computing

Project Period:
Spring Semester 2022

Project Group:
-

Participant(s):
Simon Rostami Mosen

Supervisor(s):
Stefania Serafin (internal)
Emanuele Parravicini (external)

Copies: 1

Page Numbers: 84

Date of Completion:
May 25, 2022

Abstract:

This thesis presents a hybrid algorithm for efficient dynamic rendering of room acoustics. To find the optimal perceptual trade-off between resolution and computational efficiency, perceptual listening experiments on the significance of different orders of early reflections were conducted. Based on the findings, a hybrid tool was designed and implemented in JUCE, combining dynamically rendered first-order reflections with convoluted late reverberation. A usability evaluation and a heuristic interview, proved the performance of the designed algorithm to be very effective. Moreover, a signal comparison and a perceptual comparison against a measured room impulse response, proved that while the proposed algorithm shares many attributes with the real room, some differences (possibly from the measurement of the room impulse response) are present.

Contents

Preface

This thesis was made as the final project in the 'Sound and Music Computing' (SMC) master's program at Aalborg University Copenhagen. The earliest seeds were sown throughout an internship in Audio Modeling in Italy in fall/winter 2021. During the internship, research on efficient modeling of room acoustics was initiated.

I would like to express my most sincere gratitude to Stefania Serafin for supervising this project as well as being an inexhaustible resource for guidance and inspiration.

Moreover, I wish to express my gratitude to Emanuele Parravicini and Stefano Lucato of Audio Modeling for granting me the opportunity of working with them, on a highly interesting topic, throughout my internship and during my master thesis. Finally, the entire team at Audio Modeling deserves my deepest appreciation for always being willing to share knowledge with me, taking aspects of this project to new levels.

Aalborg University Copenhagen, May 25, 2022

Simon Rostami Mosen
<srosta17@student.aau.dk>

Chapter 1

Introduction

We are surrounded by different acoustic experiences. While most of us do not consciously pay attention to room acoustics, many everyday experiences include the assessment of room acoustics. We quickly judge if the spoken word is easily understood in meeting rooms, restaurants, office spaces, etc. The prominent reverberation in a concrete staircase, the roaring sound of driving through a long tunnel, or singing in the bathroom, are all examples of everyday experiences given to us by differences in room acoustics. While the negative effects of (poor) room acoustics such as poor speech intelligibility are obvious, even for the untrained ear, the positive effects of (good) room acoustics can be slightly more subtle. Reverberation, a product of sound reflecting and scattering on different surfaces, is no new topic and has long been used for musical purposes (the first research in acoustics is often credited to the Greek philosopher Pythagoras, 6th century BC).

Until about 1950, reverberation on music recordings was just a by-product of the space. The longer the distance between the microphone and the sound source, the more reverberation ended up on the recorded signal [22]. The first electronic reverb was the *spring reverb* developed by Bell Labs for the Hammond organ in 1939, consisting of two long metal springs excited by an audio input. The vibration from the springs was transformed into an electrical signal with an iconic metallic reverberation sound [22]. In 1947 Bill Putnam Sr. developed the concept of the *echo chamber*, essentially playing back a recording in a reverberant room and capturing this with a microphone [22]. 10 years later, in 1957, the German company EMT developed an electronic reverberator based on a vibrating thin metal plate of 2 x 3 m in a steel frame, namely the first *plate reverb* which allowed for a much more realistic imitation of room reverberation.

Skipping 65 years ahead, artificial reverberation is widely used in most music recordings. As the physical modeling algorithms by Audio Modeling used for

their SWAM (Synchronous Waves Acoustic Modeling) instruments are anechoic in nature, this project aimed to investigate an efficient way of rendering realistic room acoustics, which would allow for unique early reflections depending on the location of the virtual instrument in the virtual room.

Chapter 2

Background

The following chapter will cover the theoretical background for this project covering the theoretical aspects of room acoustics, classical algorithms for artificial reverberation as well as algorithms based on geometrical acoustics.

2.1 Basics of Room Acoustics

When a loudspeaker is turned on, it radiates sound energy in a certain level [3]. When the loudspeaker is turned off, a certain time will pass before the sound level decays to inaudibility [3]. What is left behind after the sound is turned off is called reverberation, and is considered an important factor in the acoustic quality of a room [3]. The following section will describe the basics of room acoustics.

2.1.1 Reflection & Absorption

When a sound is emitted from a sound source, it propagates through air at a given speed [16]:

$$c = (331.4 + 0.6\theta)\mathrm{m/s} \tag{2.1}$$

where θ is the temperature in degrees centigrade.

When the traveling sound wave encounters a surface, it is reflected [3]. The essential idea of reflection on a flat surface is relatively simple [3]. The reflection of a point-source sound wave on a rigid flat surface is returned to the source, as light reflections in a mirror, known as specular reflections, described by Snell's law [3]. Similarly to light, the angle of incidence is equal to the angle of reflection, as seen in figure 2.1. The reflected sound is perceived as if it was originating from the image source (a virtual sound source), being located the same distance behind the reflecting surface as the original source is in front of the surface [3].

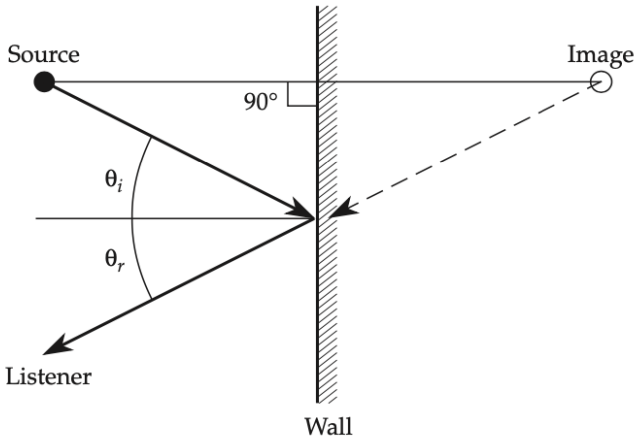


Figure 2.1: A specular reflection. Figure from [3].

When sound waves hit more than one surface, multiple reflections are created, as image sources of image sources, will exist [3]. For a rectangular, shoebox room, there are six surfaces and, therefore, the sound source has six image sources. Moreover, the reflection order is the number of surfaces the sound is reflected on before reaching the listener. The concept of image sources and how these are calculated is elaborated in section 2.2.2.

When a sound ray encounters a solid surface within an enclosure, part of the sound energy is usually reflected while another fraction of the sound energy is 'absorbed', either by conversion into heat or by being transmitted to the outside by the walls [16]. The amplitude and the phase of the reflected wave will differ from the incident wave and the changes in amplitude and phase can be expressed by the complex reflection factor R, being a property of the wall [16]:

$$R = |R| \cdot e^{j\phi} \tag{2.2}$$

where R is the complex reflection factor and $e^{j\phi}$ is the phase function.

It is given that the intensity of a plane wave is proportional to the square of the pressure amplitude, why the intensity of the reflected wave is scaled by a factor $|R|^2$ compared to the incident wave [16]. This, naturally, leads to the energy lost during reflection, namely the absorption coefficient of the wall which is defined as:

$$\alpha = 1 - |R|^2 \tag{2.3}$$

Absorption coefficients are essentially used to rate a material's effectiveness in absorbing sound [3].

2.1.2 Scattering

Most walls are not entirely smooth but contain irregularities, bumps, etc. [16]. When these irregularities are very small, they do not interfere with the specular reflections, however, once they become larger (compared to the wavelength) each of their faces can be considered a plane, which reflects the sound ray [16].

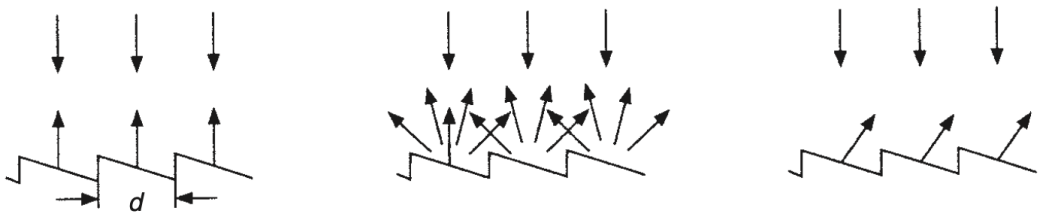


Figure 2.2: Scattering dependent on wavelength. Figure from [16].

Kuttruff describes how the relative amount of diffuse energy increases with the order of the reflections [16]. This results in virtually all reflected energy being converted into diffuse energy after just a few reflections [16]. This effect is seen in figure 2.3.

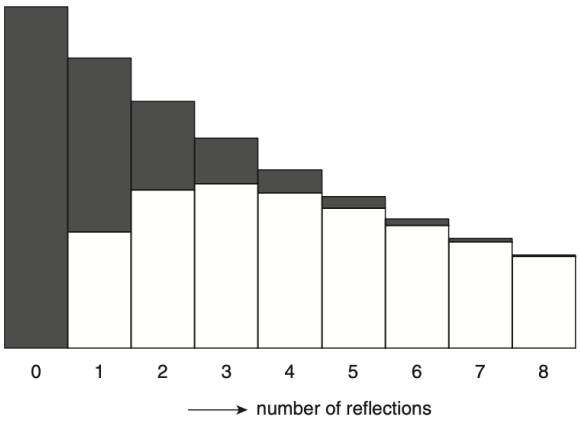


Figure 2.3: Relative amount of diffuse energy per reflection. Figure from [16].

2.1.3 Impulse Responses

Impulse Responses (IR) are considered one of the most important concepts in room acoustics [38]. It is used to describe the behavior of an (assumably) linear time-invariant (LTI) system, such as an acoustical space. Plotting IR arrival times of the different reflections are seen as perpendicular lines across the horizontal time axis, while the heights of the lines describe the relative energy in each reflection. The

impulse response is often split into three parts: the direct sound, early reflections (ER), and late reverberation [38]. This is seen in figure 2.4.

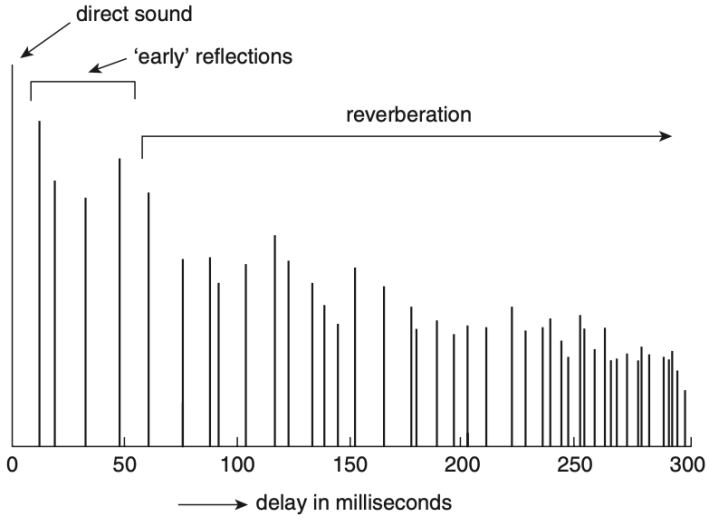


Figure 2.4: Illustration of direct sound, early reflections and (late) reverberation in an IR. [3].

The direct part is the sound reaching the listener first, basically traveling in a straight line from source to listener [38]. The early reflections are the first reflections reaching the listener after reflecting off a single surface and are usually not perceived as individual reflections by the human ear, but as an integrated part of the direct sounds [38]. The early reflections contribute to the perception of sound color and room size [38]. Late reverberation is the part of the IR where the reflection density has reached a level so high that individual reflections cannot be seen in the response [38]. The late reverberation indicates both room size and distance to the sound source.

Measuring an IR of an acoustical system essentially comes down to applying a known input signal and measuring the output from the system [33]. It is difficult to produce a perfect impulse (infinitely short and even amplitude across the entire frequency spectrum). To work around this problem, it is common to excite the system using either deterministic, wide-band signals such as a maximum-length sequence (MLS) and inverse repeated sequence (IRS) both using pseudo-random white noise or time-stretched pulses and sine-sweeps both using varying frequency signals [33]. Table 2.1 describes the disadvantages and advantages of the above-mentioned methods based on a comparison by Stan et al. [33].

Method	Advantages	Disadvantages
MLS (IRS)	High immunity to all kinds of noise. Week optimum output sound level. Timbre. Interesting method for occupied rooms.	Tedious calibration. Appearance of distortion peaks.
Time-stretched pulses	No distortion peaks.	Non-linearities can be superimposed with IR. Timbre. High optimum output.
Sine-sweep	Perfect rejection of harmonic distortions. Excellent signal-to-noise ratio. Does not require tedious calibration.	Not recommended in occupied rooms.

Table 2.1: Comparison of IR measurement methods

2.1.4 Convolution

Convolution is the mathematical operation of processing a signal going through the system using the IR [35]. Essentially, each input sample *x[n]* is (element-wise) sent through the system *h[n]*, producing the output signal *y[n]* [35]. A simple block diagram of the convolution process is seen in figure 2.5.

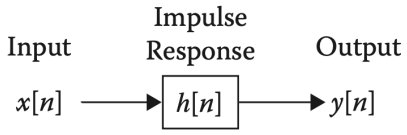


Figure 2.5: Block diagram of convolution. Figure from [35].

The mathematical equation for convolution is given as:

$$y[n] = x[n] * h[n] = \sum_{m=1}^{M} x[n-m+1] \cdot h[m], \text{ where } \mathbf{M} = \text{length}(h) \tag{2.4}$$

where $y[n]$ is the convolved output signal, $x[n]$ is the input signal and $h[h]$ is the IR of a system.

2.2 Modeling Room Acoustics

The following section describes different methods for modeling room acoustics. The section covers classical algorithms for artificial reverberation as well as geometrical methods used for room acoustics modeling.

2.2.1 Classical Algorithms

A selection of iconic algorithms for classical digital reverberation is covered in the following section.

Schroeder Reverb

In 1961 Manfred Schroeder of Bell Telephone Laboratories introduced the idea of artificial reverberation based on digital signal processing [30, 37]. Schroeder's implementation introduced the digital feedback comb filter and all-pass filter, widely used in reverb algorithms today, and used these as building blocks in series and parallel to build up repetitions of the input signal to achieve the characteristic echo density as known from reverberation [37].

Schroeder's algorithm consist of four comb filters in parallel feeding into two APFs in series as seen in figure 2.6 [26]. .

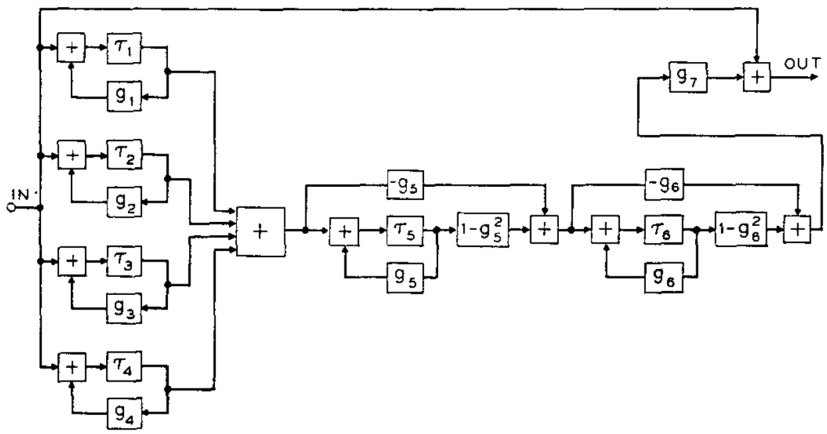


Figure 2.6: The original block diagram of Schroeder's algorithm. Figure from [30].

The comb filter is a special case of an IIR digital filter, as there is feedback from the delayed output to the input [26]. The comb filters imitate the behavior of a sound reflecting between parallel walls and thereby produce a series of echos [26]. A block diagram of the comb filter is seen in figure 2.7.

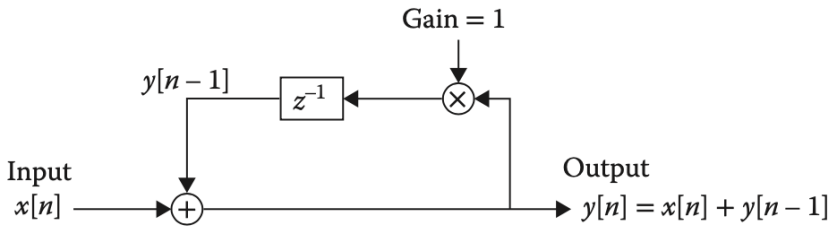


Figure 2.7: Block diagram of comb filter. Figure from [35].

By adjusting the gain of the feedback loops, the decay time of the reverberation can be controlled. Changes in the delay length of the comb filter may be used to

adjust the perceived room size [26]. One downside of the comb filter is that the distance between adjacent echos is defined by just one parameter. This results in periodicity to the output of the filter, essentially introducing a metallic ringing to the signal. To impede this effect, Schroeder connected the parallel comb filters to two all-pass filters in series. A block diagram of an APF is seen in figure 2.8.

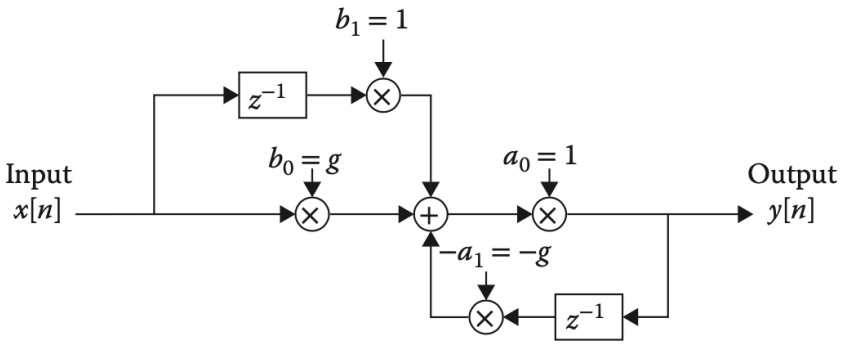


Figure 2.8: Block diagram of APF. Figure from [35].

An APF is a system that does not affect the relative amplitude of frequencies in the input signal, but instead introduces a frequency-dependent phase shift to the signal [35]. APFs in artificial reverberation are occasionally called "impulse diffusers", as they transform each input sample from the previous stage into an entire infinite IR [26]. This essentially leads to a higher echo density, resembling diffuse reflections (as described in 2.1.2). While APFs do not provide a physically accurate representation of diffuse reflections, the expansion of single reflections into multiple reflections shares perceptual qualities with the natural concept of diffusion [26].

Moorer Reverb

The next significant advancement in algorithmic reverberation is the work of James Moorer in 1979 [26]. Moorer's research builds upon the idea of Schroeder's reverberator by introducing a tapped delay line simulating early reflections and by inserting all-pole filters to control reverberation time as a function of frequency. This essentially allow frequencies to decay at different rates [18, 26, 37]. A block diagram of Moorer's algorithm is seen in figure 2.9.

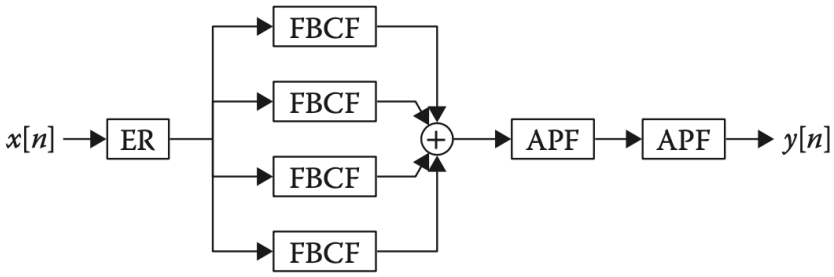
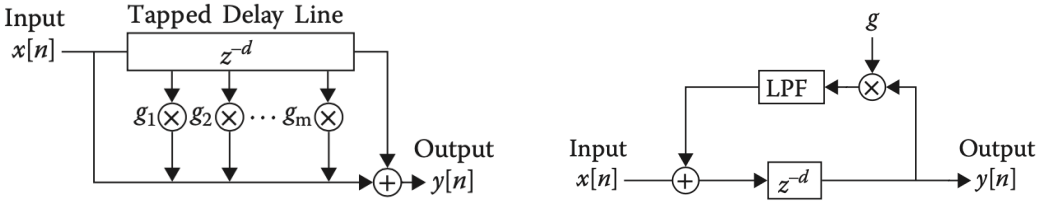


Figure 2.9: Block diagram of Moorer's algorithm. Figure from [35].

In figure 2.10a and figure 2.10b the early reflections block and the (lowpass) FBCF block are expanded.



(a) Early reflections block

(b) (Lowpass) FBCF block

Figure 2.10: Early reflections block and (lowpass) FBCF block. Figure from [35].

Moorer's addition of one-pole lowpass filters to the delay lines came from the idea that different frequencies decay at different rates and while comb filters were useful for modeling the decaying IR, it did not allow for frequency-dependent decay rates [26]. The addition of lowpass filters to the delay lines reduced the metallic sound and yielded a more natural sounding reverberation [26].

Feedback Delay Network

Introduced by Stautner & Puckette in 1982 and further refined by Jot in 1992, the feedback delay network (FDN) is the youngest algorithm in this section [14, 34]. Like Moorer's algorithm, the FDN further extends the feedback approach introduced by Schroeder [35]. In an FDN the output from each delay block is fed back to both the delay block itself, as well as any other delay block in the system [14]. This is achieved by interconnecting all delay lines in a feedback loop employing a feedback/scattering matrix that determines the input at which the output is fed back to [38]. A block diagram of an FDN with four delay lines is seen in figure 2.11.

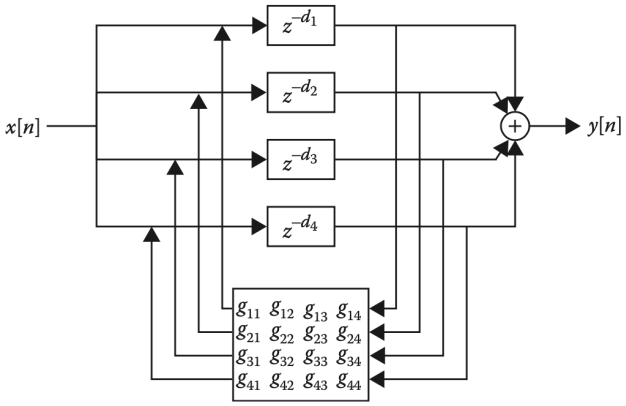


Figure 2.11: Block diagram of an FDN with four delay lines. Figure from [35].

Conceptually, the interconnected delay lines imitate how reflections travel in a room [35]. While the front and back wall, sidewalls, and floor and ceiling form feedback loops, they do not reflect in isolation and could therefore reflect in new feedback paths [35]. This is simulated through the feedback matrix, such as the ones seen in figure 4.6a, 4.6b, and 4.6c.

$$\begin{bmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \end{bmatrix}$$

(a) Stautner & Puckette Matrix [34, 35].

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

(b) Hadamard Matrix [35].

$$\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 \end{bmatrix}$$

(c) Householder Matrix [35].

Figure 2.12: Feedback/scattering matrices.

Each entry of the matrices essentially contains the various feedback gains for each possible feedback scenario [35].

2.2.2 Classical Geometrical Acoustics

The following section covers classical methods for modeling room acoustics based on room geometry. For simplicity, these methods often employ a limited case of very small wavelengths and are therefore considered a high-frequency approximation [16]. This approximation is justified if the dimensions of the room and its walls are large compared to the wavelenghts of the sound under consideration [16].

In geometrical acoustics, the concept of waves is replaced by sound rays [16]. A

sound ray, essentially, is a fraction of a spherical wave with a vanishing aperture [16]. Moreover, as sound rays are assumed to propagate in straight paths, diffraction is, generally, neglected in geometrical room acoustics [16].

Image Source Method

The Image Source method (IM) is a popular method for generating simulated RIRs and was originally presented by Allen and Berkley in 1979 [1, 26]. The algorithm is based on the principle that a specular reflection can be constructed geometrically by mirroring the source in the plane of the reflecting surface (as described in section 2.1.1) [27]. The approximate number of image sources within a radius of ct is:

$$N_{refl} = \frac{4\pi c^3}{3V} t^3 \tag{2.5}$$

This is an estimate of the number of reflections arriving at the receiver-position up to the time t after sound emission [27].

The image source method is accurate at estimating specular reflections in rectangular rooms, but for other room shapes, the case is slightly more complex [27]. Having n surfaces will result in n possible image sources of first-order [27]. Each of these can create $n-1$ second-order image sources. The number of possible image sources up to the reflection order i, is given by:

$$N_{sou} = 1 + \frac{n}{(n-2)} \left((n-1)^i - 1 \right) \approx (n-1)^i \tag{2.6}$$

The image source method is not very practical for higher orders for two reasons:

1. The exponential increase with reflection order makes it very computationally complex [27].

2. Considering figure 2.3, it is clear that the relative diffuse energy increases with reflection order. As the image source method assumes specular reflections, this assumption is more applicable for the earliest orders of reflections.

Ray Tracing Method

Ray tracing is a popular offline algorithm for modeling sound, as it allows for more advanced modeling compared to the image source method [38]. Where the image source method is deterministic and is guaranteed to find all the possible specular reflection paths between a source and a receiver, ray-tracing is stochastic and uses Monte Carlo sampling to estimate the reflection paths. The principle of ray tracing is illustrated in figure 2.13. At a known time, t, a sound source releases numerous sound particles in all directions. The sound particles travel in straight paths and

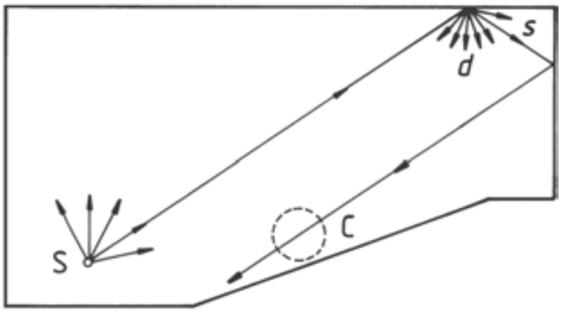


Figure 2.13: Principle of ray tracing. S = sound source, C = counting sphere, s = specular reflection and d = diffuse reflection Figure from [16].

are traced around the room, losing energy at each reflection [27]. When a particle hits a wall, a new direction, based on Snell's law, is calculated [27]. As opposed to the image source method, reflections can be either specularly reflected, in which the angle of incidence is equal to the angle of reflection (as in the image source method), or diffusely reflected, in which the direction of the reflected ray is randomized [3]. The output of the ray tracing algorithm is essentially calculated by determining when sound particles, emitted from the source, hit the receiver [27]. As seen in figure 2.13, the receiver (counting sphere) has a certain volume. This is necessary to allow the receiver to catch particles traveling by [27].

By adding volume to the receiver, there is a risk of registering false reflections and that some reflection paths are not found [27]. However, if the number of rays is sufficiently large and the receiver is sufficiently small, the ray-tracing method may yield a good approximation of the room impulse response (RIR) [25]. The minimum number of N rays necessary for room estimation is given by [27]:

$$N \geq \frac{8\pi c^2}{A} t^2 \tag{2.7}$$

where c is the speed of sound in air, A is the area and t is the time.

While the computational complexity of the image source method grows exponentially with the reflection order, this does not apply for ray tracing [37]. It is thus equally useful for higher order reflections, however, it should be noted that the accuracy of modeling decreases with increasing reflection order [37]. The efficiency of the ray tracing algorithm furthermore strongly depends on the desired time and spatial resolution [3].

Chapter 3

State Of The Art

The following chapter will investigate the state of the art in simulating room acoustics in real-time, to shed light on the strengths and weaknesses of these algorithms.

3.1 Digital Waveguide Mesh

The digital waveguide mesh (DWM) is a multidimensional extension of the digital waveguide. It is utilized for simulating wave propagation in a multidimensional system, making it useful for simulating room acoustics [9, 36]. A DWM consists of discretely spaced 1D digital waveguides arranged along each perpendicular dimension, interconnected at all intersections [9]. Each node in the grid is connected to its six neighboring nodes using unit delays [9]. At each lossless node connecting lines of equal impedance, the following two conditions must be met [9]:

1. *Volume velocity flows add to zero:* The sum of inputs must be equal to the sum of outputs.

2. *Continuity of pressure:* The sound pressures at each intersecting waveguide must be equal at the node.

Based on above-mentioned conditions, a differential equation for the nodes of a N-dimensional rectangular mesh is given as:

$$p_k(n) = \frac{1}{N}\left[\sum_{l=1}^{2N} p_l(n-1)\right] - p_k(n-2) \tag{3.1}$$

where p is the sound pressure at a node at time step n, k is the position of the node to be calculated, and l is all neighbors of k.

The propagation of waves in a 2D DWM is seen in figure 3.1.

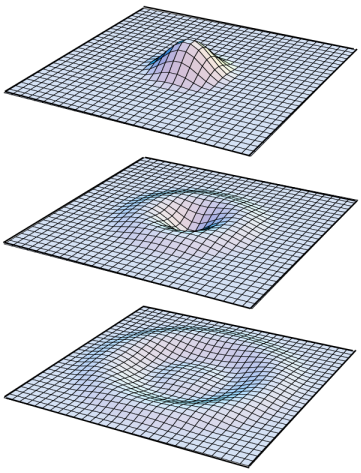
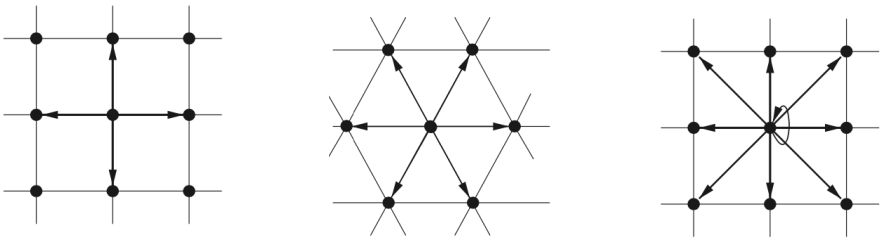


Figure 3.1: Wave progragation in a 2D digital waveguide mesh. Figure from [36].

The DWM approach to room acoustics suffers from direction- and frequency-dependent dispersion of wavefronts [9]. This has the effect that high-frequency signals propagating in parallel with the coordinate axes are delayed, while high-frequency signals traveling diagonally in the mesh, propagate undistorted [9].

Different mesh topologies can be used for the DWM. The two-dimensional rectangular mesh, being the most simple structure, is seen in figure 3.2a. Other structures, such as the triangular grid in figure 3.2b and the interpolated rectangular grid in figure 3.2c have been developed to reduce the direction-dependent dispersion, but all topologies suffer from some sort of dispersion [9]. While the interpolated mesh can be applied in both 2D and 3D scenarios, the three-dimensional counterpart to the two-dimensional triangular mesh is known as the tetrahedral mesh.

As the dispersion error is frequency-dependent, it was found that using frequency warping could remarkably reduce the error [9, 28]. In practice, a first-order all-pass filter is applied to both the input and output of the mesh, shifting frequencies to compensate for the dispersion error [9].



(a) Two-dimensional rectangular grid.

(b) Triangular grid.

(c) Interpolated rectangular grid.

Figure 3.2: Different mesh topologies. Figure from [9].

For the DWM the reflection characteristics of the surfaces in the room, are given by the boundary conditions and can generally be represented using digital filters [9, 10].

3.2 Scattering Delay Network

More recent development of real-time room acoustics, building on the idea of the IM, is the Scattering Delay Network (SDN). As the computational load of the IM grows exponentially as the order of reflection increases, this introduces certain limitations in terms of real-time applications. To solve this issue, De Sena et. al proposed an acoustic reverberator consisting of delay lines connected via scattering nodes to simulate the acoustics of an enclosure, ensuring that each significant reflection has a corresponding reflection in the synthesized algorithm [2]. The concept of the SDN is visualized in figure 3.3.

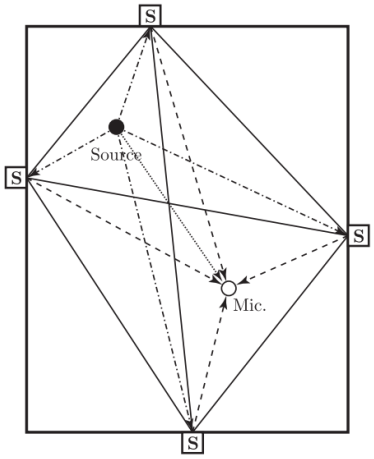


Figure 3.3: Conceptual visualization of SDN. Figure from [2].

The black lines in figure 3.3 denote bidirectional delay lines, connecting wall nodes. The nodes are denoted by **S** blocks on each wall. The dash-dotted line denotes unidirectional absorptive delay lines connecting the source to the nodes. Dashed lines denote the absorptive unidirectional delay lines from nodes to the microphone. The dotted line in the figure denotes the direct path from source to microphone.

The SDN differs from algorithms such as the IM, by rendering the first-order reflections correctly in both timing and amplitude (essentially matching the result from the IM), whereas higher-order reflections are increasingly (following the order) approximate [2]. This is achieved through a range of simple building blocks: scattering nodes, source-to-node connections, node-to-node connections, and node-to-mic connections.

Scattering nodes are junctions placed at the location of the first-order reflections, allowing higher-order reflections to be approximated by re-using the scattering nodes from the first-order reflections [2]. Each node carries out a scattering operation on the inputs from all the other nodes, based on a scattering matrix. This results in an order-dependent approximation for higher-order reflections. The source-to-node connections take care of the input to the system by connecting the source to all nodes using unidirectional absorbing delay lines. The node-to-node connections consist of bidirectional delay lines modeling the propagation delay between each scattering junction. Lastly, the node-to-microphone connections connect each node to the microphone through a unidirectional absorbing delay line [2].

3.3 Concerning Hybrids

Different techniques, such as the above-mentioned, have their strengths and weaknesses. For this reason, many hybrid models have been proposed, combining the strengths of different models into one technique [37]. Moorer's reverberator from 1979 (section 2.2.1), was essentially proposed as a hybrid structure, combining a tapped delay line with delay lengths calculated using the image source method (early reflections), with FBCFs and all-pass filters (late reverberation) [18].

As described in section 2.2.2, geometrical acoustics is a high frequency approximation, performing poorly in the lowest frequency range [37]. Therefore, a wave-based model such as (the computationally expensive) finite difference time domain (FDTD) method, can be necessary for low-frequency simulations, if the aim is accurate modeling [37]. Such model has been proposed by Murphy et al., where FDTD and ray-tracing are combined [19].

Figure 3.4 shows the different algorithms for modeling room acoustics and their optimum time and frequency area.

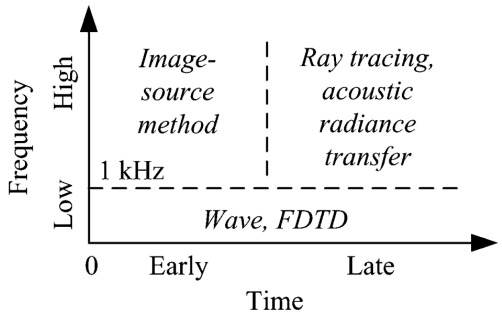


Figure 3.4: . Figure from [37].

Chapter 4

Design

This chapter documents the consideration that led to the final design of the real-time algorithm and interface. The aim was to implement a real-time VST plugin, artificially rendering room acoustics through a hybrid approach based on the previously described theory. The design of the audio algorithm, as well as the graphical user interface (GUI), was the product of discussions with the collaborators at Audio Modeling along with inspiration from relevant theory, traditional reverberation algorithms and state-of-the-art methods. Moreover, through iterative testing, different aspects were evaluated to identify optimal design choices. The following chapter will document both the design of the audio algorithm as well as the real-time interface used for interaction between the user and the audio algorithm.

4.1 Audio Algorithm

The audio algorithm aimed to conceive a computationally efficient way of realistically rendering room acoustic. From the onset of this project, it was, through discussion with collaborators at Audio Modeling, decided to use a hybrid approach where early reflections were rendered synthetically and late reverberation was applied through convolution. The idea was to dynamically render early reflections (possibly as few as one order) to allow for real-time adjustment of source- and microphone position to only spend computational power on rendering the early reflections. The late reverberation would then be applied through real-time convolution. Evaluation of the perceptual importance of early reflections was conducted to identify the number of orders necessary for a perceptually satisfactory result (section 7.1 and 7.2).

As described in section 2.1.1, not all sound is reflected after encountering a surface, why it was decided to include frequency-dependent wall absorption in the algorithm. The different materials included in the implementation and their re-

flection factor is seen in table 4.1. As described in section 2.1.1, most reflections in

Material	**125**	**250**	**500**	**1k**	**2k**	**4k**
Carpet: heavy on concrete	.02	.06	.14	.37	.60	.65
Carpet: heavy on 40-oz hair felt	.08	.24	.57	.69	.71	.73
Acoustical tile, avg, 1/2-in thick	.07	.21	.66	.75	.62	.49
Concrete block, coarse	.36	.44	.31	.29	.39	.25
Floor: wood	.15	.11	.10	.07	.06	.07
Glass: large panes, heavy glass	.18	.06	.04	.03	.02	.02
Plywood panel: 3/8-in thick	.28	.22	.17	.09	.10	.11
Perforated panel, perf: 8.7%	.27	.84	.96	.36	.32	.26

Table 4.1: Absorption coefficients at different frequency bands for different materials. Values from [3].

the real world are *not* specular. To accommodate for this, inspired by Schroeder's and Moorer's use of all-pass filters to resemble diffusion and thereby obtain higher echo density (as described in section 2.2.1), 26-order all-pass filters were applied to each delay line.

The early reflections are rendered through the following audio pipeline in JUCE:

1. Based on room size, source position, and microphone position, the first-order reflection points on each wall are calculated.

2. Knowing the positions of source, microphone, and reflections points as well as the speed of sound, the delay length of six delay lines, one per first-order reflection, can be calculated.

3. The amplitude of each reflection is calculated per channel (left/right) and is affected by distance traveled and the relative position between source and microphone.

4. The spectral and temporal content of each delay is affected by wall absorption filters and all-pass filters.

As the hybrid algorithm uses convolution with an IR of the late reverberation (measured from a room, further described in section 5), the RIR is prepared using the following pipeline in MATLAB:

1. The silence at the beginning of the RIR, coming from the distance the direct sound has to travel, is removed.

2. The theoretical last third-order reflection is calculated using the Image Source method.

3. All sound until this reflection is removed.
4. The first half of a Hann window is generated.
5. The window is applied to the beginning of the IR.

While the early reflections are dynamically rendered in real-time, the windowed IR is applied to the signal using convolution in JUCE. The IR is introduced at the position of the last first-order reflection. The entire process is visualized in figure 4.1.

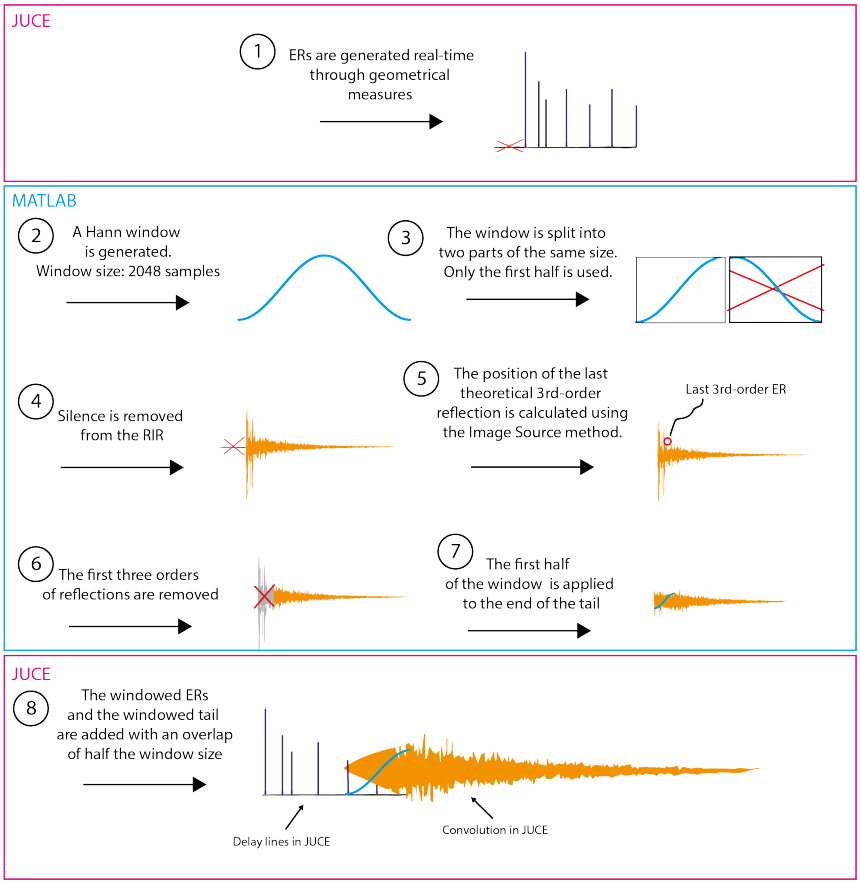


Figure 4.1: Diagram of the hybrid algorithm.

4.2 Graphical User Interface

While the previous section described the design of the audio algorithm, this section covers the design of the GUI. The final GUI is seen below in figure 4.2. The interface can be divided into 8 different sections: wall materials, source, and microphone

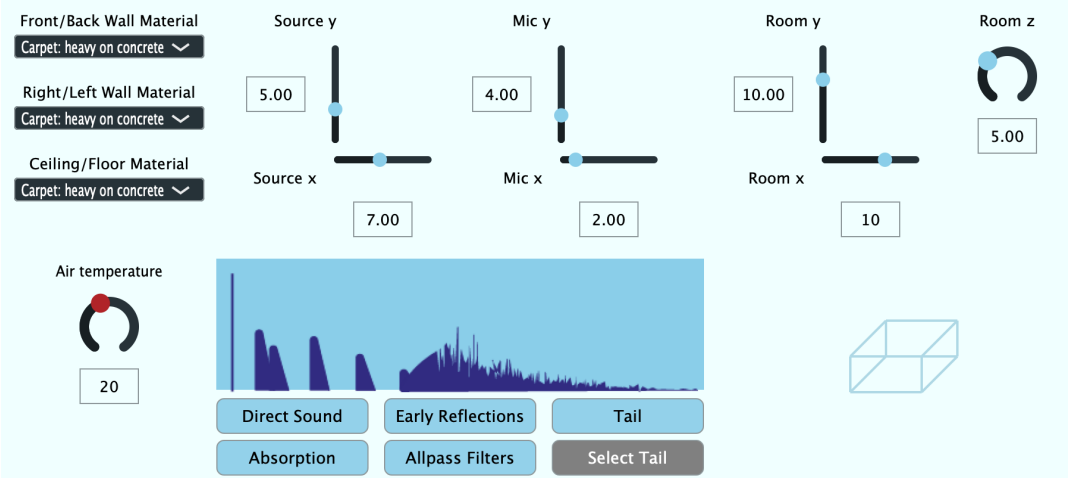
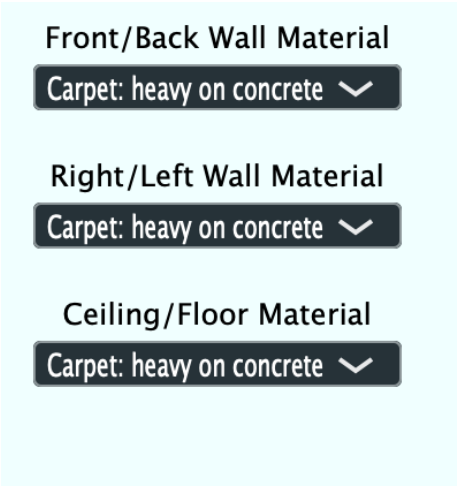


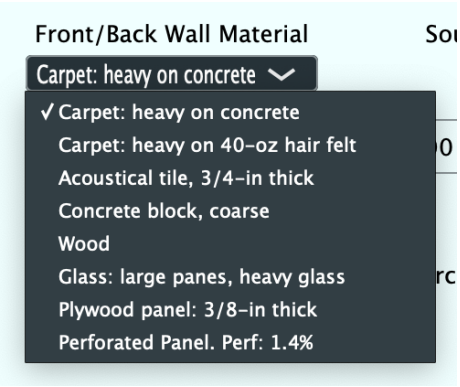
Figure 4.2: The final graphical user interface.

location, room size, air temperature, mute- and effect buttons, impulse response visualization, and room dimensions visualization.

Wall materials As the audio algorithm allowed for different wall materials, this was implemented in the GUI through drop-down menus. Parallel walls were paired to one menu to reduce the number of actions needed to set wall materials. The dropdown section is seen below in figure 4.3.



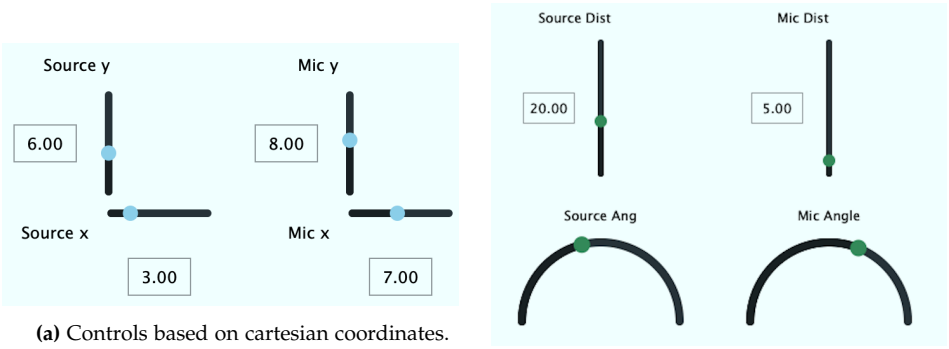
(a) Drop-down menu closed.



(b) Drop-down menu open.

Figure 4.3: Drop-down menus for selection of wall materials.

Source and microphone location To adjust the location of the sound source and the microphone, sliders were used to represent the cartesian coordinate system, as seen in figure 4.4a. Through discussion with the collaborators at Audio Modeling, informal experimentation with the polar coordinate system was conducted. The interface with polar controls is seen in figure 4.4b. While allowing for adjustment of location, based on angles and magnitudes, it was found to be difficult to determine the position of source and microphone (at least without visualization of source and microphone), why the cartesian coordinate system was used in the final implementation.



(a) Controls based on cartesian coordinates.

(b) Controls based on polar coordinates.

Figure 4.4: Different controls for position of sound source and microphone.

Room size The controls for room size did not develop significantly through the development of the GUI. The only aspect that was changed through the process was the labeling of depth and height controls. Initially, the depth slider was labeled as y, while height was labeled as z. The usability evaluation (section 7.5.1) shed light on confusion regarding these controls, why the label for the depth slider was changed to z *(depth)* and the label for the height slider was changed to y *(height)*.

Air temperature As the audio algorithm uses the speed of sound to calculate the delay lengths for the early reflections, the option to control air temperature seemed reasonable, due to the relation between the speed of sound and air temperature (as described in section 2.1.1).

Mute & effects buttons The audio algorithm treats the three parts of the IR independently (direct sound, early reflections, and late reverberation). Controls for muting each of these sections were included in the GUI. Furthermore, it was decided to allow the user to disable the effect of the wall absorption filters as well as the all-pass filters. While the effect of these controls is, naturally, heard in the audio output, it is furthermore visualized graphically in the GUI.

Impulse response visualization To visualize the effect of the mute button as well as the effect buttons, an arbitrary impulse response was visualized in the GUI. The visuals were designed in separate parts in Adobe Illustrator. The effect of a selection of the controls is seen in figure 4.5. By comparing figure 4.5a and figure 4.5b it can be seen how the effect of the wall absorption is visualized by rounding the sharp edge of the early reflections. This is, clearly, not an accurate representation of the given early reflections but an abstract visualization of the concept. Inspecting figure 4.5b against figure 4.5c shows visualization of the diffuse effect of the all-pass filters.

(a) Direct sound: ON, early reflections: ON, tail: ON, wall absorption: ON, APF: ON

(b) Direct sound: ON, early reflections: ON, tail: ON, wall absorption: OFF, APF: ON

(c) Direct sound: ON, early reflections: ON, tail: ON, wall absorption: OFF, APF: OFF

(d) Direct sound: OFF, early reflections: ON, tail: OFF, wall absorption: ON, APF: ON

Figure 4.5: Impulse response visualizations

Room dimensions visualization To provide the user with a visual representation of room size, a simple 3D cube was included in the GUI. The dimensions of the cube are dynamically changed when room size controls are adjusted. The visual effect of different room sizes is seen in figure 4.6

(a) Room size: width = 5, height = 3, depth = 15.

(b) Room size: width = 15, height = 3, depth = 15.

(c) Room size: width = 8, height = 15, depth = 2.

Figure 4.6: The visual effect of different room dimensions.

A video demonstration of the VST plugin is found in appendix VII.

Chapter 5

Implementation

This chapter describes the implementation of the hybrid reverberation algorithm described in the previous section, as a real-time VST. The VST is implemented in C++ using JUCE[1]. The entire source code and a VST3 file is found in appendix VIII.

5.1 Overall Structure

The implementation consists of the following classes:

- `Reflections`: This class is the backbone of first-order reflections. By instantiating the `Junction` class, positions for reflections are defined. Delay lines are instantiated with delay lengths based on the distance sound rays have to travel. This class, furthermore, handles the distance attenuation of the reflections.
- `Absorption`: This class handles the wall absorption filters. It takes user input from the GUI and handles the different filter coefficients for different wall materials.
- `Allpass`: This class handles the all-pass filters used for imitating diffuse reflections. The class instantiates a series of cascading filters to obtain a 26-order all-pass filter.
- `Junction`: This class handles the calculations of the different reflection points (junctions). This class is instantiated through the `Reflections class`.
- `Position`: This class is used to define positions in the implementation. Instead of using float x, y, and z to define positions for source, mic, junctions,

[1] https://juce.com/

etc., the objects of this class can be instantiated with x, y, and z coordinates through the constructor.

- `Predelay`: This class is used to delay the late reverberation to be introduced at the time of the last first-order reflection.

The convolution with the late reverberation is handled through separate classes, utilizing Graham Barab's implementation of low-latency convolution based on time distributed FFTs, as described in section 5.3. A block diagram of the hybrid algorithm is seen in figure 5.1.

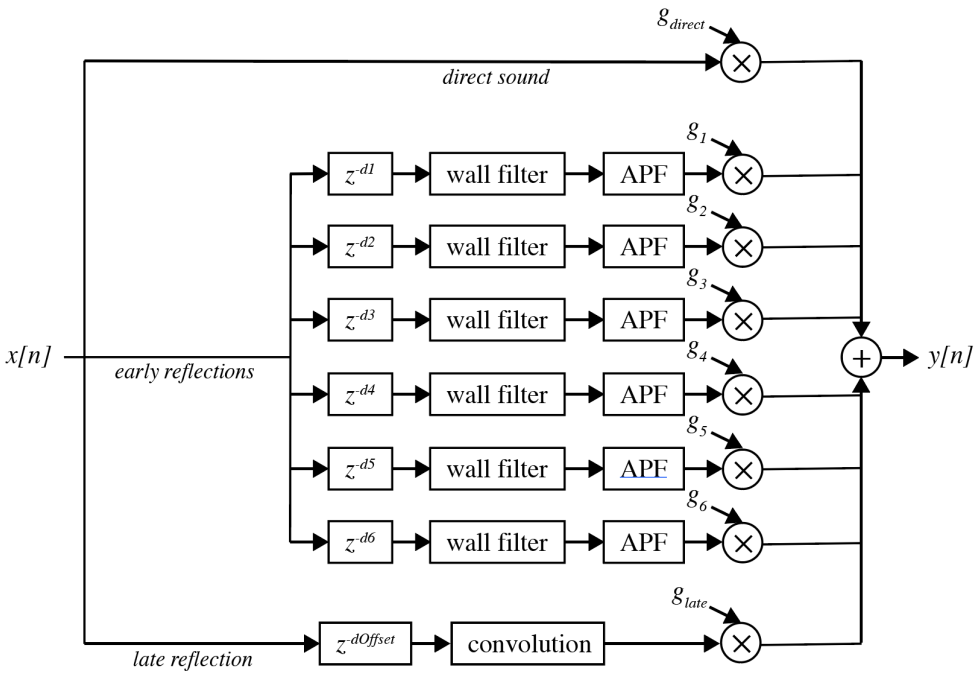


Figure 5.1: Block diagram of the hybrid algorithm.

5.2 Early Reflections

As the overall structure is now defined, the following section will look into the calculation of each of the six early reflections.

5.2.1 Reflections

To determine the delay lengths for each reflection, the position of the point of reflection on each wall was calculated using the following equation.

$$x = \frac{x_1 \cdot y_2 + x_2 \cdot y_1}{y_1 - y_2} \tag{5.1}$$

where x_1 and y_1 as well as x_2 and y_2 are sets of Cartesian coordinates. Inspired by the MATLAB implementation *Scattering Delay Network* by De Sena [2] the above equation is then used to calculate an "x" and "z" coordinates for each wall in the room.

The distance each reflection will travel is calculated by knowing the position of the sound source, the microphone and the points of reflection. The distance can be separated into two distances: source P_s to point of reflection P_j and point of reflection P_j to microphone P_m. The equation for the distance between two points in 3D space is seen below:

$$\begin{aligned} \left|P_s P_j\right| &= \sqrt{\left(x_j - x_s\right)^2 + \left(y_j - y_s\right)^2 + \left(z_j - z_s\right)^2} \\ \left|P_j P_m\right| &= \sqrt{\left(x_m - x_j\right)^2 + \left(y_m - y_j\right)^2 + \left(z_m - z_j\right)^2} \end{aligned} \tag{5.2}$$

The combined distance each reflection has to travel is given below:

$$d_{reflection} = \left|P_s P_j\right| + \left|P_j P_m\right| \tag{5.3}$$

One can calculate the delay length of each reflection in seconds by dividing the known total distance with the speed of sound:

$$t_{reflection} = \frac{d_{reflection}}{c_{air}} \tag{5.4}$$

where $t_{reflection}$ is the time (in seconds) it takes sound to travel the distance of one reflection, $d_{reflection}$ is the distance (in meters) of one reflection, and c_{air} is the speed of sound in air. To convert the time in seconds to samples, the time is multiplied with the sampling rate of the system:

$$s_{reflection} = t_{reflection} \cdot F_s \tag{5.5}$$

where $s_{reflection}$ is the number of samples of one reflection, $t_{reflection}$ is the time of one reflection, and F_s is the sampling rate of the system.

5.2.2 Attenuation

The three parts of the IR were attenuated individually. Initially, the inverse-distance law was used to determine the amplitude of the direct sound and each of the early reflections, given as [3]:

$$P = \frac{k}{r} \tag{5.6}$$

where P is the sound pressure, k is a constant, and r is the distance from the source.

While the direct sound falls off inversely proportional to the distance, the reverberant level generally varies very little throughout the room [15], why the tail was kept at a constant gain. The gain value was determined through informal listening tests and through time-domain plots (similar to figure 7.11).

5.2.3 Directivity

As the sound source and microphone can be placed around the room, the gain for the left and right channels should be calculated individually, based on the following equation [24].

$$\begin{aligned} g_l(\theta) &= \frac{1 - \sin(\theta)}{\sqrt{2\left(1 + \sin^2(\theta)\right)}} \\ g_r(\theta) &= \frac{1 + \sin(\theta)}{\sqrt{2\left(1 + \sin^2(\theta)\right)}} \end{aligned} \tag{5.7}$$

where g_l and g_r are amplitude scalars for left and right stereo channels, respectively, and θ is the position's azimuth relative to the y-axis.

5.2.4 Wall absorption

The implementation of wall absorption filters is described in the following section. As described in section 4, several materials were selected. Filter coefficients were estimated using the `yulewalk`[2] function in MATLAB based on [6] from the absorption coefficients given in table 4.1. As the `yulewalk` function expects a value for 0 Hz and Nyquist, these were set to match their closest neighboring measurement.

To simulate the material absorption at the walls, floor, and ceiling, IIR filters (Transposed Direct Form II) were created using the DSP-module in JUCE (block diagram in figure 5.2. The filters are created by instantiating the `Absorption` class for each wall.

[2] https://se.mathworks.com/help/signal/ref/yulewalk.html

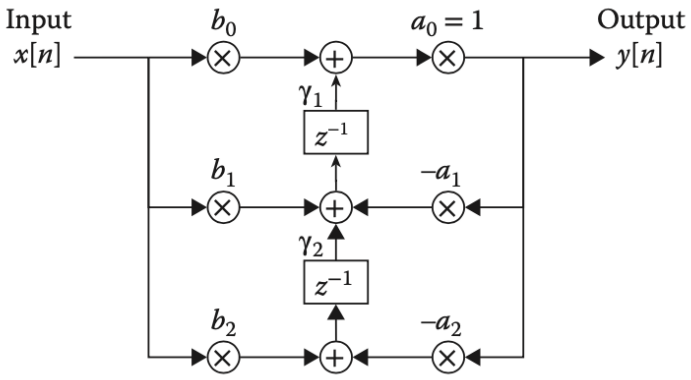


Figure 5.2: Transposed Direct Form II IIR filter. Figure from [35].

The block diagram above yields the following update equation [35].

$$\begin{aligned} y[n] &= b_0 \cdot x[n] + \gamma_1 \\ \gamma_1 &= b_1 \cdot x[n] + (-a_1) \cdot y[n] + \gamma_2 \\ \gamma_2 &= b_2 \cdot x[n] + (-a_2) \cdot y[n] \end{aligned} \tag{5.8}$$

where $y[n]$ is the output, n is the current sample and 1 and 2 are two samples.

5.2.5 Diffusion

As described in section 4.1, all-pass filters were used to add diffuseness to the early reflections. An `Allpass` class was created using all-pass filters from the DSP-module in JUCE in series allowing for a higher filter order. In this case, a filter order of 26 was selected. The all-pass filters from the DSP module are also IIR filters in Transposed Direct Form II. The effect of the all-pass filter is seen below in figure 5.3.

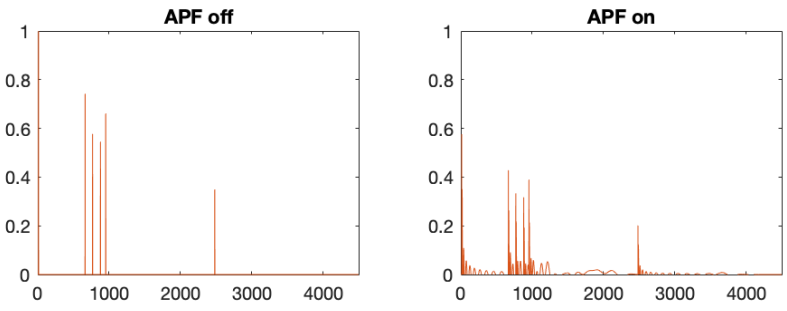


Figure 5.3: Diffusive effect of the all-pass filters.

5.3 Convolution

The tail/late reverberation of the algorithm is applied to the input using convolution. As real-time, or near real-time, convolution is computationally intensive, a combination of uniform and non-uniform time distributed FFT was used for low latency convolution [11]. The implementation was realized in JUCE based on Graham Barab's implementation[3]. The windowed tail used throughout the development of the VST is found in appendix IX.

5.3.1 Measuring RIRs

To have a baseline for comparison as well as the necessary RIRs for experiments with the hybrid reverberation algorithm (see section 6.2 and 6.3), RIRs were measured. These RIRs were, moreover, utilized in the implementation of the hybrid real-time algorithm. Based on the comparison of measurement methods described in section 5.3.1, the sine-sweep method was considered the best method for this condition, as the room was not occupied and the method required minimal calibration, while maintaining an excellent signal-to-noise ratio. The following section will describe the theoretical aspects of the sine-sweep method followed by the practical method used for the measurements used in this implementation.

Sine-sweep method

Firstly a logarithmic sine-sweep (eq. 5.9) is used to excite the system and the response is recorded [4].

$$x(t) = \sin\left[\frac{\omega_1 \cdot T}{\ln\left(\frac{\omega_2}{\omega_1}\right)} \cdot \left(e^{\frac{t}{T}\cdot\ln\left(\frac{\omega_2}{\omega_1}\right)} - 1\right)\right] \tag{5.9}$$

where ω_1 is the initial radian frequency of the sweep, ω_2 is the final radian frequency of the sweep, and T is the duration of the sweep [33].

The response is deconvolved using an inverse filter $f(t)$ to transform the sweep into a delayed Dirac delta function [33]. The deconvolution is then realized by convolving the recorded response $y(t)$ with the inverse filter $f(t)$, as seen in eq. 5.10 [33].

$$h(t) = y(t) \otimes f(t) \tag{5.10}$$

The inverse filter is generated in following way: the logarithmic sine-sweep is reversed along the time-axis and delayed to remain in the positive part of the x-axis. Convolution between the initial sweep and this inverse filter will introduce a

[3] https://github.com/grahman/RTConvolve

squared magnitude. The magnitude spectrum of the resulting signal is divided by the square of the magnitude of the initial logarithmic sine-sweep. The beginning and end of the excitation signal are exponentially faded in and out [33].

Equipment

The measurements were conducted using the following equipment: RØDE NT2-A microphone (omnidirectional polar setting), 3x Dynaudio BM5 mkII speakers (see subsection 5.3.1), Steinberg UR44C audio interface, Macbook Pro, Ableton Live 11 Suite[4], IR Measurement Tool[5], Sound Meter (Android app) by Abc Apps[6], microphone stand, table trolley, marking tape, measuring tape.

As given by ISO 3382:1:2009(E) standard on the measurement of room acoustical parameters, the microphone used for recording IRs should be omnidirectional and the speaker should be *as close to* omnidirectional as possible [12]. As the microphone used for this measurement, RØDE NT2-A, has an omnidirectional polar pattern this fulfilled the standard. It proved more challenging to find an omnidirectional speaker. The most common implementation of a speaker producing an omnidirectional field is a dodecahedron speaker [21]. As no dodecahedron speaker was accessible, a "less mono-directional" speaker was built using three Dynaudio BM5 mkII speakers, positioned in a triangular shape with 120 degrees between each speaker. The speaker layout is found in appendix B.

The sound pressure level of each speaker was measured and calibrated to a sound pressure of -80 dB @1 meter distance, 1000Hz sine wave (+- max. 0.5 dB).

Procedure

1. To have reference points for measuring IRs at different positions, the floor of the room was measured and marked with marking tape for each meter in each dimension.

2. Computer and audio interface was placed outside the room to avoid any unnecessary effect on the room acoustics (and to protect the ears of the researcher).

3. Several test measurements were conducted to confirm that everything was working correctly.

[4] www.ableton.com/en/live/
[5] www.ableton.com/en/packs/convolution-reverb/
[6] shorturl.at/pwDY6

4. The speaker and microphone were placed at different positions in the room and an IR was measured using a 60-second logarithmic sine-sweep.

A picture from the measurement is seen below in figure 5.4. For more pictures from the measurement setup, see appendix B. All measured RIRs are found in appendix X.

Figure 5.4: IR measurement setup.

5.4 Visualization Of Room Dimensions

The visualization of room dimensions, described in section 4.2, and visualized in figure 4.6 was implemented by dynamically drawing 12 lines on the GUI, based on the room dimension sliders. Conceptually, the building block needed to draw the cube was the following: `posX` and `posY` defining the position of the illustration in the GUI and `offsetWidth`, `offsetHeight` and `offsetDepth`, being the room dimensions from the sliders on the GUI. The implementation can be divided into three separate parts: front rectangle, back rectangle, and connecting lines and can be seen in pseudo-code in appendix A.

Chapter 6

Evaluation

The following sections will describe the different evaluations conducted in this project. The evaluations amounted to three individual perceptual listening experiments, a technical signal comparison, a real-time usability evaluation as well as a heuristic interview with the collaborators.

6.1 Perceptual Significance Of Isolated Early Reflections

When designing algorithms for realistic rendering of acoustics, a trade-off is often found between realism and computational load. To shed light on the eventual importance of different orders of early reflections, a pair of subjective listening experiments were conducted. The interface for the evaluation is seen in figure 6.1.

Design This study used a within-subjects design. There were three independent variables: Orders of reflections (with four levels: 3 orders, 2 orders, 1 order, and anechoic), room size (with two levels: small room and large room), and type of stimuli (with two levels: soprano vocal and drum loop). The dependent variable was 'perceived similarity' to a reference sound, where 100 indicated identical stimuli and 0 indicated bad similarity.

Participants There were 24 participants aged between 22 and 49 years (Mdn 26, SD 5.7). Participants had between 0 and 24 years of musical experience, such as playing an instrument, producing music, etc. (Mdn 3, SD 7.7). 45.8% of the participants stated that they had professional experience with audio (working with audio, professionally producing music, studying a sound-related education). Convenience sampling was used to select participants for the evaluation.

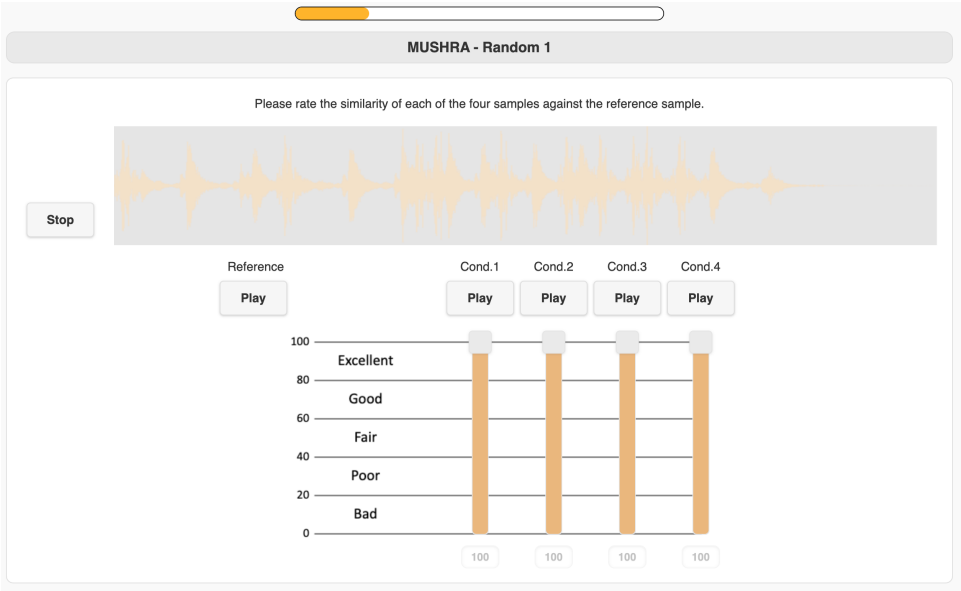


Figure 6.1: Interface used for subjective listening evaluations.

Apparatus & Stimuli The evaluation was conducted using the *MUltiple Stimuli with Hidden Reference and Anchor* (MUSHRA) paradigm, as standardized in [32]. The experiments were facilitated using webMUSHRA running on a local PHP server [29]. The evaluation was conducted on a Macbook Pro. Comments from the participants were noted down to provide qualitative data for a nuanced discussion of the results. Participants were listening to stimuli using AIAIAI TMA-2 headphones.

The stimuli used for this experiment were generated through the following pipeline:

1. IRs for two acoustical spaces[1] were generated using the Image Source method, based on Habet's MATLAB implementation[2] for first-order reflections only, two orders of reflections, and three orders of reflections.

2. The silence from the direct distance (speaker to mic distance) was removed from the synthesized ERs.

3. The silence after the last reflection of the synthesized ER was removed.

[1]Room 1: width: 8.8 m, depth: 8.8 m, and height: 3.4 m. Room 2: width: 44.0 m, depth: 44.0 m, and height: 17.0 m.

[2]https://github.com/ehabets/RIR-Generator

4. Two anechoic stimuli, a soprano vocal[3] and a drum loop[4], were convolved with IRs from both acoustical spaces. The anechoic version of each stimulus was, moreover, included in the evaluation yielding 14 different stimuli.

5. All 14 stimuli were normalized to -25 dBFS RMS. The stimuli from the evaluation is found in appendix I.

Procedure The experimenter invited one participant at a time to attend the experiment. Firstly, each participant was asked to read a short introduction concerning the following experiment and regarding informed consent. Each participant was instructed that the experiment consisted of eight pages where the participant should rate to which degree an audio sample was similar to an audio reference on a scale from 0 to 100 (bad similarity to excellent similarity) as seen in figure 6.1. Each page had four audio samples and an audio reference. The participants were, furthermore, instructed that each page essentially consisted of five instances of the same sounds, but played in different acoustical spaces. The order of pages and the order of audio samples on each page was randomized among participants. At the end of the evaluation, the participant was asked to share their age, years of musical experience (e.g. playing an instrument, working with audio, producing music, etc.), and if the participant works, or has worked, professionally with audio or studies an audio-related education.

[3]Mozart's "Donna Elvira" Aria (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://odeon.dk/downloads/odeon-zip-archives/

[4]SO_PD_90_drum_loop_dorset (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://splice.com/sounds/splice-originals/so_pocket_drums_corey_fonville

6.2 Perceptual Significance Of Early Reflections In Complete Impulse Responses

Design This study used a within-subjects design. There were two independent variables: Orders of reflections (with four levels: 3 orders, 2 orders, 1 order, and tail only), and type of stimuli (with two levels: soprano vocal and drum loop). Different room sizes were neglected in this evaluation, as the synthesized reflections should match the size of the measured RIRs. The dependent variable was 'perceived similarity' to a reference sound, where 100 indicated identical stimuli and 0 indicated bad similarity.

Participants There were 16 participants aged between 22 and 52 years (Mdn 26, SD 8.4). Participants had between 0 and 23 years of musical experience, such as playing an instrument, producing music, etc. (Mdn 0, SD 7.2). 18.8% of the participants stated that they had professional experience with audio (working with audio, professionally producing music, studying a sound-related education). Convenience sampling was used to select participants for the evaluation.

Apparatus & Stimuli Similarly to the ER only evaluation in section 6.1,this evaluation was conducted using the MUSHRA paradigm, as standardized in [32]. The experiments were facilitated using webMUSHRA running on a local PHP server [29]. The evaluation was conducted on a Macbook Pro. Comments from the participants were noted down to provide qualitative data for a nuanced discussion of the results. Participants were listening to stimuli using AIAIAI TMA-2 headphones.

The stimuli used for this experiment were generated through the following pipeline:

1. The silence from the direct distance (speaker to mic distance) was removed from the synthesized ERs.
2. The silence after the last reflection of the synthesized ER was removed.
3. A Hann window of 2048 samples was created to cross-fade between synthesized ERs and the recorded RIR tail.
4. The window was split in two and applied to ERs and RIR.
5. To isolate the "tail" from the RIR, three orders of ERs were removed from the RIR by calculating the time of the last 3rd-order reflection using the Image Source method (as described in section 2.2.2) and removing the number of samples from the RIR.

6. To respect the original envelope from measured RIR in the hybrid IR, an envelope follower was used on the RIR and multiplied to the hybrid IR [23].
7. The generated IRs were convolved with a soprano vocal[5] and a drum loop[6].
8. All 8 stimuli were normalized to -25 dBFS RMS. The stimuli from the evaluation is found in appendix II.

Procedure The experimenter invited one participant at a time to attend the experiment. Firstly, each participant was asked to read a short introduction concerning the following experiment and regarding informed consent. Each participant was instructed that the experiment consisted of four pages, as opposed to eight pages in the ER-only evaluation in section 6.1. Each participant was instructed to rate to which degree an audio sample was similar to an audio reference on a scale from 0 to 100 (bad similarity to excellent similarity), as seen in figure 6.1. Each page had four audio samples and an audio reference. The participants were, furthermore, instructed that each page essentially consisted of five instances of the same sounds, but played in different acoustical spaces. The order of pages and the order of audio samples on each page was randomized among participants. At the end of the evaluation, the participant was asked to share their age, years of musical experience (e.g. playing an instrument, working with audio, producing music, etc.), and if the participant works, or has worked, professionally with audio or studies an audio-related education.

[5]Mozart's "Donna Elvira" Aria (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://odeon.dk/downloads/odeon-zip-archives/

[6]SO_PD_90_drum_loop_dorset (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://splice.com/sounds/splice-originals/so_pocket_drums_corey_fonville

6.3 Perceived Quality Of Different Reverberation Algorithms

Design This study used a within-subjects design. There were two independent variables: Reverberation algorithm (with three levels: RIR/convolution, SDN, the hybrid algorithm proposed in this project), and type of stimuli (with two levels: soprano vocal and drum loop). The dependent variable was 'perceived similarity' to a reference sound, being the RIR/convolution reverberation, where 100 indicated identical stimuli and 0 indicated bad similarity.

Participants There were 17 participants aged between 21 and 35 years (Mdn 26, SD 3.7). Participants had between 0 and 26 years of musical experience, such as playing an instrument, producing music, etc. (Mdn 8, SD 8.1). 64.7% of the participants stated that they had professional experience with audio (working with audio, professionally producing music, studying a sound-related education). Convenience sampling was used to select participants for the evaluation.

Apparatus & Stimuli The evaluation was conducted using the MUSHRA paradigm, as standardized in [32]. The experiments were facilitated using webMUSHRA running on a local PHP server [29]. The evaluation was conducted on a Macbook Pro. Comments from the participants were noted down to provide qualitative data for a nuanced discussion of the results. Participants were listening to stimuli using AIAIAI TMA-2 headphones.

The stimuli used for this experiment were generated through the following pipeline:

1. An RIR was recorded as described in section 5.3.1, and convolved with a soprano vocal [7] and a drum loop[8].

2. Hybrid stimuli were generated by matching the room size[9] and wall materials[10] as close as possible. 38-order all-pass filters were applied. The tail was generated based on the same RIR[11] as used in the RIR condition. The RIR was processed MATLAB to remove ERs and to apply a window to the beginning of the tail, as described in figure 4.1.

[7] Mozart's "Donna Elvira" Aria (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://odeon.dk/downloads/odeon-zip-archives/

[8] SO_PD_90_drum_loop_dorset (first 300000 samples 6.8 seconds @ fs = 44.1 kHz) from https://splice.com/sounds/splice-originals/so_pocket_drums_corey_fonville

[9] Room size: 8.8 m, 8.8 m, 3.4 m

[10] Wall materials (estimated): Front/back: acoustical tile 3/4-in thick, right/left: glass, large panes, heavy glass, ceiling/floor: plywood panel, 3/4-in thick

[11] Location: source: 3,6, microphone: 7,8

3. The SDN stimuli were generated using Enzo De Sena's MATLAB implementation [2]. The room size was matched to the RIR. The implementation did not allow for the selection of wall materials but a wall attenuation variable was adjusted until the RT60 (calculated using[12] based on [31]) of the SDN stimuli matched the RT60 of the recorded RIR. The stimuli from the evaluation is found in appendix III.

Procedure The experimenter invited one participant at a time to attend the experiment. Firstly, each participant was asked to read a short introduction concerning the following experiment and regarding informed consent. Each participant was instructed that the experiment consisted of four pages, where the participant should rate to which degree an audio sample was similar to an audio reference on a scale from 0 to 100 (bad similarity to excellent similarity) as seen in figure 6.1. Each page had four audio samples and an audio reference. The participants were, furthermore, instructed that each page essentially consisted of four instances of the same sounds, but played in different acoustical spaces. The order of pages and the order of audio samples on each page was randomized among participants. At the end of the evaluation, the participant was asked to share their age, years of musical experience (e.g. playing an instrument, working with audio, producing music, etc.), and if the participant works, or has worked, professionally with audio or studies an audio-related education.

[12] https://se.mathworks.com/matlabcentral/fileexchange/1212-t60-m

6.4 Signal Comparison

Design The signal comparison evaluation intended to evaluate temporal as well as spectral differences between different reverberation algorithms.

Stimuli The stimuli to be analyzed were impulse responses from three different algorithms:

1. An RIR was recorded as described in section 5.3.1.
2. A hybrid IR was generated by matching the room size[13] and wall materials[14] as close as possible. 26-order all-pass filters were applied. The tail was generated based on the same RIR[15] as used in the RIR condition. The RIR was processed in MATLAB to remove ERs and to apply a window to the beginning of the tail, as described in figure 4.1.
3. The SDN IR was generated using Enzo De Sena's MATLAB implementation [2]. The room size was matched to the RIR. The implementation did not allow for the selection of wall materials but a wall attenuation variable was adjusted until the RT60 of the SDN stimuli matched the RT60 of the recorded RIR.

Apparatus The signal analysis was executed using MATLAB. Time-domain plots were generated using the `plot()` function, while spectral plots were generated using the `spectrogram()` function for a Short-Time Fourier Transform (STFT). For the spectral plots, a Hann window of size 512 was used. All IRs were normalized to -25 dBFS before generating the plots.

[13] Room size: 8.8 m, 8.8 m, 3.4 m

[14] Wall materials (estimated): Front/back: acoustical tile 3/4-in thick, right/left: glass, large panes, heavy glass, ceiling/floor: plywood panel, 3/4-in thick

[15] Location: source: 3,6, microphone: 7,8

6.5 Usability Of Real-Time VST

Design This study intended to evaluate the usability aspects of the real-time VST developed in this project. Through a series of instructions, problems encountered by the participants, pathways participants took to solve a given task, and additional comments and recommendations from the participants, were observed and noted.

Participants There were 7 participants aged between 21 and 32 years (Mdn 27, SD 3.0). Participants had between 0 and 26 years of musical experience, such as playing an instrument, producing music, etc. (Mdn 10, SD 8.7). 42.9% of the participants stated that they had professional experience with audio (working with audio, professionally producing music, studying a sound-related education). The participants were selected by inviting the first 7 participants of the evaluation in section 7.3 to participate in this evaluation upon completion of the listening evaluation.

Apparatus The evaluation was conducted on a Macbook Pro. Ableton Live 11 was used to host the plugin, built as a VST3. Participants were listening to the sound output using AIAIAI TMA-2 headphones. Comments from the participants were noted down to provide qualitative data for a nuanced discussion of the results.

Procedure The experimenter invited one participant at a time to attend the experiment. Participants were informed that the audio plugin they were going to try was a plugin creating artificial reverberation. Each participant was instructed that they would receive several different instructions. Participants were instructed to complete the task as well as possible and to "think-out-loud" while performing the tasks. The tasks are listed below:

1. Please try to load a room tail.
2. The tail you have just selected comes from a room with the following dimensions: width: 8.8 m, depth: 8.8 m, and height: 3.4 m. Please try to adjust the plugin to match the room size.
3. Please try to adjust the location of the sound source and microphone.
4. Please try turning off and on the tail, and then the direct sound.
5. Please try turning off and on the absorption and all-pass filters.
6. Please try changing the different wall materials.
7. What did you find challenging and what did you find straightforward?
8. Do you have any other feedback?

6.6 Heuristic Interview

Design An interview with one of the collaborators, the CTO from Audio Modeling, Emanuele Parravicini was conducted to evaluate the overall performance of the plugin as well as to discuss design, implementation, and further work. Feedback was noted and the session was screen-recorded. The documented comments and quotes were confirmed by Parravicini.

Apparatus The interview was conducted online through Zoom. The plugin was hosted in Ableton Live 11 on the interviewer's Macbook Pro.

Procedure The interviewee was, firstly, updated on the current state of the project and the plugin. The different functionalities, as well as back-end aspects, were described. The interviewee was invited to express any positive as well as negative feedback. The different functions of the plugin were showcased through Zoom, whereafter the auditory output and usability aspects were discussed. The interviewee was invited to give instructions for the interviewer to perform in the plugin. Ultimately, after the walkthrough, an overall discussion on the project was conducted.

Chapter 7

Findings

This chapter covers all findings from the different experiments described conducted throughout this project, as described in the previous section. The findings from each experiment are firstly, documented and secondly, discussed.

7.1 Perceptual Significance Of Isolated Early Reflections

The following section presents the findings from the evaluation of perceptual significance of isolated early reflections as described in section 6.1.

7.1.1 Results

Figure 7.1 shows boxplots of each condition (anechoic to 3 orders) from the evaluation along the x-axis. The y-axis describes the participants' rating of similarity between the given number of reflections and the audio reference (three orders of reflections). Wilcoxon signed-rank tests were used to check for statistical significance between the conditions. The similarity to the reference was significantly higher between all neighboring conditions. Between the three-order reference (Mdn = 96.5) and the two-order condition (Mdn = 84.0); T = 2660.5, $p < 0.001$, r = 0.51, considered above Cohen's threshold for a large effect size [5]. Between the two-order condition (Mdn = 84.0) and the first-order condition (Mdn = 52.0); T = 338.5, $p < 0.001$, r = 0.82, considered a large effect size. Between the first-order condition (Mdn = 52.0) and the anechoic condition (Mdn = 30.0); T = 811.0, $p < 0.001$, r = 0.78, considered a large effect size.

Small room isolated Stimuli was generated from both a small room and a large room, to shed light on the effect of room size. Figure 7.2 contains the data split into the two room sizes. For the small room (figure 7.2a), between the three-order reference (Mdn = 100) and the two-order condition (Mdn = 84.5); T = 646, $p < 0.001$,

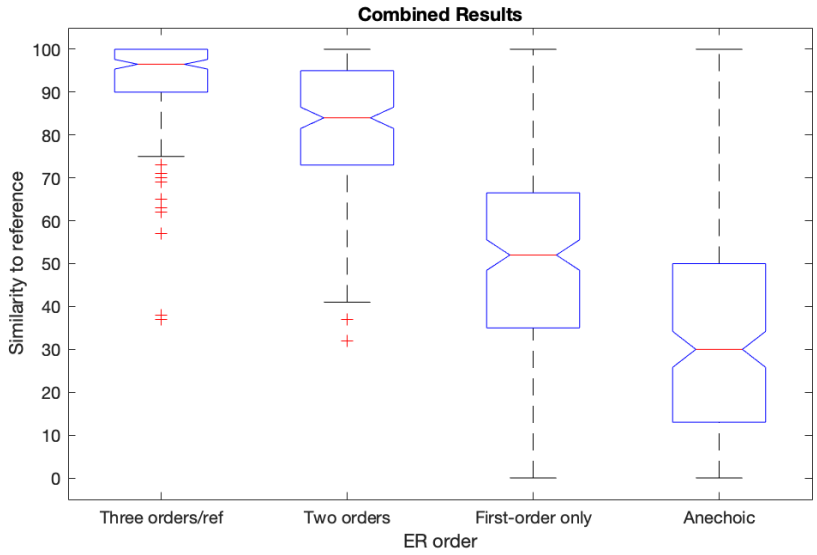
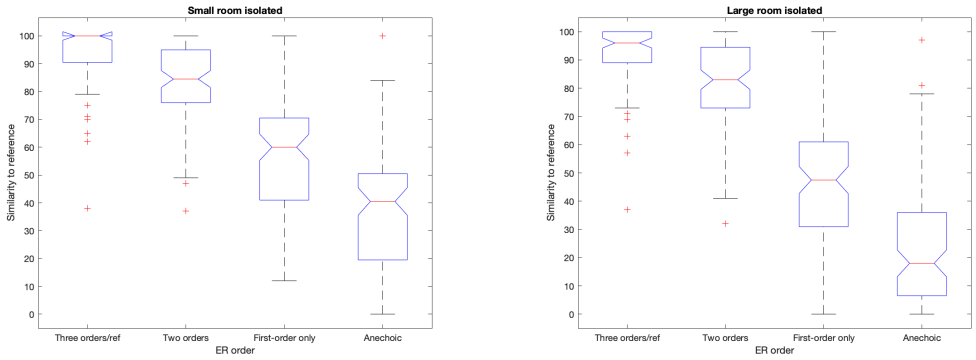


Figure 7.1: Boxplots of results from evaluation of perceptual significance of early reflections

r = 0.51, being just above Cohen's threshold for a large effect size[5]. Between the two-order condition (Mdn = 84.5) and the first-order condition (Mdn = 60); T = 115.5, $p < 0.001$, r = 0.80, considered a large effect size. Between the first-order condition (Mdn = 60) and the anechoic condition (Mdn = 40.5); T = 242, $p < 0.001$, r = 0.75, considered a large effect size.

Large room isolated The results from the large room condition is seen in figure 7.2b. For this condition, the similarity to the reference was also significantly higher between all neighboring conditions. Between the reference (Mdn = 96.0) and the two-order condition (Mdn = 83.0); T = 165, $p < 0.001$, r = 0.80, being just above Cohen's threshold for a large effect size [5]. Between the two-order condition (Mdn = 83.0) and the first-order condition (Mdn = 47.5); T = 65.5, $p < 0.001$, r = 0.83, considered a large effect size. Between the first-order condition (Mdn = 47.5) and the anechoic condition (Mdn = 18.0); T = 707.0, $p < 0.001$, r = 0.50, considered a large effect size.

Small/large room compared Considering the difference between the two conditions, the median values between the small room and the large room deviated with 4.0 for the three order stimuli: small room (Mdn = 100.0) and large room (Mdn = 96.0). Furthermore, the median values for the two-orders stimuli deviated by 1.5: small room (Mdn = 84.5) and large room (Mdn = 83.0). For the first-order only stimuli the median deviated by 12.5: small room (Mdn = 60.0) and large room (Mdn = 47.5). Finally, for the three order stimuli, the median deviated by 22.5 for



(a) Small room isolated.

(b) Large room isolated.

Figure 7.2: Results split based on room size.

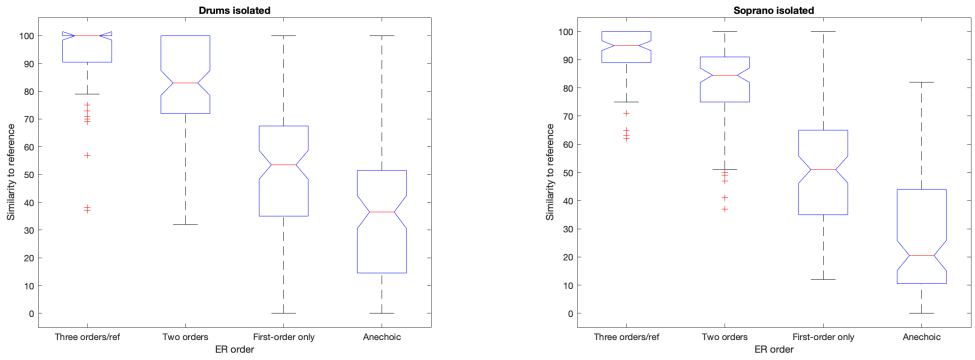
the anechoic stimuli: small room (Mdn = 40.5) and large room (Mdn = 18.0).

Drums isolated Figure 7.3 contains the same data as in figure 7.1, but split into drum stimuli and soprano stimuli. For the drum stimuli condition, the similarity to the reference was, as in the combined condition, significantly higher between all neighboring conditions. Between the reference (Mdn = 100.0) and the two-order condition (Mdn = 83.0); T = 629.5, $p < 0.001$, r = 0.70, considered a large effect size [5]. Between the two-order condition (Mdn = 83.0) and the first-order condition (Mdn = 53.5); T = 132.0, $p < 0.001$, r = 1.12, considered a large effect size. Between the first-order condition (Mdn = 53.5) and the anechoic condition (Mdn = 36.5); T = 449.5, $p < 0.001$, r = 0.94, considered a large effect size.

Soprano isolated For the soprano stimuli condition, the similarity to the reference was also significantly higher between all neighboring conditions. Between the three-order reference (Mdn = 95.0) and the two-order condition (Mdn = 84.5); T = 737.5, $p < 0.001$, r = 0.72, considered a large effect size [5]. Between the two-order condition (Mdn = 84.5) and the first-order condition (Mdn = 51.0); T = 43.0, $p < 0.001$, r = 1.18, considered a large effect size. Between the first-order condition (Mdn = 51.0) and the anechoic condition (Mdn = 20.5); T = 4.0, $p < 0.001$, r = 1.22, considered a medium effect size.

Drum/soprano compared Considering the difference between the two conditions, the median values between the drum condition and the soprano condition deviated with 5.0 for the three-order reference stimuli: drum condition (Mdn = 100.0) and soprano condition (Mdn = 95.0). For the two-order stimuli the median deviated by 1.5: drum condition (Mdn = 83.0) and soprano condition (Mdn = 84.5). Furthermore, the median values for the first-order stimuli deviated by 2.5: drum condition

(*Mdn* = 53.5) and soprano condition (*Mdn* = 51.0). Finally, for the anechoic stimuli, the median deviated by 16.0: drum condition (*Mdn* = 36.5) and soprano condition (*Mdn* = 20.5).



(a) Drums isolated.

(b) Soprano isolated.

Figure 7.3: Results split based on type of stimuli.

Professionals/non-professionals compared Splitting the results into participants with and without professional experience yielded the following results. The median values deviated with 6.0 for the three-order reference stimuli: participants without professional experience (*Mdn* = 100.0) and with experience (*Mdn* = 94.0). Furthermore, the median values for the two-orders stimuli deviated by 1.5: participants without professional experience (*Mdn* = 84.5) and with experience (*Mdn* = 83.0). For the first-order only stimuli the median deviated by 6.5: participants without professional experience (*Mdn* = 50.0) and with experience (*Mdn* = 56.5). Finally, for the anechoic stimuli, the median deviated by 4.5: participants without professional experience (*Mdn* = 26.5) and with experience (*Mdn* = 31.0). The results are visualized by boxplots in figure 7.4.

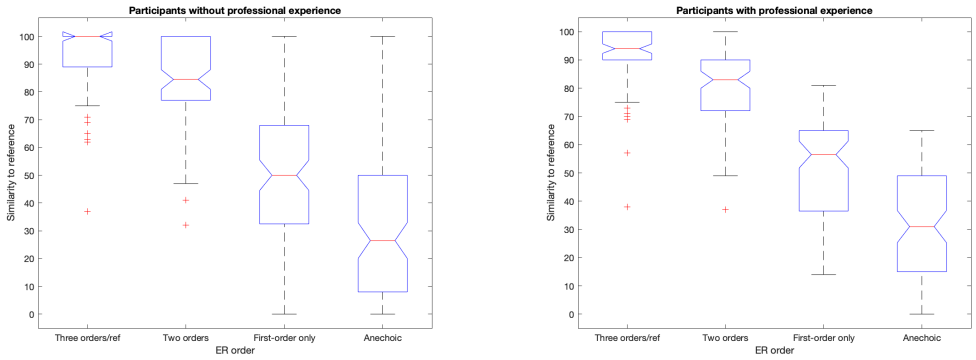
Raw results, stimuli, and MATLAB scripts for analysis are found in appendix I.

7.1.2 Comments

"*They all sounded the same to me.*"
"*You have to remind yourself to rate the similarity and not which one you prefer.*"

7.1.3 Results Discussion

The combined condition revealed that significant differences were found between all neighboring conditions. This indicates that there will be a perceptual trade-off stepping down from three orders to two orders, to one order, and finally the



(a) Results from participants without professional experience.

(b) Results from participants with professional experience.

Figure 7.4: Results split based on professional experience.

anechoic stimuli. Inspecting the median values between the different stimuli, it is clear that the difference between third-order and second-order stimuli is not very profound being 12.5 (three orders scoring higher than two orders). Between the two-order and three-order stimuli, a difference in median of 32.0 is found, indicating a significant jump in perceptual quality. Finally, between the first-order and the anechoic stimuli, a difference in median of 22.0 was found.

Splitting the results into stimuli from a small room and stimuli from a large room proved that significant differences were also found between all neighboring conditions for both small and large rooms. Comparing the median values of each condition across room size showed similarity in the three-order and two-order conditions, namely a difference in median of 4.0 and 1.5 while bigger differences were found in the first-order and anechoic conditions, namely 12.5 and 22.5. The large difference in the anechoic condition is interesting, as the stimuli presented in both conditions are exactly the same, as the anechoic sample was convolved with any IR. This sheds light on the fact that the MUSHRA scale is very arbitrary and that poor similarity, good similarity, etc. is very subjective. The lower values in the large room condition, indicate that the larger room size is more revealing. This could be an unsurprising, effect of the delay lengths being longer in the larger room condition, why individual reflections are more clearly perceived.

Considering drums stimuli and soprano stimuli individually also yielded significant differences in all neighboring conditions, as found in the combined condition. Splitting the results into drums and soprano resulted in very similar findings, as the median of all conditions differed by less than 5.0 between drums and soprano, except for the anechoic condition, which deviated by 16.0 (soprano scoring lower

than drums). This indicates that the type of stimuli (rhythmic vs. harmonic) did not have any major effect on the perceived similarity to the third-order reference.

As 45.8% of the participants stated that they work or have worked professionally with audio. It was interesting to isolate these two conditions to shed light on eventual differences between professionals and non-professionals. The findings showed no major differences in medians between the two conditions, the largest difference being 6.0. This indicates that perceiving differences in early reflections is not an easy task, and certainly not something most non-professionals nor professionals are used to. This is, moreover, backed up by comments from a subject stating that: *"they all sounded the same to me"*. Arguable the nature of this listening experiment could create confusion among some participants. It might not be clear for all participants what to listen for, as the vast majority of the auditory stream will be the same signal in all conditions. The differences are subtle and if participants are not aware of what to listen for, this could explain the high variance in most conditions. Another explanation for the high variance could be confusion regarding the objective of the task. Participants were instructed to rate the different sounds' similarity to a reference sound, which also seemed broadly understood, although one subject stated *"you have to remind yourself to rate the similarity and not which one you prefer"*.

Across all conditions, it is clear, that there is a perceptual difference between using three orders of reflections against fewer orders. As described in section 2.2.2, the number of delay lines necessary for the reproduction of the given orders of early reflections increases exponentially. Considering this, depending on the computational power accessible, three orders or two orders of reflections seems like a sensible choice as the perceptual difference between the two is marginal while the step down to first-order reflections is more pronounced.

7.2 Perceptual Significance Of Early Reflections In Complete Impulse Responses

The following section presents the findings from the evaluation of perceptual significance of early reflections in complete impulse responses, as described in section 6.2.

7.2.1 Results

Figure 7.5 shows boxplots of each condition (tail only to 3 orders) from the evaluation along the x-axis. The y-axis describes the participants' rating of similarity between the given number of reflections and the audio reference (three orders of reflections). Wilcoxon signed-rank tests were used to check for statistical significance between the conditions. No significant difference was found between the three-order reference stimuli (Mdn = 93) and the two-order stimuli (Mdn = 90) ; T = 500, p = 0.50, r = 0.08. On the contrary, a significant difference was found for the two-order stimuli (Mdn = 90) and the first-order stimuli (Mdn = 80) ; T = 305.5, $p < 0.001$, r = 0.47, considered a medium effect size. Moreover, a significant difference was found between the first-order stimuli (Mdn = 80) and the anechoic stimuli (Mdn = 64.5) ; T = 240.5, $p < 0.001$, r = 0.57, considered a large effect size.

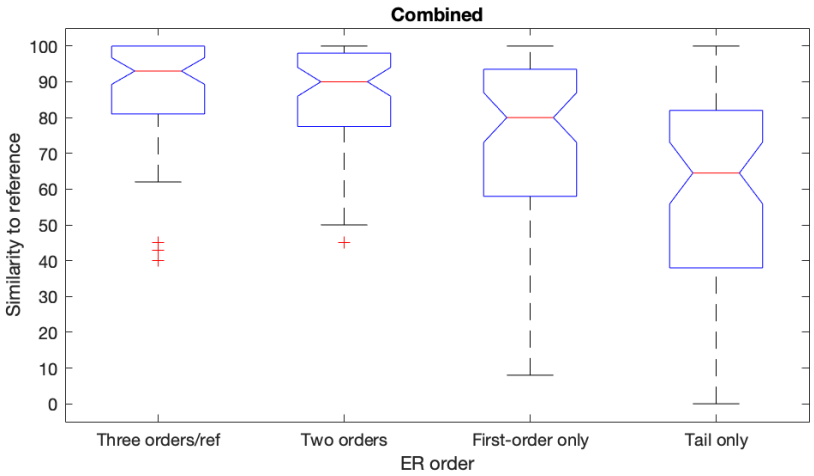


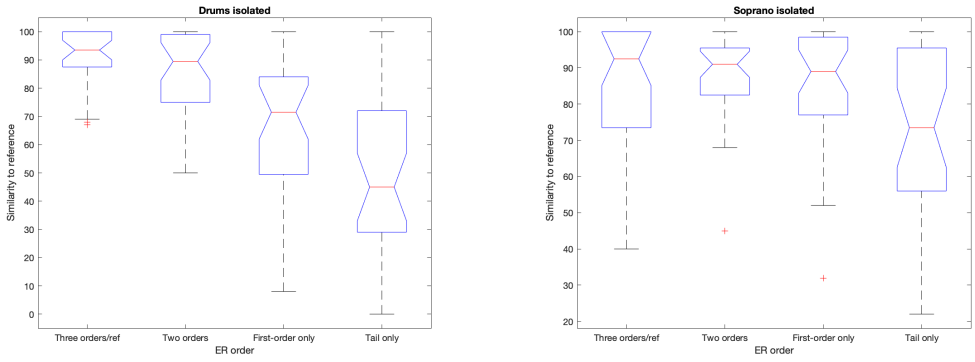
Figure 7.5: Boxplots of results from evaluation of perceptual significance of early reflections

Drums isolated Figure 7.6 contains the same data as in figure 7.5, but split into drum stimuli and soprano stimuli. For the drum stimuli condition, no significant difference was found between the third-order reference (Mdn = 93.5) and the two-order condition (Mdn = 89.5); T = 101.0, p = 0.16, r = 0.25. A significant differ-

ence was found between the two-order condition (Mdn = 89.5) and the first-order condition (Mdn = 71.5); T = 27.0, $p < 0.001$, r = 0.69, considered a large effect size. Moreover, a significant difference was found between the first-order condition (Mdn = 71.5) and the anechoic condition (Mdn = 45.0); T = 41.0, $p < 0.001$, r = 0.70, considered a large effect size.

Soprano isolated For the soprano stimuli condition, no significant difference was found between the third-order reference (Mdn = 92.5) and the two-order condition (Mdn = 91.0); T = 152.5, p = 0.66, r = 0.8. Likewise, no significant difference was found between the two-order condition (Mdn = 91.0) and the first-order condition (Mdn = 89.0); T = 145.5, p = 0.30, r = 0.18. In addition to this, no significant difference was found between the third-order reference (Mdn = 92.5) and the first-order condition (Mdn = 89.0); T = 177.0, p = 0.70, r = 0.07). On the contrary, a significant difference was found between the first-order condition (Mdn = 89.0) and the anechoic condition (Mdn = 73.5); T = 77.5, $p < 0.1$, r = 0.44, considered a medium effect size.

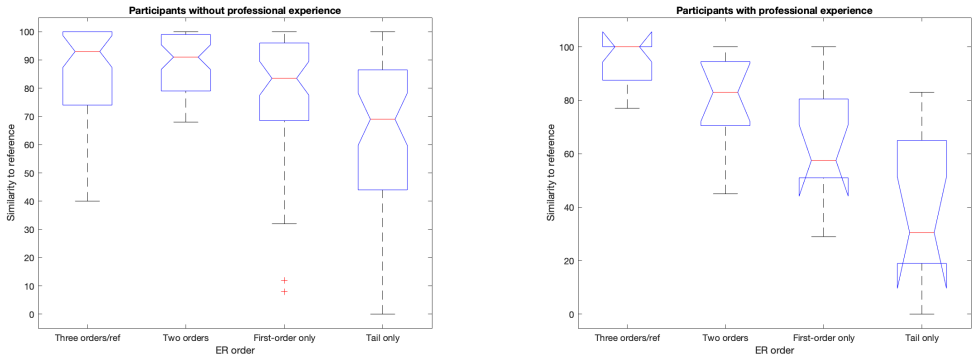
Drum/soprano compared Considering the difference between the two conditions, the median values between the drum condition and the soprano condition deviated with 1.0 for the three-order reference stimuli: drum condition (Mdn = 93.5) and soprano condition (Mdn = 92.5). For the two-order stimuli the median deviated by 1.5: drum condition (Mdn = 89.5) and soprano condition (Mdn = 91.0). Furthermore, the median values for the first-order stimuli deviated by 17.5: drum condition (Mdn = 71.5) and soprano condition (Mdn = 89.0). Finally, for the anechoic stimuli, the median deviated by 28.5: drum condition (Mdn = 45.0) and soprano condition (Mdn = 73.5).



(a) Drums isolated. **(b)** Soprano isolated.

Figure 7.6: Results split based on type of stimuli.

Professionals/non-professionals compared Splitting the results into participants with and without professional experience yielded the following results. The median values deviated with 7.0 for the three-order reference stimuli: participants without professional experience (93.0) and with experience (*Mdn* = 100). Furthermore, the median values for the two-orders stimuli deviated by 8.0: participants without professional experience (*Mdn* = 91.0) and with experience (*Mdn* = 83.0). For the first-order stimuli the median deviated by 26.0: participants without professional experience (*Mdn* = 83.5) and with experience (*Mdn* = 57.5). Finally, for the tail only stimuli, the median deviated by 38.5: participants without professional experience (*Mdn* = 69.0) and with experience (*Mdn* = 30.5). The results are visualized by boxplots in figure 7.7.



(a) Results from participants without professional experience.

(b) Results from participants with professional experience.

Figure 7.7: Results split based on professional experience

Raw results, stimuli, and MATLAB scripts for analysis are found in appendix II.

7.2.2 Comments

"*I couldn't hear any difference. Maybe less reverb on the first one compared to the last one.*"
"*It's the room I'm rating, right?*"
"*One of them sounds far from the reference.*"
"*On the opera one I couldn't hear any difference between them but in this one [drums] I could clearly hear differences.*"
"*One of them has a hi-hat that I can't hear in none of the others.*"
"*With the soprano there was this echo that didn't sound real. It made it difficult.*"
"*When she is singing it is difficult.*"
"*In the soprano one I couldn't hear any difference.*"

7.2.3 Results Discussion

The combined condition revealed no significant difference between three order stimuli and two order stimuli. Compared to the evaluation of early reflections isolated, where a significant difference was found between all neighboring conditions, this could indicate that the inclusion of an identical tail in all conditions, masks some auditory differences between each condition. This indication is, furthermore, backed up by comparing the average similarity rating in the evaluation of the isolated early reflections (section 7.1) with the average similarity rating from this evaluation being 12.8 higher in this evaluation (ER-only = 64.7, ER and tail = 77.5). A significant difference was found between the remaining conditions, indicating that a perceptual difference is still profound between two orders of reflections and first-order reflection, as well as between first-order reflections and no reflections. This indicate that subjects hear a clear difference between sparse amount of reflections, (tail only = 0 unique reflections, first-order = 6 unique reflections, two orders = 24 unique reflections, three orders = 120 unique reflections).

Isolating the drums stimuli yielded similar results as the combined condition, just with bigger effect sizes in all conditions. Looking at the soprano results individually proved no significant difference between three orders and two orders, as well as two orders and first-order. Only a significant difference was found between the first-order and the anechoic stimuli. This indicates that it was more difficult for participants to perceive a difference through harmonic stimuli against rhythmic stimuli. This is, moreover, backed up by comments from subjects such as *"On the opera one I couldn't hear any difference between them but in this one [drums] I could clearly hear differences" as well as "In the soprano one I couldn't hear any difference"* and finally *"When she is singing it is difficult"*.

As 18.8% of the participants stated that they work or have worked professionally with audio, it was interesting to isolate these two conditions to shed light on eventual differences between professionals and non-professionals. While professionals and non-professionals rated the three order reference close to 100, the differences increased as the order decreased (professionals rating lower than non-professionals in the other conditions). The findings across the two conditions deviated by a noticeable amount, but it is worth noting that the professional conditions contained data from only three participants as opposed to 13 in the non-professional condition. Therefore, it is difficult to generalize these results.

It is clear from the combined condition, that there is a perceptual difference between using three orders of reflections against fewer orders. Similar to the findings from the ER-only evaluation, depending on the computational power accessible, three orders or two orders of reflections seems like a sensible choice as the per-

ceptual difference between the two is marginal while the step down to first-order reflections is more pronounced. As this project investigates efficient rendering of room acoustics for real-time physically modeled instruments such as strings or woodwinds, it is worth considering the soprano stimuli as an indication of the perceived quality of the algorithm on harmonic stimuli such as string instruments or woodwinds. As no significant difference was found between the three-order reference stimuli and the first-order stimuli (medians deviating by only 3.5), this seems like a sensible trade-off between computational complexity and perceived similarity to three orders.

7.3 Perceived Quality Of Different Reverberation Algorithms

The following section presents the findings from the evaluation of the perceived quality of different reverberation algorithms, as described in section 6.3.

7.3.1 Results

Figure 7.5 shows boxplots of each condition (RIR, SDN, hybrid) from the evaluation along the x-axis. The y-axis describes the participants' rating of similarity between the samples from each reverberation algorithm and the audio reference (RIR). Wilcoxon signed-rank tests were used to check for statistical significance between the conditions. A significant difference was found between the RIR (reference) (Mdn = 100.0) and the SDN algorithm (Mdn = 51.0); T = 2346.0, $p < 0.001$, r = 0.87, considered a large effect size. Moreover, a significant difference was found between the RIR (reference) (Mdn = 100) and hybrid algorithm (Mdn = 45); T = 2278.0, $p < 0.001$, r = 0.86, considered a large effect size. Finally, a significant difference was also found between the SDN algorithm (Mdn = 51) and the hybrid algorithm (Mdn = 45); T = 1480.5, p = 0.03, r = 0.26, considered a small effect size.

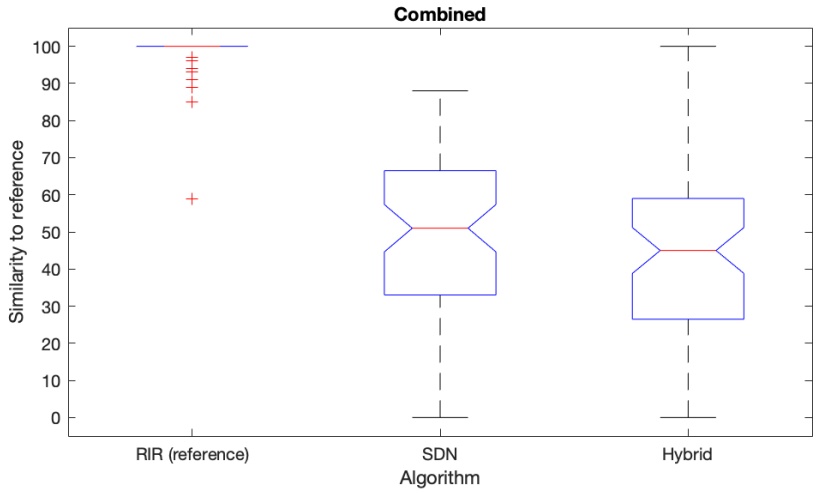


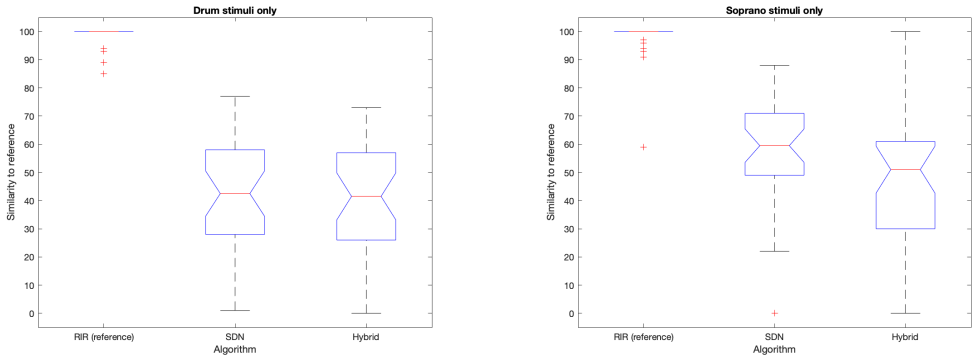
Figure 7.8: Boxplots of results from evaluation of different reverb algorithms.

Drums isolated Figure 7.9 contains the same data as in figure 7.8, but split into drum stimuli and soprano stimuli. For the drum condition, a significant difference was found between the RIR (reference) (Mdn = 100) and the SDN algorithm (Mdn = 42.5); T = 595.0, $p < 0.001$, r = 0.62, considered a large effect size. Moreover, a significant difference was found between the RIR (reference) (Mdn = 100.0) and hybrid algorithm (Mdn = 41.5); T = 595.0, $p < 0.001$, r = 0.62, considered a large

effect size. On the contrary, no significant difference was found between the SDN algorithm (Mdn = 42.5) and the hybrid algorithm (Mdn = 41.5); T = 314.5, p = 0.54, r = 0.07.

Soprano isolated For the soprano condition, a significant difference was found between the RIR (reference) (Mdn = 100) and the SDN algorithm (Mdn = 59.5); T = 595.0, $p < 0.001$, r = 0.62, considered a large effect size. Moreover, a significant difference was found between the RIR (reference) (Mdn = 100.0) and hybrid algorithm (Mdn = 51.0); T = 561.0, $p < 0.001$, r = 0.61, considered a large effect size. Finally, a significant difference was found between the SDN algorithm (Mdn = 59.5) and the hybrid algorithm (Mdn = 51.0); T = 435.5, p = 0.018, r = 0.29, considered a small effect size [5].

Drum/soprano compared Considering the difference between the two conditions, the median values between the drum condition and the soprano condition deviated with 0.0 for the RIR (reference) stimuli: drum condition (Mdn = 100.0) and soprano condition (Mdn = 100.0). For the SDN stimuli the median deviated by 17.0: drum condition (Mdn = 42.5) and soprano condition (Mdn = 59.5). Finally, for the hybrid stimuli, the median deviated by 9.5: drum condition (Mdn = 41.5) and soprano condition (Mdn = 51.0).

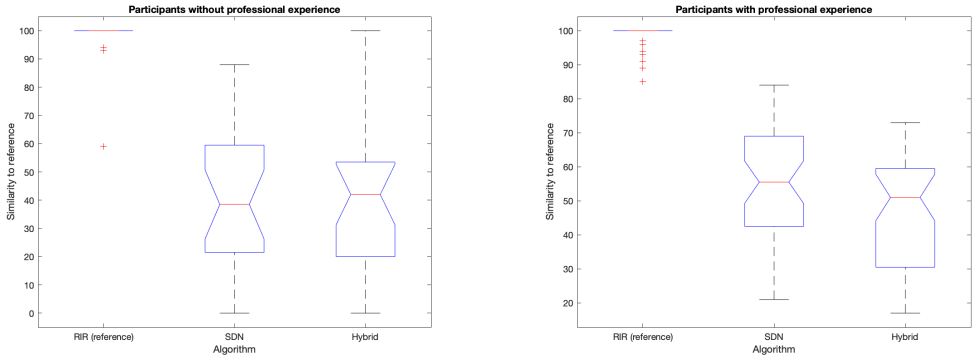


(a) Drum stimuli only. **(b)** Soprano stimuli only.

Figure 7.9: Results split into drum stimuli and soprano stimuli.

Professionals/non-professionals compared Splitting the results into participants with and without professional experience yielded the following results. The median values for the RIR stimuli were identical: participants without professional experience (Mdn = 100.0) and with experience (Mdn = 100). For the SDN stimuli the median deviated by 17.0: participants without professional experience (Mdn = 38.5) and with experience (Mdn = 55.5). Furthermore, the median values deviated

with 9.0 for the hybrid stimuli: participants without professional experience (*Mdn* = 42.0) and with experience (*Mdn* = 51.0). The results are visualized by boxplots in figure 7.10.



(a) Results from participants without professional experience.

(b) Results from participants with professional experience.

Figure 7.10: Results split based on professional experience

Raw results, stimuli, and MATLAB scripts for analysis are found in appendix III.

7.3.2 Comments

"*I felt like one was the reference, drum was difficult. One of them was a bit closer to the reference whereas one was less close.*"
"*I felt like both sounds that were not the reference were very similar. Maybe I guessed a bit.*"
"*Both are clearly way lower than 100 [in similarity to the reference].*"

7.3.3 Results Discussion

The combined results proved a significant difference between all three reverb algorithms indicating a perceptual difference between all reverberation algorithms. It is clear from the results that neither the SDN nor the hybrid algorithm was found very similar to the RIR reference. This is, furthermore, confirmed by comments from the subjects such as "I felt like both sounds that were not the reference were very similar. Maybe I guessed a bit" and "Both are clearly way lower than 100 [in similarity to the reference]".

While the combined results proved a significant difference between all three reverb algorithms, isolating the drums stimuli proved no significant difference between SDN and hybrid stimuli, while being perceived relatively far from the RIR condition (RIR = 100, SDN = 42.5, hybrid = 41.5). Isolating the soprano stimuli yielded

similar results as the combined condition. All similarity ratings were higher in the soprano condition, except for the RIR stimuli being 100 in both conditions, which could confirm the indications from section 6.2 of differences in reverberation being harder to perceive through harmonic stimuli as opposed to rhythmic stimuli.

As 64.7% of the participants stated that they work or have worked professionally with audio, it was interesting to isolate these two conditions to shed light on eventual differences between professionals and non-professionals. Inspecting each condition individually revealed no major difference in the combined case. The largest difference was found in the SDN stimuli (median deviating by 17.0, higher in the professional condition).

Naturally, it is worth noting that this evaluation sheds light on different reverb algorithms' similarity to a recorded RIR, why this does not reflect the quality of the algorithms, but the perceptual similarity to the RIR. The quality of the RIR, as described in section 5.3.1, is a product of the measurement process. As several aspects of the measurements process could be improved (such as using a dodecahedron speaker and taking the average of multiple measurements), the quality of the RIR was possibly not optimal. This could have the effect, that algorithms with objectively higher sound quality (lower signal-to-noise ratio, fewer artifacts, etc.) could be rated as less similar while still maintaining high quality.

7.4 Signal Comparison

The following section covers the findings from the signal comparison evaluation as described in section 6.4. Stimuli and MATLAB scripts for analysis are found in appendix IV.

7.4.1 Results

Time Domain

The time domain plots of the three IRs is seen in figure 7.11.

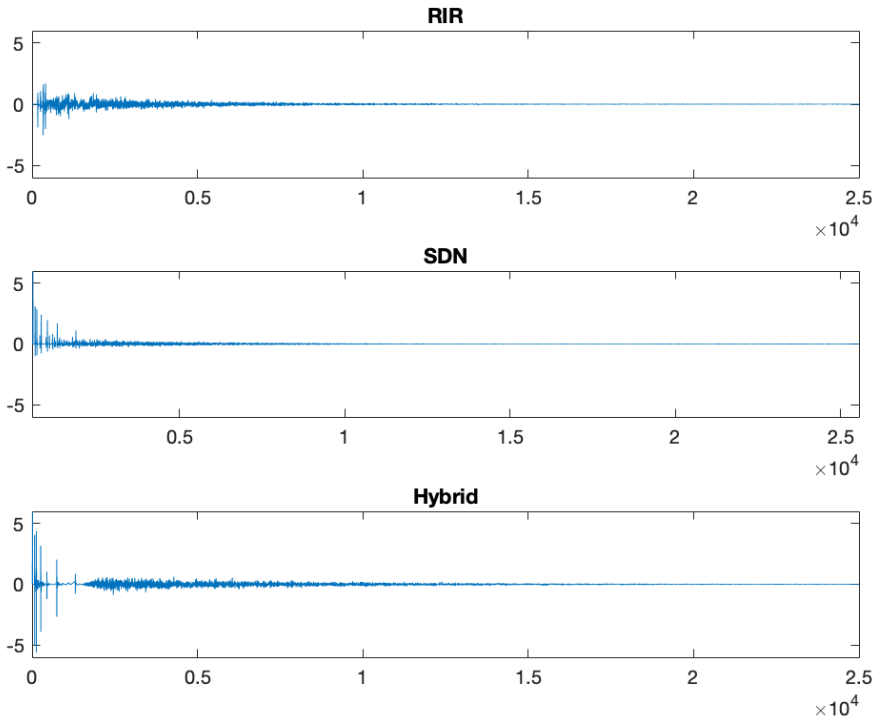


Figure 7.11: Time domain plots of IRs.

Frequency Domain

The spectral plots of the IRs is seen in figure 7.12.

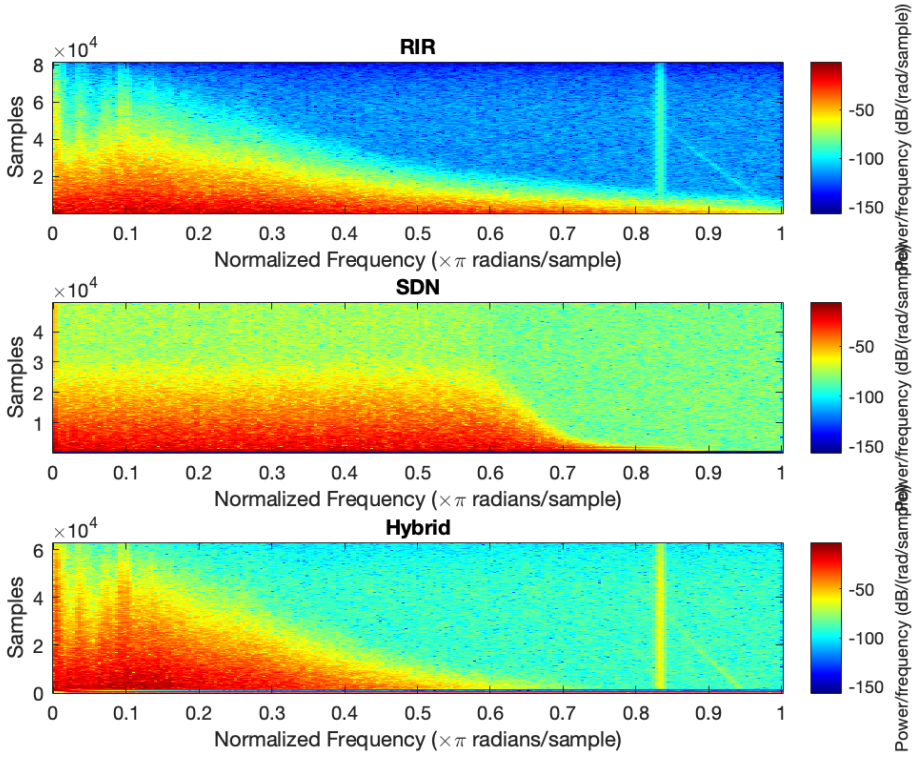


Figure 7.12: Spectrograms of IRs.

7.4.2 Results Discussion

The time-domain plots indicate that all three algorithms share a range of similarities. The early reflections are clearly seen as individual reflections at the beginning of each signal while being slightly less pronounced in the RIR condition. Inspecting the plots, the change from early reflections to late reverberation in the hybrid algorithm is noticeable. This could be due to the truncated nature of the hybrid algorithm, consisting of separate parts connected. It is, moreover, seen in the plots, that the early reflections of the RIR and the SDN are more densely distributed, while the early reflections of the hybrid algorithm are more sparsely spread. This difference is a natural side-effect of the hybrid algorithm using only first-order reflections, as opposed to the other algorithms.

Inspecting the spectrograms, it is clear that the RIR and the hybrid algorithm share many similarities. The notches in the spectral content of low frequencies are seen in both RIR and hybrid conditions, as well as similarities in the spectral decay over time. This is unsurprising, as a large part of the hybrid stimuli originates from the same RIR as seen in the top plot. The SDN condition differs from the other algo-

rithms, as the spectral decay of frequencies is different. In the SDN spectrogram, it is seen that frequencies below 0.6 $\times\pi$ radians/sample (13230 Hz) decay at the same rate, while higher frequencies decay at a different, much faster rate.

In both the RIR and the hybrid spectrograms, an artefact is seen around 0.84 $\times\pi$ radians/sample (18522 Hz). Supposedly, this high-frequency artifact is an effect of some mistake in the recording process. Considering the very low amplitude of the artifact, and as no participants from any of the perceptual experiments noticed any abnormalities, it is considered negligible.

Generally, it is clear from both time-domain and frequency-domain plots, that the RIR and hybrid algorithm share many similarities. While this is, most likely, the product of sharing part of the same IR, this should not be neglected.

7.5 Usability Of Real-Time VST

The following section describes the findings from the usability evaluation of the real-time VST, as described in section 6.5.

7.5.1 Results

Topic	Quote
Loading a tail	"I don't know what a tail is."
Room size	"It was not clear to me if z controlled room depth or room height." "Uh, can you just type a number? Cool." "Relatively easy to adjust room size." "Why is the room z slider not oblique?" "All was intuitive. The only thing would be source mic room sliders. Not clear if y and x is the floor or z is height or depth. Once you move them it makes sense."
Source & mic	"It is a bit unclear that you listen from the microphone position." "When you put them in the same position you have more of the direct signal and less reverb." "The position of source and mic is not clear. Maybe have an x/y matrix with coordinates with both source and mic in the same matrix." "Everything is clear, with labels, it's neat and clear what you are doing. You can hear immediately what you are affecting." "You could have a grid with a dot for mic and source to control position."
Direct & tail	"Especially the buttons made sense from the beginning." "Easy to turn on/off ER, direct sound and tail." "Maybe dry/wet instead of on/off buttons."
Absorp. & APF	"It seems a little louder without absorption." "When APF is off, it is more significant. More clear beat." "It's like it's dampened when on. It's like it's absorbing sound. With APF maybe the sound gets a bit thinner but honestly I struggle to hear a difference." "All was very clear and logical. Simple to turn effects on/off." "Allpass filters sound weird. They have value from artistic point of view but it doesn't sound realistic."
Materials	"Not a very big difference when changing wall materials". "I heard a clear difference between wood and carpet". "It was difficult to hear a difference when changing materials" "It was clear, but having one button to change all wall materials at once would be nice." "It was obvious how to select materials." "Materials are not so different. In real life, there would be more difference"
Visual feedback	"It was not clear that the orange color meant out of bounds, maybe use a more alarming color. But it makes sense when you know it. Maybe color the whole slider or just write 'outside boundary'." "It was intuitive that orange meant out of bounds." "I would like visuals of the mic and source." "It would be nice to have visuals of location of the source and mic. It's hard to imagine." "Might be cool to have a popup when mouse over with information about the function." "It is clear from visuals that you have direct sound, ER and tail, materials were clear, source, mic, room, all clear. But I've also worked a lot with reverb."
Other comments	"What does air temperature affect?" "How big an effect does air temperature have?"

Table 7.1: Selected quotes from real-time evaluation

The raw notes from the evaluation is found in appendix V.

7.5.2 Results Discussion

While most aspects of the usability evaluation of the real-time VST went effortless, a few aspects are worth discussing.

Firstly, it was clear that participants with music experience had a more intuitive interaction with the software. Concepts such as *loading a tail* seemed straightforward for subjects with music experience, while, as expected, it was obvious that technical terms were unknown for subjects without music experience. However, these subjects still manage to solve the task by visually inspecting the GUI and acting based on signifiers and feedback. Similarly, while invited to disable all-pass filters in the plugin, a participant stated: *"With APF maybe the sound gets a bit thinner but honestly I struggle to hear a difference"*, which is symptomatic of how some subjects without music experience struggled to hear the subtle effect of frequency-dependent phase shifting, causing slight confusion.

Adjusting room size generally seemed intuitive to participants. Quotes such as *"It was not clear to me if z controlled room depth or room height" and "Not clear if y and x is the floor or z is height or depth. Once you move them it makes sense"* indicate some confusion with the depth/height controls. As seen in figure 4.2, y was used to control the room depth, and z was used to control the room height. This indicates confusing labeling of the controls, and by investigating relevant literature on 3D modeling [8, 13, 20], it was confirmed that z usually denotes depth and not height. While this might have caused confusion, it was, furthermore, mentioned that all confusion had cleared when sliders were moved and the responsive visualization of the room adjusted to the controls. Another solution to this problem could be to include the parameter's description in the label of the slider, such as *room x (depth)*.

Adjusting the position of the source and microphone was clear to all participants. Comments such as *"When you put them in the same position you have more of the direct signal and less reverb"*, indicate that participants had a clear idea of the acoustical mechanisms in play. Several participants, furthermore, requested to have a visual representation of the source and microphone. This is well summarized by quotes such as *"It would be nice to have visuals of location of source and mic. It's hard to imagine"* and *"The position of source and mic is not clear. Maybe have an x/y matrix with coordinates with both source and mic in the same matrix"*. Some participants requested the source and microphone to be visualized dynamically as the current visualization of room size in the GUI, while others suggested replacing x/y sliders with one coordinate system/grid containing two dots representing the source and microphone. Moreover, it was stated that it was confusing that the listening position of the subject was based on the microphone position. Conversely, it was stated that this could be solved by dynamically visualizing both the source and microphone in the room.

While most subjects heard an effect of the wall absorption filters, it was stated by several participants that they expected a bigger effect from replacing room ma-

terials. The auditory effect of the filters is not very pronounced, as these filters only affect the early reflections while leaving the tail unaffected, making the perceived difference very insignificant. The same applies to the adjustment of room size, which also only affects the early reflections. These comments could indicate a design flaw in the user interface, as the dynamic controls invite users to adjust parameters in real-time, while the controls were intended to be adjusted to the specifications of the room from which the tail was recorded. While this could be avoided by prompting the user with a popup message asking for room size and wall materials upon loading the tail, it would limit the creative aspects of the VST.

In general, as also mentioned earlier, more visual feedback was requested by the participants. While one subject commented *"It was intuitive that orange meant out of bounds"* another subject stated, *"It was not clear that the orange color meant out of bounds, maybe use a more alarming color. But it makes sense when you know it. Maybe color the whole slider or just write 'outside boundary'"*. The latter quote indicates that some subjects desired more visual feedback from the GUI. As proposed by the subject, changing the orange color (currently indicating a source or microphone placed outside of the room) to a more alarming color could be a simple way of solving this issue, although this issue would likely not remain relevant if the controls for positioning source and microphone would be replaced with a coordinate system/grid, clearly indicating when an object is placed out of the room.

7.6 Heuristic Interview

The following section covers the findings from the heuristic interview with Emanuele Parravicini, CTO at Audio Modeling, as described in section 6.6.

7.6.1 Results

The following section shows the results of the heuristic interview. The comments from the interviewee are summarized in table 7.2.

Topic	Comment
General	"It is very effective."
Optimization	"An improvement could be to calculate the fade-in of the tail in the VST instead of MATLAB." "It would be nice with an accurate representation of the ER, tail, etc." "From where is the IR recorded? It is a good idea to mention this." "The CPU can be reduced using a hybrid convolution which uses part of the IR with TDM and the rest with FFT."
Commercial products	"If I should choose for a commercial product, I would let the user disable/enable the direct/reverberant relationship. The gain of reverb in real life is more or less constant." "In a commercial product, we can have a button to switch between high-quality vs low-quality to switch between first-order only and the two-orders." "In a commercial product, we need CPU as low as possible while maintaining the quality as much as possible. 1st order plus tail is very a good compromise." "Audio Modeling already uses this [hybrid convolution] in their commercial products. The CPU usage is about one-fifth of the TDM, depending on block size, etc.."
Future work	"Consider how to adapt the VST to have many sources. We could make a multi-source VST on the bus to do all processing and then client VSTs send position and audio from each track."

Table 7.2: Selected quotes from the heuristic interview.

The raw notes from the heuristic interview is found in appendix VI.

7.6.2 Results Discussion

While it was clear that Parravicini of Audio Modeling was satisfied with the product, the interview yielded much constructive feedback. Parravicini confirmed that the focus on maintaining a computationally lightweight algorithm (such as choosing first-order reflections instead of higher orders), was considered very important, stating that *"in a commercial product, we need CPU as low as possible while maintaining the quality as much as possible. 1st order plus tail is very a good compromise"*. Furthermore, it was added that for the commercial application a switch in the GUI could be implemented to allow users with high CPU power to switch between 'low quality' (first-order) and 'high quality' (two orders). Parravicini, moreover, described how Audio Modeling currently uses a hybrid convolution algorithm that handles part of the IR through time-division multiplexing (TDM) and the rest with Fast Fourier Transform (FFT) and that this algorithm uses one-fifth CPU power compared to a TDM based algorithm. He pointed out that using the hybrid convolution algorithm in this application would, most likely, reduce the CPU usage significantly.

Parravicini pointed out that an improvement to the application would be to do all 'tail preparation' in the VST instead of in MATLAB as well as including an accurate representation of the early reflections, tail, etc. instead of the current visualizations. These are considered valuable points, as the changes would increase the overall usability of the software, potentially allowing users a more intuitive experience. Parravicini, furthermore, advised considering how to adapt this implementation to allow for multiple sound sources, as the commercial implementation would require this. He proposed the implementation of a multi-source VST and place on a bus in the DAW while each track had a client VST sending audio and position data to the multi-source VST. This would be a natural next step in the development of this software.

Chapter 8

Discussion

In the process of designing, implementing, and evaluating a hybrid algorithm for efficient rendering of room acoustics in real-time, a few aspects are worth discussing. While the findings were discussed individually in the previous chapter, this chapter aims to discuss more general aspects of the project from a broader perspective.

The MUSHRA test was used for several evaluations in this project. As MUSHRA tests output ordinal data (and not interval or ratio data) it does not fulfill the requirements for parametric tests [17]. As data is ranked for non-parametric tests, some information about the magnitude of difference between scores is lost, why non-parametric tests are less powerful than the parametric counterparts [5]. This increases the change of a Type II error, i.e. a bigger chance of accepting no difference between groups when, in reality, a difference exists [5].

As computational efficiency is an important factor, it could have been beneficial to evaluate the computational complexity of running the hybrid algorithm. As the implementation of the algorithm in this project is in a prototypical state, the implementation is not optimized for CPU efficiency. This would require a non-negligible amount of optimization of the code. On the contrary, the efficiency is indicated by the simplicity of the algorithm. Consisting of only six delay lines, absorption filter, and all-pass filters, the dynamic rendering of early reflection, is assumed to be very lightweight. Furthermore, as Parravicini described in section 7.6.1, using the hybrid convolution algorithm from Audio Modeling would allow the convolution with the tail to be performed in a very efficient manner. Based on this, it is fair to assume that the hybrid algorithm from this project, potentially, is very computationally effective.

As this project has demonstrated effectiveness at rendering efficient room acoustics

in real-time, it is worth looking into the ability of the project to expand in future directions. As discussed with Parravicini in section 7.6.1, allowing for multiple sound sources in the same room, with the same microphone position, would be a natural extension of this project. It would, most likely, require a new general structure of the implementation as well as communication between different VSTs. There is no apparent reason, why this should not be possible. Moreover, the current implementation of the hybrid algorithm only considers shoebox rooms. For a complete future implementation, it seems natural to allow the user to use late reverberation from more complex room shapes. While the image source method can be adjusted for polyhedric rooms of any shape [7], it could be challenging to create an interface for the user to define more complex rooms. It could, moreover, be interesting to conduct perceptual experiments to shed light on the perceptual effect of approximating complex room geometry with simpler shapes such as shoebox rooms or polyhedrons with a limited number of faces.

Chapter 9

Conclusion

It is concluded that an effective real-time VST for realistic rendering of room acoustic was developed. Through evaluation of the perceptual significance of isolated early reflection, it was found that a perceptual trade-off was present between all neighboring conditions (three orders of reflections to one order of reflections). From the evaluation of perceptual significance of early reflections in complete impulse responses, it is concluded that no significant difference was found between three orders of reflections and two orders of reflection. Additionally, considering only harmonic (soprano) stimuli, no significant difference was found between first-order stimuli and three-order stimuli, why it is concluded that first-order reflections are sufficient for applications such as this one, as the content of the virtual instruments of Audio Modeling is highly harmonic. From the evaluation of perceptual quality of different reverb algorithms, it is concluded that a significant difference was found between the recorded RIR and the proposed hybrid algorithm, as well as between the hybrid method and the SDN algorithm. However, these conclusions should not be taken as definite, as the comparative nature of the evaluation sheds light on the similarity to the quality of the recorded RIR and not the quality of the reverberation, which means that artifacts from the recording process, spectral coloration from microphones, speaker, etc. could affect the perceived similarity. The usability evaluation proved the interface of the plugin to be intuitive for the user to maneuver. It was, furthermore, concluded that more visual feedback in the GUI was desired by most subjects. Ultimately, the hybrid algorithm was presented to Parravicini of Audio Modeling and a heuristic interview was conducted. Through the heuristic interview, it was concluded that the product was found to be very satisfactory and that focusing on maintaining a computationally efficient algorithm was a correct decision. While the computational efficiency of the algorithm was not quantitatively investigated due to the prototypical nature of the implementation, the simple infrastructure of the proposed algorithm is concluded to potentially be very effective.

Bibliography

[1] Jont B Allen and David A Berkley. "Image method for efficiently simulating small-room acoustics". In: *The Journal of the Acoustical Society of America* 65.4 (1979), pp. 943–950.

[2] Enzo De Sena et al. "Efficient synthesis of room acoustics via scattering delay networks". In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23.9 (2015), pp. 1478–1492.

[3] F.A. Everest and K. Pohlmann. *Master Handbook of Acoustics*. McGraw-Hill Education, 2009. ISBN: 9780071603331.

[4] Angelo Farina. "Simultaneous Measurement of Impulse Response and Distortion with a Swept-Sine Technique". In: *Journal of the Audio Engineering Society* (2000).

[5] A. Field and G. Hole. *How to Design and Report Experiments*. SAGE Publications, 2003. ISBN: 9780761973836.

[6] Benjamin Friedlander and Boaz Porat. "The Modified Yule-Walker Method of ARMA Spectral Estimation". eng. In: *IEEE transactions on aerospace and electronic systems* AES-20.2 (1984), pp. 158–173. ISSN: 0018-9251.

[7] M. Gensane and F. Santon. "Prediction of sound fields in rooms of arbitrary shape: Validity of the image sources method". In: *Journal of Sound and Vibration* 63.1 (1979), pp. 97–108. ISSN: 0022-460X. DOI: https://doi.org/10.1016/0022-460X(79)90380-8. URL: https://www.sciencedirect.com/science/article/pii/0022460X79903808.

[8] Karen Goulekas. *Visual Effects in a Digital World: A Comprehensive Glossary of Over 7000 Visual Effects Terms*. Elsevier, 2001.

[9] David Havelock, Sonoko Kuwano, and Michael Vorländer. *Handbook of signal processing in acoustics*. Vol. 1. Springer, 2008.

[10] Jyri Huopaniemi, Lauri Savioja, and Matti Karjalainen. "Modeling of reflections and air absorption in acoustical spaces a digital filter design approach". In: *Proceedings of 1997 Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE. 1997, 4–pp.

[11] Jeffrey Hurchalla. "A Time Distributed FFT for Efficient Low Latency Convolution". In: *Journal of the Audio Engineering Society* (2010).

[12] *ISO 3382:1:2009(E), Acoustics — Measurement of room acoustic parameters — Part 1: Performance spaces*. Standard. Geneva, CH, June 2009.

[13] Wallace Jackson. *VFX Fundamentals: Visual Special Effects Using Fusion 8.0*. Springer, 2016.

[14] Jean-Marc Jot and Antoine Chaigne. "Digital delay networks for designing artificial reverberators". In: *Audio Engineering Society Convention 90*. Audio Engineering Society. 1991.

[15] Mark Kahrs and Karlheinz Brandenburg. *Applications of digital signal processing to audio and acoustics*. Springer Science & Business Media, 1998.

[16] Heinrich Kuttruff. *Room acoustics*. Crc Press, 2016.

[17] Catarina Mendonça and Symeon Delikaris-Manias. "Statistical tests with MUSHRA data". In: May 2018.

[18] James A Moorer. "About this reverberation business". In: *Computer music journal* (1979), pp. 13–28.

[19] Damian Murphy et al. "Hybrid room impulse response synthesis in digital waveguide mesh based room acoustics simulation". In: *Proceedings of the 11th International Conference on Digital Audio Effects (DAFx-08)*. Citeseer. 2008, pp. 129–136.

[20] Todd Palamar. *Mastering Autodesk Maya 2014: Autodesk Official Press*. John Wiley & Sons, 2013.

[21] Nikolaos M Papadakis and Georgios E Stavroulakis. "Review of Acoustic Sources Alternatives to a Dodecahedron Speaker". In: *Applied Sciences* 9.18 (2019), p. 3705.

[22] Karl Pedersen and Mark Grimshaw-Aagaard. *The Recording, Mixing, and Mastering Reference Handbook*. English. United Kingdom: Oxford University Press, 2019. ISBN: 9780190686635.

[23] Leevi Peltola et al. "Synthesis of hand clapping sounds". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.3 (2007), pp. 1021–1029.

[24] Ville Pulkki. "Virtual sound source positioning using vector base amplitude panning". In: *Journal of the audio engineering society* 45.6 (1997), pp. 456–466.

[25] Ville Pulkki and Matti Karjalainen. *Communication acoustics: an introduction to speech, audio and psychoacoustics*. John Wiley & Sons, 2015.

[26] Joshua D Reiss and Andrew McPherson. *Audio effects: theory, implementation and application*. CRC Press, 2014.

[27] Jens Holger Rindel. "The use of computer modeling in room acoustics". In: *Journal of vibroengineering* 3.4 (2000), pp. 219–224.

[28] Lauri Savioja and Vesa Valimaki. "Reducing the dispersion error in the digital waveguide mesh using interpolation and frequency-warping techniques". In: *IEEE Transactions on Speech and Audio Processing* 8.2 (2000), pp. 184–194.

[29] Michael Schoeffler et al. "webMUSHRA—A comprehensive framework for web-based listening tests". In: *Journal of Open Research Software* 6.1 (2018).

[30] Manfred R Schroeder. "Natural sounding artificial reverberation". In: *Audio Engineering Society Convention 13*. Audio Engineering Society. 1961.

[31] Manfred R Schroeder. "New method of measuring reverberation time". In: *The Journal of the Acoustical Society of America* 37.6 (1965), pp. 1187–1188.

[32] B Series. "Method for the subjective assessment of intermediate quality level of audio systems". In: *International Telecommunication Union Radiocommunication Assembly* (2014).

[33] Guy-Bart Stan, Jean-Jacques Embrechts, and Dominique Archambeau. "Comparison of different impulse response measurement techniques". In: *Journal of the Audio engineering society* 50.4 (2002), pp. 249–262.

[34] John Stautner and Miller Puckette. "Designing multi-channel reverberators". In: *Computer Music Journal* 6.1 (1982), pp. 52–65.

[35] E. Tarr. *Hack Audio: An Introduction to Computer Programming and Digital Signal Processing in MATLAB*. Audio Engineering Society presents. Routledge, 2018. ISBN: 9781138497559.

[36] Scott A Van Duyne and Julius O Smith. "Physical modeling with the 2-D digital waveguide mesh". In: *Proceedings of the international computer music conference*. INTERNATIONAL COMPUTER MUSIC ACCOCIATION. 1993, pp. 40–40.

[37] Vesa Välimäki et al. "Fifty years of artificial reverberation". In: *IEEE Transactions on Audio, Speech, and Language Processing* 20.5 (2012), pp. 1421–1448.

[38] Udo Zölzer et al. *DAFX-Digital audio effects*. John Wiley & Sons, 2002.

Appendix A

Pseudo-Code For Visualization Of Room Dimension

```
Line 1:
Point A: (posX-offsetDepth-offsetWidth, posY+offsetDepth-offsetHeight)
Point B: (posX-offsetDepth+offsetWidth, posY+offsetDepth-offsetHeight)
Line 2:
Point A: (posX-offsetDepth-offsetWidth, posY+offsetDepth+offsetHeight)
Point B: (posX-offsetDepth+offsetWidth, posY+offsetDepth+offsetHeight)
Line 3:
Point A: (posX-offsetDepth-offsetWidth, posY+offsetDepth+offsetHeight)
Point B: (posX-offsetDepth-offsetWidth, posY+offsetDepth-offsetHeight)
Line 4:
Point A: (posX-offsetDepth+offsetWidth, posY+offsetDepth+offsetHeight)
Point B: (posX-offsetDepth+offsetWidth, posY+offsetDepth-offsetHeight)
```

Listing A.1: Front rectangle

```
Line 5:
Point A: (posX+offsetDepth-offsetWidth, posY-offsetDepth-offsetHeight)
Point B: (posX+offsetDepth+offsetWidth, posY-offsetDepth-offsetHeight)

Line 6:
Point A: (posX+offsetDepth-offsetWidth, posY-offsetDepth+offsetHeight)
Point B: (posX+offsetDepth+offsetWidth, posY-offsetDepth+offsetHeight)

Line 7:
Point A: (posX+offsetDepth-offsetWidth, posY-offsetDepth+offsetHeight)
Point B: (posX+offsetDepth-offsetWidth, posY-offsetDepth-offsetHeight)

Line 8:
Point A: (posX+offsetDepth+offsetWidth, posY-offsetDepth+offsetHeight)
Point B: (posX+offsetDepth+offsetWidth, posY-offsetDepth-offsetHeight)
```

Listing A.2: Back rectangle

```
Line 9:
Line 9:
Point A: (posX-offsetDepth-offsetWidth, posY+offsetDepth-offsetHeight)
Point B: (posX+offsetDepth-offsetWidth, posY-offsetDepth-offsetHeight)

Line 10:
Point A: (posX-offsetDepth-offsetWidth, posY+offsetDepth+offsetHeight)
Point B: (posX+offsetDepth-offsetWidth, posY-offsetDepth+offsetHeight)

Line 11:
Point A: (posX-offsetDepth+offsetWidth, posY+offsetDepth+offsetHeight)
Point B: (posX+offsetDepth+offsetWidth, posY-offsetDepth+offsetHeight)

Line 12:
Point A: (posX-offsetDepth+offsetWidth, posY+offsetDepth-offsetHeight)
Point B: (posX+offsetDepth+offsetWidth, posY-offsetDepth-offsetHeight)
```

Listing A.3: Connecting lines

Appendix B

Images from the RIR measurement setup

8
O
-10
-5
0
40
80


2.0.028

CHAN 1