

Master's Thesis

# Replicating real-world audio experiments in virtual reality

**Dorđe Perinović, BSc**

Matr. number: 01431685

Aessor: O.Univ.Prof. Mag.art. DI Dr.techn. Robert Höldrich

Supervisor: Dr. Matthias Frank

Master's Degree Programme: Electrical Engineering and Audio Engineering

Graz University of Technology

University of Music and Performing Arts Graz

Institute of Electronic Music and Acoustics

Graz, 30.01.2023





# Acknowledgments

This project was an ending of a long and thrilling journey. It consisted of incorporating all of my, during the university gathered, engineering knowledge with two hobbies, being the 3D modeling and the video game programming. I am therefore very thankful to my parents and my sister supporting me both financially and emotionally throughout all of my educational days, and providing me the possibility to study and develop the mentioned hobbies.

I am very thankful to God guiding me through the most difficult times of my whole education, especially through the years affected by Covid, and making me choose the correct career path.

Since this project was based on experiments, I am thankful to all of 31 participants, making the progression of the project possible. I am thankful to the professor who prepared me for entrance exam of my study programme in 2015 and who also evaluated the experiment together with few of his colleagues.

Thanks to the supervisor Dr. Matthias Frank I was able to finish certain parts of the project more easily and to correctly understand the thesis. I am grateful to the assessor O.Univ.Prof. Mag.art. DI Dr.techn. Robert Höldrich and to Dipl.-Ing. Dr.techn Martin Hagmüller from Technical University of Graz who accepted to additionally support and grade the thesis.

I am thankful to my girlfriend who believed in me even in periods of more challenging obstacles. I also say "thank you" to all of my family and friends from Graz and Banja Luka who respected my passion and backed me through the years of my education.

Finally, I'm much obliged to all of the professors, teachers and other people who expanded my knowledge and helped me to direct myself into the right life path, which brought me to this very moment.

## **Statutory declaration**

I declare that this thesis was written entirely by me. Other than in the work mentioned literature, no external material was quoted or used. Therefore I state that this thesis, as a continuation of my recently published audio-engineering project which it may use material from, is an original document.

A handwritten signature in blue ink, consisting of a stylized 'D' and 'P' followed by a flourish.

Graz, 30.01.2023

Perinović Đorđe, BSc



### **Kurzfassung**

Wahrnehmungsexperimente spielen eine wichtige Rolle in der Forschung und Entwicklung der Audiotechnologie. Bis vor kurzem waren sie fast immer ortsabhängig, was bedeutet, dass ein Experiment nur in bestimmten Räumen oder Orten durchgeführt werden konnte, was zu einer reduzierten Reproduzierbarkeit führte. In den letzten Jahren haben uns Entwicklungen in Computergrafik, virtueller Akustik und Elektronik ermöglicht, virtuelle Realität (VR) zu erleben. Theoretisch sollte ein detailliertes audiovisuelles Modell eines Raums eine VR-Umgebung mit 6 Freiheitsgraden bieten, in der reale Experimente mit reduziertem Hardwareaufwand auf der ganzen Welt nachgebildet werden könnten. Diese Masterarbeit erstellt ein detailliertes visuelles Modell des IEM CUBE und seines umgebenden Lautsprechersystems sowie ein akustisches Modell basierend auf verteilten Messungen und Simulationen. Schließlich wird ein Experiment, das vor einigen Jahren im CUBE durchgeführt wurde, in VR repliziert, um die Plausibilität der Modelle zu untersuchen.

### **Abstract**

Perceptual experiments play an important role in research and development of audio technology. Until recently, they were almost always location-dependent, which means that an experiment could be done only in certain rooms or places resulting in a reduced reproducibility. In the last years, development in computer graphics, virtual acoustics, and electronics enabled us to experience virtual reality (VR). Theoretically, a detailed audiovisual model of a room should provide a VR environment with 6 degrees of freedom, where real-world experiments could be replicated with reduced hardware effort all around the world. This thesis creates a detailed visual model of the IEM CUBE and its surrounding loudspeaker system, as well as an acoustic model based on distributed measurements, and simulations. Finally, an experiment that has been done in the CUBE some years ago will be replicated in VR to investigate the plausibility of the models.

## Contents

<b>1</b>	<b>Introduction</b>	<b>8</b>
1.1	History and project idea . . . . .	8
1.2	Preparation . . . . .	11
<b>2</b>	<b>Modeling</b>	<b>13</b>
2.1	Visual model . . . . .	13
2.2	Acoustic model . . . . .	17
2.2.1	Diffuse reverberation . . . . .	19
2.2.2	Early reflections and additional attenuation . . . . .	24
2.2.3	Levels . . . . .	26
<b>3</b>	<b>Experiment setup</b>	<b>30</b>
3.1	General . . . . .	30
3.2	Unity . . . . .	32
3.3	Reaper . . . . .	34
<b>4</b>	<b>Experiment layout</b>	<b>38</b>
4.1	Personal introduction . . . . .	38
4.2	Tutorial . . . . .	39
4.3	Experiment . . . . .	42
<b>5</b>	<b>Result evaluation</b>	<b>47</b>
5.1	Comparison of participant groups . . . . .	49
5.2	All results . . . . .	52
5.3	IEM audio engineers . . . . .	54
<b>6</b>	<b>Conclusion and further development</b>	<b>58</b>
<b>7</b>	<b>Appendix A</b>	<b>62</b>
<b>8</b>	<b>Appendix B</b>	<b>63</b>

<i>Dorđe Perinović: Replicating real-world audio experiments in virtual reality</i>	7
<b>9 Appendix C</b>	<b>64</b>
<b>10 Appendix D</b>	<b>65</b>

# 1 Introduction

## 1.1 History and project idea

Recent decades were the prime years of technological advancements in the computer graphics and audio reproduction. From the brief prototypes and first versions of displaying 3D video for video games to the complete and even wireless 4K-resolution virtual reality (VR) equipment, now it's possible to deliver the various artistic and scientific ideas totally location-independent.

This means that the projects previously possible to be carried out only at certain rooms or places can now be completely reproduced virtually. That gave us the possibility of having 6 degrees of freedom in any surrounding that is available as a 3D model, which will bring us to the first task of the this master's thesis.

Together with the virtual replication of the visual 3D space, the development of audio followed accordingly. The main task of the virtualization equipment was to trick our brain into thinking that it's in a real environment and to eventually ignore the fact that something is used to generate what we are experiencing with our senses. Taken into account that the acoustics is much more complicated to understand and to calculate than optics, it's correspondingly more difficult to generate plausible virtual acoustics. Moreover, the visual part of VR was able to accomplish 6 degrees of freedom, while the 3D audio typically had only 3 degrees of freedom, which will be explained in detail below.

### 3D Audio

In order to simulate the 3D audio environment, one way could be to listen in the center of a (hemi)sphere of audio sources. By controlling the output of every surrounding loudspeaker, audio signals can be precisely heard from every direction. While being sufficient for the most uses, stereo and 5.1 surround sound reproduction can't deliver the right sound direction for certain situations where the accurate audio localization is needed [1]. On the other hand, for such environment a whole room with many loudspeakers and the right acoustics is needed which is not easily available. One of such rooms is built at the "Institute for Electronic Music and Acoustics" (*IEM*) of the "University of Music and Performing Arts" (*KUG*) in Graz (Austria), which is used for ambisonics experiments, concerts, and lectures [2]. Until now, this room has been used for various experiments on the development of the ambisonic listening [3]. To make the room available to others outside Graz, this real loudspeaker system surrounding has to be somehow simulated and attached to the listener's ears, in the way that the listener can move around the VR and experience the corresponding audio at the same time.

Since today, the commercial use of 2-channel binaural is widely spread, it has been found out that bringing the ambisonic acoustic surroundings to the headphones would be a good step in the audio research. For that, an *binaural decoder* has to be used [4, 5]. One of plug-ins of the *IEM Plug-in Suite*<sup>1</sup> is able to do that, which will be explained in detail later in this thesis.

Being able to hear the ambisonic audio on headphones and to wear the VR headset at the same time was crucial for the further experiments in the VR, since now both the visual and the audio part of the VR has 6 degrees of freedom. This means that the experiments in the VR can be created and that the rooms can finally be brought to the virtual world both visually and acoustically. However, the simulation of ambisonic acoustic model consists of many parts (see section 3.3) which could create a deviation from the real-life acoustic model. How well then the VR sound represents the real-life acoustics will be examined in this thesis.

That question caused many further acoustic experiments to appear, also in VR. One significant experiment from K. Enge [6] also done at *IEM* was about the resolution of the ambisonic reproduction and it compared a virtualization of a single loudspeaker and the real one. It is possible to do the real-time simulations of room acoustics and to run the VR at the same time, but the practice showed that the accurate acoustic simulations needed for these experiments are computer-resource demanding. This experiment focused on reducing resources usage as well and it showed that the 3rd order of binaural ambisonics decoding was as plausible as 7th order. This discovery will be important for this project.

One other important discovery was done shortly after the mentioned project where first order of binaural ambisonics reproduction was rated as sufficient for diffuse reverberation [7]. All these experiments built confidence in using efficient ambisonics of relatively low orders simulation as such for the scientific experiments.

Another question for VR and audio reproduction was how well the visual VR surroundings and the ambisonics simulation correlate. Although already partially answered by the mentioned thesis, it didn't focus completely on the audio reverberation, distance, room size, etc. This created the idea of the recent experiment which was done completely in VR [8]. That project, aside from the other interesting findings, revealed how well the ambisonics simulation matches the visual surroundings.

---

1. freely available on [plugins.iem.at](http://plugins.iem.at)

**Project foundation**

Experiments on matching the sound and vision were nothing new at this point. In some of them, people were matching the heard sound reproduction with pictures [9], but the VR and the simulated acoustics gave us a way to really feel the surroundings both visually and acoustically, without the influence of the real-world surroundings.

Until then, the room divergence effect has not been investigated in virtual environments [10]. It happens while listening to the anechoic recording, while being in the room with the longer reverberation. First hints for its relevance in VR were found in the experiments by K. Enge [6].

Discoveries from his experiments inspired the idea of matching sound and vision in virtual reality, which was carried out in the following project [8]. One of the 3D models of the previous project was transferred for it to be the virtual experiment room (Mumuth hall in Graz [11]). This experiment consisted of the introduction to VR, tutorial in VR where the participants had time to get used to the equipment and VR itself and 6 experiments parts of in total 30 trials. Participants could change the values according to the stimuli using a virtual tablet. For example, they would adjust the room size to the reverberation they hear and vice versa, loudspeaker distance to the heard source distance and the same with the loudspeaker/source rotation. Interesting results were gathered and certain questions were answered such as the confirmation that the distances under 10 m tend to be overestimated and the ones over 10 m underestimated [12, 13, 14, 15]. As it happens with every experiment it raised some new questions. Since it was an audio research project, the visual quality and the correspondence to the real-world wasn't discussed, although it would be interesting to repeat such experiment with new 4K-resolution VR equipment.

One of the important questions was how accurate is this acoustic simulation in the first place. In order to correctly make a correlation between the visual and the auditory part of VR it is necessary for them to correctly represent their real-life counterparts. Although the audio of the previous project was calibrated, it wasn't evaluated in comparison to the real-life which brought us to the concept of this thesis. One way to do that was to replicate an existing real-life experiment in virtual reality and afterwards compare the results. For that, a sweet spot evaluation experiment from 2017, done in real-life CUBE was chosen [3]. This project builds upon the previous project [8], as it uses the same equipment, the parts of programming code and the improved version of the previous acoustic simulation.

## 1.2 Preparation

### Setting up the VR equipment

As in the previous project, an HTC Vive VR set was used, since most of the programming code for understanding its functionality has already been written.

The minimum setup of the VR set consists of 2 base stations which emit infrared beams that are used for positioning the headset and the 2 hand controllers in the virtual 3D space. In order to achieve the medium movement area of around 2x3m (so called "*safe zone*") 2 base stations are sufficient. For larger areas, a number of base stations has to be increased to 4, but for this project, the mentioned area meets our needs. The best way is to put the base stations high enough so that they see the controllers and the headset.

The VR Headset is basically the video projection on 2 video screens slightly horizontally shifted on the each eye. This technology replicates our vision where this slight difference in what our eyes see makes us recognize the depth visually. When the headset is moved around the safe zone or rotated, its motion sensors, as well as the interference with the base stations' infrared beam, generates the position and rotation values for the projections on the eyes. The tracking of 2 hand controllers works the same way (see figure 1 and online article [16]).

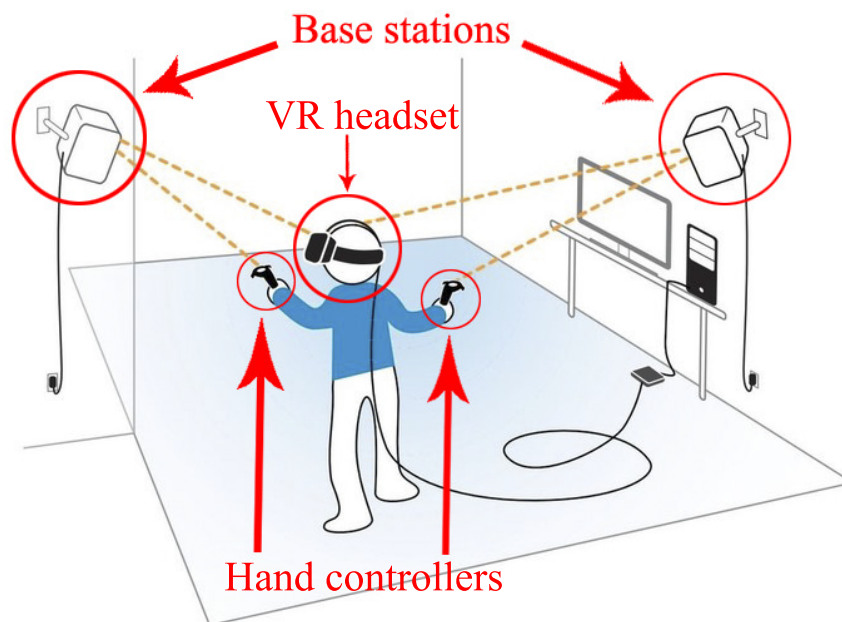


Figure 1 – Virtual Reality (VR) equipment and its usage

**Planning**

After analyzing the needed steps, the following plan for the development of the master's project was created:

1. Unlike previous project, for this one the 3D model of the room [2] wasn't available. This wasn't seen as a big obstacle in the beginning since a potential way of modeling a room using smartphone or a tablet camera was recently created [17]. After some research on which applications could be used to capture a 3D model, it was concluded that this technology wasn't sufficient for our needs. The 3D model needed to be versatile where every surface could have different color or texture. Since this project was going to be worked on by an audio engineering student, experience in 3D modeling was purely hobby-related and it was highly uncertain if the camera-captured model could be manipulated and later even imported to the experiment programming software correctly. Therefore it was decided that it was less difficult and more accurate to build the 3D model from scratch using SketchUp [18] (see subsection 2.1).
2. The calculation of the parameters (reverberation time, spectrum, reflection properties, etc.) of the acoustical model could be done by analyzing 720 impulse responses (IRs) recorded in CUBE (see subsection 2.2<sup>2</sup>).
3. The calibration of the acoustic model could be implemented in Reaper DAW (digital audio workstation) based on the analyzed calculation (see subsection 3.3). Measurement of the adjusted acoustics by rendering the impulse responses (IRs) and analyzing them in Matlab. Two types of the reverberation (FDN and BRIR reverberation) had to be adjusted to produce the mathematically same results.
4. The programming of the communication between the Reaper DAW and Unity application using C# programming language, as well as programming of the experiment itself could be done in the Unity game engine (see subsections 3.1 and 3.2).
5. The Gathering of the participants and evaluation of the experiment trials was done (section 4).
6. Finally, the results were analyzed and compared to the real-life experiment.

---

2. freely available at: <https://phaidra.kug.ac.at/view/o:104435>



## 2 Modeling

In this section it will be explained how the visual and the acoustic model were created to represent the real environment in the best possible way.

### 2.1 Visual model

After deciding to create a 3D model from scratch, a slot in the real room was booked for the measurements. Idea was to create a functional model for later uses on the institute and not just for this project. In reality, the IEM CUBE consists of almost no parallel surfaces, which influences the accuracy of the modeling negatively. Since the room in the floor plan represents a trapeze with one curved base surface, first step was to measure the angle between the front wall (see figure 2) and the side walls, which was around 100 degrees. Afterwards the angles between window segments on the back wall were measured (see figure 3). Every segment was around 2 to 3 degrees, so that the help of the university published floor-plan was used to increase accuracy of finding such a small angle.

As already mentioned in the previous section, *SketchUp* [18] was used for modeling of the CUBE for virtual reality. It was decided not to include other objects inside the room such as piano, chairs and trussing since these objects only increase the number of the surfaces and therefore the graphics card's usage later in the application itself. All of the audio-engineering necessary objects, like absorbers, windows, doors were taken into account. After creating the whole model and measuring of the values inside the editor, it appeared that they correspond with the real-life measurements.

The only important object left to be modeled was the loudspeaker ("Tannoy System 1200") and everything was ready to be exported for Unity (figure 4). The reason why the loudspeakers were not built inside the CUBE 3D model was because it was much easier to rotate and move the loudspeakers inside the Unity than in the SketchUp, because every modification done in the SketchUp had to be exported again for Unity. Another challenge was the scaling between the SketchUp and the Unity and that the Unity doesn't accept all 3D formats in the the same way. In the end, the .fbx format showed the least differences in the scaling and it was the easiest for the import into Unity.



Figure 2 – CUBE front wall view and its 3D model



Figure 3 – CUBE back wall view and its 3D model



Figure 4 – Tannoy System 1200 3D model

## 2.2 Acoustic model

When creating an experiment using virtual acoustics, it is important to make sure that it represents the real acoustics as accurate as possible. Otherwise, the experiment results will be obsolete. However it's not possible to confirm that the virtual room sounds the same on each end. It is however possible to check if the virtual room characteristics match the original measurement values at a few real measurement positions.

There were multiple ways to create an accurate acoustic model, but the computer resource use had to be taken into account. One of these tools was *SPARTA* with the *6DoFconv* plug-in [19]. This plug-in works by doing an interpolation between 1st order IRs at multiple positions. However, it can only virtualize one loudspeaker and it means that 24 simultaneously working plug-ins would be needed for this project, which wouldn't be achievable for the current computational resources. A similar tool was built at the *IEM* institute by the colleague M. Kaspar [20]. A variant less demanding for the computational resources which can achieve 6 degrees of freedom at the same time was chosen, being the *Reaper DAW* with the *IEM* plug-ins. Through the *IEM* plug-ins it was possible to create 2 variants of the acoustic simulation in *Reaper*. Although less demanding, these variants still needed 2 separate computers for the calculation of the visual model (1st computer running *Unity*, section 3.2) and the acoustic model (2nd computer running *Reaper*, section 3.3).

Before mentioning more complex parts of the acoustic model, the first condition was to correctly simulate the real-life "Tannoy 1200 System" loudspeaker's directivity pattern (see Appendix A and Appendix B). For doing so, a freely available *IEM* plug-in "mcfx\_conv" was used to make a convolution between the mono audio signal and the corresponding IRs from the ".wav" file based on the information from the accompanying ".conf" configuration preset file [21]<sup>3</sup>. This creates a 16-channel 3rd order ambisonics track, which represents one of the *CUBE*'s loudspeakers and can be then rotated with a plug-in "SceneRotator" and/or moved with a plug-in "RoomEncoder".

The *first variant* was to simulate the direct sound and early reflections of every loudspeaker using the image-source-model in the plug-in "RoomEncoder" and the late part of the diffuse reverberation through the **FDN** reverberation [22, 23], using the plug-in "FDNReverb". An important characteristic of the plug-in "RoomEncoder" is that it calculates the acoustic model based on the real-time controlled listening position, respecting the 6 degrees of freedom. Even so, the "Number of reflections" parameter (seen on figure 15), which defines how many early reflections will be generated for each

---

3. A document about "Tannoy 1200 System" directivity can be found on the link: <https://phaidra.kug.ac.at/view/o:68229>

loudspeaker, had to be reduced to 40 from the maximum number of 236, which was used in the founding project [8]. In the previous project, the only one loudspeaker (thus the only one "RoomEncoder" plug-in) had to be simulated and it didn't make much difference regarding the computational resources even when the single computer runs both the visual and the acoustic simulation. According to [24], it is perceptually sufficient to compute 40 position-dependent reflections and the later ones position independent.

The *second variant* was using the same 24 loudspeaker tracks and the "RoomEncoder" plug-ins, but with **BRIR** reverberation [25]. The BRIR reverberation is the convolution between an IR created from the recording of the frontal loudspeaker 1 with the dummy-head on ear height in the center of the CUBE and the dry sound signal (e.g. speech recording [26]). This creates a sound signal with a reverberation from the real-life CUBE. This way, the BRIR reverberation doesn't simulate only the diffuse reverberation, but also the early reflections. For doing so the "Number of reflections" of the 24 "RoomEncoder" plug-ins has to be reduced to 0, leaving them to simulate only the direct sound. This variant is less demanding for the computational resources, but it's mathematically correct regarding early reflections and diffuse reverberation only when the participant is in the center of the room. Therefore, this thesis will reveal if this less demanding acoustic model is sufficient in regards to similarities to the real-life acoustics and if not, if at least the *first variant* corresponds to the real-life acoustics.

*Matlab* was chosen as a tool to analyze and calculate everything in the project as well as to generate the plots and figures. What was provided for this project are impulse responses of 30 receiver and 24 source positions (making altogether 720 impulse response .wav files) that were recorded in the real-life CUBE [27].

The parameters will be listed in the overview in the subsection 2.2.3 and table 1 to be later used in Reaper to tune the acoustic simulation. The plan of the calibration was like the following:

1. Frequency-dependent reverberation, by analyzing all real-life 720 impulse responses;
2. Diffuse reverberation spectrum by analyzing only the late part of 720 impulse responses (part after 100 ms after the direct sound);
3. Frequency-dependent early reflection attenuation, by using Sabine's reverberation formula [28];
4. Additional attenuation of every surface in the room, by analyzing specific parts of the impulse responses;
5. Using the calculated reverberation values for the FDN reverberation inside Reaper and adjusting its level to match the C80 value (clarity index) from real-life measurements;



6. Adjusting the level of a track with BRIR reverberation to match the same C80 value.

### 2.2.1 Diffuse reverberation

The parameters of the diffuse reverberation were needed for some other acoustic parameters and therefore it was the first part of the audio calibration. These values were needed to adjust the FDN reverberation as seen in the figure 14 using the parameter "Reverberation Time" in the "FDNReverb" plug-in. This reverberation is basically an algorithm that employs a 64x64 frequency-dependent feedback delay network [22, 23] and it has to be tuned in order to represent the real-life reverberation correctly. In reality, reverberation can be slightly different for every position in the room, which will also be discussed in the calibration plots.

In order to account for the frequency-dependency of the measured reverberation time from figure 5(top) there were only 2 filters available to adjust the reverberation curve in the "FDNReverb". Additional checks were needed, so that the Matlab-generated impulse was played over every virtual loudspeaker in Reaper and then rendered to 24 .wav files, which were IRs of the virtual room. This process needed to be repeated many times, until the tuned virtual reverberation (figure 5(bottom)) matched the measured one (top).

To represent many reverberation times in the figures(top, of 720 measured impulse responses and bottom, of 24), the **red** curve represents the median value of all of the curves which are in this case invisible. Now if we analyze the rest of the values below and above the median values and take the curve representing 20% and 80% percentile, we will get the **blue** and **orange** curve. Dotted lines below and above the red curve represent  $\pm 10\%$  from the median value to indicate the just-noticeable-difference (JND) [29]. If the majority of impulse responses doesn't deviate away from this area, which is fulfilled here, it permits us to use this variant of acoustic simulation for all listening positions in the model regarding FDN reverberation (figure 14 and figure 7). It means that the room in reality is diffuse enough, so that reverberation doesn't differ significantly in regards to where the sound is being heard.

Reverberation like any other sound has a specific timbre. In order to adjust it correctly, the later part (everything after 100 ms after the direct sound) has to be analyzed in the same way.

The dashed line in the figure 6 represents the median of all spectra. The tuning principle, just like in the previous case, had to be adjusted many times until the curve matched the

measured one. This time, another plug-in was used ("MultiEQ", as seen in the figure 14 and the figure 7), which had many more filters and therefore, the spectrum was easier to tune more accurately.

To completely adjust the "FDNReverb", a correct delay to the direct sound and early reflections has to be set up in order to calculate a correct C80 (clarity factor) value, which will be discussed in the section 2.2.3.



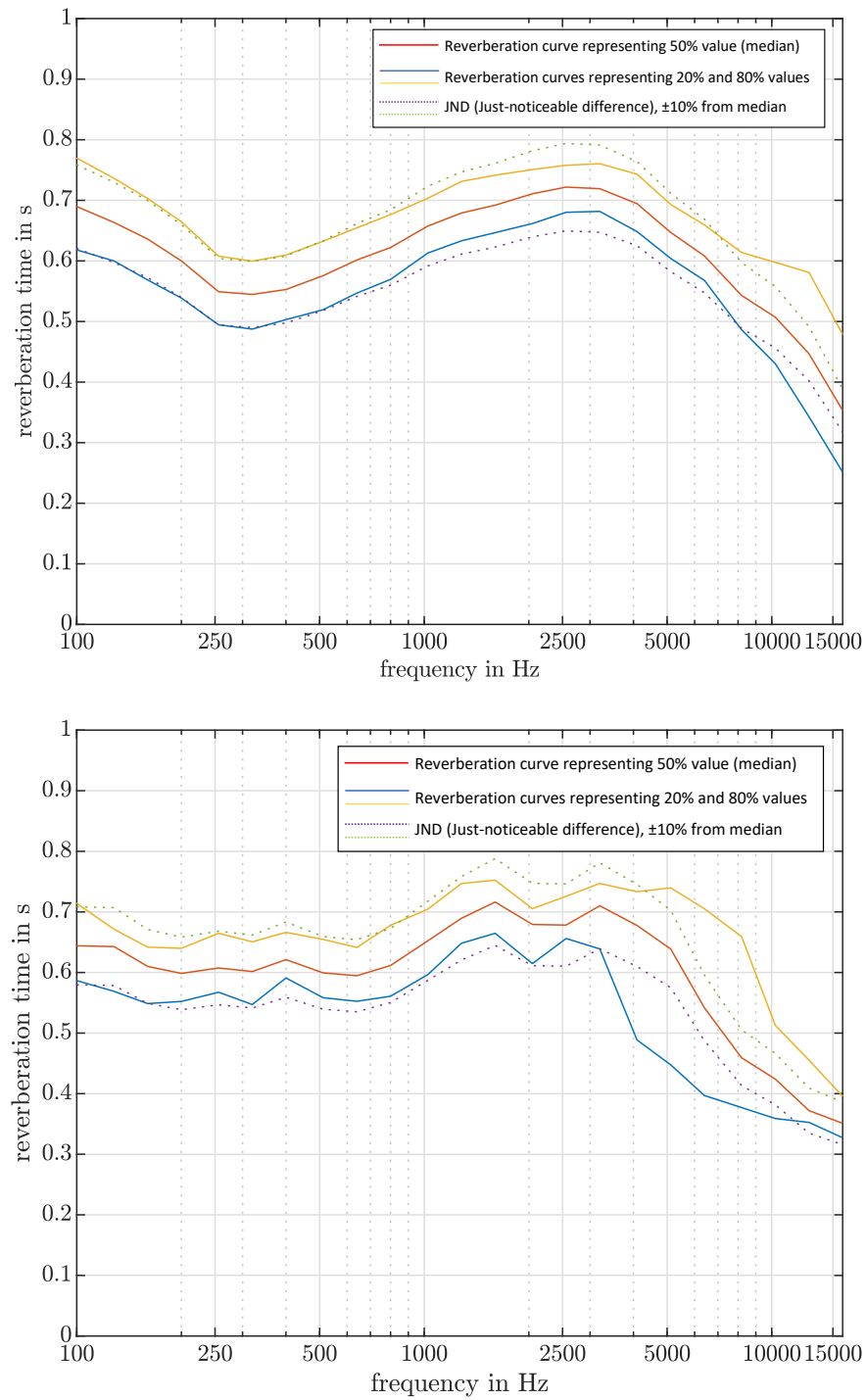


Figure 5 – Measured(top) and simulated(bottom) diffuse reverberation

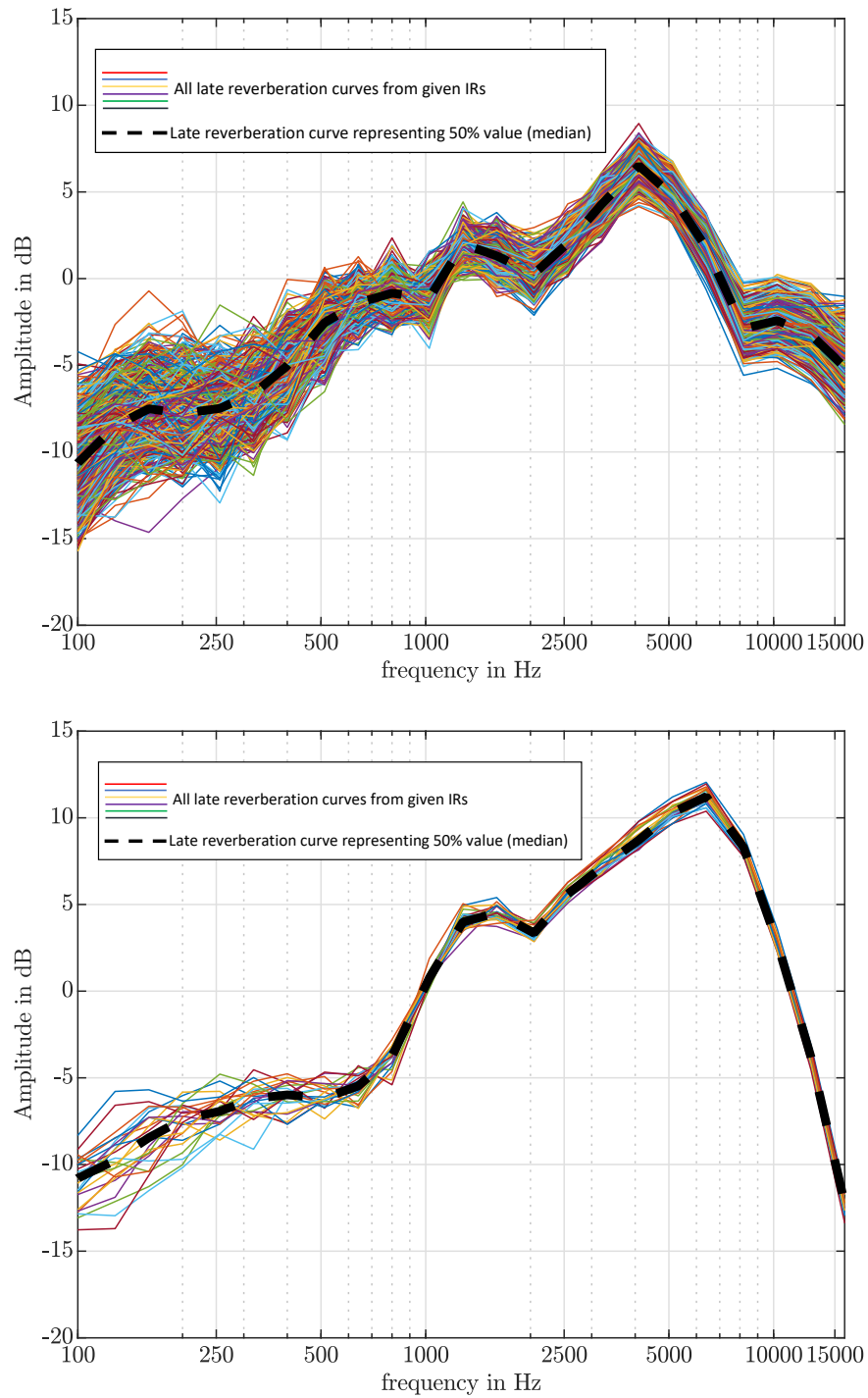


Figure 6 – Measured(top) and simulated(bottom) spectrum of the reverberation

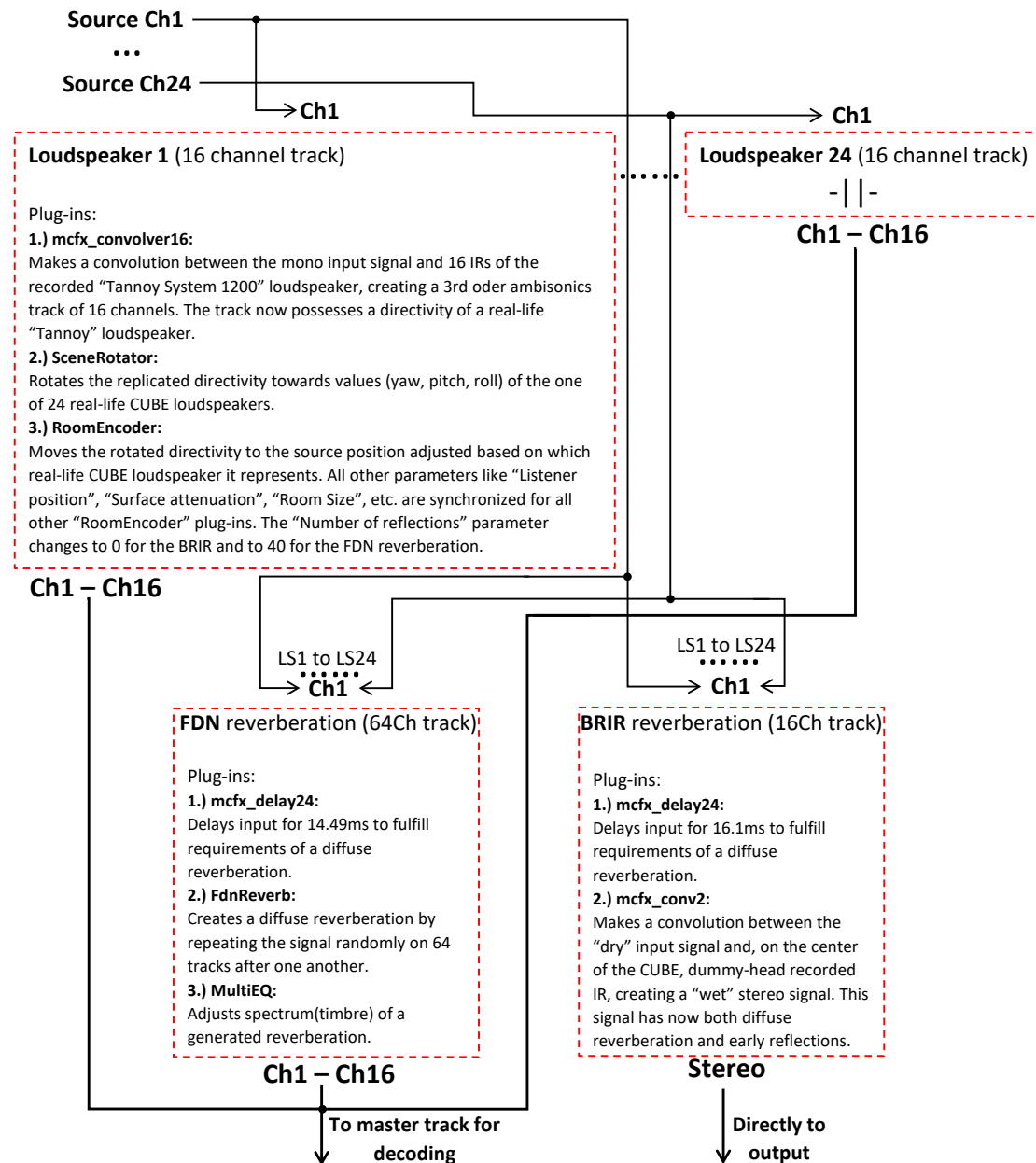


Figure 7 – Flowchart and differences between the FDN- and BRIR reverberation.

### 2.2.2 Early reflections and additional attenuation

The properties of the early reflections were dealt with inside the IEM plug-in "RoomEncoder" that is based on image-source model of a rectangular room. Although the attenuation is frequency-dependent, there were only 2 filters available to tune this parameter. It could be calculated with the Sabine's reverberation formula [28]:

$$T = 0.163 \cdot \frac{V}{A \cdot \alpha} \cdot \text{m}^{-1}[\text{seconds}], \quad (1)$$

where  $T$  is the reverberation,  $V$  the room volume and  $A$  the sum of all of the surfaces. For the calculation, the already found reverberation times from figure 5 were used. It was simplified to 0.6 seconds for the part below 5 kHz and the 0.5 seconds above. In order to find the wanted absorption factor  $\alpha$ , formula 1 had to be reshaped. The volume was calculated by approximating the 3D model and the value of  $V = 480 \text{ m}^3$  was calculated. The area was read directly from the 3D model in SketchUp which showed that  $A = 435 \text{ m}^2$ .

As a results for frequencies below 5 kHz (reverberation 0.6 s) the  $\alpha \approx 0.3$  was calculated and for the rest the  $\alpha \approx 0.36$ . To get the reflection factor, these values needed to be subtracted from 1. The corresponding dB values are then -1.55 dB for the part below 5 kHz and -1.94 dB for the rest. Therefore the value -1.55 dB was used as "overall attenuation" in the plug-in and additional -0.39 dB were added for the frequencies above 5 kHz.

This way the early reflections now represent the real-life CUBE correctly, but only if every surface in the room is of the same material and shape. In order to be even more detailed, the "additional attenuation" for every surface was added, since the floor is from wood, ceiling from concrete, there are curtains on the back wall, etc. This causes a different attenuation for every surface and this can be determined by analyzing the specific impulse responses. Since the project was provided with 30 measurement positions of 24 loudspeakers, some of them were suitable to find out specific surface attenuations [27].

Let's say that the front wall attenuation is wanted. One way would be to choose the loudspeaker that plays the sound perpendicular to the front wall so that one of the microphones finds itself in the line of both the direct sound and the first reflection. In this case it would be the loudspeaker 7 and the microphone 27 (refer to the figure 8(top)).

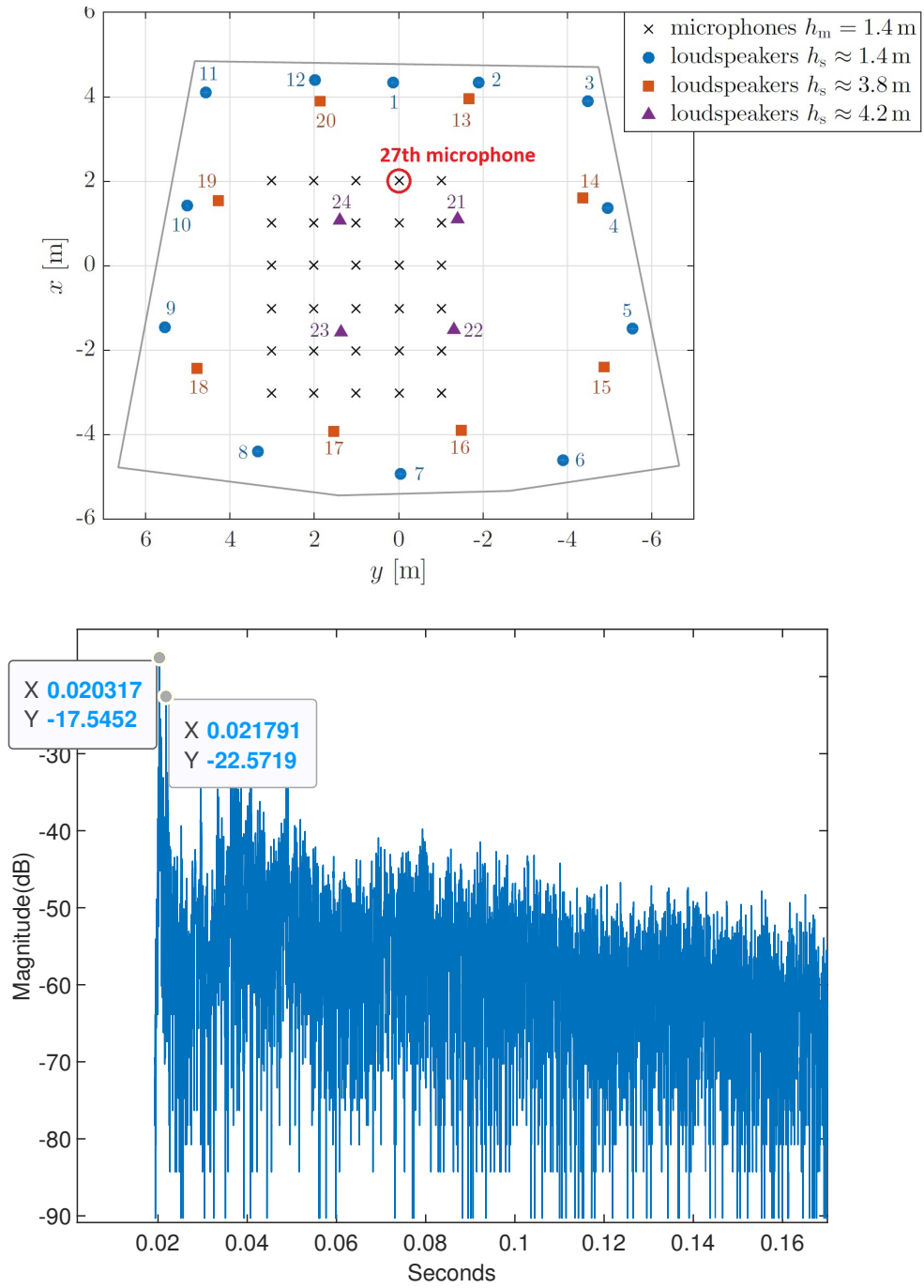


Figure 8 – Impulse response from 7th loudspeaker and 27th microphone with 2 first impulses marked (details in top figure representing floor plan of a CUBE)

It's possible to read the difference in dB directly from the plot and here it would be around 5 dB difference between the direct sound and from the front wall reflected sound. However, there are two important points needed to be taken into account. This 5 dB is the difference between two samples, but it's not correct to assume that the complete attenuation comes only from the reflection itself. Therefore, 25 samples above and below the samples seen in figure 8(bottom) were arithmetically averaged to get the more accurate difference. Additionally, there is also a distance drop in dB that has to be subtracted from this value.

The value that was calculated for the front wall attenuation was then -4.55 dB. However, overall attenuation of -1.55 dB was already set inside the "RoomEncoder" plug-in (figure 15). A front wall additional attenuation then has to be -3 dB in order to achieve the mentioned value. In the same way the attenuation of all other surfaces was calculated and the only thing that was left to do is to make sure that the ratio of the direct sound/early reflections and the reverberation level is like in the real-life.

### **2.2.3 Levels**

The last part of the acoustic calibration was to make sure that the reverberation is in the correct level ratio with the direct sound and the early reflections, otherwise the room would be too echoic or anechoic resulting in the deviation in the results and making the experiments implausible.

This part consisted of measuring the impulse response energy of the front loudspeaker and the microphone in the center between first 80 ms of the IR and the rest. After the value has been found out ( $C80 = 11.61$  dB), the levels in reaper need to be adjusted until the same value is measured from the acoustic simulation. For that, the track had a raw impulse generated in matlab with "listener position" in the middle of the room and only the 1st loudspeaker playing with the reverberation. The level of the FDN reverberation track (as seen in figure 14) was increased first in big steps and then in smaller until the  $C80$  measured from Reaper equaled the one measured from the recorded IR. The same principle was used for the track of BRIR reverberation and the acoustic simulation was finally fully calibrated. The resulting impulse responses of the level adjustment can be seen in the figures 10 and 11 and compared with the IR from the real measurement at the same position in figure 9.

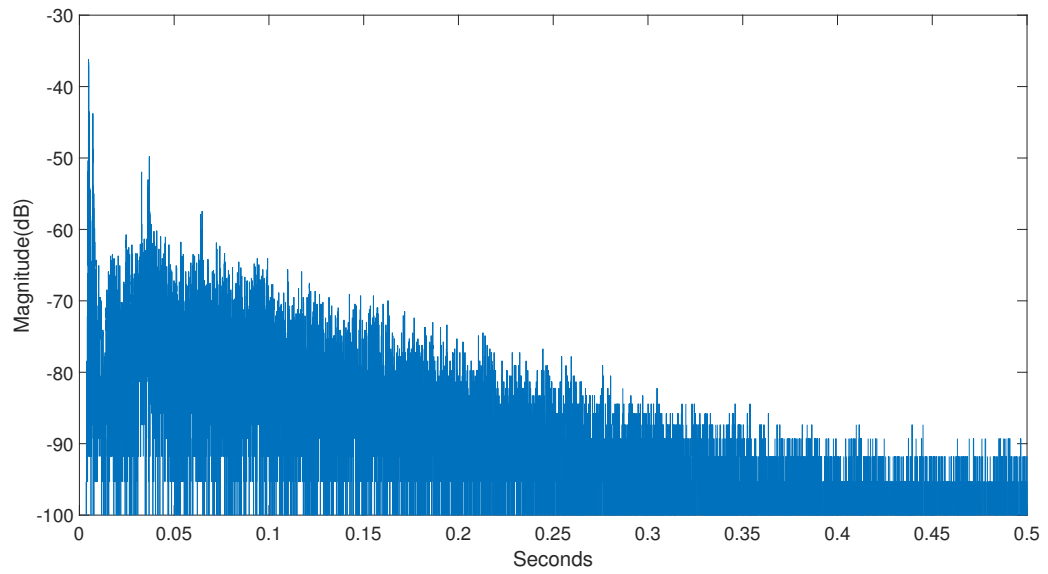


Figure 9 – Impulse response of the 17th microphone (central listening position) and 1st loudspeaker (real-life measuring position at the center of the CUBE).

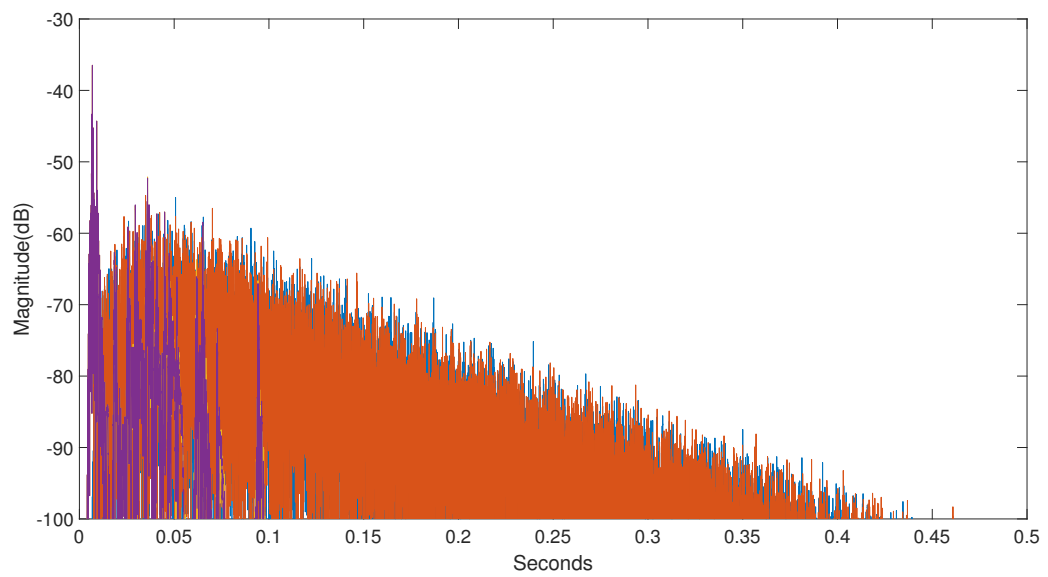


Figure 10 – Impulse response of FDN + RoomEncoder of 40 reflections (same simulated position as in the figure 9).

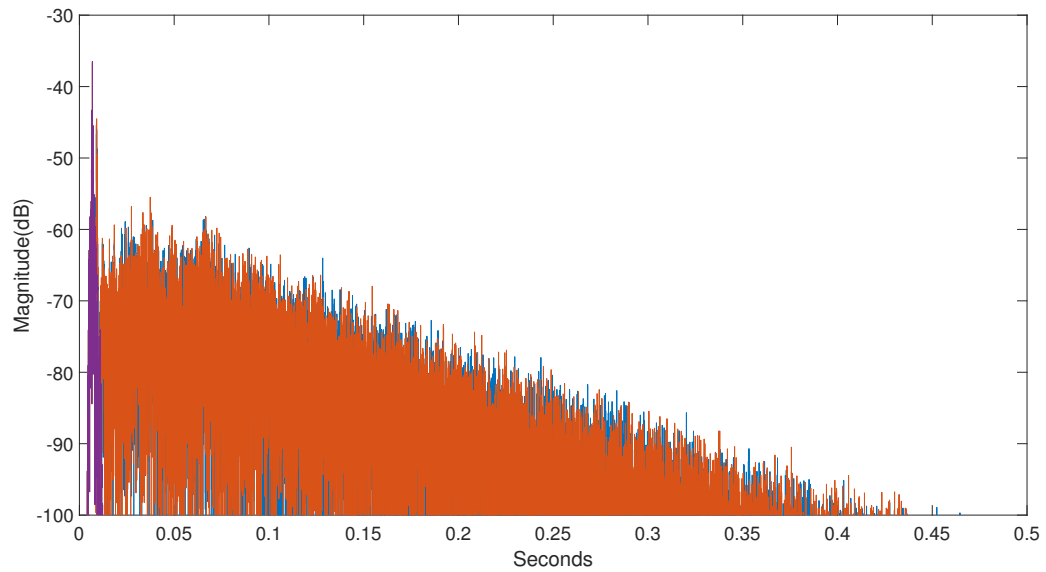


Figure 11 – Impulse response of BRIR + RoomEncoder of 0 reflections (same simulated position as in the figure 9).

The overview of all calculated or predefined acoustic parameters can be seen below in the list and the table 1:

1. "FDNReverb" track level: -16.22 dB;
2. "BRIRconv" track level: -32.8 dB;
3. All other tracks were adjusted to the level of 0 dB.



Plug-in	Parameter	Values
mcfx_convolver (tracks "LS1 to "LS24)	Convolver preset and IR	"Tannoy.conf" and "Tannoy.wav" (1 to 16ch) used for creating loudspeaker directivity pattern of a 3rd order (see Appendix A or Appendix B).
SceneRotator ( -    - )	Yaw, pitch, roll angle (figure 8(top))	Different for every of 24 tracks depending on which angle a virtual loudspeaker should be turned, based on its position in the room.
RoomEncoder ( -    - )	Reflection attenuation level (section 2.2.2)	Overall attenuation: -1.55 dB High-Shelf on 10 kHz: -0.4 dB
-    -	Additional attenuation (section 2.2.2)	Front wall: -3 dB Back wall: -0.99 dB Left wall: -1.64 dB Right wall: -1.64 dB Floor: -0.05 dB Ceiling: -0.1 dB
-    -	Other parameters	Room Size: 10 m x 11.5 m x 5 m (based on measurments from the section 2.1) Listener position: Generated in Unity based on where the participant is standing. Source position: Different for every loudspeaker position (see figure 8(top))
mcfx_delay24 (track "FDNReverb")	Delay (section 2.2.1)	14.49 ms
FDNReverb ( -    - )	General settings	Rev. Time: 0.5 s , Dry/Wet: 100% wet Fade In: 0.1 s , Room Size: 20 FDN Size: 64 (section 2.2.1)
-    -	Filter settings (section 2.2.1)	1st at 30 Hz with Q=0.9 and +6 dB/s 2nd at 10.068 kHz with Q=0.3 and -25.7 dB/s
MultiEQ ( -    - )	Equalizer filters (section 2.2.1)	High-pass(12 dB/oct): 20 Hz, Q=0.45 Peak: 59 Hz, Q=0.05, -1 dB Peak: 737 Hz, Q=0.75, -4.4 dB Peak: 1400 Hz, Q=0.9, +6.5 dB Peak: 1805 Hz, Q=1.65, -3.6 dB Low-pass(12 dB/oct): 6676 Hz, Q=1.45
mcfx_delay2 (track "BRIRconv")	Delay (section 2.2.1)	16.1 ms
mcfx_convolver2 ( -    - )	Convolver preset and IR	"BRIRCube.conf" and "CUBE.wav" used for creating a "wet" stereo diffuse reverberation, based on in CUBE dummy-head recorded impulse.
SceneRotator (Master track)	Yaw, pitch, roll angle	Based on head rotation of the participant.
BinauralDecoder ( -    - )	Order	3rd order with no "headphone equalization"

Table 1 – Overview of all acoustic model parameters

## 3 Experiment setup

In this section a detailed information-flow inside the experiment will be explained, as well as in which way the experiment tools were used.

### 3.1 General

The before-mentioned experiments in VR (see [30] and [8]) were programmed in the *Unity* real-time development platform, because of its versatility and possibilities. This experiment was made on the same platform, as many parts could be reused. From the audio-engineering project, the following parts were transferred:

1. Tutorial layout with a few added slides and an option to change between Serbian and English language, since more than a half of the participants found it easier to understand. The tutorial was this time transferred into the experiment room directly and the virtual tablet was removed, since the participants could evaluate the stimuli in a correct way even if the progression controls in the experiment were at the fixed frontal-placed screen.<sup>4</sup>
2. Rays and interaction of the hand controllers;
3. OSC messages: Head rotation and play/pause command;
4. Result exporting system. However, it needed completely other values for the export;

The *IEM Plug-in Suite* consists of many necessary plug-ins to simulate a room using ambisonics. For that, an audio workstation that can use VST type of plug-ins had to be used, thus the *Reaper* DAW was chosen, because of its ability to process 64 channels in one track and to read OSC messages. *OSC* (open sound control) is a protocol that runs over a network and can transfer messages on chosen IP ports. Two computers were connected with the network cable and the IP addresses were edited in order for the OSC messages to be successfully received.

The *HTC Vive* virtual reality set was provided by *IEM* and connected to computer running the *Unity* application and for the sound playback, the overear headphones *Sennheiser HD380 Pro* were connected to computer 2 running *Reaper*.

---

4. In the previous audio-engineering project [8] participants had to walk around the "safe area" and carry the virtual tablet on which were the controls for the experiment evaluation like the buttons "Next", "Back" and the slider for adjusting certain values based on the stimulus. This could've also been done by using one of functionalities of the hand controllers, which would include much less programming, but this way participants had a visual feedback of what they were doing and it was much easier to understand. Here, participants were almost always looking at the L-C-R triplet which was below the mentioned screen.

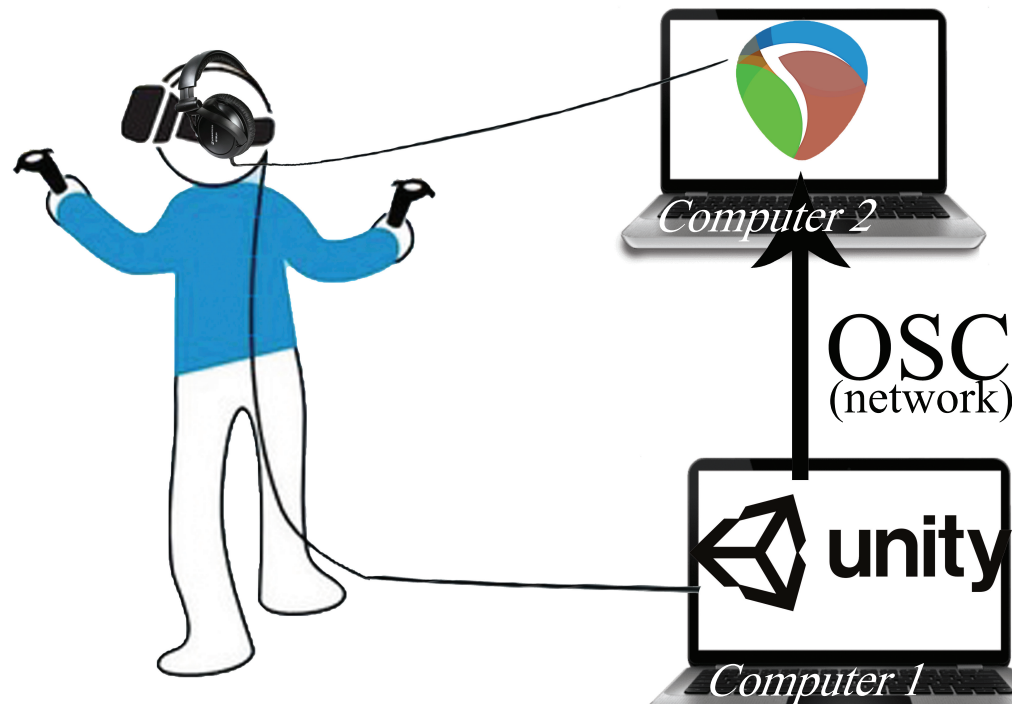


Figure 12 – Connection between Reaper and Unity

Processing of the information can chronologically be described as following (refer to the figure 12):

1. The user moves his/her head or hands wearing HTC equipment which is then registered over the HTC Vive interface. Positions in regards to the HTC base stations, rotation and commands are then translated for the *SteamVR* application which is necessary for the virtual reality to function.<sup>5</sup>
2. *SteamVR* then sends the information to the Unity game engine for handling everything regarding vision and graphics of the project.
3. Information about the sound changes is sent over the OSC protocol to the second computer handling the Reaper project.
4. Reaper then listens to the information from the first computer and tunes plug-ins and other options accordingly. Additionally, it was necessary to install the *SWS extension* [31], so that the specific time regions in the Reaper can be selected and played via OSC commands from the Unity application.

---

5. HTC Vive interface is a small externally powered electronic device which is the brain of the VR set and it exchanges information between the computer (connected via USB) and the headset. It also processes the video and sound received over the DisplayPort or HDMI to the headset.

5. Finally, the ambisonic surroundings are decoded for the headphones and the sound is played back creating the complete audiovisual experience (see figure 16 for a graphical representation).

## 3.2 Unity

Usually used for the programming and designing of video games (therefore so called video game engine), Unity is a piece of software that makes creating video simulations and games much easier. Because of that, game designers and developers didn't need to waste time building physics, UI systems or graphics shaders from scratch. It can be seen as a tool to build games on a computer, like artists use brushes and colors as tools to paint pictures. It even has built-in methods (the `Start()` being called on the first frame and the `Update()` on every frame), sound handling and much more of what a developer might need. This C# object-oriented editor, although amazing for programming and video, doesn't come with an audio system that could fulfill the needs of this project.

One of these built-in parts is the "XR Plug-in" which, when installed, provides a complete set of VR interaction scripts, 3D models and controls. By replacing a regular "Camera" with e.g. "CameraRig" from plug-ins folder, virtual reality can already be used (visible inside "Hierarchy" in figure 13).

In order for Unity to understand the OSC, certain definitions and scripts need to be imported [32]. The next task would be to separately add dummy objects to hold the OSC script for the every plug-in/control. Some of the new options then become available on the objects and every receiving port inside Reaper can have a corresponding object in the Unity application (see figure 16).

Two other important scripts are the "SendPositionOnUpdate" and the "SendRotationOnUpdate" and they will be used to control the listener position of the IEM plug-in "Room Encoder" and the rotation of the "Scene rotator" plug-in (see section 2.2 and figure 16). Basically, with OSC, Unity can manipulate Reaper completely and that gives us the freedom to use two computers which was necessary for this project. For example, in figure 13 can be seen that an object "OscHeadRotation" was created to control values of the above mentioned "Scene rotator" (how the sound will change when the participant rotates his/her head).

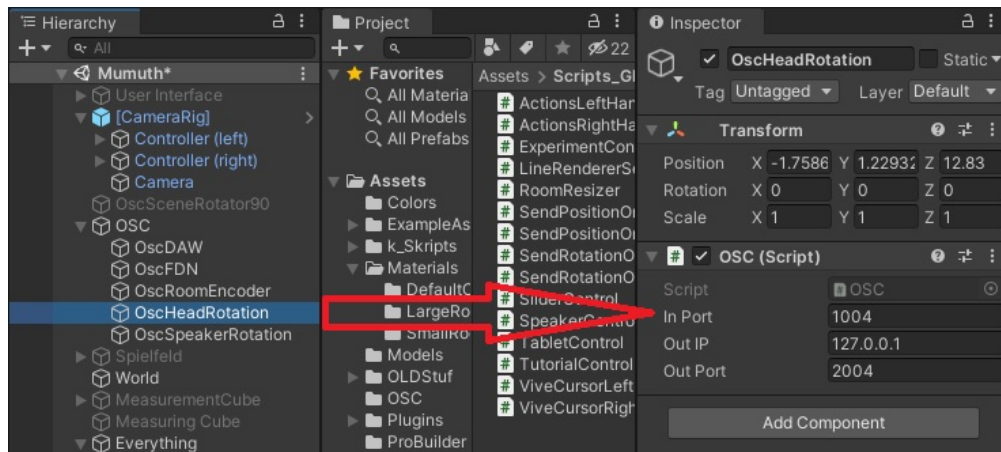


Figure 13 – Example of an implementation of the OSC protocol in Unity. 127.0.0.1 is the loopback IP address when using the OSC communication between two programs on the same computer.

### 3.3 Reaper

When an OSC message is received, next step is in the hands of Reaper. As the original real-life experiment employed 24 Tannoy loudspeakers, 24 tracks were added representing each of the loudspeakers. Three 24-channel source tracks were encoded from a professional lossless mono speech recording [26] into 3 different ambisonics orders (1st, 3rd, and 5th) with a panning on the 1st virtual loudspeaker (see the figure 8(top) or 22). Two more tracks were used to generate the FDN and BRIR reverberation. Additional tracks were added only for the playback of the tutorial speech, which was directly sent to the output, without going through the master track. The signal flow can then be explained as following (also graphically shown in the figure 16):

1. The 24-channel encoded source track of a lossless male speech [26] is the top track of the reaper project. It isn't sent to the master, but it is used to be the source for every loudspeaker playback track ("LS1" to "LS24") or reverberation ("FDNReverb" or "BRIR conv"). It holds the frontal speech signal decoded to the 24 loudspeakers of the CUBE in three different orders using max-rE weighting (1st, 3rd, and 5th after one another).
2. A "kronlachner" mcfx convolver plug-in [33] is used to make a convolution between the track that is being played and the specific impulse responses. Here it was used for each of the 24 loudspeaker tracks and it caused that the sources playing in a simulated room have a directivity based on IRs of the "Tannoy System 1200" loudspeaker [21] and are not omnidirectional anymore. This brings us to the next plug-in: "SceneRotator".
3. The "SceneRotator" is used to make a fixed rotation of the virtual loudspeaker based on the information provided in the measurement document [27]. Before the loudspeaker has a directivity, it's an omnidirectional sound source and no difference can be heard, regardless of which angle is set in the "SceneRotator". Yaw, pitch and roll rotation of every of the 24 loudspeaker tracks had to be adjusted separately.
4. The sound is then passed to the "RoomEncoder" which does most of the job of all plug-ins and it uses the most of the computer's resources. As seen in figure 15, it consists of the listener and source position information, the room dimensions and specific reflection attenuations. The loudspeaker that previously only had a directivity pattern and the rotation, can be moved around the room with correctly set dimensions. The room dimensions were in this case approximated to fit the volume and the dimensions of the rectangular version of the CUBE, because the plug-in doesn't support nonparallel surfaces yet. The only two parameters of the "RoomEncoder" plug-ins controlled by Unity were "Listener Position" (which is basically the listener's head position in VR) and the "Number of reflections" which will equal 40 for the FDN and 0 for the BRIR reverberation based on

the part of the experiment (as explained in the 2.2). Additionally, it should be mentioned that the option "Synchronize Room Settings" was useful and saved a lot of time setting up the project since every of the 24 tracks representing virtual loudspeakers should excite the same acoustic surrounding, only with the different "Source Position" parameters.

5. Every loudspeaker track output is then sent to the "Master" track consisting of another "SceneRotator" (for the rotation of the subject's head in the VR) and the "BinauralDecoder" plug-in. In the previous experiments [7, 34] it was concluded that 7th order of ambisonics (highest) makes no noticeable difference to the 3rd order for the diffuse reverberation. For the early reflections [35] even the lower order can be used. This plug-in uses a state-of-the-art magnitude-least squares approach to make a binaural rendering from a 3rd order ambisonics [5, 4], which is then played over the headphones.
6. The audio source is simultaneously being received by the reverberation tracks "FDNReverb" and "BRIR conv". Both tracks receive all of the source track channels to their first channel and they are delayed with "mcfx\_delay" correspondingly so that they play right amount of time after the direct sound from the "RoomEncoder".

**FDN reverberation** is tuned inside the "FDNReverb" plug-in and corrected with an equalizer ("MultiEQ" IEM plug-in) to represent the timbre of the real reverberation and sent to master track with an adjusted level, so that the C80 value of the virtual and real acoustics are the same at the central listening position (see section 2.2).

**BRIR reverberation** uses "mcfx convolver" to make a convolution between BRIR impulse response and audio playback from the "AudioSource" creating corresponding early reflections and diffuse reverberation. Because this track is ready for the stereo playback it's sent directly to the output also with a correct level to respect the measured C80 value, but without going through the master track.

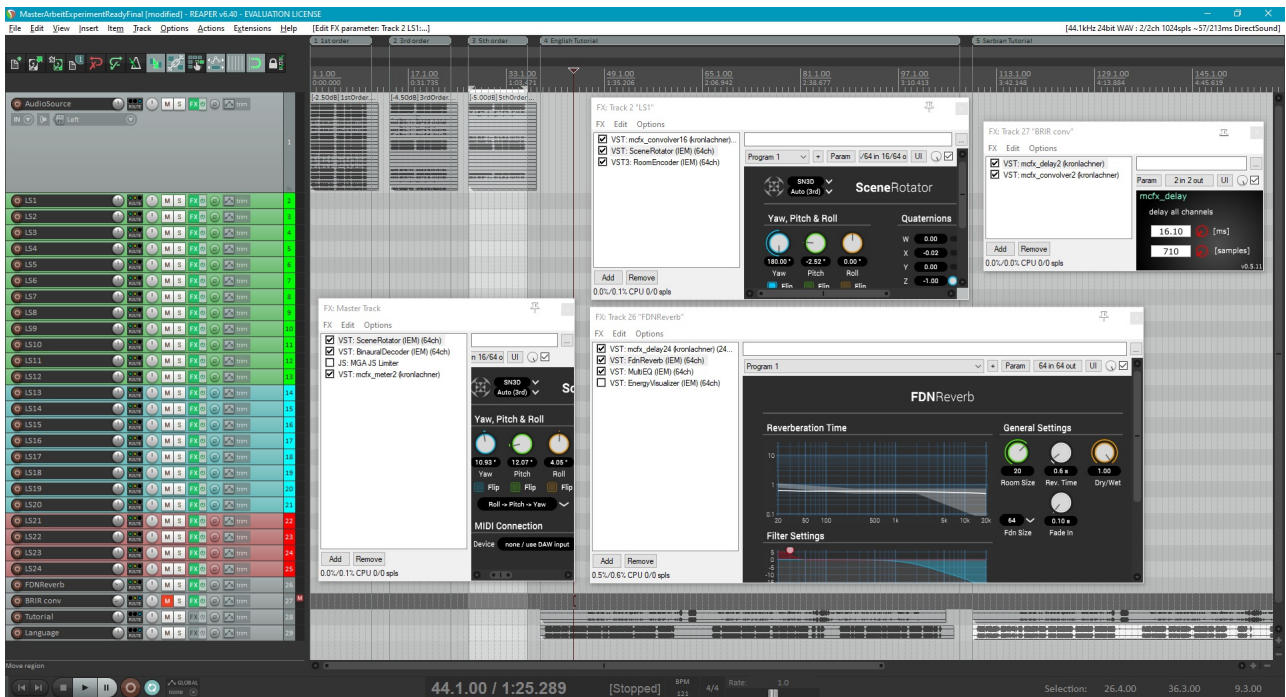


Figure 14 – Reaper project

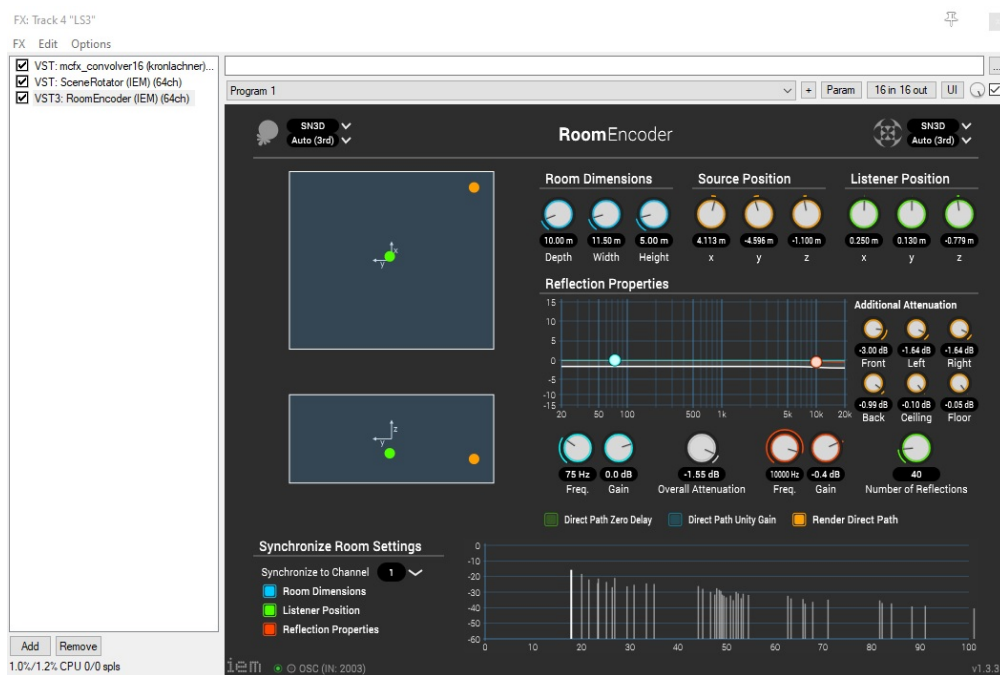
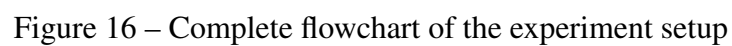


Figure 15 – "RoomEncoder" plug-in on the 3rd virtual loudspeaker





## 4 Experiment layout

In this section, it will be described how the experiments looked like and of which parts they consisted. Because of the already done experiments from the founding project [8], many participants already had experience with the virtual reality. However, this experiment differed in many ways from the previous one, so it needed a thorough introduction. The entire experiment consisted of 4 parts of total duration of around 48 minutes (see section 4.3 regarding duration):

1. Real-life introduction to the VR and what to expect (~3 minutes);
2. VR tutorial on the virtual CUBE projector screen about detailed use and functionality of the experiment (~5 minutes);
3. First part of VR experiment evaluating every direction and all of 3 ambisonics orders(1st, 3rd, and 5th) with the **FDN** reverberation (33 trials, ~20 minutes);
4. Second part of VR experiment evaluating every direction and all of 3 ambisonics orders(1st, 3rd, and 5th) with the **BRIR** reverberation (33 trials, ~20 minutes);

### 4.1 Personal introduction

In this part, participants were introduced to equipment and functionality of VR in case that they didn't already have experience with it. This experiment consisted of walking and teleportation inside the virtual CUBE which was confusing to some participants, so that additional explanations were needed before entering the virtual world. The evaluation of the experiment trials was also mentioned, but the thorough explanation was inside the tutorial, which will be explained in the next section. Afterwards, they were assisted in putting on headphones and VR equipment and the experiment application "Masterarbeit.exe" was executed, loading the tutorial after the Unity splash screen.

## 4.2 Tutorial

After the application is started, the participant sees the virtual CUBE and instructions on the screen (figure 17):

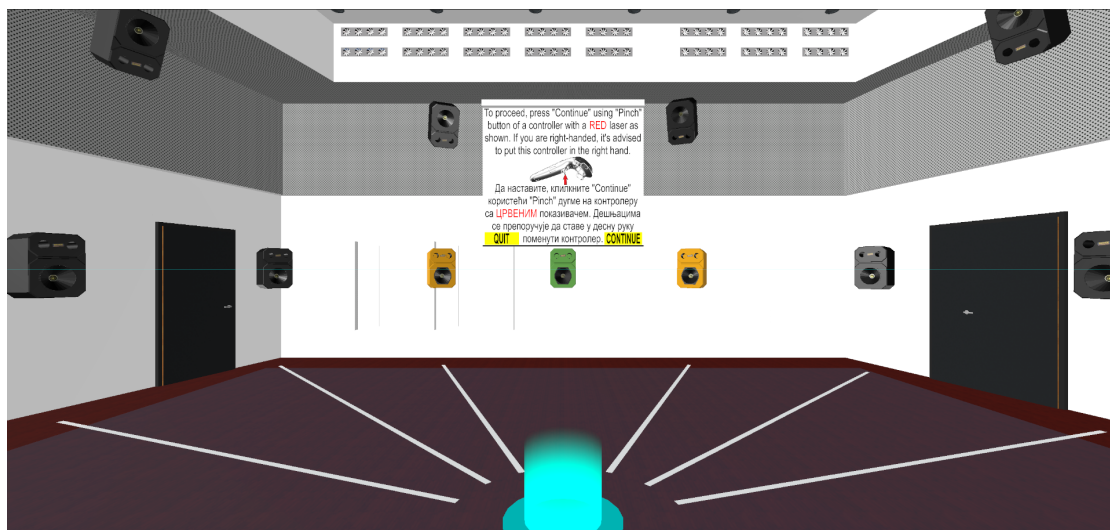


Figure 17 – CUBE in Unity (first slide of tutorial)

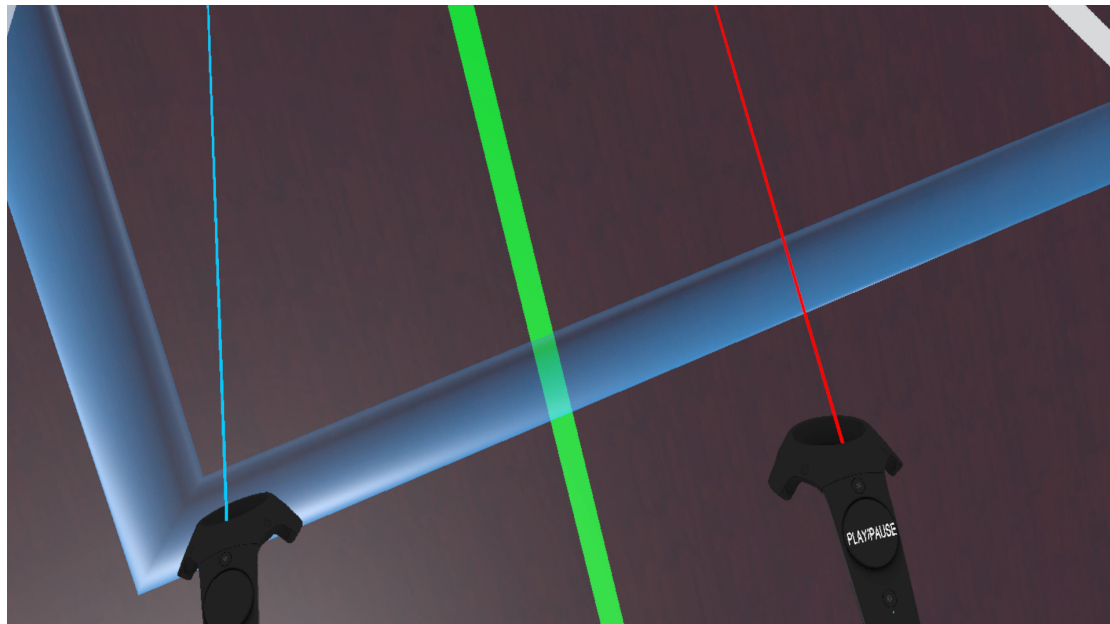


Figure 18 – Participant in VR checking position on the floor with controllers visible

In the very beginning, participants could see that their controllers project a laser beam. One of them is red, the other one blue and the controllers themselves were then being called accordingly "red" and "blue" controller throughout the tutorial and experiment (figure 18). When the lasers hit the interactable objects, they change color to cyan or yellow, depending on which controller has hit the target. The "red controller" was used for interaction with all yellow buttons visible in the experiment, thus being the "main controller" used for progression in the experiment (e.g. "Quit" and "Continue" on figure 17).

The "blue controller" on the other hand wasn't any less important, because it was used for teleportation around the virtual CUBE by clicking on the cyan surface of the floor. This surface was stretching over almost whole virtual CUBE, so that participants couldn't accidentally leave the room through a wall and into the void. The only way of doing the experiment correctly was to teleport over the whole area, because the VR setup and the boundaries of the real rooms used for the experiment could support a "play area" of maximum 2x3 meters, which would be around 1/20th of the actual CUBE.

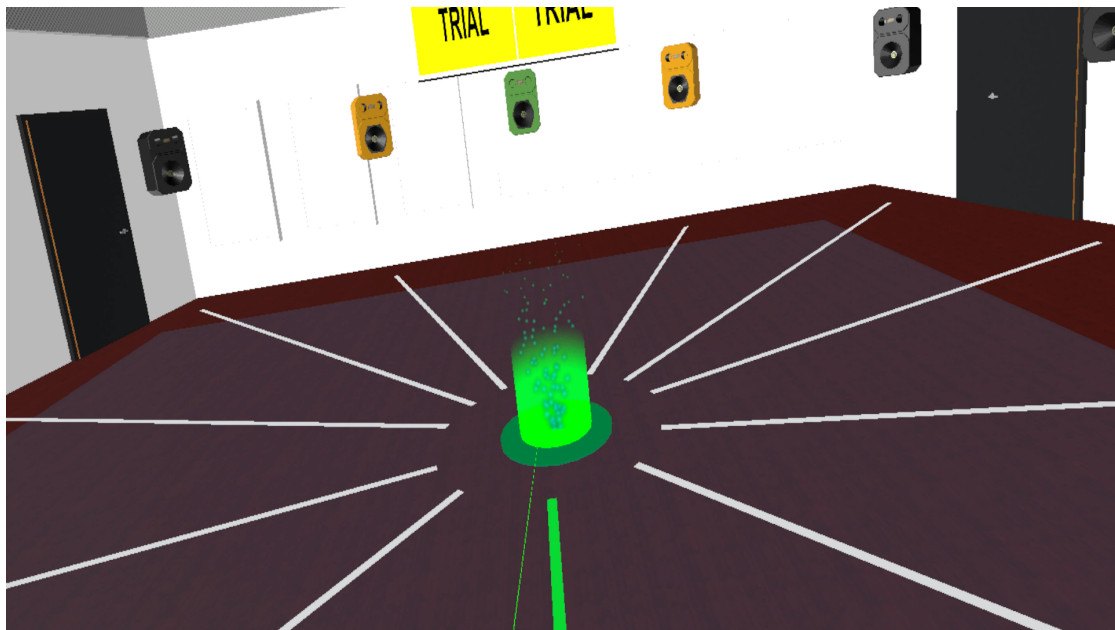


Figure 19 – Participant about to teleport to the center of the CUBE

Regardless if blue or red controller, when the laser hits an interactable object and changes the color, a participant can click the "pinch" button under index finger, which will turn the laser green and execute the corresponding command. By clicking the button "Continue" like in figure 17, the next tutorial slide is being selected and so on until

the experiment itself is reached.

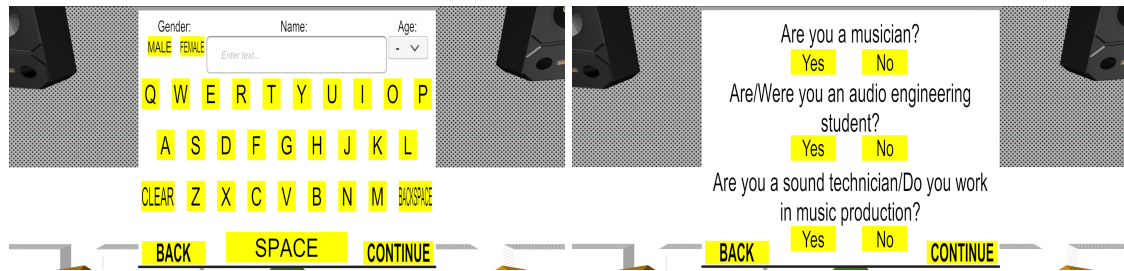


Figure 20 – 4th tutorial slide (information input)

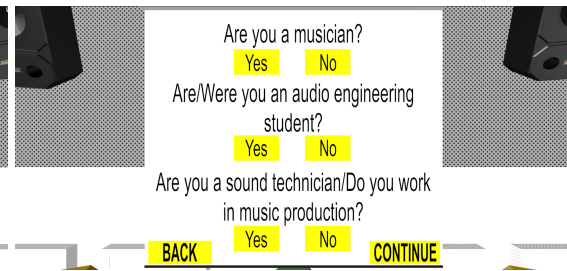


Figure 21 – 5th tutorial slide (questions)

After the participant has been introduced to how to progress through the tutorial slides, few slides will follow where the participant must input information like name, age, gender and experience with audio and music. This information will be useful later for the better analysis if the experience has any influence on the experiment results and if so, if it is going to influence results positively or negatively (figures 20 and 21). This information is necessary for the export of the result files and participants can't progress through slides if this information is not entered.

Afterwards, participants reach a part of the tutorial where they need to get used to tele-transportation by teleporting a few times around the virtual CUBE until they can progress to the next slide. Throughout this part they also get used to the "floor reference" in VR, which is a simple rectangle telling them where they are in regards to the real-life surroundings. Then they are watching a video demonstration on the screen regarding the evaluation procedure of the experiment. They listen to the speech explaining what exactly to do in the experiment that will follow, while listening to the stimuli from the demonstration at the same time (played correspondingly over track "Language" and "Tutorial" visible in figures 14 and 16).

Here they learned the meaning of the term "sweet spot" and how to evaluate the trials correctly. Participants can click the button "Repeat video" if they still didn't understand the experiment completely or proceed to the last slide, where they will be informed that the experiment will be started after clicking the last "Continue".

### 4.3 Experiment

When the experiment is entered, the first trial begins and the application randomly selects one of 11 directions towards each of the horizontal loudspeakers except for the front loudspeaker (marked "C" or "1" in figure 22). All directions except the first one (corresponding to the loudspeaker 1) are marked on the floor in the virtual CUBE like white lines and one of the directions, that the first trial is connected to, will change its color to green (can be seen in figure 19).

The reason why the first direction wasn't even marked is because the ambisonics sweet spot towards this loudspeaker can't be correctly evaluated. In order to exit the area of the sweet spot, which is this experiment's stimulus, a participant has to move outside the direction of the loudspeaker where the ambisonics has been weighted to (being the loudspeaker 1). In the case of evaluation of the first direction, a participant would have to move towards the loudspeaker 1, contradicting the experiment's procedure.

The experiment itself consisted of 2 parts. The *First part* consisted of 33 trials on evaluating sweet spot size of 3 different ambisonic orders, of which every order gets 11 trials, while using FDN reverberation (virtual version of the experiment in the reference [3]). The *Second part* consisted of the same amount of trials, but with BRIR reverberation. The overview of a trial index distribution can be seen in table 2.

Trial index	Direction	Amb. order	Exp. part	Trial index	Direction	Amb. order	Exp. part
[1, 2, 3]	2	[1, 3, 5]	FDN	[34, 35, 36]	2	[1, 3, 5]	BRIR
[4, 5, 6]	3	-    -	-    -	[37, 38, 39]	3	-    -	-    -
[7, 8, 9]	4	-    -	-    -	[40, 41, 42]	4	-    -	-    -
[10, 11, 12]	5	-    -	-    -	[43, 44, 45]	5	-    -	-    -
[13, 14, 15]	6	-    -	-    -	[46, 47, 48]	6	-    -	-    -
[16, 17, 18]	7	-    -	-    -	[49, 50, 51]	7	-    -	-    -
[19, 20, 21]	8	-    -	-    -	[52, 53, 54]	8	-    -	-    -
[22, 23, 24]	9	-    -	-    -	[55, 56, 57]	9	-    -	-    -
[25, 26, 27]	10	-    -	-    -	[58, 59, 60]	10	-    -	-    -
[28, 29, 30]	11	-    -	-    -	[61, 62, 63]	11	-    -	-    -
[31, 32, 33]	12	-    -	-    -	[64, 65, 66]	12	-    -	-    -

Table 2 – Overview of all experiment directions and belonging trials.

Let's assume that the first randomly selected trial index was the 17th trial (refer to direction diagram from figure 22). As the participant enters the experiment, 7th direction will light up in green and the 3rd ambisonics order from Reaper will start playing immediately. If the participant was not in the center of the room at the moment the trial was called out, a sentence on the screen will appear telling the participant to go back to the center of the room (situation would look exactly like in the figure 19). Afterwards, the participant would walk along the marked direction towards the corresponding loudspeaker until he/she hears that the sweet spot has been left.

The so called "leaving from the sweet spot" was thoroughly explained in the personal introduction (section 4.1) and the tutorial itself (section 4.2): The sweet spot is left when the sound is localized outside the L-C-R triplet or when the localization is collapsed into a single loudspeaker. The "L" and "R" (or the second and 11th) loudspeaker were then marked orange and the first loudspeaker green for a better recognition. Participants also had a possibility to pause the playback of the sound to ask a question or for additional explanation if needed. The same conditions mentioned above were taken directly from the real-life experiment from [3], which is the foundation of the VR version of the experiment and only so the results could be compared.

When the participant has moved to the position where he/she thinks to leave the sweet spot for the given direction, the "Next trial" button (figure 23) has to be clicked which will activate the following trial. The application will automatically take the previous trial's position of the participant's head in the virtual CUBE for the later export. Since the previous trial in this case was number 17, the next trial will be randomly chosen based on the remaining trials from that direction (16th and 18th trial index representing 1st and 5th order, respectively). The algorithm built for randomizing trial indices is shown in Appendix C.<sup>6</sup>

---

6. This code can be used only in Unity, since it uses a method from the built-in class "Random".



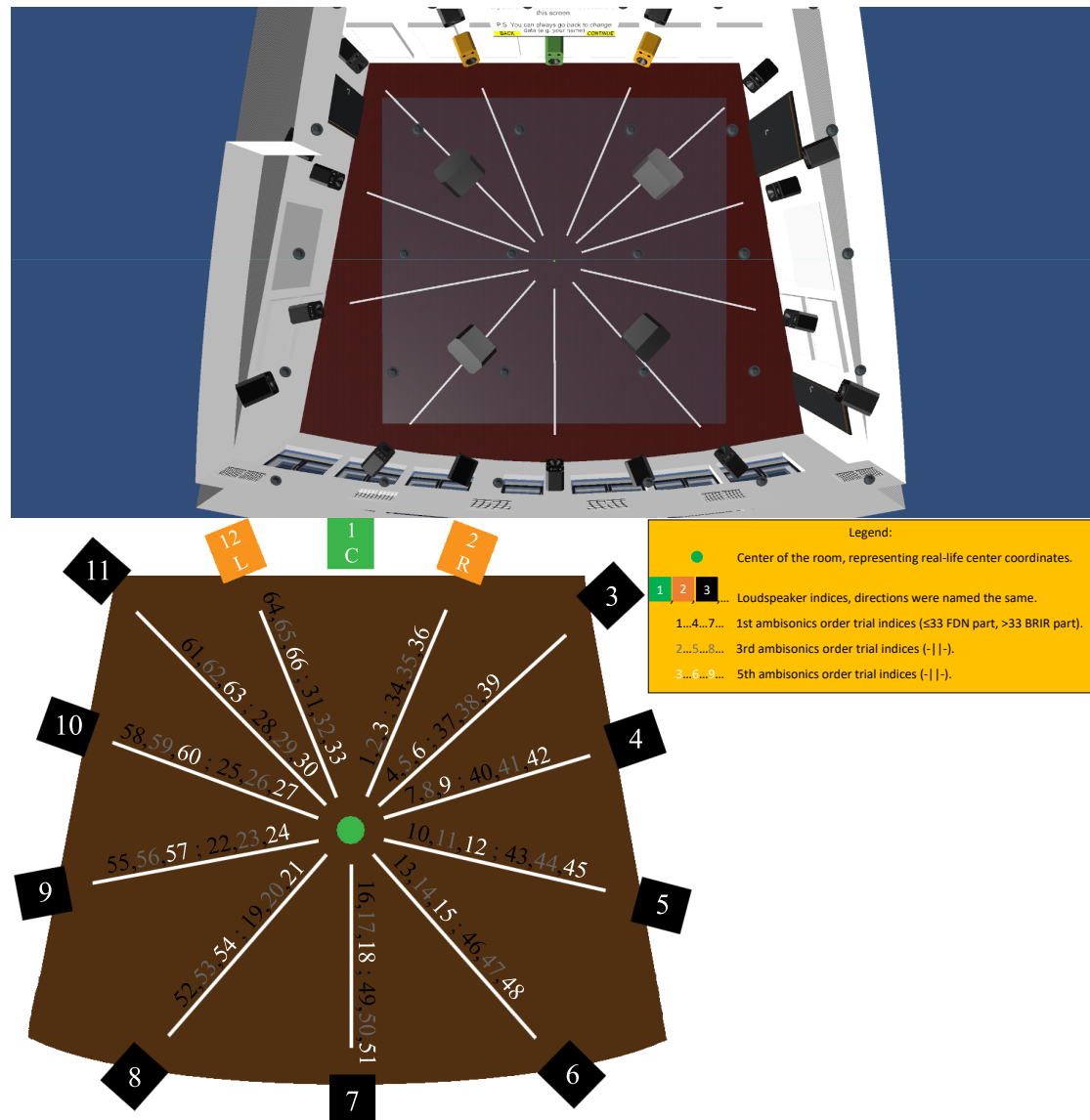


Figure 22 – Unity floor plan view of CUBE and its direction/trial diagram

After all of the trials (or ambisonics orders) from the 7th direction have been evaluated, the next direction clockwise of the *first part* trials is chosen, being the trials of the 8th direction (refer to figure 22 or table 1). After the next 3 trials have been evaluated, the next direction's trials are randomly selected and so on until the whole lap is completed. In this case, the lap is completed when the last trial of the 6th direction's trials is evaluated (13th, 14th or 15th). After all *first part* trials are evaluated, the experiment moves onto the *second part* trials in the way described in the flowchart on figure 16.



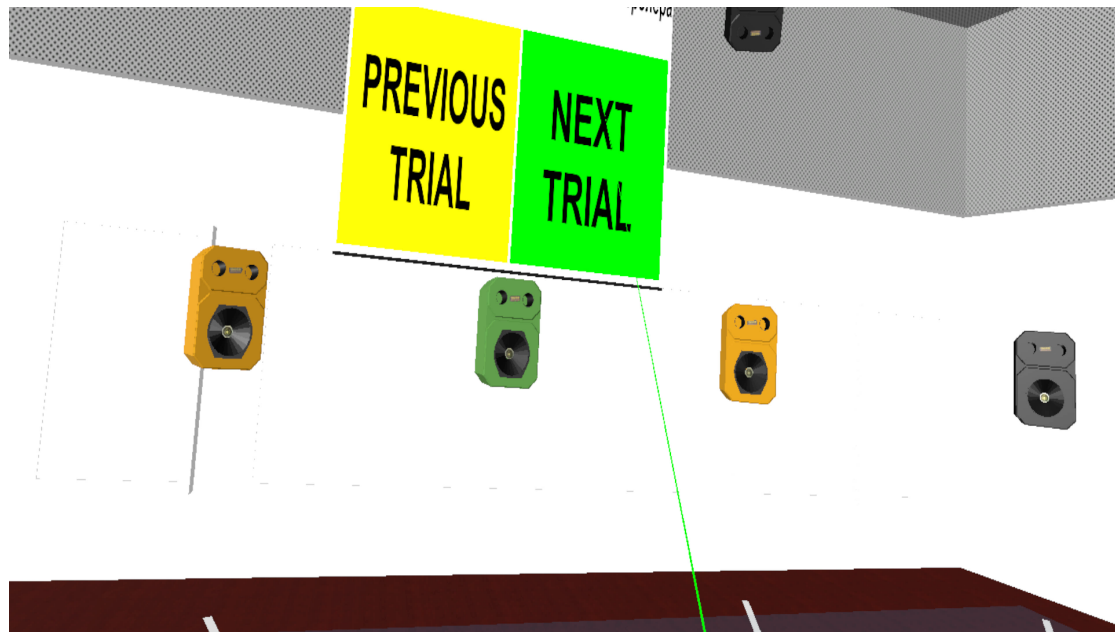


Figure 23 – Participant about to evaluate the stimuli from one of the trials.

During this transition, the track of the FDN reverberation is muted, the previously muted track of the BRIR reverberation in the *first part* is now unmuted, the "Number of reflections" from all loudspeaker "RoomEncoder" plug-ins is changed from 40 to 0 and the new lap of trials begins, which in this case will be one of the 6th direction *second part* trials (49th, 50th or 51st). After these trials are evaluated the lap continues clockwise just like in the *first part* of the experiment until all of the trials are finally evaluated.

The screen, previously showing only the "Previous trial" and "Next trial" buttons, changes its content to inform the participant that the experiment is finished. It also shows the button "Back" to go to the last trial and the button "Finish" which, when clicked, generates the results .txt file and closes the application (one of the participant's result files can be seen in Appendix D).

In the folder "Results" in the application's directory, the new file appears named after the participant's entered information, which would be in a format: 'Subject's Name'+ 'Gender'+ 'A (if audio engineer)' + 'S (if sound technician)' + 'M (if musician)' + 'Subject's age' + '.txt'. For the author, a 26 years old male participant with all positively answered questions on figure 21, a results file with a name: "DJORDJE PERINOVIC M ASM 26.txt" is generated.



## 5 Result evaluation

The experiments were performed by altogether 31 participants, of which 26 were male and 5 were female. 21 participants did the experiments in Serbian language in one of the rooms of the company "AAP-electronic" in Banja Luka, Republic of Srpska, Bosnia and Herzegovina. The other 10 have chosen the English language and these participants did the experiments in the *Regieplatz Petersgasse* of *IEM* at KUG in Graz, Austria. Altogether, 22 participants had experience in music or singing of which 17 were audio engineers or sound technicians. From the mentioned number, 11 participants were "IEM audio engineers", meaning that they studied or were studying the degree programme "Electrical and Audio Engineering" in Graz, Austria.

In addition to the generated results file, after every participant finishes the experiment, the "Times.txt" file is generated where is written how many seconds were needed to finish the experiment. Experiment participants needed on average 48 minutes to finish the experiment (as explained in detail in the section 4), where the longest experiment lasted 81 minutes and the shortest 29 minutes.

Five participants needed to pause the evaluation to get a rest from VR, three of which had a problem with dizziness during the experiment. What is interesting is that by coincidence, these three participants did the evaluation of the founding experiment from [8] as well and didn't have any difficulties although the exact same visual model equipment was used. The only difference regarding graphics was that in this project, a "Universal Render Pipeline" was used in Unity, which could maybe impact the VR experience in some way. On the other hand, this experiment was much longer in duration, but the problem could also lie in the VR 3D model which had many white surfaces and was much smaller in size. In addition, the mentioned participants reported occasionally evaluating trials with eyes closed which helped them to make less breaks during the experiment, but because of the lack of medical knowledge, this topic will not be discussed more detailed in this thesis.

Every results file had a title line "Experiment results:" and the numbers afterwards (like in Appendix D), which represented the trial evaluations chronologically. This means that, although randomly showed in experiment itself, they were later exported in the right order, so that the analysis can be done correctly. Every number shows how much (in meters) the participant was radially away from the center of the room.

The next step would be to analyze the results from the experiments and compare them to the real-life results from [3], once again, with a help of *Matlab*. Median values will be generated for all conditions based on different participant's experience and p-

values (based on Bonferroni-corrected Wilcoxon signed-rank tests) will additionally tell if there was a significant difference between ambisonics orders.<sup>7</sup>

Figure 24 is the graphical representation of the size of the evaluated sweet spot that was done in the original real-life experiment [3]. This diagram was the reference for the quick analysis of all generated VR results, so that they can be easily ranked, by analyzing the same type of diagrams generated for VR in the next sections. An important point was that the mean value from the symmetrical loudspeakers (e.g. "L" and "R" or 11 and 2) were taken as a final result, since the CUBE is a symmetrical room.

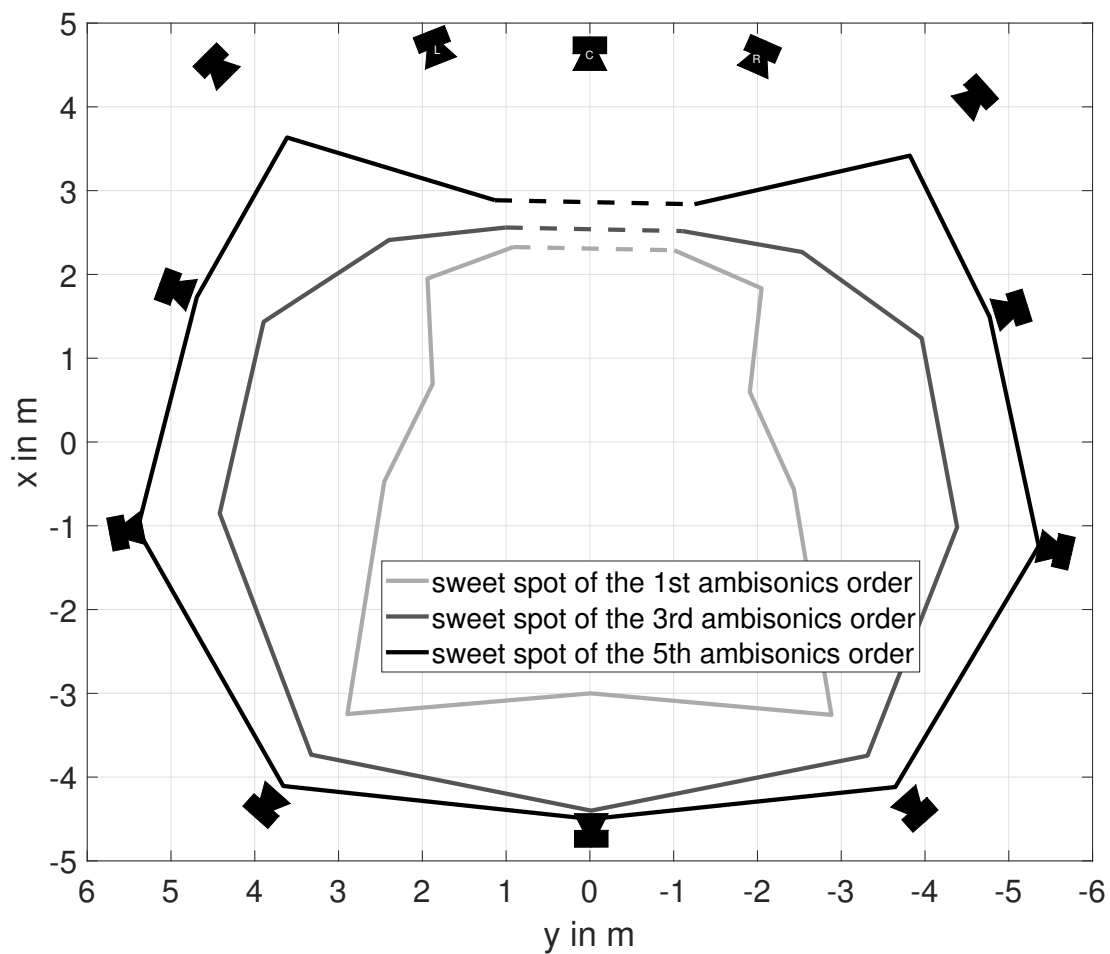


Figure 24 – Sweet spot diagram of the real-life experiment results.

7. A significant difference is fulfilled when the  $p < 0.05$

## 5.1 Comparison of participant groups

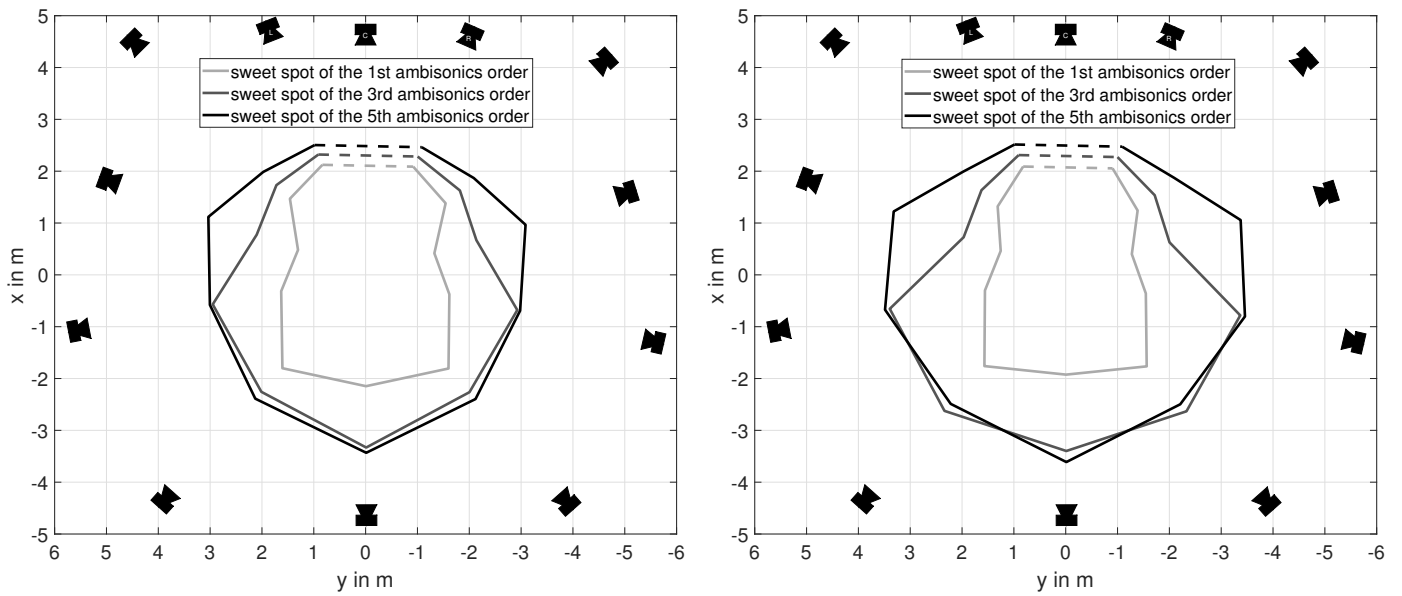


Figure 25 – First experiment part ("FDN") sweet spot diagrams of different participant groups.  
Left: *All participants* Right: *Musicians/Audio engineers*

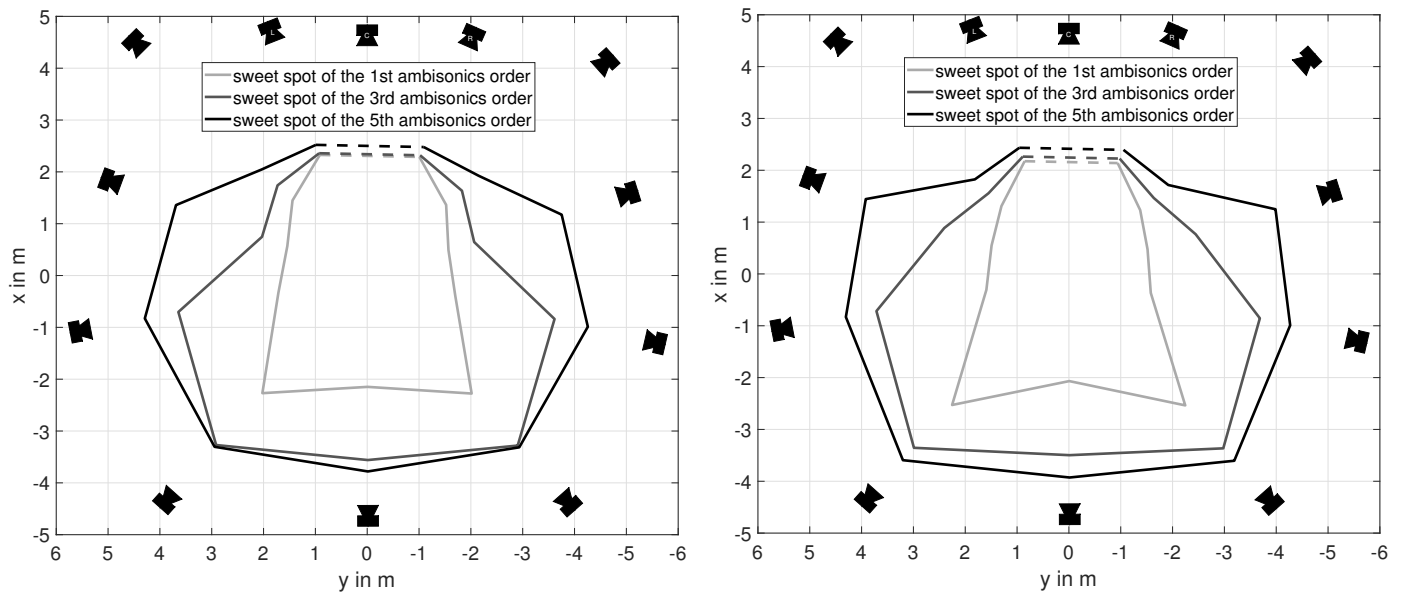


Figure 26 – First experiment part ("FDN") sweet spot diagrams of different participant groups.  
Left: *Audio engineers* Right: *IEM audio engineers*

First of all, a results group of *all* participant was analyzed giving a whole picture of how participants evaluated the stimuli (figure 25(left)). Because of different audio experience, it was expected that some results will correlate better to the real-life results than the others. From the very beginning of analysis, it could be roughly seen that the "FDN" sweet spots of any group were evaluated closer to the real-life results than the "BRIR" sweet spots, which will be also confirmed based on p-values in sections 5.2 and 5.3. Therefore, a "FDN" part results were used to rank specific groups more accurately (see figures 25 and 26).

Secondarily, the group "Musicians and audio engineers" was analyzed, which has shown an increase of the sweet spot area, but the results have still shown a big deviation from the real-life results (figure 25(right)). Next group was the "Audio engineers", consisting of all of the audio experienced participants both from Graz and Banja Luka, whose results were even more similar to the real-life results (figure 26(left)), but the most plausible results, because of the argument mentioned above, were generated by "IEM audio engineers" (figure 26(right)).

Although every evaluation was observed, it still can't be concluded if the group "IEM audio engineers" did more or less "correct" way of evaluating stimuli. It has been observed that "IEM audio engineers" walked a bigger distances until the stimuli were evaluated, which is also shown in the results, but if it corresponds to the real-life more accurately, one more point has to be mentioned.

Real-life results from the [3] were also done by the audio engineers that were used to doing experiments and majority of them was indeed from "IEM" institute, just like the mentioned group from this project. Although not the exact same group of participants, it can be concluded that they had a more trained hearing for doing the audio experiments. Since both of the mentioned groups consisted of participants with the similar audio experience and because the group of "IEM audio engineers" of this project had the highest similarity to the real-life experiments, it can be concluded that this tendency of the results is, in fact, correct.

To make a better decision on which participant groups to thoroughly compare to the real-life results, an additional check using p-values of every participant group and every order increase was analyzed (see table 3).

	$p, Fdn, Area_{13}$	$p, Fdn, Area_{35}$	$p, Fdn, Radius_{13}$	$p, Fdn, Radius_{35}$
All participants	$\ll 0.001$	0.0067	$\ll 0.001$	$\ll 0.001$
Musicians/Audio engineers	$\ll 0.001$	0.0023	$\ll 0.001$	$\ll 0.001$
Audio engineers	$\ll 0.001$	0.0374	$\ll 0.001$	$\ll 0.001$
IEM audio engineers	0.0029	0.0059	$\ll 0.001$	$\ll 0.001$

Table 3 – First experiment part p-values of different participant groups.

Based on "FDN" part p-values from table 3, every order increase of every participant group produced a significant difference. Therefore, these values showed no additional arguments on the group choices, which was the reason to additionally analyze p-values of the second experiment part ("BRIR" p-values in table 4).

	$p, Brir, Area_{13}$	$p, Brir, Area_{35}$	$p, Brir, Radius_{13}$	$p, Brir, Radius_{35}$
All participants	0.0048	0.1644	$\ll 0.001$	0.0449
Musicians/Audio engineers	0.0052	0.1998	$\ll 0.001$	0.0589
Audio engineers	0.0060	0.1238	$\ll 0.001$	0.0361
IEM audio engineers	0.0088	0.0088	$\ll 0.001$	$\ll 0.001$

Table 4 – Second experiment part p-values of different participant groups.

From table 4 can be concluded that certain values don't pass the signed rank test. Out of all experiment groups, the only group that passed the signed rank test for both "FDN" and "BRIR" part was the group "IEM audio engineers", which confirms previous assumptions.

Additionally, the group of *all* participant will be thoroughly analyzed, to uncover more arguments for discussion and comparison with the most plausible results.

## 5.2 All results

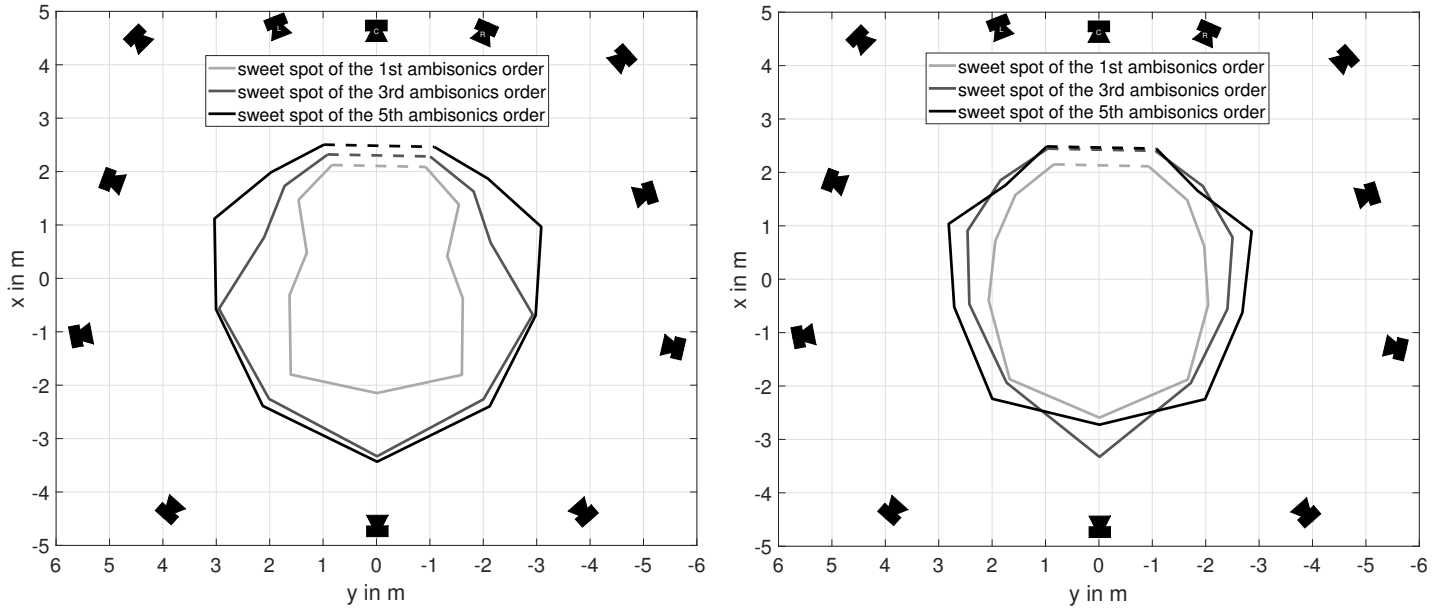


Figure 27 – Sweet spot diagrams of *all* participants.

Left: FDN results (first part)

Right: BRIR results (second part)

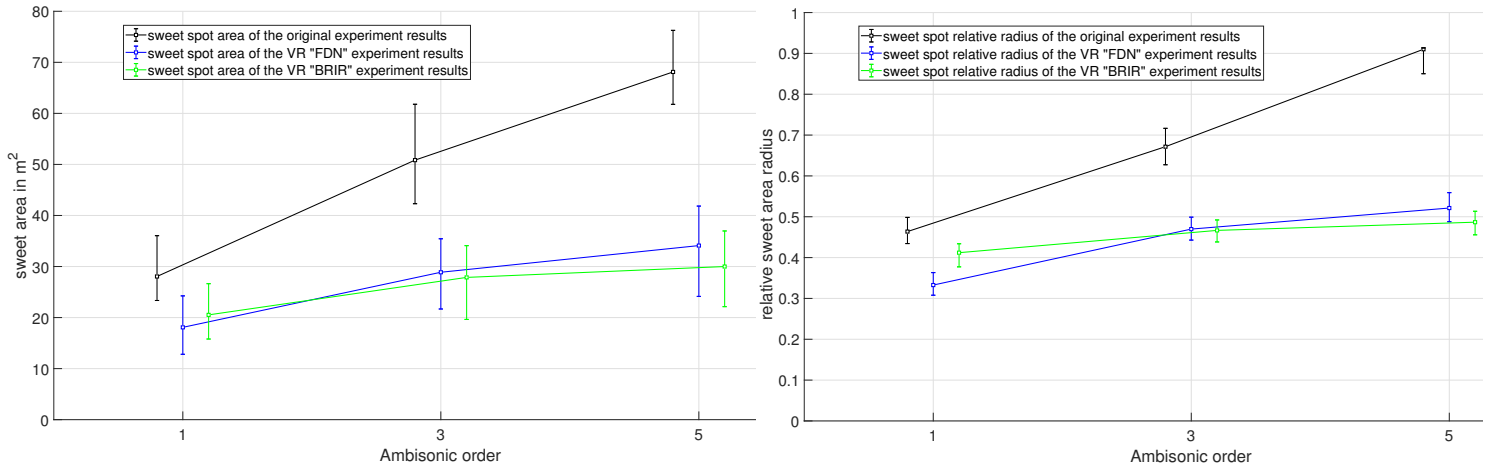


Figure 28 – Errorbar plots with medians of the group *all* participants and whiskers representing 25 and 75 percentile values.

Left: Sweet area size

Right: Relative radius size



The results from sweet spot area in  $\text{m}^2$  from figure 28(left) were calculated by using median values of every direction's radial distance and interpolating the value to the next direction's result to create a circle division. An area of each of the divisions was calculated and added in order to get the shown area value. The same principle was done for 25 and 75 percentile values (plot whiskers).

By analyzing the area and the radius of the real-life and VR results from figure 28 can be seen that real-life area is approximately 29% bigger for the 1st ambisonics order, 45% for the 3rd and 53% for the 5th ambisonics order, which was the first observation of the results:

With an increasing ambisonics order, the difference between sweet spot areas between the real-life and VR results also increases, which can also be seen when comparing results from the graphically shown sweet spots from figures 24 and 27.

The calculated Wilcoxon signed rank values were Bonferroni-corrected giving the most strict tests regarding the difference significance. When analyzing all of the 31 results, almost all of the p-values between the 1st and 3rd, and the 3rd and 5th order produced values much smaller than  $p=0.05$  (e.g.  $p, FDN, Area_{35} \ll 0.001$  or  $p, FDN, Radius_{13} \ll 0.001$ ).

Certain order increases for the evaluation of the "BRIR" produced rather borderline results (see table 4, first row). The increase from the 3rd to the 5th order of the relative radius (figure 28(right), green curve), created the value of  $p, BRIR, Radius_{35} = 0.045$ , which can also be confirmed from the plot. The same increase of the order for the area of the sweet spot gave the value of  $p, BRIR, Area_{35} = 0.16$ . From there can be concluded that regarding all of the results, participants found it difficult to differentiate between the 3rd and 5th order, which could be due to the limiting of the encoding and decoding. The BRIR results didn't pass the signed rank test for all orders, which still confirms that the FDN orders were easier to differentiate regardless of participant's audio experience.

This deviation from the real-life results was expected, however such a large difference for the 3rd and especially 5th order raised questions about the reliability of the participants and the quality of the acoustic simulations. Thus, it was decided to separate the results of different participant groups.

The following results only show results from the group "IEM audio engineers" (11 results), since this group has shown the highest similarity to the real-life results and will therefore be thoroughly analyzed. The reason and process of choosing this group of results was discussed in the section 5.1.

### 5.3 IEM audio engineers

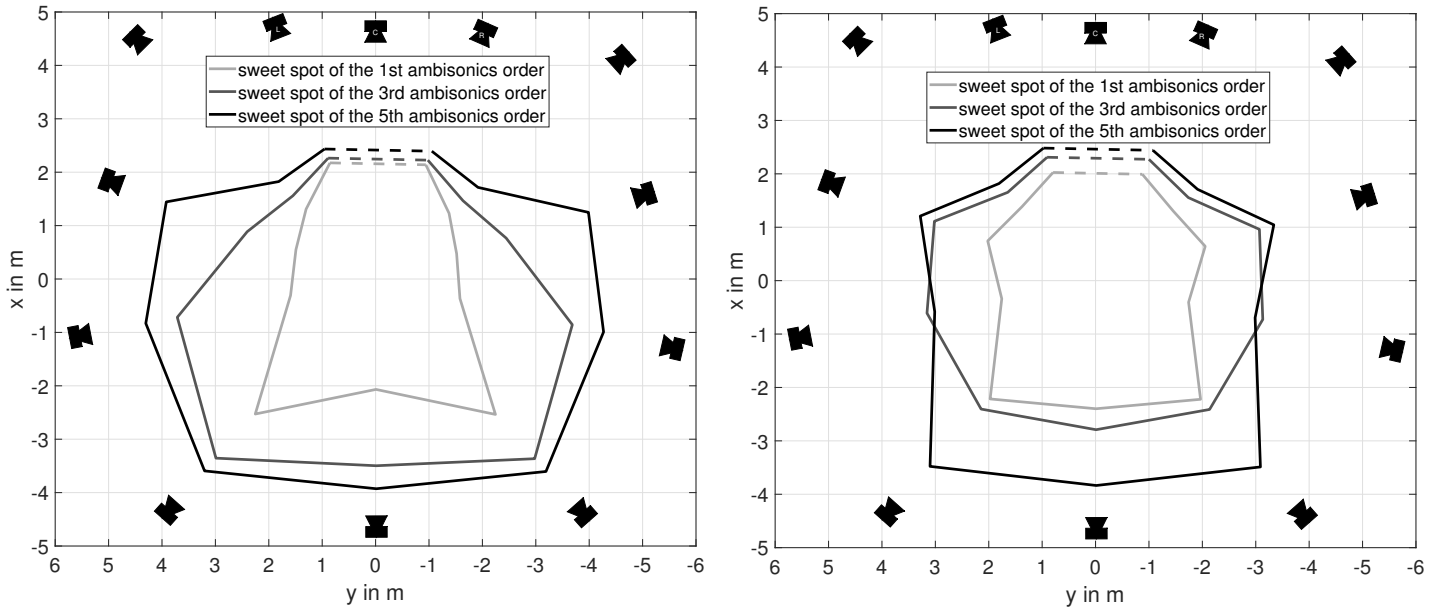


Figure 29 – Sweet spot diagrams of *IEM audio engineers*.

Left: FDN results (first part)

Right: BRIR results (second part)

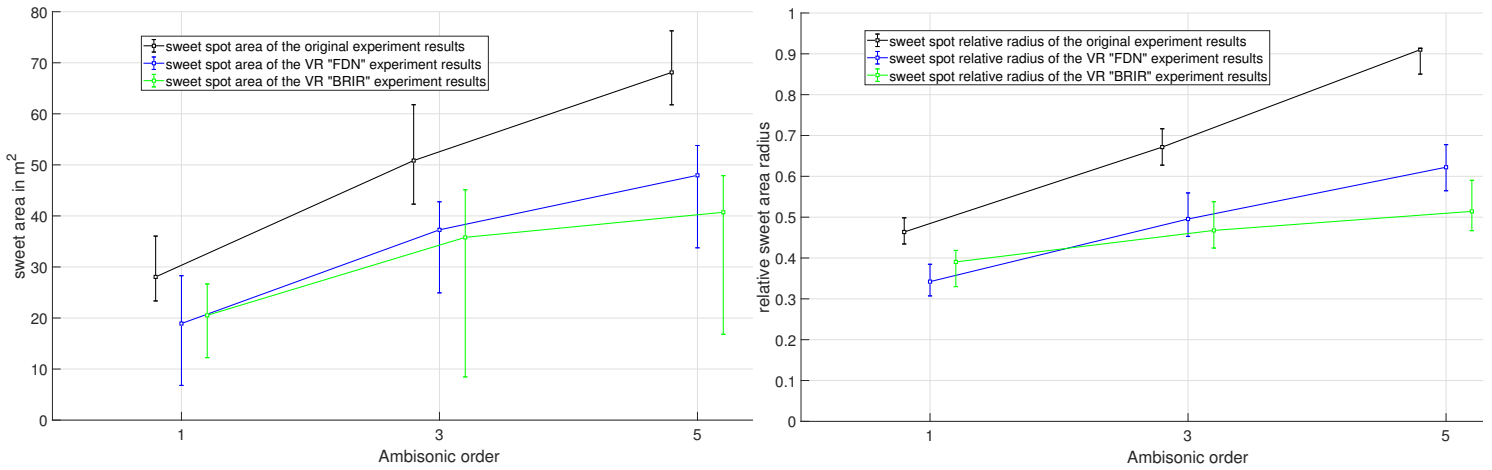


Figure 30 – Errorbar plots with medians of the group *IEM audio engineers* and whiskers representing 25 and 75 percentile values.

Left: Sweet area size

Right: Relative radius size

The first observation of these results can be seen when comparing figures 27 and 29. It is clear that the group "IEM audio engineers" produced results with much lesser deviations from the real-life results in figure 24. However, there was still a noticeable deviation from the real-life results that needs to be analyzed.

What can be seen from figure 30 is that the deviation of the median of the 1st order "FDN" stimuli was approximately 34% and of the "BRIR" stimuli 27%. The arithmetic middle of these values is around the value of the all participants from the last section, meaning that the evaluation of the 1st order could be clearly heard by all participants regardless of their audio experience. An interesting observation is that the whiskers of the plots from this group in figure 30 are larger than in the one from *all* participants in figure 28, especially for the absolute values (left plots). That means that 25 and 75 percentile values from *all* participants were closer to the mentioned median value than the group "IEM audio engineers", which indicates that the increasing number of participants evaluating the sweet spot of the first order ambisonics, makes the evaluation more accurate, regardless of participant's audio experience.

When evaluating 3rd order ambisonics, the deviation of the "FDN" stimuli is at the value of 26% and of the "BRIR" stimuli at the value of 29%. For 5th order ambisonics, "IEM audio engineers" evaluated "FDN" stimuli with a deviation of 29% and "BRIR" stimuli with a deviation of 40% to the real-life results.

The p-values from the results of "IEM audio engineers" were predominantly within values of  $p \ll 0.001$ , while the rest of the values was higher but still in the boundaries of  $p \leq 0.05$ . A decreased difference of "BRIR" results between the 3rd and 5th order of ambisonics can be seen from figure 30, which can be confirmed for the results of the group "*all* participants" as well. However, in this group, the values are not borderline. The highest p-values in this case are the  $p, BRIR, Area_{35} = 0.0088$  and  $p, BRIR, Area_{13} = 0.0088$ , meaning that the group "IEM audio engineers" did notice the difference between the orders more accurately (see table 4, fourth row).

An interesting observation regarding absolute (figure 30(left)) and relative (figure 30(right)) results is that in both cases of 1st order "FDN" stimuli (blue curve) they are evaluated with higher deviation to the real-life results (black curve) than the 1st order "BRIR" stimuli (green curve). The "FDN" reverberation, being more computer-resource demanding algorithm, is expected to have less deviation to the real-life results, which can be concluded for the 3rd and 5th ambisonics order (even for all results). Why the first order creates an unexpected result and makes a "BRIR" reverberation better in this case lies behind the way the "BRIR" reverberation works:

Since this type of a reverberation is, as already mentioned in previous sections, a convolution between the raw signal and the recorded IR at the center of the room, it makes it completely correct when the participants stand in the area around the center of the room. In the case of 1st order, participants tend not to walk a long distance from the center, because the sweet spot is here the smallest. They would mostly walk around an area of 2.5 m x 2.5 m around the center of the room and evaluate the stimuli there. That means that participants don't walk a long distance away from the area where the "BRIR" reverberation is completely correct. The "FDN" reverberation, although being synthetic, is also calibrated towards the center of the room. On the other hand, "BRIR" in this area gives a recorded variant of the early reflections and the diffuse reverberation, making it indisputably more correct than some algorithm that tries create a simplified model of the environment.

As soon as the ambisonics order increases, participants will tend to walk longer distances away from the center of the room where "BRIR" reverberation loses accuracy. What exactly happens is that from the "RoomEncoder" calculated direct signal for specific loudspeakers shifts more towards central recorded early reflections and diffuse reverberation. For evaluations with extreme distances from the central listening position, it could happen that the direct signal comes to the listener's ears at the moment when the early reflections or even diffuse reverberation are being heard, which would distort the sound localization. In the case of the first experiment part, the "RoomEncoder" generated direct signal shifts together with the early reflections giving the more plausible acoustic model for such distances. This observation can be also recognized from the sweet spot diagrams of the "BRIR" and "FDN" results (figures 27 and 29).

When comparing the VR "FDN" results of the "IEM audio engineers" from figure 29(left) with the real-life results from figure 24 an interesting and maybe the most important observation of this project can be seen:

It appears that the size of the sweet spot for the 5th ambisonics order of the VR "FDN" results has approximately the same area as the 3rd order of the real-life results. When analyzing the "FDN" evaluations of the area and the relative radius of the 5th ambisonics order in figure 30, it can be seen that their value deviates from the real-life 3rd ambisonics order only about 8%, confirming the already mentioned observation.

There are many hypotheses why this happens. As mentioned in sections 2.2 and 3.3, there were many parts of the acoustic model that were limited to the 3rd order because of the previous researches in this area [6] [7] in order to reduce the usage of the computer resources, such as:

- ◆ 3rd order of the "Tannoy System 1200" directivity pattern in every loudspeaker track in the plug-in "mcfx\_conv" and
- ◆ 3rd order of the binaural decoding.

When comparing the directivity patterns from the Appendix A and Appendix B, it can be observed that with an increasing order for the frequencies as of 2 kHz, the directivity of the loudspeaker narrows to between  $\pm 30$  degrees. The mentioned 10th order of directivity is approximately the one of the real-life "Tannoy Sytem 1200" loudspeakers, which were used in the original experiments. The difference in this project increases its significance, because every of the 24 virtual loudspeakers receives the directivity as in Appendix A, which could drastically impact the acoustic model that participants evaluate.

When increasing the loudspeaker directivity order, the virtual loudspeaker tracks will automatically have more channels, which means that every "SceneRotator" and "RoomEncoder" plug-in will have more calculations to perform.

The resolution of decoding is also an important part in the acoustic simulation. Binaural decoding, needed to bring ambisonics surroundings to the participant's headphones was also of the order 3. The increase of this order would cause more use of the computer's CPU, which also needs to be considered.

An important feedback from some participants was about a large source width of the direct sound, even when listening to the 5th-order playback. This increased source width could be caused by the limitation to 3rd order binaural decoding.

## 6 Conclusion and further development

This project's idea was created after the founding experiment of the audio-engineering project [8]. An idea was to compare the VR ambisonics to the real-life ambisonics by replicating the real-life experiment inside VR and with simulated ambisonics. A founding project focused more on plausibility of the audio-visual correspondence, while this project focused on the plausibility of the simulated ambisonics audio.

A visual and acoustic model had to be created to represent the real-life environment of the actual experiment. The visual model of the CUBE was created separately from the "Tannoy System 1200" loudspeaker in order to increase visual versatility inside the Unity project. The acoustic model also consisted of two parts, both using the frontal encoded 24-channel ambisonics track as source.

The first acoustic model consisted of the direct sound and 40 early reflection generated by the "RoomEncoder" plug-in and the diffuse reverberation generated by the "FDN" algorithm. The second model consisted of the same direct sound generated by the "RoomEncoder" plug-in with the early reflections turned off. The early reflections, together with the diffuse reverberation, were generated by convolving the source track with the central recorded dummy-head IR ("BRIR" reverberation), creating a less accurate acoustic model regarding complete CUBE surface, but completely correct acoustic model at the central listening position. First acoustic model was, on the other hand, more demanding for the computer resources, but it was yet to be confirmed if any acoustic model corresponds to the real-life and if so, which corresponds better.

The experiment itself consisted of evaluating the sweet spot size inside the virtual CUBE, by hearing the first acoustic model ("FDN") and evaluating every of the 11 required directions for the randomly chosen ambisonics order (1st, 3rd, or 5th). The participant would start from the center of the room and walk towards one of the marked direction's loudspeaker until the L-C-R triplet can't be localized anymore, or the localization concentrates on the one of the other loudspeakers. The same principle will be then repeated for remaining two ambisonics orders and then again for all of the following directions (in the clockwise order). Afterwards, the participant would do the same for the second part of the experiment ("BRIR") after which the results will be generated representing radial distance from the center for every of the trials.

Then, the results were gathered and separated into many groups to find the most plausible evaluations, which were then used to analyze and discuss both the first model's ("FDN") and second model's ("BRIR") similarity to the real-life ambisonics environment.

After the analysis of the results it had been concluded that the group "IEM audio engineers" evaluated the results in the most correct way and these results were taken as a reference for comparison with the real-life results. Although the ambisonics orders have been successfully differentiated between each other (based on p-values), the results have indisputably shown that the sweet spot area could be at most the size of the 3rd order ambisonics sweet spot from the real-life experiments.

As already explained in the previous section, this outcome could have been caused by the limiting of the overall ambisonics order of the acoustic model (section 2.2), in order to reduce computer-resource usage and still respect previous research on the optimum ambisonics order. These experiments used just a single loudspeaker and it seems that for more virtual phantom-sound-sources the mentioned order is not sufficient. An obvious improvement of this VR project would be then to increase the ambisonics order of the encoding and the decoding to hopefully get the results that don't deviate less from the real-life results.

However, increasing the order for just one step (from 3rd order to 4th), will give a drastic difference in the CPU usage, since every of the plug-ins will receive more tracks, do more calculations and need more time to calculate the correct acoustic model. This acoustic model used on average 65% of the Intel Core i7-4700HQ and 90% of the Intel Core i5-4570T, which can be used as a reference to know which processor of the newer generations could support a higher order of encoding/decoding. A good idea for the improved project would be to simply supply a PC with the strongest possible CPU and to compare the results of the highest ambisonics encoding/decoding to confirm if this was indeed the reason for the deviation of the VR results. If it confirms the hypothesis, then the order can be gradually decreased until the deviation from the real-life results becomes significant, finding a new "optimal ambisonics order". One idea could be also to render the direct sound with the highest possible order and reduce the order for the early reflections based on the reference [34].

The "BRIR" was an experimental reverberation to confirm if the fact that the diffuse reverberation and early reflections of the center of the CUBE could be applied to other positions inside the room. However, the results have confirmed that "BRIR" reverberation in this case yielded smaller sweet spots. Although more accurate, the "FDN" reverberation still needs improvement. In this case, direct sound and the early reflections were generated by 24 "RoomEncoders", which increases a CPU usage for at least 20% on the mentioned CPUs. The current version of this plug-in doesn't support curved or sloping walls, as they exist in the real-life CUBE. To confirm this hypothesis, an idea could be to do the real-life experiment in the room with completely parallel walls, which would

include installment of the complete ambisonics system inside the room. Afterwards, the new room can be simulated visually and acoustically to create an accompanying VR experiment and the early reflections would now be mathematically completely correct. Another solution would be to update the "RoomEncoder" plug-in which could support a generation of the early reflections generated from these types of walls.

One important point that has to be mentioned and wasn't taken into account in this project is the shifting of the diffuse reverberation with a participant's movement. Let's take a look at the first part of the experiment ("FDN" reverberation), since the shifting of the early reflections is already calculated; When a participant moves away from the center, where the check of the acoustic parameters was done (section 2.2), the direct sound and early reflections are automatically shifted to correspond with the distance from the each sound source. The "FDN" reverberation was on the other hand delayed for the fixed amount of time. To accurately adjust the "FDN" reverberation, it should receive every loudspeaker track with a delay from the already calculated direct sound from the "RoomEncoders", however without the level change. This could be done through adding of 24 separate source tracks (instead of only one 24 channel source track) and an "mcfx\_delay24" where the delay of every of the input tracks would be delayed depending on the position the participant is standing on.

This same fact can be applied to the "BRIR" reverberation, but the problem here is that "BRIR" reverberation is not versatile. It can't be separated to create the early reflections and the diffuse reverberation on two separate tracks. In the case of the first part of the experiment, the early reflections inside every of the "RoomEncoder" plug-ins move together with the calculated direct sound, but also move between each-other to create a corresponding reflections simulation of the room, regardless of where the participant is standing. In the case of the second part of the experiment, only the delay of the direct sound was shifted together with the participant's position, which is the task of the "RoomEncoder", but here the computational resources were spared because it didn't generate early reflections as in the first part. In this case, the early reflections were fixed between each other and also to the direct sound and the diffuse reverberation.

This can be assumed to be the reason for the higher deviation of the 3rd and 5th order results, because with the higher distance to the loudspeakers, the direct sound from the "RoomEncoder" plug-ins would simply blend into the diffuse reverberation of the "BRIR" which would distort the localization of the direct sound completely. That's the reason why as soon as the participant moves from the central position in the CUBE, the direct sound of the virtual L-C-R triplet becomes harder to distinguish from the reverberation and degrades the acoustic simulation. This observation was confirmed by some participants personally as soon as they finished the experiment. This outcome could be



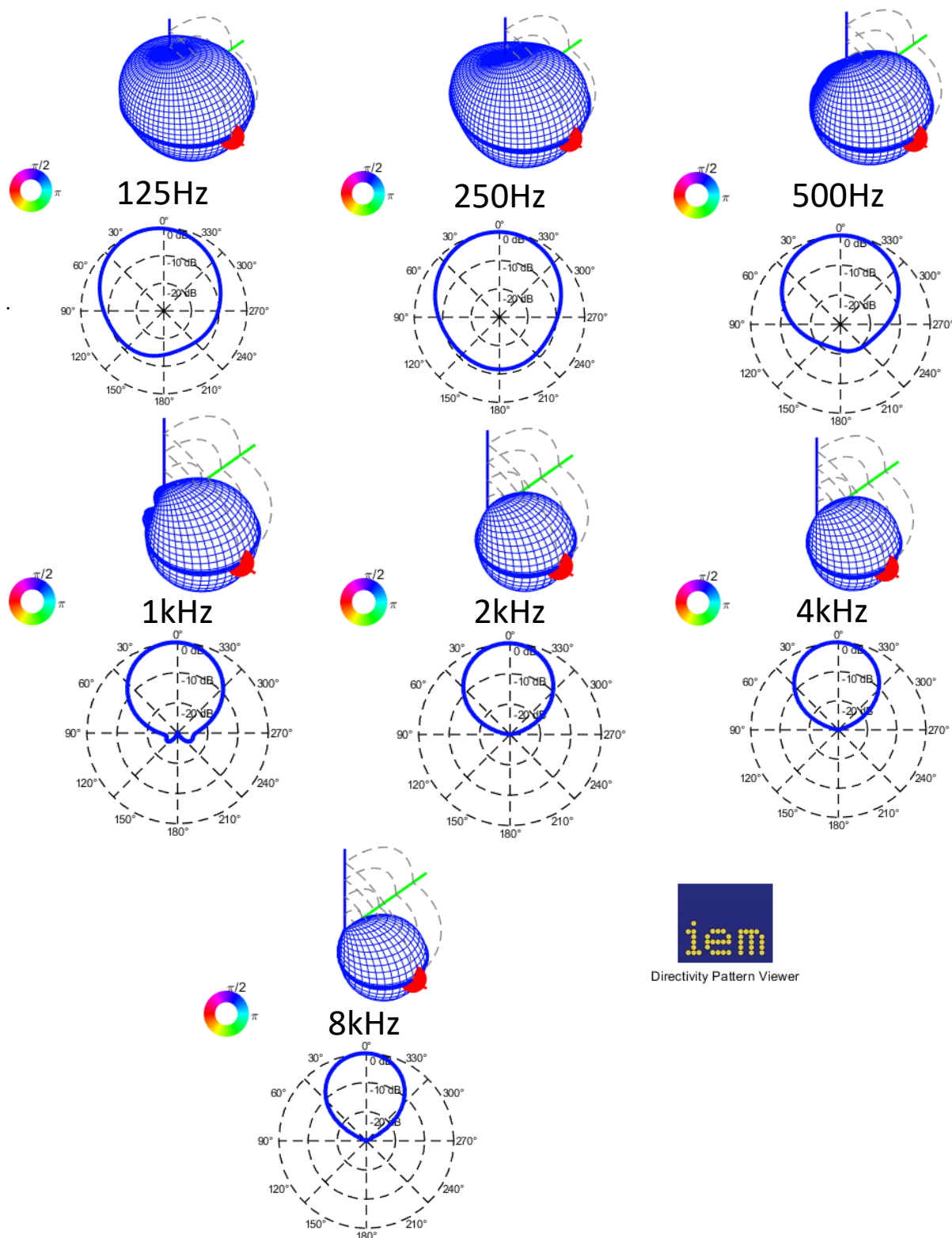
improved by shifting the "BRIR" reverberation late enough in a similar way as described above in case of the FDN.

In comparison to the previous project, the amount of time for each participant to finish the experiment was almost doubled. On the other hand, most of the participants found the experiment itself interesting regardless of how long it took to evaluate it. Participants also found this experiment much easier to understand and didn't have problems evaluating the stimuli based on their perception. One of the participants reported that the experiment could be also done with a different headphones to see if it would influence the results.

All in all, this type of experiment is fun for the most people and it can be confirmed by the fact that for both this and the previous project, we were able to gather at least 30 participants, with even more of them being interested in doing it, but they had to be rejected because of limited time. The only obstacle here is that this experiment, because of the complicated setup of equipment and use of VR, has still to be done at a certain location that provides the necessary equipment. This project also facilitates and supports a further development of VR experiments at *IEM*, by the optimization of the C# codes inside Unity scripts and the CUBE 3D model for the later use in future research.

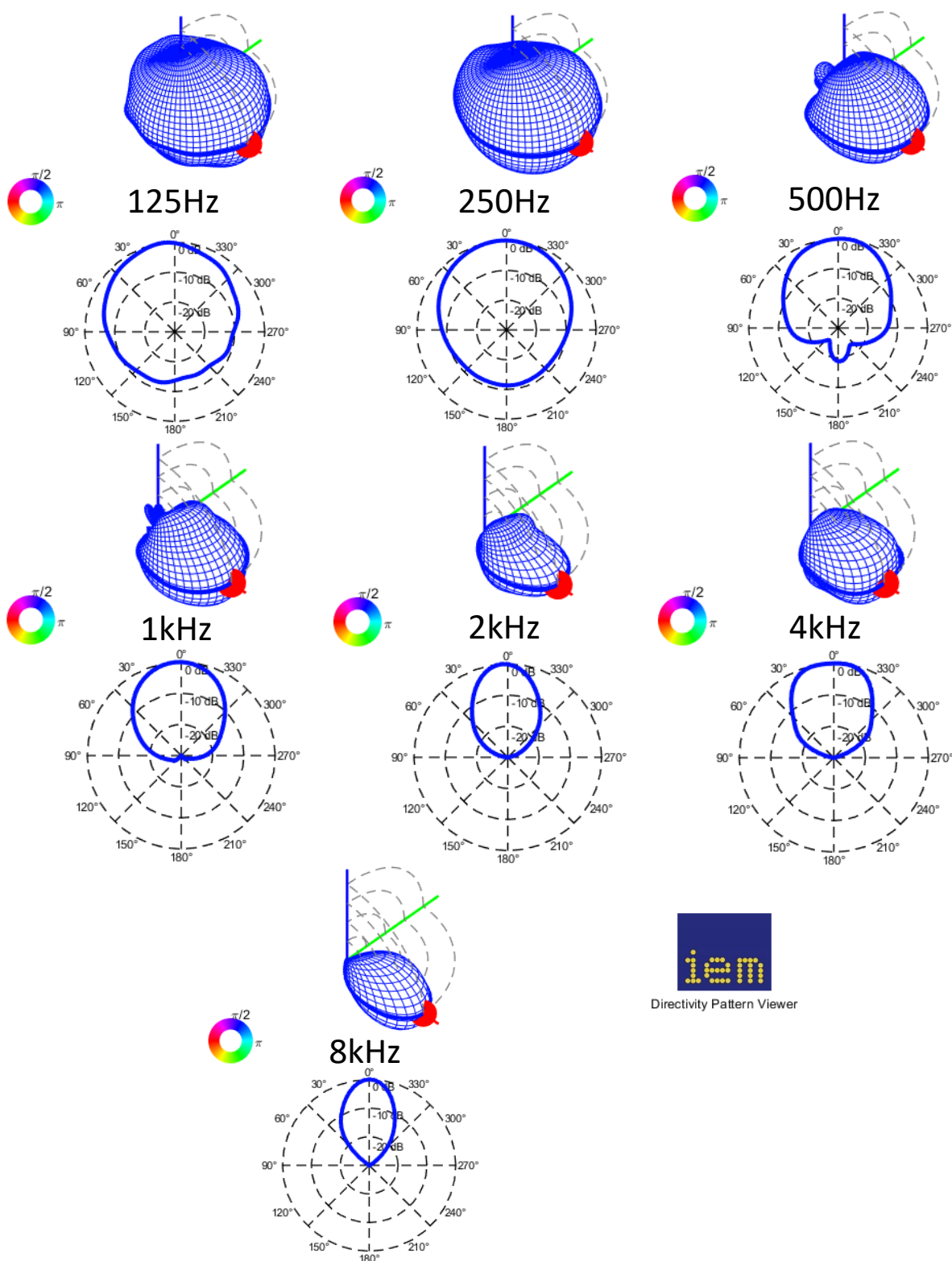
## 7 Appendix A

### Tannoy System 1200 directivity patterns (3rd order)



## 8 Appendix B

### Tannoy System 1200 directivity patterns (10th order)



## 9 Appendix C

```
//generates a random number between limits (including limits) and avoiding choosing a number from "forbiddenElements"

public static int RandomExceptList(int min, int max, int[] forbiddenElements)
{
    bool numberAllowed = true;

    for(int i=0;i<forbiddenElements.Length;i++)
    {
        if(forbiddenElements[i]<min || forbiddenElements[i]>max)
        {
            Debug.Log("All forbidden numbers have to be inside given range.");
            return min;
        }
        for (int j=i+1; j<forbiddenElements.Length;j++)
        {
            if(forbiddenElements[i]==forbiddenElements[j])
            {
                Debug.Log("Forbidden numbers have to be different between each other.");
                return min;
            }
        }
    }

    if (max<min)
    {
        Debug.Log("Minimum limit has to be lower than maximum limit...");
        return min;
    }

    if (forbiddenElements.Length>(max-min))
    {
        Debug.Log("You can't forbid so many numbers.");
        return min;
    }

    if (forbiddenElements.Length>=1)
    {
        //this part here checks indefinitely for an allowed randomly generated number (outside "forbiddenElements" array).
        //Eventually it will find it and return it and therefore terminate the loop.
        while(true)
        {
            numberAllowed=true;

            int r = Random.Range(min,max+1);

            for(int i=0;i<forbiddenElements.Length;i++)
            {
                if (r==forbiddenElements[i])
                    numberAllowed=false;
            }
            if(numberAllowed)
                return r;
        }
    }
    else
    {
        int r = Random.Range(min,max+1);
        return r;
    }
}
```

## 10 Appendix D

Experiment results:

2.397734  
2.21402  
2.395093  
2.070364  
2.147975  
2.403822  
2.048378  
3.564313  
4.035251  
2.917493  
4.080325  
4.466065  
3.87089  
4.870499  
4.918052  
2.147988  
3.559958  
4.005072  
4.702639  
3.877255  
5.281292  
3.199892  
4.284203  
4.585447  
2.059596  
4.60113  
3.577318  
1.942275  
2.680727  
3.480297  
2.774365  
2.60008  
2.870539  
2.31335  
2.579224  
2.594949  
2.264795  
1.972958  
3.062333  
2.342696  
4.268686  
3.94691  
2.642004  
4.345919  
4.110238  
5.005476  
4.774813  
4.685365  
2.128989  
2.687741  
3.866339  
3.555072  
4.255024  
5.012543  
1.557148  
4.281146  
4.560926  
2.132196  
4.387874  
2.681109  
3.014962  
3.019535  
2.988785  
2.241144  
2.850127  
2.864475

## References

- [1] M. Nussbaumer, *Aufnahme- und Wiedergabeverfahren für surround sound*. IEM, 2000. [Online]. Available: <https://phaidra.kug.ac.at/open/o:11115>
- [2] “Iem cube audio experiment room.” [Online]. Available: <https://iem.kug.ac.at/services/jahresberichte-wurde-2013-eingestellt/200001/cube.html>
- [3] F. Zotter and M. Frank, *Exploring the perceptual sweet area in Ambisonics*. AES, 2017. [Online]. Available: <https://iem.kug.ac.at/publikationen/publikation.html?q=8f78f80affd9a13fb259995e3a6b96611c1fd1016840b8e1de58cf3590e9f81f>
- [4] M. Zaunschirm, C. Schörkhuber, and R. Höldrich, “Binaural rendering of ambisonic signals by head-related impulse response time alignment and a diffuseness constraint,” *J. Acoust. Soc. Am.*, vol. 143, no. 6, pp. 3616–3627, 2018.
- [5] C. Schörkhuber, M. Zaunschirm, and R. Höldrich, “Binaural rendering of ambisonic signals via magnitude least squares,” in *Fortschritte der Akustik, DAGA*, Munich, March 2018.
- [6] K. Enge, M. Frank, and R. Höldrich, “Listening experiment on the plausibility of acoustic modeling in virtual reality,” in *Fortschritte der Akustik, DAGA*, 2020.
- [7] Đ. Perinović and M. Frank, “Spatial resolution of diffuse reverberation in binaural ambisonic playback,” in *Fortschritte der Akustik, DAGA*, Munich, March 2021.
- [8] Đ. Perinović, *Experiments on matching vision and sound in virtual reality*. IEM, 2022. [Online]. Available: <https://phaidra.kug.ac.at/view/o:127766>
- [9] C. Chen, R. Gao, P. Calamia, and K. Grauman, “Visual acoustic matching,” *arXiv preprint arXiv:2202.06875*, 2022.
- [10] S. Werner, F. Klein, T. Mayenfels, and K. Brandenburg, “A summary on acoustic room divergence and its effect on externalization of auditory events,” in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2016, pp. 1–6.
- [11] KUG, “György-ligeti-saal hall in mummuth,” 2009. [Online]. Available: <https://www.kug.ac.at/universitaet/campus/mumuth/>
- [12] Z. Pavel, B. S. Douglas, and B. W. Adelbert, *Auditory Distance Perception in Humans: A Summary of Past and Present Research*. Acta Acustica, 2005. [Online]. Available: [https://www.researchgate.net/profile/Adelbert-Bronkhorst/publication/229068125\\_Auditory\\_distance\\_perception\\_in\\_humans\\_A\\_summary\\_of\\_past\\_and\\_present\\_research/links/574c075c08ae74237f3ed74e/Auditory-distance-perception-in-humans-A-summary-of-past-and-present-research.pdf](https://www.researchgate.net/profile/Adelbert-Bronkhorst/publication/229068125_Auditory_distance_perception_in_humans_A_summary_of_past_and_present_research/links/574c075c08ae74237f3ed74e/Auditory-distance-perception-in-humans-A-summary-of-past-and-present-research.pdf)
- [13] A. J. Kolarik, B. C. Moore, P. Zahorik, S. Cirstea, and S. Pardhan, “Auditory distance perception in humans: a review of cues, development, neuronal bases,

- and effects of sensory loss,” *Attention, Perception, & Psychophysics*, vol. 78, no. 2, pp. 373–395, 2016.
- [14] P. Zahorik and F. L. Wightman, “Loudness constancy with varying sound source distance,” *Nature neuroscience*, vol. 4, no. 1, pp. 78–83, 2001.
  - [15] A. Kuusinen and T. Lokki, “Investigation of auditory distance perception and preferences in concert halls by using virtual acoustics,” *The Journal of the Acoustical Society of America*, vol. 138, no. 5, pp. 3148–3159, 2015.
  - [16] R. Holly, “The least painful way to set up htc vive lighthouses!” 2017. [Online]. Available: <https://www.vrheads.com/least-painful-way-set-htc-vive-lighthouses>
  - [17] “Photogrammetry pros and cons.” [Online]. Available: <https://www.scoutaerial.com.au/article-photogrammetry/https://www.scoutaerial.com.au/article-photogrammetry/>
  - [18] “Sketchup wikipedia.” [Online]. Available: <https://en.wikipedia.org/wiki/SketchUp>
  - [19] M. Thomas, M.-K. Nils, D. Rapolas, M. Leo, J. Sebastian, and P. Ville, *Pereceptually informed interpolation and rendering of spatial room impulse responses for room transitions*. ICA2022, 2022. [Online]. Available: <https://leomccormack.github.io/sparta-site/docs/help/related-publications/mckenzie2022perceptually.pdf>
  - [20] M. Kaspar and Z. Franz, “The perspectiveliberator - an upmixing 6dof rendering plugin for single-perspective ambisonic room impulse responses.” DAGA, October 2022. [Online]. Available: <https://arxiv.org/pdf/2210.03360.pdf>
  - [21] B. Manuel, F. Matthias, and R. Daniel, “Dirpat database and viewer of 2d or 3d directivity patterns of sound sources and receivers,” in *Audio Engineering Society Convention 144*, May 2018. [Online]. Available: <http://www.aes.org/e-lib/browse.cfm?elib=19538>
  - [22] J.-M. Jot and A. Chaigne, “Digital delay networks for designing artificial reverberators,” in *90th AES Conv., prepr. 3030*, Paris, February 1991.
  - [23] J. Stautner and M. Puckette, “Designing Multi-Channel Reverberators,” *Computer Music Journal*, vol. 6, no. 1, pp. 52–65, 1982.
  - [24] S. W. Clapp and B. U. Seeber, *Sound Localization in Partially Updated Room Auralizations*. DAGA, 2016.
  - [25] Z. Franz, F. Matthias, and Z. Markus, “Binaural rendering with measured room responses: First-order ambisonic microphone vs. dummy head,” 2020. [Online]. Available: [https://www.researchgate.net/publication/339618284\\_Binaural\\_Rendering\\_with\\_Measured\\_Room\\_Responses\\_First-Order\\_Ambisonic\\_Microphone\\_vs\\_Dummy\\_Head](https://www.researchgate.net/publication/339618284_Binaural_Rendering_with_Measured_Room_Responses_First-Order_Ambisonic_Microphone_vs_Dummy_Head)

- [26] EBU, “Ebu sqam test cd for audio evaluation,” 2008. [Online]. Available: <https://tech.ebu.ch/publications/sqamcd>
- [27] M. Kaspar, “Cube audio measurement with 24 source and 30 receiver positions,” 2008. [Online]. Available: <https://phaidra.kug.ac.at/view/o:104435>
- [28] W. C. Sabine, “Sabine reverberation formula.” [Online]. Available: <https://www.sciencedirect.com/topics/engineering/sabine-equation>
- [29] Z. Meng, F. Zhao, and M. He, “The just noticeable difference of noise length and reverberation perception,” in *2006 International Symposium on Communications and Information Technologies*. IEEE, 2006, pp. 418–421.
- [30] K. Enge, *Hoerversuch zur Plausibilitaet akustischer Modellierungen in virtuellen Umgebungen*. IEM, 2019.
- [31] Cockos, “Sws extension for additional reaper features,” 2014. [Online]. Available: <https://phaidra.kug.ac.at/view/o:104435>
- [32] T. Fredericks, “Unity osc scripts.” [Online]. Available: [\\_\\_\\_\\_\\_/https://thomasfredericks.github.io/UnityOSC](https://thomasfredericks.github.io/UnityOSC)
- [33] M. Kronlachner, “Mcfx convolver plug-in,” 2014. [Online]. Available: [\\_\\_\\_\\_\\_/http://www.matthiaskronlachner.com/?p=1910](http://www.matthiaskronlachner.com/?p=1910)
- [34] M. Frank and M. Brandner, “Perceptual Evaluation of Spatial Resolution in Directivity Patterns,” in *Fortschritte der Akustik, DAGA*, Rostock, march 2019.
- [35] M. Frank, M. Brandner, and F. Zotter, “Perceptual evaluation of spatial resolution in early reflections,” in *Fortschritte der Akustik, DAGA*, 2022.