RIRs PROCESSING SOFTWARE DEVELOPMENT

Aguirre Rodrigo

Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina
email: aguirrerodrigo1@hotmail.com

Dragone Esteban

Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina
email: estebandragone@gmail.com

Mascaro Julieta

Universidad Nacional de Tres de Febrero, Ingeniería de Sonido, Caseros, Buenos Aires, Argentina
mascaro38644@estudiantes.untref.edu.ar

This document describes the development of a software for the calculation of the main acoustic parameters that describe a room. Using Python language and Qt Designer, a graphical user interface (GUI) is created that allows the processing of monophonic and stereo impulse responses. These types of responses are the ones that allow obtaining binaural acoustic parameters, such as the interaural cross-correlation (IACC). The parameters obtained are calculated according to ~~what is defined by~~ ISO 3382 and ~~their values are~~ compared with commercial software for validation purposes. The results have shown, globally, a high similarity with commercial software that performs the same processes.The clarity parameters showed mostly inferior results and localized in high frequencies. The values of T_{20} and T_{30} presented comparable values with commercial software and with the expected trend. For the EDT, it was possible to verify the dependence and sensitivity that it has with the transition time T_t. In relation to the $IACC_{early}$, an increase in the difference with frequency was observed.

Keywords: Acoustical Parameters, Python Software, Room Impulse Response (RIR)

1. Introduction

Reverberation time is the most widely used parameter to characterize the acoustic behavior of rooms and has a long history of development in terms of theory, measurement and application. In addition to this parameter, acoustic properties are added that define the sound experience of a subject immersed in a room, such as clarity, early and late acoustic energy, lateral energy and interaural cross-correlation. The set of these parameters constitutes a complete acoustic profile of the enclosure.

~~The~~ ISO 3382 [1] series ~~of standards~~ establishes most of the guidelines and conditions for obtaining these acoustic parameters. In general, and following what was presented by Angelo Farina [2], the measurement of the reverberation time is carried out by recording a room impulse response (RIR), often indirectly from a swept sinusoidal signal. After the registration of the RIR, the derivation of the acoustic parameters of interest will be affected by the choice of the software that processes those responses.It is quite common for different types of software to be used in the world of architectural acoustics, in their paid or free versions. Therefore, at the time of processing, professionals always find a wide range of software available.

For this reason, this report aims to develop its own software using the Python programming language that allows the calculation of the following acoustic parameters: T_{20},T_{30}, EDT, C_{50}, C_{80}, T_t and EDT_t and $IACC_{early}$. The results are compared with the Aurora Acoustical Parameters plug-in [3] and EASERA [4] which is presented as a RIR processing software frequently used in the field of room acoustic study.

2. Theoretical Framework

2.1 Impulse response and background noise compensation

ISO 3382 [1] represents the standard that must be followed to calculate acoustic parameters. The impulse response h(t) is the full descriptive quantity for an LTI system, therefore it contains information on all acoustic parameters such as reverberation time and clarity in a room. Using backward integration of the squared impulse response it is possible to obtain a smooth quantity by Schroder integral [5] that is suitable for calculating them all, using equation 1 ~~below:~~

$$ETC(t) \; = \; 10 los_{10}\left[\int_{t}^{T} h^{2}(\tau) d\tau \, / \int_{0}^{T} h^{2}(\tau) d\tau \right] \qquad (1)$$

The results obtained using (1) could be sensitive to background noise, for example, the unrestricted end of the integration could allow noise to be included. These problems have been addressed by many techniques, which Venturi et al. bring together and compare in an article [6]. Lundeby's method is based on an iterative algorithm that chooses the correct time interval for which the linear regression is calculated. This will keep out unwanted contributions. The description of the algorithm is also reported in [6].

2.2 Choice and Design of Fractional Octave Filter Banks

Evaluation of room acoustic parameters is usually done by observing fractions of an octave or thirds of an octave in the frequency spectrum of interest, usually the human hearing range. To do this, the signal corresponding to the registered RIR must be filtered.

Filter banks are specified by the IEC 6120 standard [7], which details the nominal frequencies of each band, the procedures to calculate the upper and lower frequency of each band and the slope of the filter, among other parameters. It is important to remember that the process of filtering the signal in sub-bands causes a certain time delay. For this reason, ISO 3382 [1] recommends first performing a time window and then applying the filters to calculate those parameters that require a time division. This causes the algorithms to perform different procedures to correct said delay. This generates great variability in the acoustic results, for which it is considered that the design of the filters used must be carefully studied when developing and using said RIR processing software.

This is demonstrated by A. Venturi et at.[6] in his work comparing the processing of real and synthetic RIRs to evaluate octave filters with different slopes (from 36dB/octave and 144dB/octave). It is concluded that, having two filters that comply with the regulations, the results of the acoustic parameters may be different. The parameter T_{20} shows greater sensitivity to filter selectivity compared to EDT and T_{30}. The main conclusion of the article was to reach the following suggestion: use soft filters to extract "short" (EDT, T10) parameters so as not to significantly alter the response time and filters with steeper slopes for "long" parameters to reject the energy of the adjacent band in the last part of the IR. The broadband time decay could give a clue as to which slope to choose.

2.3. Acoustical Parameters definitions

2.3.1. Reverberation Time.

Campani and Farina present in their article [8] the reverberation time as the best known and fundamental parameter of acoustic science; this is defined as the time -in seconds- that the sound pressure level needs to decrease by 60 dB. The decay rate is measured by linear least squares regression of the measured decay curve from a level 5 dB below the initial level to 35 dB below, and it is called T_{30}. When there is a very noisy environment, the final value is changed to 25 dB and T_{20} is obtained. It is important to note that even if the decay rate is measured over a range of only 30 or 20 dB, the reverberation time is always expressed as the time required for a 60 dB decay. If the decay is linear, then T_{60} = T_{30} = T_{20}

2.3.2. Clarity

These parameters express a balance between early and late arriving energy, which is useful for measuring the clarity perceived by human ears. The C_{te} parameter is used with $t_e = 50\,ms$, or $t_e = 80\,ms$ depending on the room destination, listening to voice or music, respectively; C_{te} is defined as:

$$C_{te} = 10log_{10}\left[\int_{0}^{t_e} p^2(t)dt \,/\, \int_{t_e}^{\infty} p^2(t)dt\right] \quad (2)$$

where $p(t)$ is the measured impulse response

2.3.3. Transition time

The transition time T_t is defined as the instant in which the accumulated energy of the impulse response reaches 99% of its total energy [9]. This parameter, in its objective form, determines the boundary between the deterministic and stochastic regions of the impulse response.

2.3.4. Interaural Cross Correlation

Human spatial perception comes from the ability of the human auditory system to record stereophonic signals, particularly the recording of the difference between the signals that reach the two ears. If there is no difference between the left and right sounds, it means that there is no spatial information, and then a listener will not be able to locate a sound source in a scene that will appear completely flat to them. With a binaural microphone it is possible to record exactly what reaches two ears, and by means of the cross-correlation operation between these two signals the spatial degree of the information will be revealed: this is the definition of the Interaural Cross-Correlation function ($IACF$), which can be calculated according to Equation 3 ~~presented below~~:

$$IACF_{t1/t2}(\tau) = \left(\int_{t1}^{t2} pl(t) + pr(t+\tau)dt\right) / \sqrt{\int_{t1}^{t2} pl^2(t) \, . \, \int_{t1}^{t2} pr^2(t)\, dt} \quad (3)$$

where $pl(t)$ corresponds to the pressure of the left channell/ ear and $pr(t)$ to the pressure of the right channel/ear

The interaural cross-correlation coefficients ($IACC$) correlates well with spatial impressions of subjective quality in a concert hall. This coefficient is obtained as the maximum of the interaural cross-correlation function ($IACF$), according to Equation 4.

$$IACC_{t1/t2} = max\left(\left|IACF_{t1/t2}(\tau)\right|\right) \; for \; -1\,ms < \tau < +1\,ms \quad (4)$$

$IACC$ can be measured to describe the disparity in the arrival of the signal at the two ears by the first reflections ($t_1 = 0$ and $t_2 = 80\,ms$) or by the reverberant sound ($t_1 = 80\,ms$ and t_2 = a time greater than the time room reverberation). If the calculation is performed to know the first reflections, the $IACC_{early}$ parameter is obtained, while the parameter corresponding to the reverberant field will be $IACC_{late}$. In this report, only the value for $IACC_{early}$ is obtained.

3. State of Art

Most of the published works comparing the calculation of reverberation time from an impulse response, have focused on the efficiency of the algorithms, often proposing improvements or novel approaches to the calculation method. The main works correspond to Schroeder [5]; Chu, who compares the Schroeder method with an averaging method [10]; and Jacobsen and Rindel [11] with their proposal of time reversed decay measurements.

Lundeby et al. [12] and Bradley [13] conducted studies in the 1990s comparing systems for measuring and analyzing reverberation time. Lundeby's study included the entire computational measurement and analysis system, while Bradley used an electronic artificial reverberator to remove physical variability. However, both studies were comparisons of measurement and processing systems, rather than isolating just the part of the software used.

Later studies by Ikegami and Uchida [14] and Katz [15] focused solely on analysis using an impulse response as test input to software. The Katz impulse response (a recording of a balloon bursting in a large room) is of particular interest because of the large scale of the study and because the impulse response has a significant noise level. The RIR recorded by Katz was sent to researchers for analysis, using the software of his choice. The results showed a wide range of reverberation time values in the lowest octave band (125 Hz), for which the signal-to-noise ratio was the poorest. This study is of particular interest because several software used for the tests are still in use, although in updated versions.

Subsequent to Katz's work, some smaller-scale software comparison studies have been published, for example, by Tronchin et al. [16] (four software implementations), Topa et al. [17] (three implementations), Machín et al. [18] (three implementations, two external and one from the authors) and Mansilla et al. [19] (four implementations, three external and one from the authors).

The conclusions of these studies vary: Tronchin et al., using RIRs measured in an opera house, found mean deviations of T30 (reverberation time) of up to 8.3% between software, with a maximum deviation of almost 25%. ; Topa et al. tested three rooms (a school assembly hall, a church, and a university auditorium) and found that the software did not affect the results; the main objective of the study by Machín et al. was to validate the software developed by the authors through a set of measurements in an auditorium and found only minor deviations in the reverberation time between the developed algorithm and commercial software; ~~and~~ Mansilla et al. tested 59 diverse RIRs and found average octave band deviations of approximately 0.3 s at T30 among the external software tested. Alvarez-Morales et al. [20] performed a detailed software comparison based on a large set of on-site measurements in a university auditorium, using four software implementations, with extensive statistical analysis. They found that the implementations mostly matched when using the programs as recommended by the software documentation.

Therefore, the extent to which the results between different software are comparable seems to depend on the type of measurement performed; understanding the entire electroacoustic chain prior to the processing of an RIR (equipment, stimulus signals, types of enclosures and their acoustic treatment)

4. Software Development

4.1. Loading an Impulse Response and Filtering

When starting the software with its respective graphical user interface (GUI), the user can see several options on the screen that can be chosen for the processing of an impulse response. First of all, a signal corresponding to a wav format RIR must be loaded. The developed program allows the loading of three types of audio files: mono, stereo and split stereo. This last format allows the user to work with two mono files corresponding to binaural RIRs that have been recorded with two independent microphones.

For each load option, a function is generated in which the user's choice is taken into account. Taking the example of loading a monophonic RIR, *MonoParam* function was designed to load the audio, using the audioread function of the *Soundfield* Python's library, and then using the *IRm_cut* function, the signal is cut from its maximum peak. If there is an excessive length greater than 15 seconds, the excess reverberation tail is also cut in order to avoid excessively large data being processed.

In the case of stereophonic RIRs, there is a particular function for the case of a stereo file (*StereoParam*) and another function for the case of two L-R files of a stereo split (*StereoSplitParam*). When loading a stereo RIR without channel separation, it is necessary to perform said separation prior to cutting; this action is performed by the *SepStereo* function which receives a stereo RIR and outputs separate L and R channels. In this way, the cutting process can then be executed by a function called *IRs_cut* which contains the *IRm_cut* function applied on each L and R channel of the signal. Additionally, the dimensions of each channel after cutting are made compatible.

After cutting the signal, it is filtered. The user can choose between filtering the signal by octave bands or third octave bands. The filters designed correspond to the regulations IEC 6120 [6] and are IIR second order Butterworth bandpass type. Analogous to the cutting process, in the case of the filtering functions, there are monophonic (*FiltMono*) and stereophonic (*FiltStereo*) options depending on the load of the file by the user.

4.2.Smoothing of the filtered signal

Once the filtered signal is obtained, the smoothing process is applied. For this part of the processing, the user can select through the GUI between two methods: linearization by Schroeder's inverted time integral with noise compensation using the Lundeby's method or linearization using a Moving Median Filter (MMF).

The *Lund* function performs the entire process of the method exposed by Lundeby. This function consists of finding the point where the energy decay and the background noise level intersect to establish the initial limit of the inverted integral in time. To do this, the designed function performs the following steps: first, the noise level of the last 10% of the signal in dB RMS is taken; then divide the signal into intervals and obtain the average using the *mean* function of the *Numpy* library; a linear regression is then applied between the interval of 0dB and the nearest average noise energy + 10dB. If the S/N ratio is insufficient (implying a value greater than -20 dB), new time intervals are calculated for the average. From this value an iterative comparison process begins.

From the decay slope obtained by linear regression over the corresponding time interval, the initial truncation point is determined as the intersection of the slope of the energetic decay curve and the background noise level.

In the event that the user selects the moving median filter option for the smoothing process, the *MovMed* function is designed. The function allows the user to enter the size of the window to use.

4.3 Main process of calculation of acoustic parameters

Once the filtered and smoothed signal is available, the different acoustic parameters of interest are obtained. For the process of calculating the reverberation times EDT, T_{20} and T_{30} *RTparam* function is executed, which saves the results in a list for each defined variable. The index corresponding to the decay range required for each parameter is searched: from -1 to -10 for EDT, -5 to -25 for T_{20} and -5 to -35 for T_{30}. Finally, a linear least squares regression is performed and each corresponding slope is acquired.

To calculate the EDT_t and T_t , a function is developed, *Tt_EDTt*, which receives the IR signal, cuts the first 5 milliseconds, and finds the upper index of the interval in which the signal reaches 99% of its energy. This index marks the transition time in the time vector and is taken to perform the least squares linear regression to obtain the EDT_t.

The clarity parameters, C_{50} and $C_{80}$are obtained through the *Clarityparam* function, which receives as inputs the filtered IR signal and the temporal limits of the integration that must be performed for these parameters according to what is expressed in the Equation 2 in the theoretical framework section of this report.

Finally, the last parameter to calculate is the $IACC_{early}$ binaural parameter. To do this, there is the *IACCearly* function, which expects as input two signals corresponding to IR_L (left channel) and IR_R (right channel) with their corresponding sampling frequency. The function performs the calculation expressed in equations (3) and (4) for a time integral limit equal to 80 milliseconds.

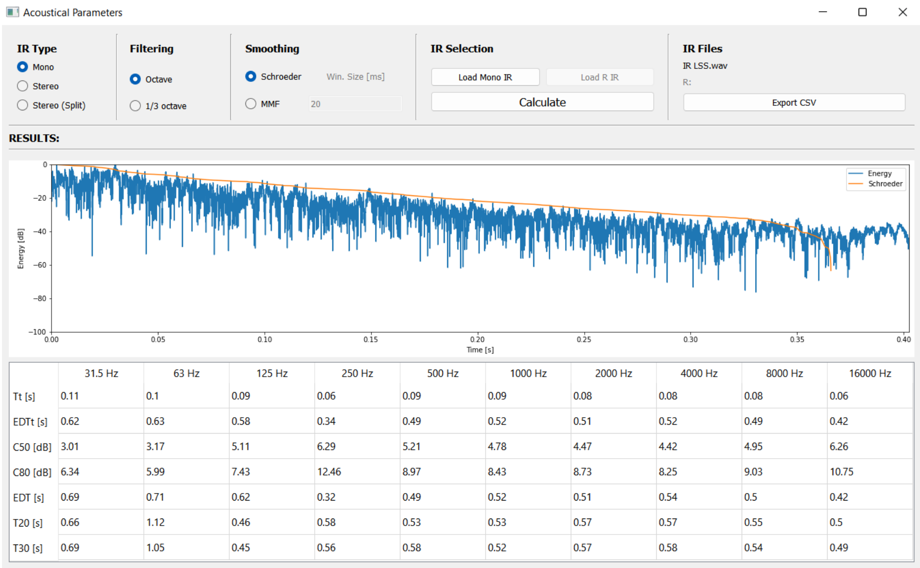
4.5. User errors covered

User errors are considered when loading the audio files of the RIRs. If a monophonic RIR is selected to be processed but a stereo audio file is loaded, the GUI is designed to print a warning to the screen indicating that a monophonic file must be loaded. It operates in the same way in the case of selecting whether to process stereo or split stereo audios and load monophonic audios.

Another error to watch for is the invalid input of window size for the moving median filter. If the user enters values of 0 or negative, the interface displays a warning to enter a valid window size in order to continue.

4.6. GUI and block diagram of the code

Figure 1 below presents the graphical user interface (GUI) developed for the code. Figure 2 shows the block diagram of the signal flow in the code processing.



	31.5 Hz	63 Hz	125 Hz	250 Hz	500 Hz	1000 Hz	2000 Hz	4000 Hz	8000 Hz	16000 Hz
Tt [s]	0.11	0.1	0.09	0.06	0.09	0.09	0.08	0.08	0.08	0.06
EDTt [s]	0.62	0.63	0.58	0.34	0.49	0.52	0.51	0.52	0.49	0.42
C50 [dB]	3.01	3.17	5.11	6.29	5.21	4.78	4.47	4.42	4.95	6.26
C80 [dB]	6.34	5.99	7.43	12.46	8.97	8.43	8.73	8.25	9.03	10.75
EDT [s]	0.69	0.71	0.62	0.32	0.49	0.52	0.51	0.54	0.5	0.42
T20 [s]	0.66	1.12	0.46	0.58	0.53	0.53	0.57	0.57	0.55	0.5
T30 [s]	0.69	1.05	0.45	0.56	0.58	0.52	0.57	0.58	0.54	0.49

Figure 1. Main window of the graphical user interface (GUI)

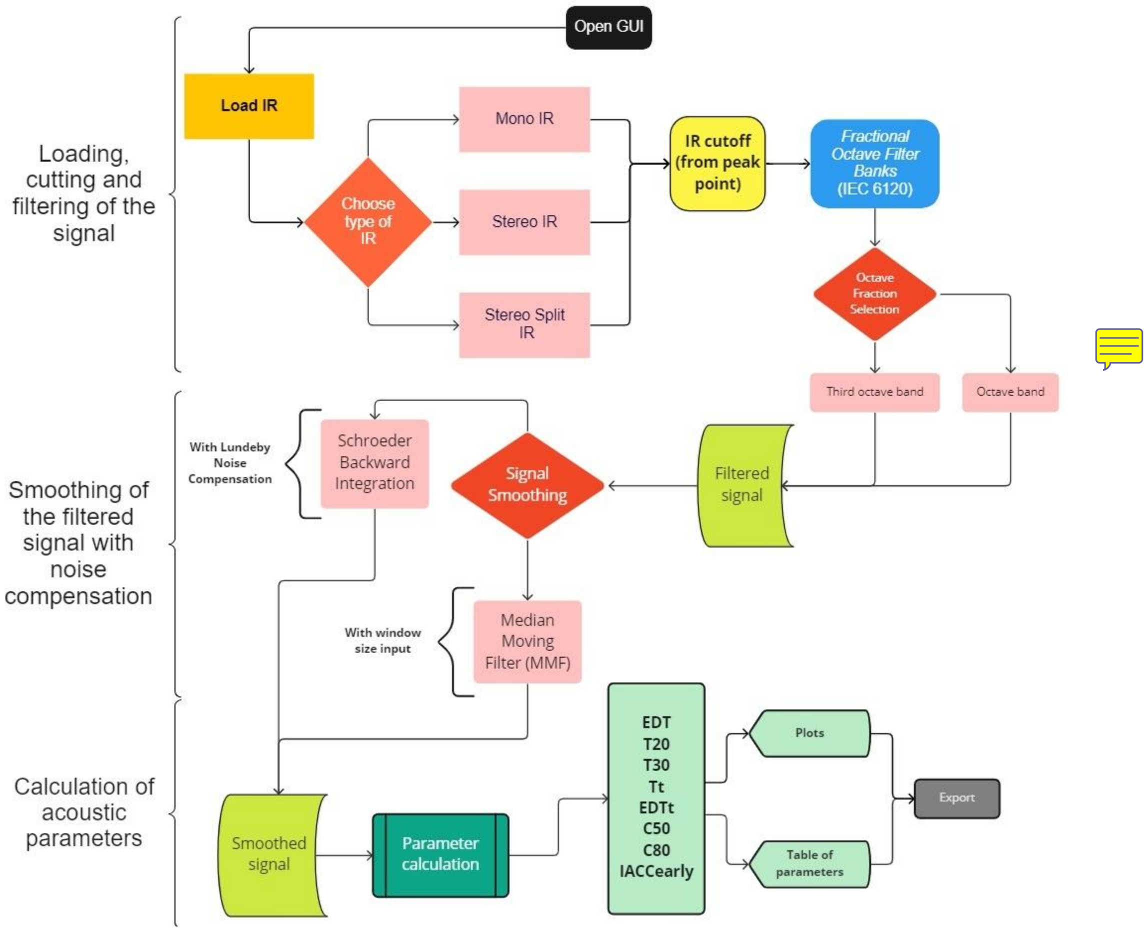


Figura 2. Signal flow of the code developed.

5. Results and Discussion

The developed code and the final graphical interface are available in an online repository for free download. The validation method applied to the developed code corresponds to the comparison of the calculated acoustic parameters with those obtained from Aurora software through the "Acoustic Parameters" function and also from the EASERA software. For all three cases, the same impulse response is applied and, in both software; the signal is windowed from its first maximum to achieve the greatest possible similarity with the code procedure. Figures 3 to 7 show the differences between the acoustic parameters calculated with the code and those obtained from Aurora and EASERA.

Although the code performs the calculation in octaves from 31.5 Hz to 16 kHz; As the EASERA software only allows the calculation from 125 to 8 kHz, in this section the graphs are presented in this interval, to make a correct comparison.

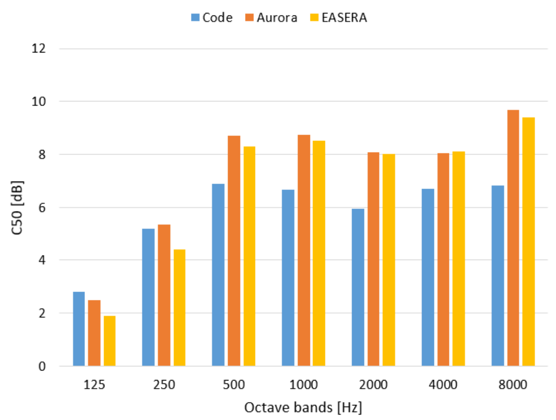


Figure 3. C50 by octave bands calculated using the code (blue), obtained from Aurora (orange) and from EASERA(yellow).

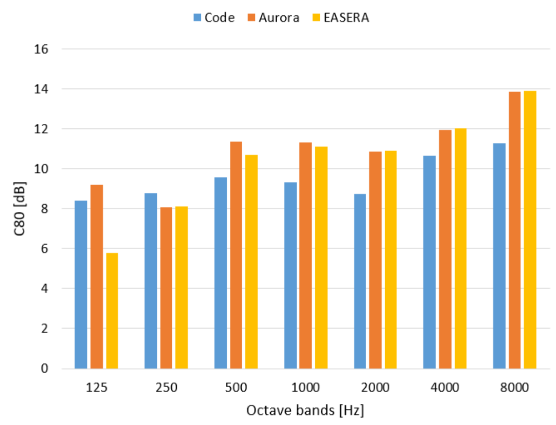


Figure 4. C80 by octave bands calculated using the code (blue), obtained from Aurora (orange) and from EASERA(yellow).

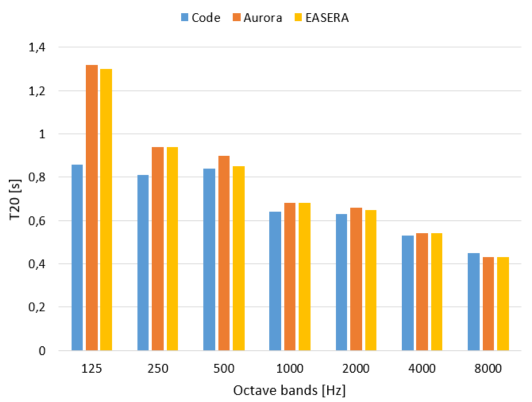


Figure 5. T20 by octave bands calculated using the code (blue), obtained from Aurora (orange) and from EASERA(yellow).

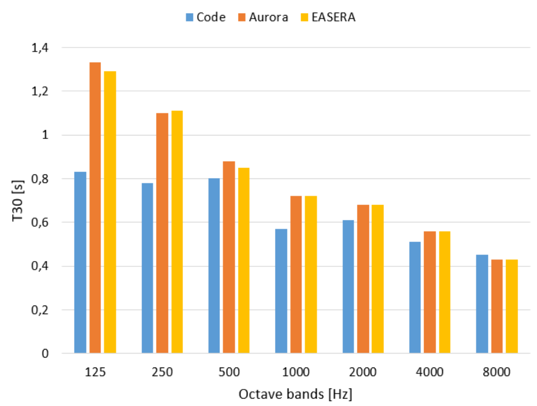


Figure 6. T30 by octave bands calculated using the code (blue), obtained from Aurora (orange) and from EASERA(yellow).

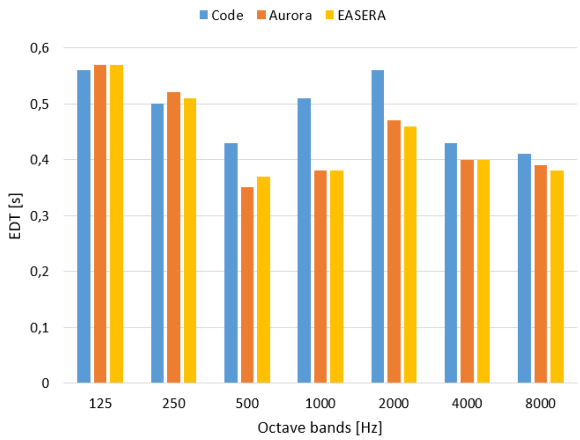


Figure 7. EDT by octave bands calculated using
the code (blue), obtained from Aurora (orange) and from EASERA(yellow).

Regarding the clarity parameters, C_{50} and C_{80} , the two software used for validation present similar results, while the developed code always presents slightly lower values in most of the frequency bands. Above the 500 Hz band, the code shows larger differences with maximum values at high frequencies; particularly for parameter C_{50} there is a difference of 2 dB in the 2 kHz band and almost 3 dB for the 8 kHz band. Similar trends are observed for C_{80}. Towards low frequencies, the result is closer to the software results.

On the other hand, regarding the parameters referred to the reverberation time, in the parameters T_{20} and T_{30}, very similar results are observed in the comparisons for high frequencies. Towards low frequencies (bands of 125 and 250 Hz), the typical RT increase curve is observed in the cases of the results obtained by software, while the code presents a more constant value when lowering the frequencies. This may be due to the fact that it is at low frequencies where the greatest uncertainty of the results is presented, due to the low modal density.

In the case of the EDT parameter, there are results with a noticeable difference between the 500 and 2 kHz bands of around 100 milliseconds between the developed code and the results of Aurora and EASERA. It can be inferred that this parameter is more sensitive to software processing because ~~obtaining~~ it depends on the time definition from which the sound field changes from being deterministic to stochastic and therefore the time chosen to perform the linear regression and obtain the corresponding slope affects the results.
The comparison of EDT_t and T_t was not possible because none of the software used calculates these parameters.

It was also possible to compare the $IACC_{early}$ parameter calculated with the code against that obtained with the EASERA software. The following figure shows the result.

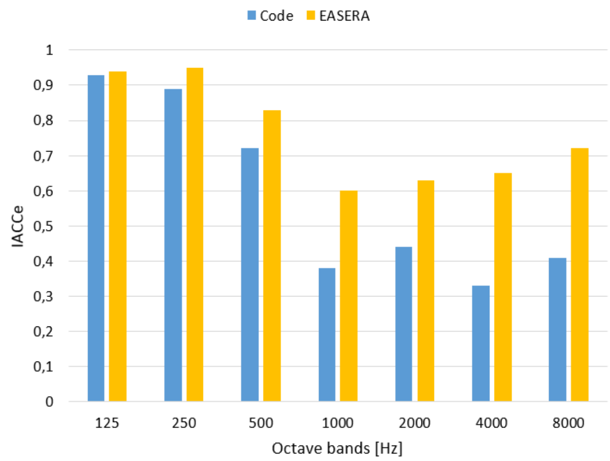


Figure 8. $IACC_{early}$ by octave bands calculated using the code (blue) and obtained from EASERA(yellow).

An increase in the difference with frequency can be observed, reaching up to a ratio difference of 0.3 for the 4 and 8 kHz octave bands. A validation could not be performed with the smoothing option using a moving average filter, MMF, because the software used for it does not have this option.

The following figures show the EDT_t and T_t parameters obtained with the code, applying smoothing using the Schroeder method and using a moving average filter (MMF). For this, a window of 15 ms was applied, considering that bands are analyzed by thirds of octaves and the lowest frequency is 31.5 Hz, a window of 15 ms allows a correct analysis up to the frequency of interest.

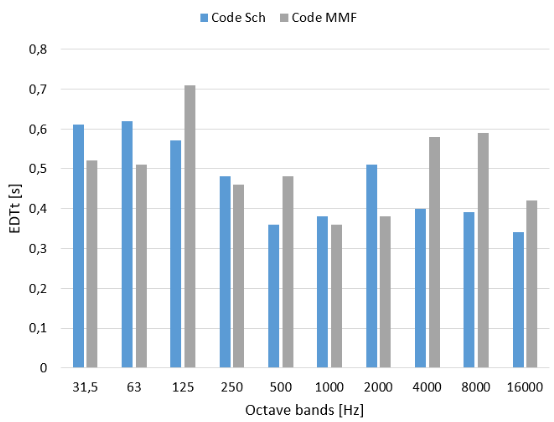


Figure 9. EDTt calculated by octave bands, with smoothing by Schroeder's method (blue) and moving average filter (gray).

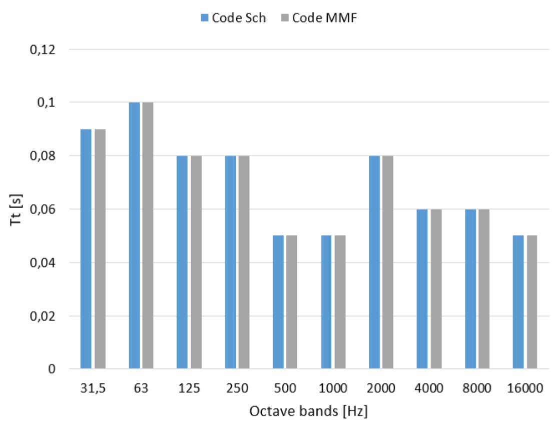


Figure 10. Tt calculated by octave bands, with smoothing by Schroeder's method (blue) and moving average filter (gray).

Similar results are observed in Figure 10 for the two cases, since the transition time parameter is calculated before performing the smoothing.

The Annexe section contains the rest of the acoustic parameters obtained with the code after smoothing using a moving average filter and using the Schroeder method, although they are not comparable. The clarity parameters, C_{50} and C_{80}, are not attached because they present the same result as for smoothing using the Schroeder method, given that this parameter is calculated before it. However, the spreadsheet is available online where the results of all the parameters obtained by the code are saved to access if it is necessary to observe any particular result.

Due to the fact that the chosen commercial software does not use another smoothing method (moving median filter for example), the comparison between linearization methods is carried out internally with the same designed code and its results are presented in the Annex section.

6. Conclusions

The development of a software for the calculation of acoustic parameters under the ISO 3382 standard was carried out successfully and the results for the parameters of reverberation time, clarity and interaural cross-correlation have shown, globally, a high similarity with commercial software. that carry out the same processes.

It was observed that the clarity parameters showed results, for the most part, lower than those calculated by Aurora and by EASERA, but always with differences of less than 3 dB and located at high frequencies. In the case of the reverberation time parameters, a high consistency could be verified with the studies carried out by Katz [15] for the values of these parameters at low frequencies; these values are the ones that expose the main vulnerabilities of an impulse response processing software because at low frequencies there is usually a poorer signal-to-noise ratio than at medium and high frequencies. Added to this is the low modal density at low frequency, which increases the uncertainty of the measurements.

In any case, with the exception of these previous details, the T20 and T30 presented values comparable to commercial software and with the expected trend. For the particular case of the EDT, it was possible to verify the dependence and sensitivity of the correct determination of the transition time Tt, since the result of the EDT will be strongly affected by said time, in which it is defined when the acoustic field goes from being deterministic to stochastic. In addition, it was possible to verify that the transition time is a function of the frequency, for which its calculation must be carried out with the correct care to obtain an EDTt value, which will represent a more reliable analysis of the acoustic characteristics of the acoustic field studied.

The development of a software for the calculation of acoustic parameters under the ISO 3382 standard was carried out successfully and the results for the parameters of reverberation time, clarity and interaural cross-correlation have shown, globally, a high similarity with commercial software. that carry out the same processes.

It was observed that the clarity parameters showed results, for the most part, lower than those calculated by Aurora and by EASERA, but always with differences of less than 3 dB and located at high frequencies. In the case of the reverberation time parameters, a high consistency could be verified with the studies carried out by Katz[ref] for the values of these parameters at low frequencies; These values are the ones that expose the main vulnerabilities of an impulse response processing software because at low frequencies there is usually a poorer signal-to-noise ratio than at medium and high frequencies. Added to this is the low modal density at low frequency, which increases the uncertainty of the measurements. In any case, with the exception of these previous details, the values of T20 and T30 presented values comparable to commercial software and with the expected trend. For the particular case of the EDT, it was possible to verify the dependence and sensitivity of the correct determination of the transition time Tt, since the result of the EDT will be strongly affected by said time, in which it is defined when the acoustic field goes from being deterministic. to stochastic. In addition, it was possible to verify that the transition time is a function of the frequency, for which its calculation must be carried out with the correct care to obtain an EDTt value, which will represent a more reliable analysis of the acoustic characteristics of the acoustic field studied.

In relation to the early interaural cross-correlation binaural parameter ($IACC_{early}$), it was possible to compare it with the EASERA software and an increase in the difference with frequency was observed, reaching a ratio difference of 0.3 for the octave bands 4 and 8 kHz. However, these results are only for smoothing by Schroeder since it has not been possible to carry out a validation with the smoothing option using a moving average filter because EASERA does not have this option. These comparisons are pending to be carried out in future investigations with a greater amount of developed software that allows calculating binary parameters.

7. References

[1] ISO 3382. Acoustics - Measurement of room acoustic parameters. International Organization for Standardization (2009).

[2] Farina, A. Advancements un impulse response measurement by sine sweeps. AES (2007).

[3] Aurora Pulg-ins. Online: http://pcfarina.eng.unipr.it/Aurora_XP/index.htm.

[4] AFMG EASERA. Version 1.2. Professional acoustic and electronic measurements Online: https://www.afmg.eu/en/afmg-easera.

[5] Schroeder, M.: New method of measuring reverberation time. J. Acoust. Soc. Am. 37, 409–412 (1965)

[6] A. Venturi, A. Farina and L. Tronchin. On the effects of pre-processing of impulse responses in the evaluation of acoustic parameters on room acoustics. Session 1aAAa: Advanced Analysis of Room Acoustics: Looking Beyond ISO 3382 I. ICA 2013 Montreal, Canada (2013).

[7] IEC 61260. Electroacoustics - Octave-band and fractional-octave-band filters. International Electrotechnical Commision (2004).

[8] Campani, S., Farina, A. A new Audacity feature: room objective acoustical parameters calculation module. Departement of Industrial Engineering - University of Parma (2014).

[9] A. Bidondo, L. Shtrepi, L. Pepino, A. Astolfi. Analysis of the room acoustic texture parameters in function of diffusers location and distribution inside a small concert hall. Proceedings of the International Symposium on Room Acoustics (2019).

[10] Chu, W.: Comparison of reverberation measurements using Schroeder's impulse method and decay-curve averaging method.J. Acoust. Soc. Am. 63, 1444–1450 (1978)

[11]Jacobsen, F., Rindel, J.: Time reversed decay measurements. J. Sound Vib. 117, 87–190 (1987)

[12] Lundeby, A., Vigran, T., Bietz, H., Vorländer, M.: Uncertainties of measurements in room acoustics. Acustica 81, 344–355 (1995)

[13] Bradley, J.S.: An international comparison of room acoustic measurement systems. Institute for Research in Construction, IRC Internal Report IRC-IR-714 (1996)

[14] Ikegame, M., Uchida, K.: Comparison of various types of acoustic indexes calculated from the same impulse response (in Japanese). In: Proceedings of the Symposium on Impulse Response Measurements and Evaluation of Acoustic Indexes, pp. 7–13, Architectural Institute of Japan (2001)

[15] Katz, B.F.G.: International round robin on room acoustical impulse response analysis software 2004. Acoust. Res. Lett. Onl. 5, 158– 164 (2004)

[16] Tronchin, L., Tarabusi, V. and Bortilitti, A.: On the uncertainties of calculations of acoustical parameters in room acoustics measurements. In: Proceeding of 6th AMTA/WSEAS Conference, Buenos Aires (2005)

[17] Topa,M.D., Toma, N., Kirei, B.S., Homana, I., Neag,M., De Mey, G.: Comparison of different experimental methods for the assessment of the room's acoustics. Acoust. Phys. 57(2), 199–207 (2011)

[18] Machín, J., Arregui, A., San Martín, R., Arana, M.: Areto 1.0: Software para la caracterización acústica de recintos. Techniacustica 2013: 44◦ Congresso Español de Acústica, Valladolid (2013)

[19] Mansilla, J., Sato, S.I., Rodiño, L., Arias, A., Masson, F., De Palma, I.C.: Comparison of room acoustic parameters calculated by different types of software. INTER-NOISE and NOISE-CON Congress and Conference Proceedings. Institute of Noise Control Engineering vol. 250(3), pp. 3333–3343 (2015)

[20] Álvarez-Morales, L., Galindo, M., Girón, S., Zamarreño, T., Cibrián, R.M.: Acoustic characterisation by using different room acoustics software tools: a comparative study. Acta Acust. United Acust. 102 (2016)

8. Annex

The following figures show the comparison of the results obtained by the code developed using two smoothing methods (Schroeder and Moving Median). In the case of the EDT parameter, in the moving median smoothing method there is an evident error in the calculation for the 63 Hz band.

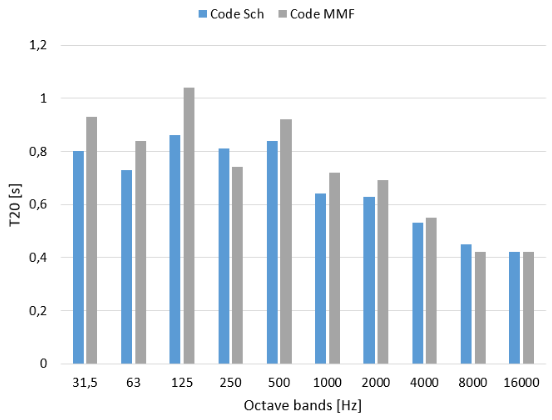


Figure11. T20 calculated by octave bands, with smoothing by Schroeder's method (blue) and moving average filter (gray)

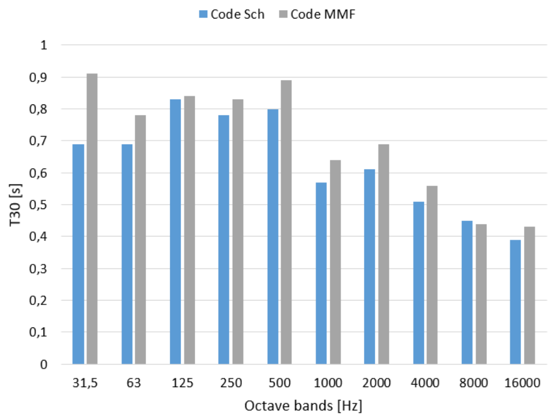


Figure12. T30 calculated by octave bands, with smoothing by Schroeder's method (blue) and moving average filter (gray)

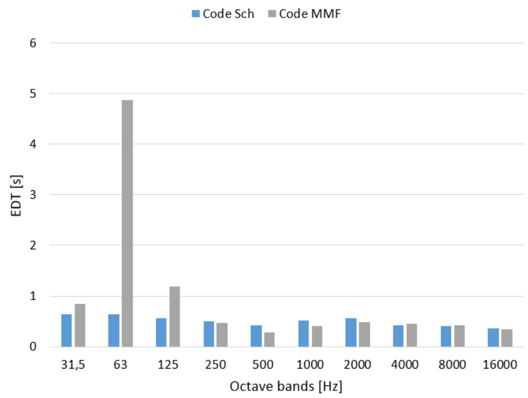


Figure13. EDT calculated by octave bands, with smoothing by Schroeder's method (blue) and moving average filter (gray)