Acoustical Instruments & Measurements
Universidad Nacional de Tres de Febrero - UNTREF

UNTREF
UNIVERSIDAD NACIONAL
DE TRES DE FEBRERO

DEVELOPMENT OF AN IMPULSE RESPONSE PROCESSING SOFTWARE BASED ON PYTHON

Leandro Gómez
Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina
email: leandro.damian93@gmail.com

Joaquin Guiñazú
Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina

email: joaquin_guinazu@hotmail.com

This paper explains the principles and procedures that were taken into account for the development of a software capable of calculating the acoustic parameters of a room from its impulse response. The software was developed on Python using WxPython framework for the GUI design and all the parameters were calculated according to the ISO-3382 standards. Finally, the results obtained with the software were compared with those calculated with other commercial software in order to verify if the signal processing was correct finding that the results delivered by the developed software where pretty accurate.

Keywords: Impulse Response, Acoustic Parameters, Python Software

1. Introduction

The response of an enclosure to any stimuli can be determined by it impulse response. An impulse its a signal with an infinitesimal duration and therefore, has information along all the spectrum. Since you cannot achieve a signal that has infinitesimal duration, there are several ways to generate a signal that serves the same purpose: firing a weapon, exploding a balloon, or even sine-sweep.

The ISO-3382 is an international standard that establish this different methods to obtain the impulse response of a room and how to calculate it characteristic parameters such as the clarity or the reverberation time. In this paper we will focus on the technique of obtaining the response from a sine-sweep.

Throughout this paper, the development of an impulse response processing software based on Python to obtain some acoustic parameters that characterize the room from it RIR will be explained. In particular, the calculated parameters are: EDT, T_{20}, T_{30}, C_{50}, C_{80}, $IACC_{Early}$, T_t and EDT_t. Finally, the obtained results will be compared with another commercial software in order to verify it quality.

2. Theorical FreamWork

2.1 Impulpse Responses

The characteristics of a linear, time-invariant component of any system are fully described by its impulse response $h(t)$. It is thus desirable to acquire the impulse response as accurately as possible.

In traditional impulse response measurement, periodic pulse and Maximal-Length Sequence (MLS) are often used as excitation signals. The impulse response for a room will typically be an oscillatory signal with a large number of periods. The envelope of the signal will be irregular but typically have a fast attack-time and an exponential decay. The impulse response may be measured as the response of the room to a very short acoustic pulse. However, it will in most cases where sources other than a loudspeaker are used, be difficult to have sufficient control of the spectral content and the directional characteristics of the excitation. To obtain the required control of the excitation signal, the impulse response is in most practical cases obtained by digital signal processing. The room is excited by a known signal for a certain time and the impulse response is calculated from the response to the excitation.[1]

The output of a time-invariant linear system is analytically determined by the convolution of the input signal with the system's impulse response (IR),

$$y(t) = x(t) * h(t) \tag{1}$$

where $x(t)$ and $y(t)$ are respectively the input and output signals and $h(t)$ represents the IR, which entirely characterizes the system in the time domain. Frequently, problems are faced in the frequency domain because it simplifies the analysis, e. g. a convolution in the time domain is equivalent to a multiplication in the frequency domain. This is done by Fourier transforming the time signal, which is usually referred to as spectrum. In this way, the output spectrum of a linear system is simply the multiplication of the input and the IR spectra,

$$Y(f) = X(f).H(f) \tag{2}$$

where $X(f)$ and $Y(f)$ denote the Fourier transforms of the input and the output signals and $H(f)$ is the spectrum of the IR, i. e. the so-called transfer function (TF) of the system. The simple division of the output and the input spectra yields the transfer function of the system,

$$H(f) = \frac{Y(f)}{X(f)} \tag{3}$$

Then, the inverse Fourier transform (IFT) of this ratio retrieves the corresponding IR.

2.2 Sine-Sweep

2.2.1 Linear Sweep

A linear (LIN) sweep is usually considered a 'white' excitation signal within the frequency range of concern. The term linear refers to the fact that the instantaneous (angular) frequency changes linearly with time:

$$\omega_{LIN}(t) = \frac{\omega_2 - \omega_1}{T} t + \omega_1 \tag{4}$$

where ω_1 and ω_1 are the lower and the upper angular frequencies of interest and T represents the duration of the signal. The sweep rate is a parameter often used to indicate how fast the instantaneous frequency changes with time. For LIN sweeps, this is always constant and equals to $\frac{BW}{T}$, where BW is the bandwidth of the signal.

2.2.2 Log-Sweep

Excitation signals with a 'white' spectrum are not always the best choice. Although all frequencies are equally excited, a white signal does not ensure the same accuracy for all frequency components unless the background noise also has a flat spectrum. In fact, background noise is usually more prominent at low frequencies. In order to counterbalance this effect, the low frequency content of the excitation signal must be emphasized. This can easily be done for sweep signals defining the proper temporal expression of the instantaneous frequency. Specifically, the sought expression must be a function that grows very slowly at the beginning and raises very rapidly afterwards. This is the main idea behind the definition of instantaneous frequency of a logarithmic (LOG) sweep. In this case,

$$\omega_{LOG}(t) = \omega_1 e^{\frac{t}{T} In(\frac{\omega_2}{\omega_1})} \tag{5}$$

The energy of a sweep signal is proportional to the time spent exciting each spectral component. Unlike LIN sweeps, the sweep rate of a LOG sweep is not constant.[2]

2.2.3 *Inverse Filter*

Most of the deconvolution methods regarding sweep signals involve the use of a time-reversed version of the excitation signal. In [3], Muller and Massarani only suggest to deconvolve the measured response of the system with a time-reversed version of the excitation signal when the latter is a LIN sweep. This is due to the fact that LIN sweeps are regarded as 'white' signals. Other authors as Farina in [4] describe the deconvolution process of LOG sweeps with a time-reversed excitation signal followed by a magnitude spectrum correction. This it's based on applying to the signal an amplitude envelope to reduce the level by $6dB/octave$, starting from $0dB$ and ending to $-6log_2(\frac{\omega_2}{\omega_1})$[2]

2.3 Acoustical Parameters

2.3.1 *Reverberation Time*

The least controversial of all physical criteria for use in rating the subjective listening quality of a room is the reverberation time of the sound field measured as a function of frequency. Reverberation time RT_{60} is defined as the length of time in seconds it takes for the energy in the steady-state sound field in a room to decay 60 dB after the source of sound excitation is turned off. When measuring RT it is in practice difficult to reach full 60 dB decay due to e.g. background noise. Instead a decay range of 20 or 30 dB are commonly used.

According to ISO 3382-2 the reverberation times given by the limited decay ranges are denoted T_{20} and T_{30}, respectively and according to the standard, when determining T_{20} and T_{30}, the evaluation does not start until the sound level has already fallen by 5 dB.

The Early decay time (EDT) is a measure of the rate of sound decay, expressed in the same way as a reverberation time, based on the first 10 dB portion of the decay. In a highly diffuse space where the decay is completely linear, the two quantities, RT and EDT, would be identical. The early decay time has been shown to be better related to the subjective judgement of reverberation, also called 'reverberance', than the traditional reverberation time.[5]

To sum up, each parameter is the measure of time of the sound decay between the following limits:

$$\begin{aligned} EDT &: (0dB; -10dB) \\ RT_{20} &: (-5dB; -25dB) \\ RT_{30} &: (-5dB; -35dB) \end{aligned} \tag{6}$$

2.3.2 *Clarity*

Speech clarity (C_{50}) is expressed in dB and it is related to the attribute clarity. It is an objective measure of the clarity or intelligibility of speech. The basis for C_{50} is the fact that late reflections are unfavorable for understanding speech because it causes speech sounds to merge making speech unclear. However, if the delay does not exceed a certain time limit, the reflections will contribute positively to the intelligibility.[5]

$$C_{50} = 10log\frac{\int_0^{50ms} h^2(t)\, dt}{\int_{50ms}^{\infty} h^2(t)\, dt} \tag{7}$$

Objective clarity or early-to-late sound index (C_{80}) relates to the balance between perceived clarity and reverberance, which can be particularly delicate for music listening. The energy within 80 ms of the direct sound includes that of the direct sound and early reflections.

$$C_{80} = 10log\frac{\int_{0}^{80ms}h^2(t)\,dt}{\int_{80ms}^{\infty}h^2(t)\,dt} \tag{8}$$

2.3.3 Inter Aural Cross Correlation

Is the measure of the difference in signals received by two ears of a person. IACC values range from -1 to +1. A value of -1 means the signals are identical, but completely out of phase. +1 means they are identical, and 0 means they have no correlation at all. The IACC will be nearly +1 for mono sources directly in front of or behind the listener, with lower values if the source is off to one side. IACC values are of interest to acousticians as a number of researches have found that large IACC values correspond to greater degrees of envelopment and overall more enjoyable listening experience in auditoriums.

$$IACF_{t_1,t_2}(\tau) = \frac{\int_{t_1}^{t_2}p_l(t).p_r(t+\tau)\,dt}{\sqrt{\int_{t_1}^{t_2}p_l^2(t)\,dt\int_{t_1}^{t_2}p_r^2(t)\,dt}} \tag{9}$$

where $p_l(t)$ is the impulse response at the entrance to the left ear canal; $p_r(t)$ is the impulse response at the entrance to the right ear canal. The inter-aural cross correlation coefficients, IACC will be:

$$IACC_{t_1,t_2} = max|IACF_{t_1,t_2}|for - 1ms < \tau < +1ms \tag{10}$$

2.3.4 Transition Time(T_t)

The transition time can be interpreted as the time after which all the reflected pulses are heavily overlapping, and therefore the envelope becomes more important than the detailed structure.[6]
From this parameter we can define the EDT_t as the rate of sound decay, expressed in the same way as a reverberation time, of the portion of the signal between $0dB$ and the level in the transition time.

2.4 State of Art

There are currently several software that have the option of performing impulse response processing. Among the commercial we can find the Aurora developed by Angelo Farina. Although it has several applications, it focuses in measuring and manipulating room acoustical impulse responses, performing analysis of acoustical parameters and auralization.[7]. Currently it can be used as a plugin for Adobe Audition and Audacity softwares and it is a close source software. This is the software that will be used in the present work to compare the obtained results. Another known software to perform this type of calculation is the Dirac[8] software from Bruel & Kjær, which, in addition to calculating some of the parameters of the ISO-3382 standard, also has the option of calculating the parameters mentioned in ISO 18233 (methods analysis) and IEC 60268-16 (speech intelligibility). The latter was not used to compare results, as it is a paid program. In addition, there are non-commercial software developed by researchers, such as the software called Acmus. This software was developed in Matlab by Bruno S. Masiero et. al[9] and allows the the user to generate and deconvolve acoustic signals to obtain the impulse response of a room and calculate the enclosure's acoustic parameters from the obtained RIR.

Unlike the commercial and non-commercial softwares mentioned, the program developed in this work is capable of calculating the Transition Time (T_t) and Early Decay Transition time (EDT_t) parameters. It was decided to develop it as a free and open source software so that the community interested in the research area can make their contributions and, therefore, the software obtained constant improvements and updates.

3. Software Development

GGRiRs software allows the user to calculate the acoustical parameters of an enclosure from it impulse response. The main structure of the software is shown in the figure 1. As it can be seen in the figure,the impulse response can be import by the user or obtained from the recorded sine-sweep used in the measurement and the corresponding inverse-filter that the user must generate. In both cases, the signal can be mono or stereo.

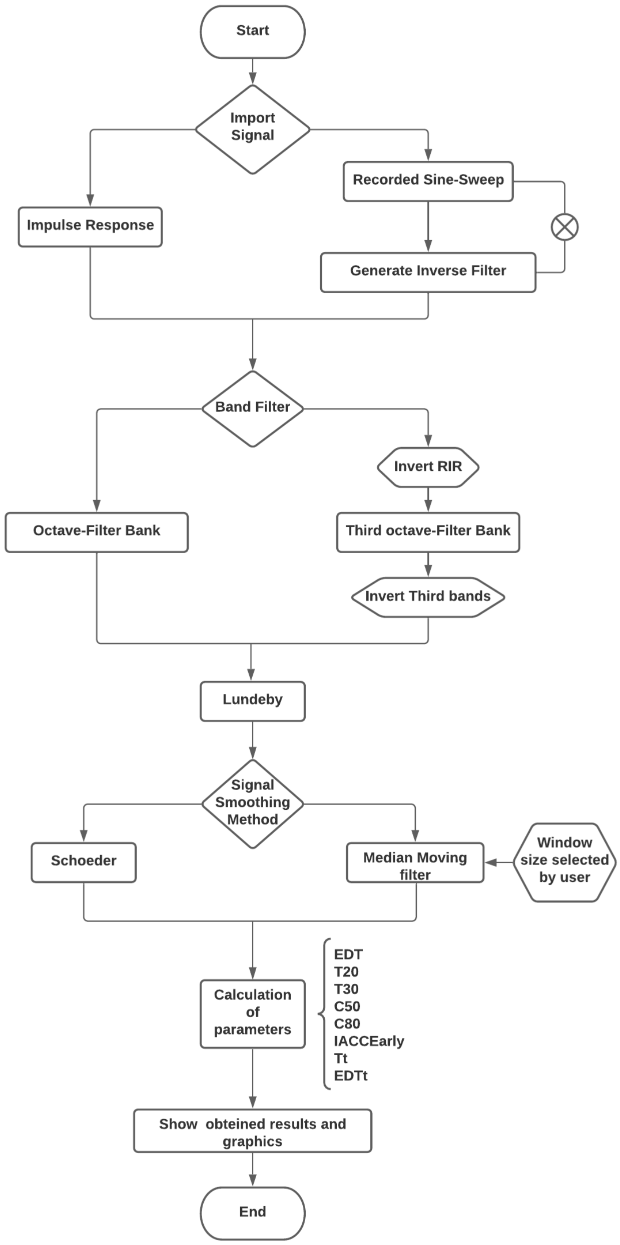

Figure 1: Software block diagram

In figure 2 the menu to apply the reverse filter to the imported recorded sine sweep is shown. The user should know and enter all the parameters of the imported sine-sweep, such as duration or if its linear or logarithmic, in order to apply the correct inverse filter and obtained an impulse response with minimum error for processing. After this, the user can choose between two procedures to smooth the signal: the inverse Schroeder integral[10] or a moving median filter. In order to find the limit of the Schroeder integral, the Lundeby[11] procedure was applied. In the case of the MMF, the user must choose a window length in ms, which, in the first place, is suggested to be the inverse of the lowest

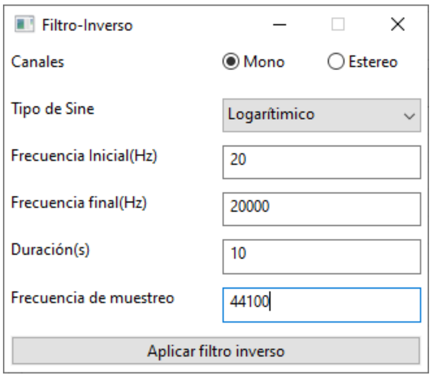

Figure 2: Menu to apply reverse filter to imported recorded sine sweep

frequency of the signal to be processed. For example, the suggested window size for a signal that starts in 20 Hz is 50 ms. Mathematically:

$$T_{Ventana}(ms) = \frac{1}{f_{inicio}} * 1000 \tag{11}$$

Then, the signal passes through an octave and thirds octave filter bank. In the case of the third octave bank, the "reversed RIR"[12] procedure is applied to minimize the uncertainty of its direct application. This procedure is based primarily on inverting the impulse response on the time axis. Then, the filter bank,either octaves or thirds of octaves to the inverted signal, is applied and finally the filtered bands are temporarily inverted again.

Then, the software calculate all the parameters as explained in the theoretical framework, being the $IACC_{Early}$ calculated only for the stereo impulse responses. Finally, the results are shown on the screen together with the graph of the signal energy and the smoothing. Figure 3 shows the GUI with the graphic of the imported RIR and the acoustical parameters obtained by octaves and third-octaves.

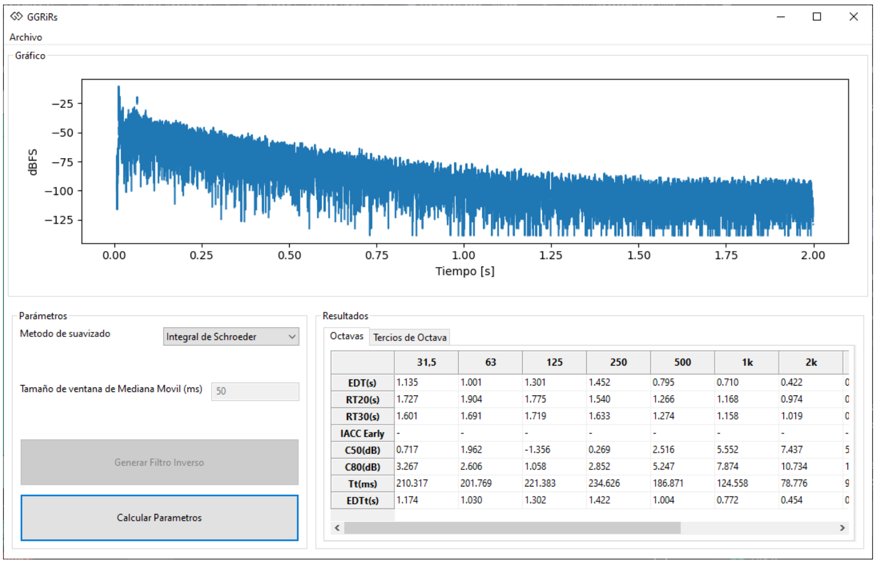

Figure 3: Calculated Parameters from an Imported Impulse Response

4. Results and Discussion

In this section the results obtained with both smoothing methods will be shown and compared with those obtained with Angelo Farina's commercial software: Aurora. For this process, a stereo impulse response of the Lady Chapel, St Albans Cathedral obtained from the OpenAir library was used. The presentation of the results of all the parameters for octaves and thirds of an octave for each case are shown in appendix B while in this section we will concentrate on some parameters in particular and only on the left channel of the signal filtered by octaves.

For a correct presentation of the results, it was taken into account that for the lower frequency bands (from 125 Hz downwards), the results presented large variations, which is not reflected in the higher frequency bands. This may be mainly due to the method used when filtering the signal (2nd order butterworth filter). This problem could be solved by implementing some other type of filter which works better at low frequencies, or even using 3 different types of filter to discriminate low, medium and high frequencies. Due to the aforementioned, data below the 125 Hz band will not be taken into account for the analysis of results, since these results will present great uncertainties.

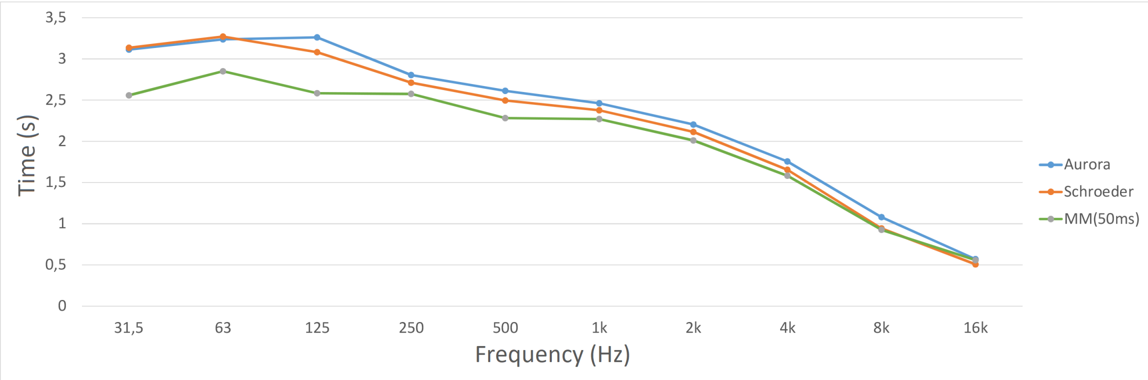

Figure 4: RT30 obtained with both methods and compared with commercial software

Figure 4 shows the results of the RT30 for the analyzed audio. It can be seen that the results are very similar for both smoothing methods relative to commercial software. But this similarity deteriorates as we go down to lower frequencies (in the range specified before as comparable), being the 125 Hz band the one that presents the greatest difference between the results obtained with the MMF and the other two points. However, it can also be seen that in this band, the difference between the results obtained with the Schroeder method and those obtained with the Aurora present a minimal difference. This clearly demonstrates the difference in signal processing that exists between the two smoothing methods.

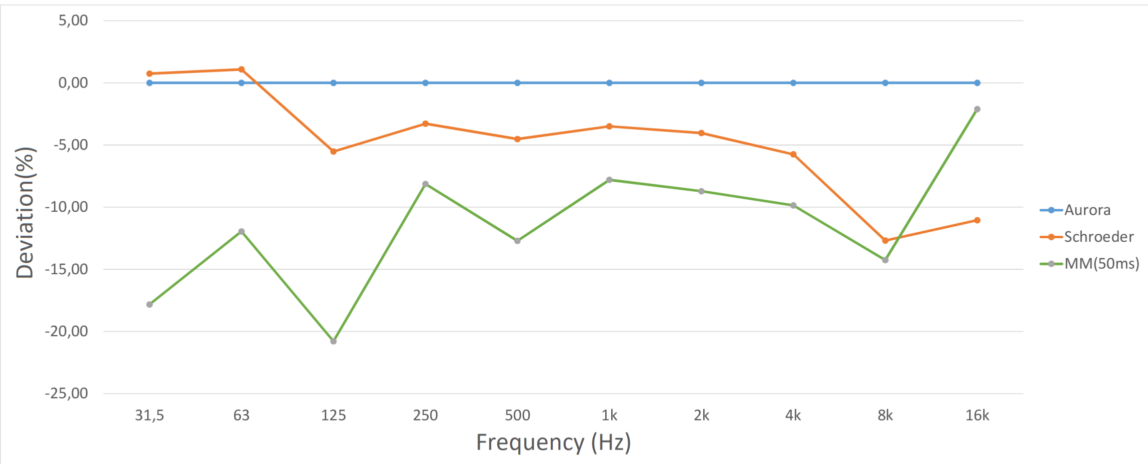

Figure 5: Deviation between own results and the obtained with commercial software

Figure 5 shows what is the deviation of the results obtained with the software in relation to those obtained in the Aurora.

This is useful to clearly observe the difference in the results obtained compared to those of the Aurora software. It can be seen that the greatest difference is found in the median moving filter smoothing method with the greatest deviation in most of the spectrum, which has an average of 10.55% . On the other hand, the Schroeder method presents an average deviation of 6.21%. This may be due to the fact that the Aurora software uses the Schroeder method to smooth the signal and, also, that it presents a more uniform smoothing than the moving median. On the other hand, the results of using the moving median as a smoothing method could be improved by increasing the size of the window, which results in a more uniform smoothing, but at the same time, it requires more processing power and therefore, the calculation of the parameters will take longer.

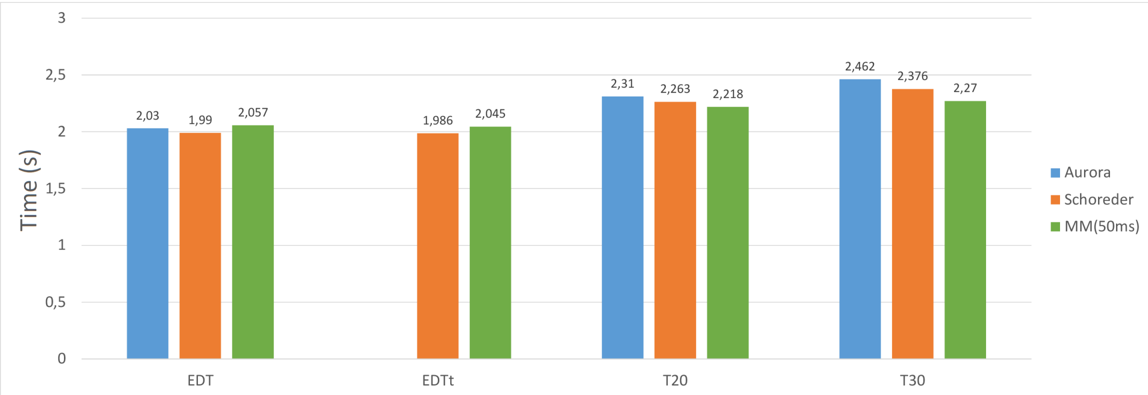

Figure 6: Reverberation Time obtained with both methods and compared with commercial software

In figure 6, we can see the results obtained with each process for the EDT, T_{20}, T_{30} and EDT_t for the 1 kHz as the most representative band. It is important to clarify that the EDT_t is only shown for the GGRiRs software processing since the Aurora does not allow to calculate this parameter. As can be seen, for this band there are no differences greater than 5% for any of the parameters calculated in relation to those of commercial software. Moreover, the T_{30}, which was previously discussed, is the one that presents the greatest difference and it has already been determined that this parameter presents an average deviation no greater than 12%, therefore, it is expected, that the deviation of the other parameter was lower. Regarding the EDT_t, similar results were obtained between the two smoothing processes, but it cannot be concluded if they are correct per se, because there is no other commercial software that calculates it to compare.

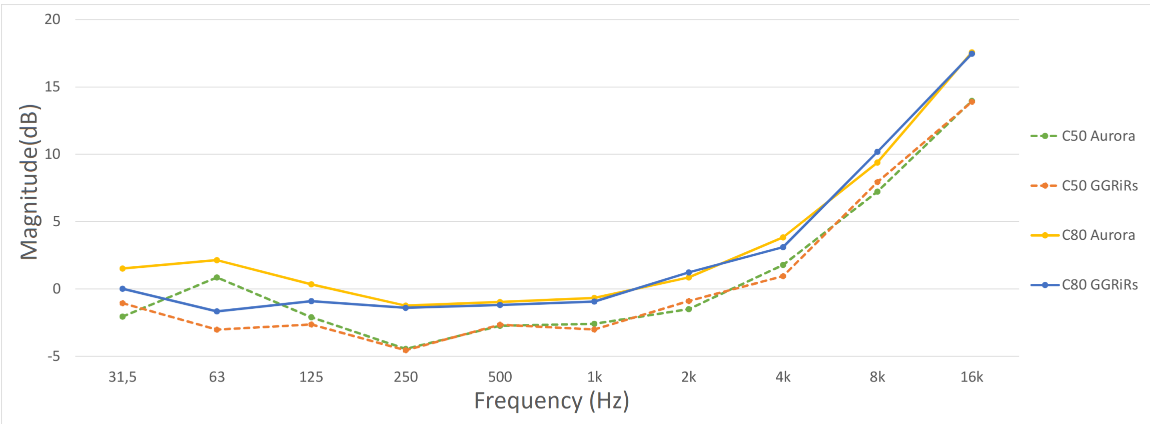

Figure 7: Clarity obtained with both methods and compared with commercial software

In figure 7 the C_{50} and C_{80} values obtained with the software developed against those obtained with Aurora can be observed. As mentioned above, the values obtained below the 125 Hz band will not be taken into account in order to show conclusive results. It can be observed that the deviation between the results is not very significant, resulting for the C_{50} an average variation of 8.27% and for the C_{80} an average variation of 8.22% with respect to the Aurora software.

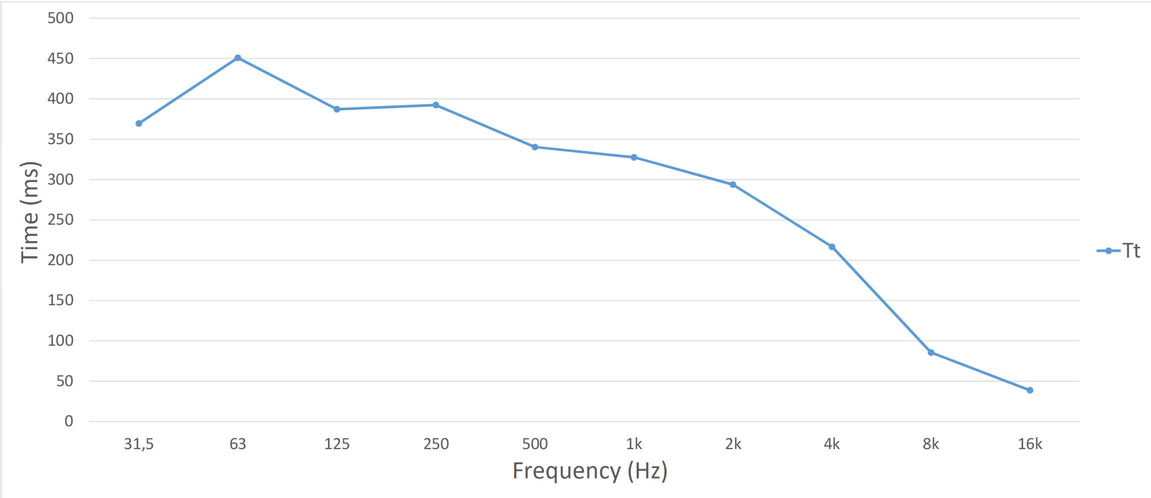

Figure 8: Transition time obtained with GGRiRs Software

In figure 8, the curve corresponding to the transition time (T_t) can be observed. Again analyzing above 125 Hz, a decay of the curve is seen as the frequency increases. This makes sense since the parameter T_t corresponds to the time that elapses between the early and late reverberant fields. High-frequency reflections are easier to attenuate with any porous material that may be found at the measurement site, including bricks or paint on the walls, as well as other types of objects that are in the place. For this reason, they lose energy more quickly than low-frequency reflections, resulting in an overlap of reflections for shorter times.

For the T_t parameter, as for the EDT_t, there is no commercial software available to make comparisons, that is why in figure 8 only the curve of the T_t parameter obtained with the software developed in octave bands is shown.

5. Conclusions

It can be concluded that the developed software works correctly, at least taking into account the bands above 125 Hz. In this range of the spectrum, the results obtained by the Schroeder smoothing method are within the expected values, with an efficient and fast processing time. Not so those obtained by the moving median method, which result in large deviations, largely due to the window size used during the comparison with Aurora. Regarding this last smoothing method, it is also concluded that it is not useful, since large window sizes are required to obtain uniform smoothing and this translates into long processing times. On the other hand, the Schroeder method continues to result in considerably less processing times and a much more uniform smoothing, which results in a calculation of acoustic parameters according to those expected.

For future versions of the software, better signal filtering methods could be considered, analyzing more variables, such as taking into account the low, mid and high frequencies, filtering each one in different ways, or giving the user the option to select which type of enclosure was measured (church, theater, control room, etc.), so that the software uses different algorithms depending on the approximate R_T that is expected. Another point to consider could be to use other methods to find the crossing point between the sound and the background noise, for example the one developed by Leonardo Pepino[13], since this method works best than Lundeby's method for impulse responses which have a lot of background noise and do not satisfy the S / N conditions specified in ISO-3382.

References

1. ISO 3382. *Acoustics - Measurement of room acoustic parameters*. International Organization for Standardization, 2009.

2. Antoni Torras Rosell. *Methods of Measuring Impulse Responses in Architectural Acoustics*. Technical University of Denmark, 2009.

3. Swen Muller. *Transfer-function measurement with sweeps*. Journal of the Audio Engineering Society, Vol. 49, No. 6, pp. 443–471, 2001.

4. Angelo Farina. *Simultaneous measurement of impulse response and distortion with a swept-sine technique*. 108th AES Convention, No. 5093, 2000.

5. Michael Barron. *Auditorium Acoustics and Architectural Design*. Taylor & Francis, 2009.

6. Alejandro Bidondo, Javier Vazquez, Sergio Vazquez, Mariano Arouxe, and Germán Heinze. *A new and simple method to define the time limit between the early and late sound fields*. Audio Engineering Society, 2016.

7. Angelo Farina. *Aurora Plugin*. URL: http://pcfarina.eng.unipr.it/Aurora_XP/index.htm.

8. Brüel & Kjær. *DIRAC*. URL: https://www.bksv.com/es/analysis-software/acoustic-analysis-software/room-acoustics-software-dirac.

9. M. Queiroz, F. Kon F. H. Iazzetta, M. H. A. Gomes, F. L. Figueiredo, B. S. Masiero, L. K. Ueda, L. Dias, M. H. C. Torres, and L. F. Thomaz. *AcMus: an open, integrated platform for room acoustics research*. Journal of the Brazilian Computer Society, vol. 14, no. 3, pp. 87–103, 2008.

10. M. R. Schroeder. *New method of measuring Reverberation Time*. Bell Telephone Laboratories, 1964.

11. A. Lundeby, T. E. Vigran, H. Bietz, and M. Vorlander. *Uncertainties of measurements in room acoustics*. Appl. Acoust. 81, 344–355, 1995.

12. Birgit Rasmussen, Jens Holger Rindel, and Helge Henriksen. *Designand Measurement of Short Reverberation Times at Low Frequencies in Talks Studios*. Journal of the Audio Engineering Society, 39(1/2), 47-57., 1991.

13. Leonardo D. Pepino and German R. Barchi. *Estimación de parametros acústicos a partir de respuestas al impulso con ruido*. IV Jornadas JAAS, 2018.

6. Appendix A

The RIR under analysis throughout the work was belong to Lady Chapel, St Albans Cathedral and it is available in the OpenAirlibrary. This section shows the graphs of all the parameters of the left channel filter by octaves obtained with the software(except for the $IACC_{Early}$ that needs the processing of both channels), compared to those obtained in Aurora. Notice that only the EDT, T_{20} and T_{30} are shown in the graphics with both smoothing process. This is because C_{50}, C_{80} and $IACC_{Early}$ presents the same values on all the octaves for Schroeder and the moving median filter. Furthermore, for the T_t and EDT_t parameters, only the results obtained with the GGRiRs software are presented, since there is no commercial software to compared with.

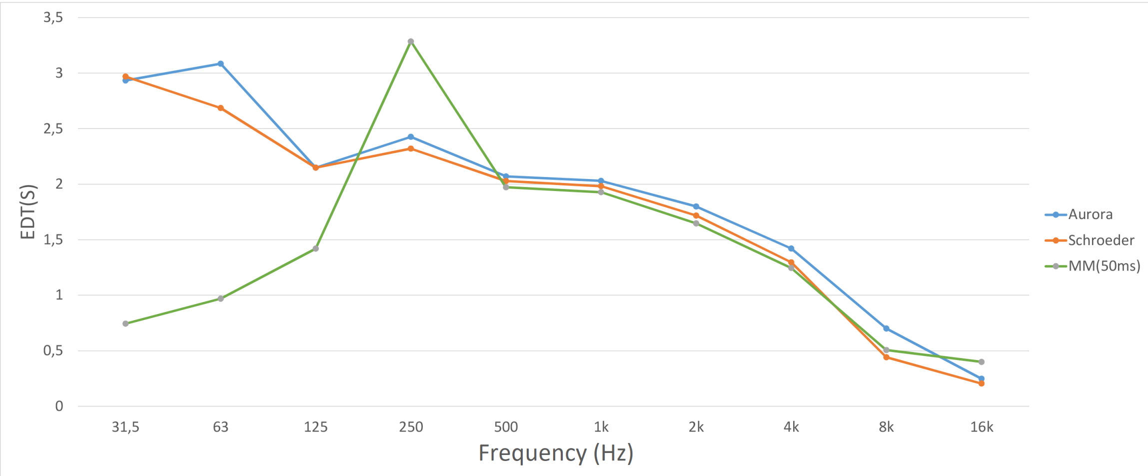

Figure 9: EDT obtained with GGRiRs and compared with commercial software

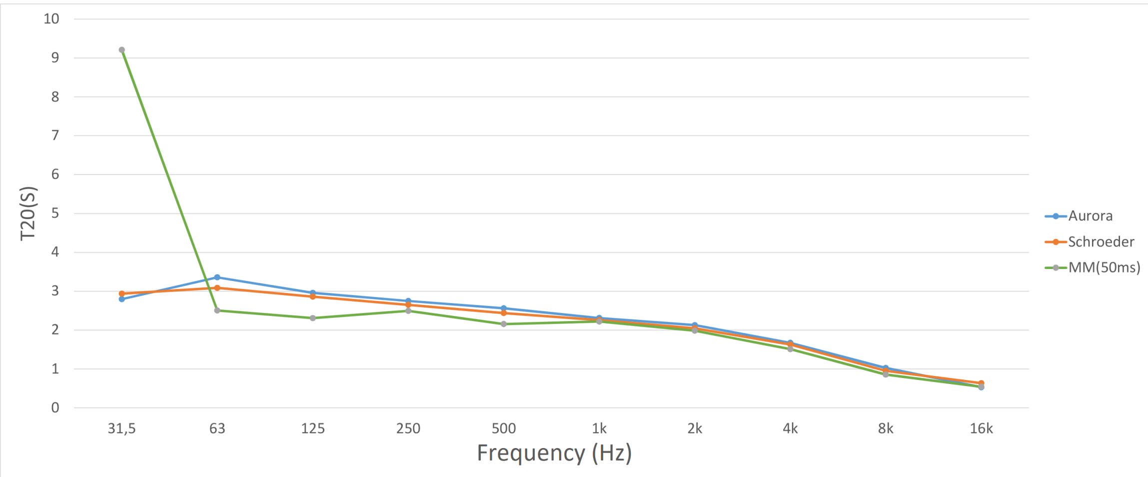

Figure 10: T20 obtained with GGRiRs software and compared with commercial software

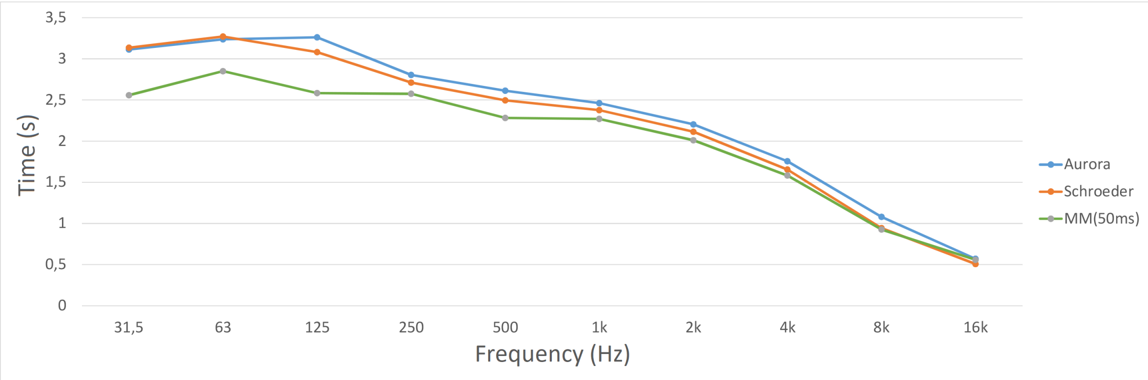

Figure 11: T30 obtained with GGRiRs software and compared with commercial software

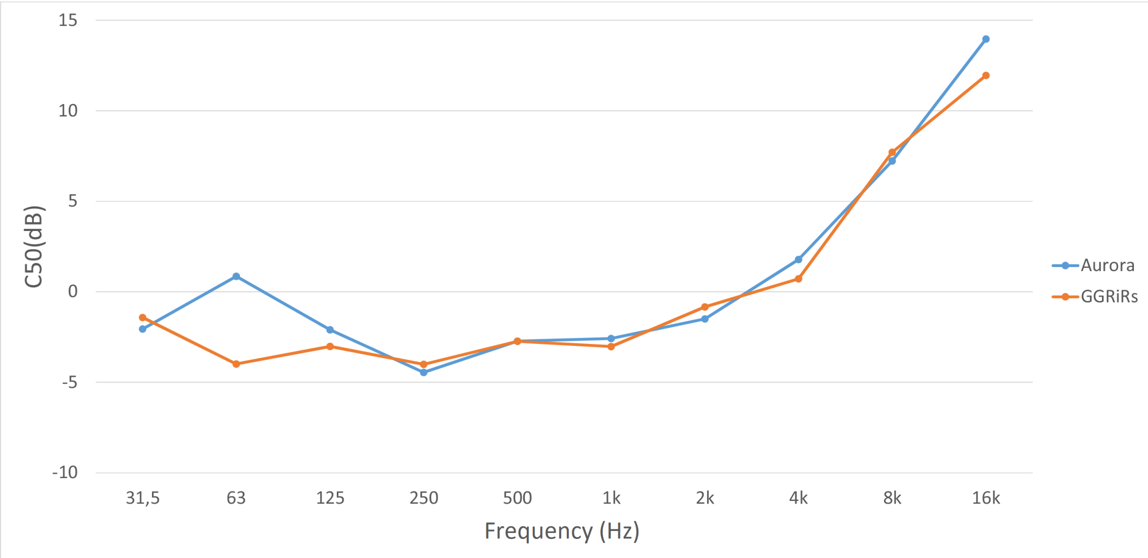

Figure 12: C50 obtained with GGRiRs software and compared with commercial software

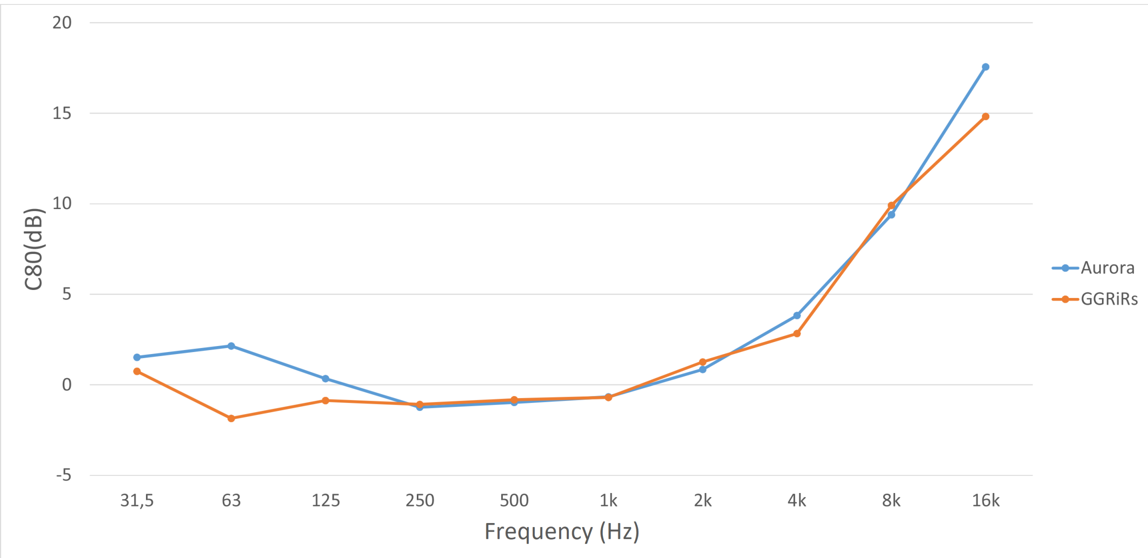

Figure 13: C80 obtained with GGRiRs software and compared with commercial software

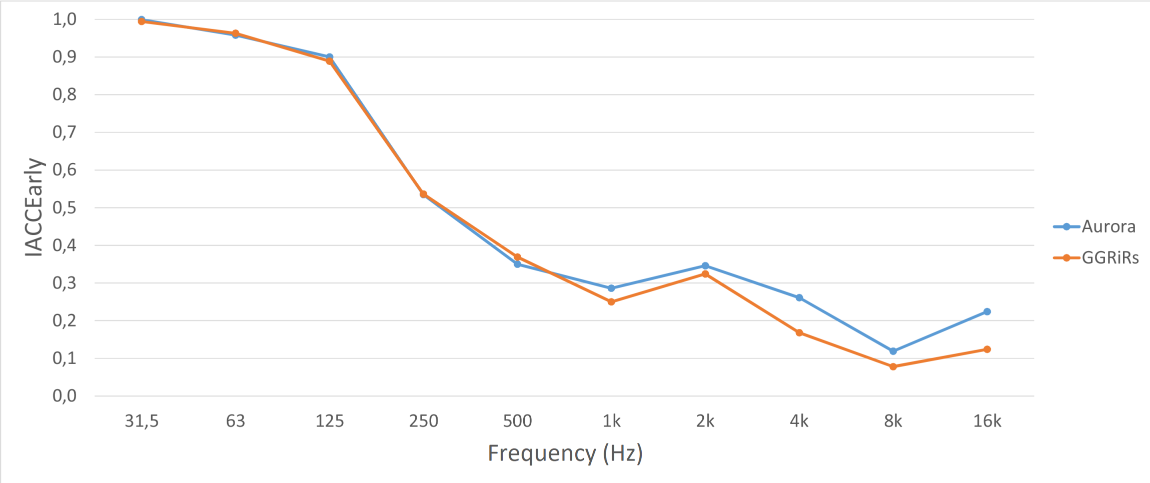

Figure 14: IACC obtained with GGRiRs software and compared with commercial software

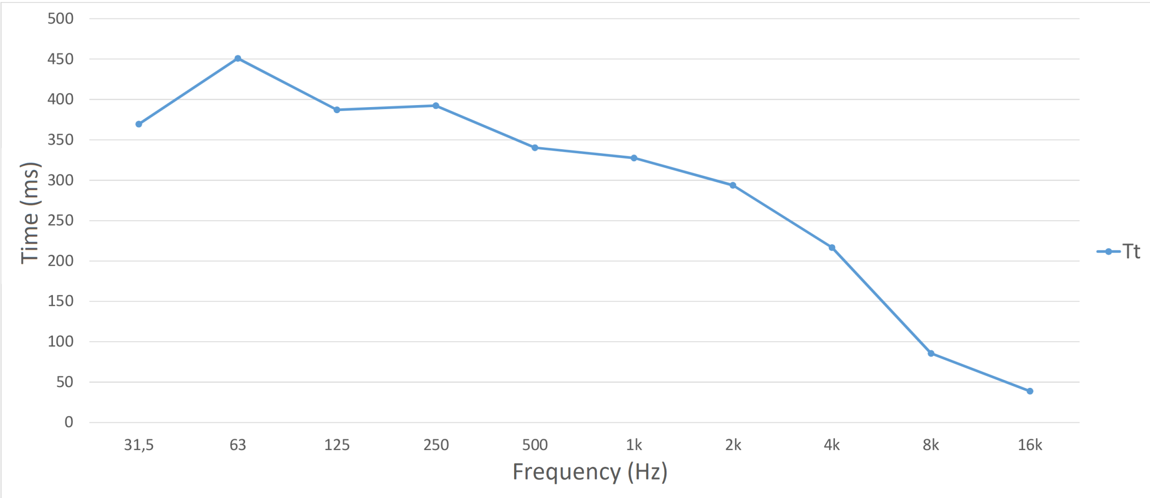

Figure 15: Transition Time obtained with GGRiRs software

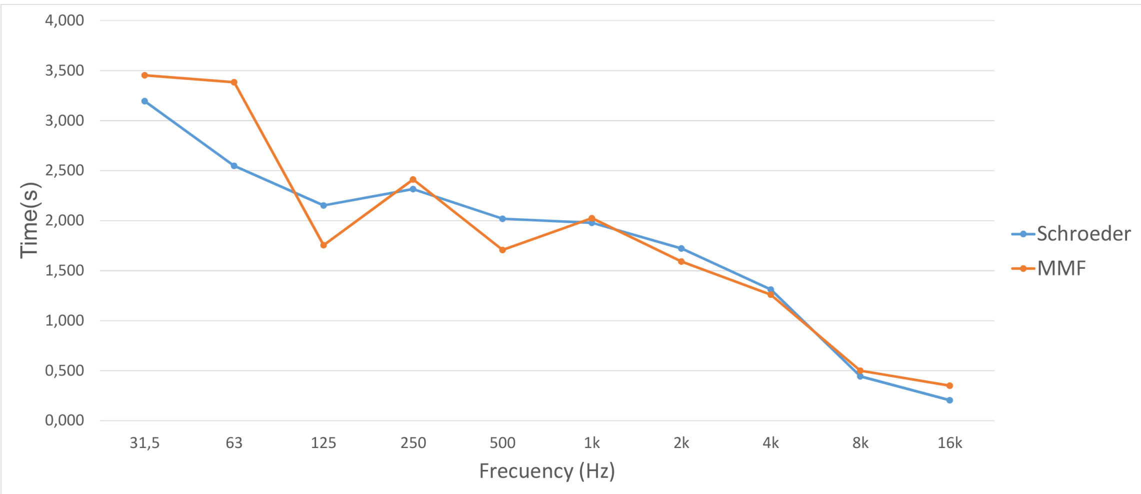

Figure 16: EDTt obtained with GGRiRs software

7. Appendix B

This section shows the tables of the parameters obtained by the GGRiRs software for both channels of the RIR filtered by octaves and third-octave bands with Schroeder smoothing.

Left	31,5	63	125	250	500	1000	2000	4000	8000	16000
EDT(s)	2,969	2,685	2,148	2,320	2,028	1,982	1,717	1,297	0,442	0,205
T20(s)	2,938	3,086	2,860	2,649	2,439	2,254	2,048	1,629	0,952	0,634
T30(s)	3,303	3,159	3,057	2,710	2,491	2,356	2,113	1,705	1,037	0,741
IACC early	0,994	0,963	0,889	0,536	0,369	0,250	0,324	0,168	0,078	0,124
C50(dB)	-1,421	-3,983	-3,017	-4,006	-2,730	-3,027	-0,834	0,720	7,715	11,954
C80(dB)	0,745	-1,852	-0,869	-1,083	-0,827	-0,698	1,260	2,838	9,911	14,824
Tt(ms)	369,501	450,907	387,188	392,313	340,272	327,528	293,719	216,644	85,556	38,798
EDTt(s)	3,195	2,548	2,153	2,316	2,019	1,979	1,722	1,311	0,444	0,206

Right	31,5	63	125	250	500	1000	2000	4000	8000	16000
EDT(s)	2,915	2,605	2,207	2,482	1,812	1,656	1,533	0,842	0,548	0,221
T20(s)	2,931	3,091	2,781	2,664	2,325	2,164	1,988	1,455	0,949	0,604
T30(s)	3,255	3,142	3,025	2,700	2,448	2,276	2,065	1,562	1,042	0,719
IACC early	0,994	0,963	0,889	0,536	0,369	0,250	0,324	0,168	0,078	0,124
C50(dB)	-1,498	-3,991	-2,965	-2,235	-1,692	0,348	1,934	4,767	6,606	11,122
C80(dB)	0,925	-2,128	-0,112	0,134	0,937	2,491	3,416	6,411	8,929	14,241
Tt(ms)	360,204	468,662	389,887	412,018	301,769	266,871	244,195	166,349	95,034	51,701
EDTt(s)	3,139	2,480	2,221	2,493	1,826	1,670	1,571	1,023	0,558	0,251

Figure 17: Octave filtered RIR results for both channels of the signal processed with GGRiRs Software

THIRD-OCTAVE Left	25	31,5	40	50	63	80	100	125	160	200	250	315	400	500	630
EDT(s)	2,401	1,720	3,689	3,012	2,740	3,379	3,137	3,604	2,836	2,947	2,811	2,526	2,761	2,692	2,393
T20(s)	3,218	3,064	3,207	3,973	3,215	3,223	3,383	3,412	3,414	2,920	2,759	2,640	2,686	2,611	2,528
T30(s)	3,494	3,300	3,379	4,278	3,175	3,169	3,189	3,612	3,445	3,094	2,984	2,843	2,705	2,562	2,605
IACC early	1,001	1,000	0,997	0,972	0,978	0,976	0,983	0,976	0,879	0,713	0,597	0,242	0,054	0,294	0,299
C50(dB)	1,798	3,307	-0,094	-4,230	-1,984	-4,756	-6,246	-4,831	-6,465	-6,305	-5,725	-1,561	-2,709	-2,784	-4,427
C80(dB)	4,874	4,461	0,661	-2,094	0,119	-3,722	-4,784	-3,455	-5,361	-5,652	-3,821	-0,062	-1,110	-0,844	-3,026
Tt(ms)	405,828	366,689	490,998	546,440	470,567	511,202	492,812	568,435	476,621	547,914	471,474	390,975	426,803	456,395	401,338
EDTt(s)	2,508	2,430	2,922	3,191	2,785	3,328	2,982	3,560	2,831	2,958	2,771	2,518	2,761	2,701	2,388

THIRD-OCTAVE Left	800	1000	1250	1600	2000	2500	3150	4000	5000	6300	8000	10000	12500	16000	20000
EDT(s)	2,363	2,135	2,268	2,275	2,135	1,984	1,892	1,675	1,523	1,238	1,027	0,846	0,778	7,480	31,726
T20(s)	2,576	2,461	2,509	2,348	2,277	2,117	1,999	1,701	1,492	1,300	1,073	0,908	1,043	21,417	20,641
T30(s)	2,579	2,507	2,513	2,400	2,290	2,156	2,017	1,792	1,553	1,333	1,121	0,973	1,384	18,642	16,422
IACC early	0,190	0,225	0,098	0,191	0,104	0,225	0,150	0,132	0,120	0,081	0,093	0,088	0,080	0,059	0,043
C50(dB)	-3,198	-3,623	-2,188	-5,784	-5,027	-2,805	-2,662	-1,601	-2,242	-1,714	-0,211	1,093	1,950	-0,741	-3,001
C80(dB)	-1,371	-1,964	-0,531	-4,117	-2,603	-0,244	-0,903	0,909	0,389	1,101	3,020	4,326	5,310	1,790	-0,882
Tt(ms)	408,503	359,433	378,458	392,517	368,980	330,862	318,005	274,286	254,127	208,299	172,245	145,125	138,163	1.368,980	3.274,535
EDTt(s)	2,367	2,135	2,269	2,273	2,135	1,981	1,893	1,675	1,521	1,238	1,027	0,847	0,781	13,240	31,508

Figure 18: Third-octave filtered RIR results for channel L of the signal processed with GGRiRs Software

THIRD-OCTAVE Right	25	31,5	40	50	63	80	100	125	160	200	250	315	400	500	630
EDT(s)	2,359	1,554	3,696	3,277	3,113	3,544	3,143	3,570	2,405	2,560	2,463	2,912	2,732	2,637	2,135
T20(s)	3,230	3,521	3,283	3,900	3,190	3,193	3,365	3,548	3,174	2,719	2,941	2,780	2,694	2,679	2,622
T30(s)	3,491	9,209	3,462	4,110	3,375	3,232	3,231	3,627	3,310	3,034	2,915	2,808	2,804	2,746	2,653
IACC early	1,001	1,000	0,997	0,972	0,978	0,976	0,983	0,976	0,879	0,713	0,597	0,242	0,054	0,294	0,299
C50(dB)	2,175	3,396	0,199	-3,989	-2,634	-10,354	-5,835	-5,532	-4,538	-3,509	-2,946	-3,953	-7,491	-6,668	-2,748
C80(dB)	4,951	4,445	0,747	-2,116	-1,322	-9,425	-4,315	-3,934	-3,138	-2,343	0,329	-2,195	-2,765	-3,270	-1,535
Tt(ms)	404,150	361,927	403,741	564,535	524,444	566,100	469,501	513,175	452,041	474,717	408,458	445,442	446,440	415,170	353,379
EDTt(s)	2,483	2,335	3,466	3,409	3,134	3,497	3,045	3,261	2,460	2,557	2,451	2,903	2,729	2,646	2,122

THIRD-OCTAVE Right	800	1000	1250	1600	2000	2500	3150	4000	5000	6300	8000	10000	12500	16000	20000
EDT(s)	2,381	2,535	2,437	2,143	2,414	2,085	1,827	1,723	1,573	1,277	1,016	0,887	0,798	5,133	31,661
T20(s)	2,718	2,480	2,429	2,432	2,295	2,123	2,026	1,793	1,523	1,314	1,070	0,905	1,046	21,301	20,743
T30(s)	2,717	2,654	2,486	2,421	2,325	2,166	2,011	1,789	1,543	1,348	1,108	0,978	1,459	18,348	16,434
IACC early	0,190	0,225	0,098	0,191	0,104	0,225	0,150	0,132	0,120	0,081	0,093	0,088	0,080	0,059	0,043
C50(dB)	-4,052	-3,074	-5,822	-2,980	-3,975	-4,115	-2,816	-2,361	-2,438	-0,869	-0,013	1,030	2,062	-0,219	-3,278
C80(dB)	-0,981	-1,918	-3,191	-1,232	-1,740	-1,602	-0,637	-0,214	-0,456	2,101	3,175	4,351	5,214	2,173	-1,418
Tt(ms)	403,016	402,721	392,245	372,971	406,190	344,014	314,649	286,803	264,036	211,837	168,639	146,531	141,383	978,027	3.510,499
EDTt(s)	2,391	2,504	2,433	2,145	2,413	2,085	1,832	1,722	1,570	1,278	1,016	0,887	0,807	8,400	30,670

Figure 19: Third-octave filtered RIR results for channel R of the signal processed with GGRiRs Software