A HIGHLY PARAMETRIZED SCATTERING DELAY NETWORK IMPLEMENTATION FOR INTERACTIVE ROOM AURALIZATION

Marco Fontana, Giorgio Presti, Davide Fantini and Federico Avanzini

Laboratory of Music Informatics (LIM)
University of Milan
Milan, Italy
{name}.{surname}@unimi.it

Arcadio Reyes-Lecuona

Telecommunication Research Institute (TELMA)
University of Málaga
Málaga, Spain
areyes@uma.es

ABSTRACT

Scattering Delay Networks (SDNs) are an interesting approach to artificial reverberation, with parameters tied to the room's physical properties and the computational efficiency of delay networks. This paper presents a highly-parametrized and real-time plugin of an SDN. The SDN plugin allows for interactive room auralization, enabling users to modify the parameters affecting the reverberation in real-time. These parameters include source and receiver positions, room shape and size, and wall absorption properties. This makes our plugin suitable for applications that require real-time and interactive spatial audio rendering, such as virtual or augmented reality frameworks and video games. Additionally, the main contributions of this work include a filter design method for wall sound absorption, as well as plugin features such as air absorption modeling, various output formats (mono, stereo, binaural, and first to fifth order Ambisonics), open sound control (OSC) for controlling source and receiver parameters, and a graphical user interface (GUI). Evaluation tests showed that the reverberation time and the filter design approach are consistent with both theoretical references and real-world measurements. Finally, performance analysis indicated that the SDN plugin requires minimal computational resources.

1. INTRODUCTION

Reverberation can be defined as the persistence in time of sound caused by the reflections of sound waves on surfaces. Artificial reverberation was first studied in the 1920s for music broadcasting and recording using analog methods [1]. The emergence of digital methods in the early 1960s [2] played a crucial role in spreading artificial reverberation to a variety of fields, including film audio post-production, sound design, video games, and extended reality (XR). Since then, several digital methods have been proposed. One possible three-group classification is [3]: convolution-based techniques, physically-based models, and delay networks.

Convolution-based techniques model a reverberant environment as a linear and time-invariant (LTI) system. The reverberation effect is simulated by convolution with a room impulse response (RIR) recorded in the environment of interest. These techniques are accurate, but lack flexibility in control and are computationally expensive, especially for long RIRs. Physically-based approaches simulate the propagation of sound waves from the source to the listener, accounting for the acoustic phenomena occurring in a given environment [4]. These approaches require knowledge of the physical properties of the environment such as the shape, size, and surface materials. Physically-based methods are highly controllable and accurate, but also computationally demanding, and thus can be impractical for real-time applications. Examples of physically-based methods include numerical approximations to the acoustic wave equation, such as finite-difference time-domain (FDTD) [5], and geometrical acoustics techniques like ray tracing [6], and image-source method [7]. The third category of artificial reverberation techniques is represented by delay networks, which are networks of delay lines, digital filters, and feedback connections. Delay networks provide an efficient and controllable solution to artificial reverberation. The simulated reverberation effect is perceptually plausible [8] but not based on a given room geometry with certain physical properties. Therefore, delay networks, as well as convolution techniques, gained popularity in music and art [3]. Conversely, physically-based approaches are well-suited for XR applications as coherence with the visual environment and flexibility for source and receiver positions are required. Notable examples of delay networks include the first one proposed by Schroeder and Logan [2], including later extensions [9, 10], the feedback delay network (FDN) proposed by Jot and Chaigne [11], and the scattering delay network (SDN) [12–14]. SDN is a recent architecture with parameters tied to the physical properties of the room, preserving the computational efficiency of delay networks.

Several implementations of plugins for interactive artificial reverberators have been proposed in the literature. These works have contemplated FDNs [15], digital waveguide meshes (DWMs) [16] and convolution-based reverberation [17,18], among others. SDNs have been scarcely investigated in this regard. Yeoward et al. [19] developed a real-time binaural room modeling plugin using SDN for reverberation, but no public application featuring this method has been released. Additionally, the interactivity of their plugin is limited to three degrees of freedom (DOF) for receiver rotation, and material absorption is modeled by only a single attenuation coefficient. Thus, there is currently no publicly available real-time implementation of an SDN, to the best of our knowledge. This work proposes such an implementation, which is freely available as both a VST3 (virtual studio technology 3) plugin and a standalone application[1]. Our SDN plugin provides interactive room auralization, allowing real-time adjustment of parameters that affect reverberation. These parameters regard the source and receiver, the room dimensions, and the wall absorption properties. These features make our plugin suitable for various applications

[1] https://github.com/LIMUNIMI/Real-time-SDN

involving spatial audio, such as creating virtual rooms or emulating real-world environments for virtual (VR) or augmented reality (AR), respectively. Additionally, this work presents a method for dynamically designing filters based on the absorption coefficients per octave band of given materials. Other features include air absorption modeling, support for various output formats , OSC for controlling source and receiver parameters, and a graphical user interface (GUI).

This paper is organized as follows. Section 2 provides a theoretical introduction to SDNs. Section 3 describes our implementation of the SDN plugin, while Section 4 reports the evaluation tests performed with the plugin. Section 5 concludes the paper.

2. SCATTERING DELAY NETWORKS

2.1. Background

As physically-based models can be computationally demanding, some researchers have investigated the design of delay networks controlled by parameters with a physical interpretation. Examples include the circulant feedback delay networks (FDN) [20] and the digital waveguide networks (DWN) [21]. A DWN is a collection of multiple digital waveguides interconnected by scattering junctions. Digital waveguides are bidirectional delay lines at some impedance, while the scattering junctions model the signal scattering that occurs between waveguides. A DWN with few junctions to model a rectangular room [22] inspired the design of the scattering delay network (SDN) structure.

2.2. SDN overview

The scattering delay network (SDN) is an artificial reverberator originally proposed by De Sena et al. [12] and later refined [13, 14] and extended [23–26]. The authors' goal was to provide an efficient, interactive, and scalable synthesis of room reverberation for video games, while providing explicit control over physical properties of the room. SDN's parameters are inherited directly from the room geometry and the absorption properties of the walls. SDN correctly renders in time and space the first-order reflections and the line-of-sight path between the source and the receiver. On the other hand, the accuracy of higher-order reflections decreases as the reflection order increases. This results from the SDN structure (see Fig. 1) which models a room having a source node, a receiver node, and a scattering node placed on each wall where the first-order reflections occur. The scattering nodes are connected to each other, as well as to the source and receiver nodes. Additionally, the source and receiver nodes are connected by a unidirectional delay line for the line-of-sight path. This section assumes that the enclosure modeled by the SDN is a rectangular room, although extensions to arbitrary polyhedra are possible. For the sake of simplicity, Fig. 1 depicts the SDN structure in a two-dimensional rectangular room, however, our implementation supports 3D shoebox rooms, but not arbitrary polyhedra.

2.3. SDN structure

2.3.1. Scattering nodes

In SDNs, scattering nodes are connected to each other by bidirectional delay lines to approximate the paths of higher-order reflections. Each scattering node multiplies the input signal $\mathbf{p}^+ = [p_1^+, \ldots, p_K^+]^\intercal$ received from the other K scattering nodes by the $K \times K$ scattering matrix $\mathbf{S}$ to give the output $\mathbf{p}^- = [p_1^-, \ldots, p_K^-]^\intercal$:

$$\mathbf{p}^-[n] = \mathbf{S}\mathbf{p}^+[n] \tag{1}$$

Given the sample rate f_s and the speed of sound c, the delay line length $D_{k,j}$ between the nodes at locations x_k and x_j is:

$$D_{k,j} = \lfloor f_s \|x_k - x_j\|/c \rfloor. \tag{2}$$

In the scattering node, energy absorption is modeled by the random-incidence absorption coefficient α, which is defined in the ISO 354 standard [27] and is known for many materials [28]. The wall absorption effect can be expressed as $\mathbf{S}^*\mathbf{Y}\mathbf{S} = (1-\alpha)\mathbf{Y}$, where $\mathbf{Y}$ is the Hermitian positive-definite matrix [20] and $(\cdot)^*$ is the conjugate transpose. Given the reflection coefficient $\beta = \sqrt{1-\alpha}$, $\mathbf{S}$ can be expressed as $\mathbf{S} = \beta\mathbf{A}$, where $\mathbf{A}$ is a lossless scattering matrix (see Eq. (6)). This lets the energy loss at the SDN walls match the corresponding physical room resulting in an energy decay rate consistent with the Sabine [29] and Eyring [30] equations. To model the frequency-dependent nature of the absorption properties of real materials, we can use the scattering matrix $\mathbf{S} = \mathbf{H}(z)\mathbf{A}$, where $\mathbf{H}(z)$ is a diagonal matrix composed of the wall filter $H(z)$. $H(z)$ is defined by the absorption coefficients of a given material, which are usually defined per octave band.

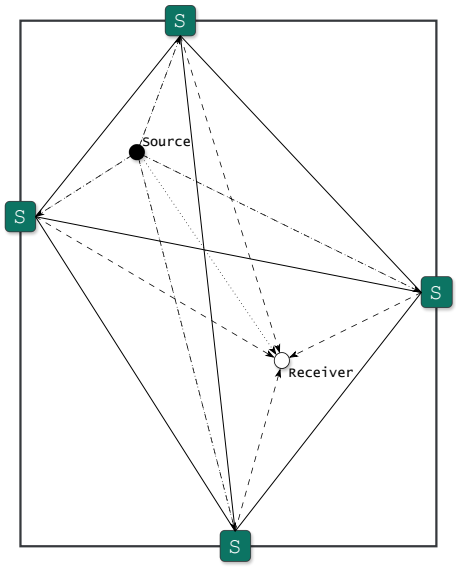


Figure 1: *Two-dimensional schema of the SDN structure adapted from [14, Fig. 4]. The blocks S correspond to the SDN wall nodes. Solid lines represent bidirectional delay lines connecting the wall nodes to each other. Dash-dotted lines represent the unidirectional delay line connecting the source to the wall nodes. Dashed lines represent the unidirectional delay lines connecting the wall nodes to the receiver. The dotted line represents the line-of-sight path.*

2.3.2. Source to scattering nodes connection

The input of the system is represented by the source node, which is connected to the scattering nodes through unidirectional absorbing delay lines. The delay line length $D_{S,k}$ between the source at position x_S and the k-th scattering node at position x_k is computed

as in Eq. (2). The attenuation $g_{s,k} = 1/\|x_S - x_k\|$ is also applied for reproducing spherical spreading.

2.3.3. Scattering nodes to receiver connection

The scattering nodes are connected to the receiver node through unidirectional absorbing delay lines. As previously stated, the outgoing wave variables $\mathbf{p}^-[n]$ of a scattering node are transmitted to neighboring scattering nodes. On the other hand, the signal $p_e[n] = \mathbf{w}^T\mathbf{p}^-[n]$ is the output from the scattering node to the receiver node, and is obtained by a linear combination of the wave variables $\mathbf{p}^-[n]$ with the weights $\mathbf{w}$.

The delay line length $D_{k,R}$ between the k-th scattering node at position x_k and the receiver at position x_R is computed as in Eq. (2). Further, the attenuation is defined as follows to obtain the correct value for the first-order reflections [14]:

$$g_{k,R} = \frac{1}{1 + \frac{\|x_k - x_R\|}{\|x_S - x_k\|}}. \tag{3}$$

3. IMPLEMENTATION

3.1. Features and user interface

This work presents a real-time implementation of an SDN that models a 3D shoebox room, exposing several parameters and with low computational cost. The SDN plugin has been implemented using the C++ JUCE framework[2] and is available as both a VST3 and a standalone application. Our plugin allows for interactive room auralization, enabling the real-time adjustment of parameters that affect reverberation. These parameters include the source and receiver positions, the receiver orientation (yaw, pitch, roll), the three dimensions of the shoebox room, and the absorption coefficients of the walls. Absorption coefficients are defined per-octave, and some example presets are provided in the implementation. The position and rotation parameters are smoothed over 20 ms to avoid artifacts. Further, the plugin allows to mute the direct path rendering and to receive OSC messages for controlling receiver and source parameters. The graphical user interface (GUI) shown in Fig. 2 provides a user-friendly interaction with the plugin.

The plugin natively supports mono, stereo and first to fifth order Ambisonics as output formats. Additionally, the binaural rendering toolbox (BRT) [31] is used to obtain binaural output format. With this, custom head-related transfer functions (HRTFs) can be loaded in the plugin using the SOFA format [32]. Additionally, the rendering features of the 3D Tune-In Toolkit [33] algorithms are used. These features include ear parallax correction for sources at a distance different from that of the measured HRTF, and smooth interpolation of time-aligned head-related impulse responses (HRIRs) followed by the addition of the corresponding ITD to avoid comb filtering. In our SDN plugin, we used the Eigen C++ library[3] to handle matrix and vector algebra as well as quaternion interpolation.

3.2. Wall filters

Different methods can be used to describe the impact of specific materials on the acoustics of a room. The most common one encompasses the definition of absorption coefficients by octave band [34]. These coefficients represent the ratio between the absorbed and incident energies and are typically measured in octave bands ranging from 125 Hz to 4 kHz, while the 8 and 16 kHz bands are less frequently considered.

We designed a single, third-order, infinite impulse response (IIR) filter simulating the acoustics absorption of a given material. This type of filter provides a close match of the absorption coefficients of real-world materials (see Section 4.2), resulting in a natural-sounding output, while maintaining a low computational cost at runtime. The filter design approach used the weighted least-squares method through a C++ adaptation of the MATLAB function `invfreqz`[4], which is based on the work of Levi [35] and Dennis et al. [36]. This function has been proven to provide a good fit for sparsely sampled data [34].

To achieve satisfactory results, the absorption coefficients α must be processed in two steps. First, the coefficients α are converted to reflectance R through the following relation:

$$|R(j\omega)| = \sqrt{1 - \alpha(\omega)}. \tag{4}$$

Secondly, to cover the entire spectrum, the values at the Nyquist frequency and at 0 Hz are extrapolated from the 16 kHz and 125 Hz points, respectively. Linear interpolation is used to create a denser set of points, which facilitates filter design. To obtain a minimum phase filter, we computed the minimum phase frequency response by extracting its cepstrum, folding it around time zero, and adding the cepstrum anti-causal part to its causal part [37].

As recommended by Huopaniemi [38], we used weights based on psychoacoustical properties of human hearing to improve the fitting method. We computed the weight for each frequency point f_c as $1/f_{ERB}$ where f_{ERB} is the equivalent rectangular bandwidth (ERB) defined as follows:

$$f_{ERB} = 24.7(4.37 f_c + 1). \tag{5}$$

Our C++ adaptation of `invfreqz` produces only real-valued filters, unlike the original MATLAB implementation. The plugin feature that yields the highest CPU usage is the filter design step, which needs to be performed whenever the absorption coefficients are changed. Even still, it does not impede seamless real-time sound processing.

3.3. Air absorption

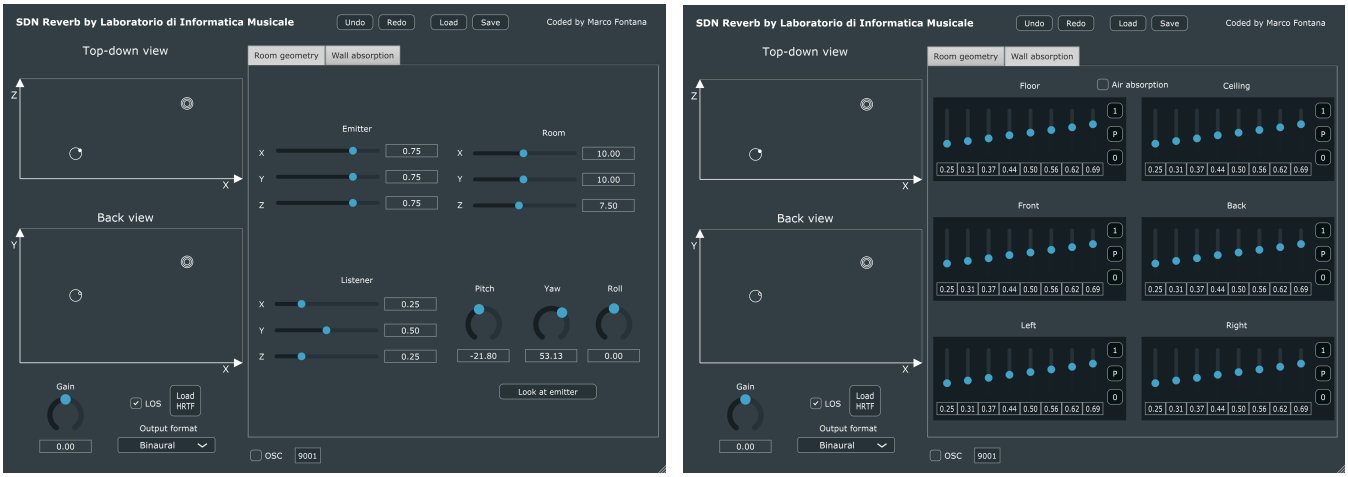
In our SDN, we simulated the frequency-dependent effect of air absorption to achieve a more realistic reverberation effect following an existing approach [38]. This approach is suitable for reverberators not based on convolution, in contrast to other methods [39, 40]. To this end, we integrated low-pass filters into the system's waveguides, as recommended by De Sena et al. [14]. Each waveguide is equipped with a filter whose attenuation curve depends on the length of the waveguide propagation path. The damping for each frequency band was estimated using the ISO 1993 standardized equations for sound absorption in air [38]. Temperature of 20°C and humidity of 20% were assumed.

The air absorption filters were designed following the same approach described in Section 3.2. Third-order IIR filters were used as they are sufficient to accurately fit the desired response. The filter coefficients are precomputed at system startup with a resolution of 1 meter to reduce runtime computations.

[2] `https://juce.com/`

[3] `https://eigen.tuxfamily.org`

[4] `https://www.mathworks.com/help/signal/ref/invfreqz.html`



(a) Room geometry view

(b) Wall absorption view

Figure 2: *Graphical user interface (GUI) of the implemented SDN plugin.*

3.4. Code structure

The C++ code for our SDN plugin was developed using the JUCE framework as a set of classes that interact with each other, namely `Room`, `Source`, `Receiver`, and `ScatteringNode`. The main container of the simulation is the `Room` class, which includes information on the room's geometry, as well as the scattering node and waveguide instances. The bidirectional waveguides of the node-to-node connection are implemented as pairs of unidirectional connections. We modeled the delay lines of the node-to-node connections using circular buffers, with fractional delays handled with a first-order all-pass filter approach. The `Source` class implements the sound origin. It collapses multichannel input to a mono signal by summing all channels. The amplitude of the input signal is then adjusted according to the gain parameter.

The `ScatteringNode` class implements the scattering junctions. This class accounts for incoming and outgoing waveguides, as well as signal scattering and material absorption filtering. As suggested by De Sena et al. [14], we used the Householder isotropic scattering matrix $\mathbf{A}$ for the scattering operation:

$$\mathbf{A} = \pm\frac{2}{K}\mathbf{1}\mathbf{1}^{\intercal} - \mathbf{I}, \tag{6}$$

where $\mathbf{1} = [1, \ldots, 1]^{\intercal}$ and $\mathbf{I}$ is the identity matrix. With this scattering matrix, the incoming wave from a connected node is uniformly reflected, sending equal amounts of energy to the remaining scattering nodes through the outgoing connections. The filters are implemented as IIR direct form II filters.

The `Receiver` class models the receiver node and handles output format initialization, encoding and switching. For the purpose of spatialization, each node-to-receiver connection is treated as a distinct virtual source. For stereo encoding, each source is individually placed in the stereo panorama using the square root panning function. Similarly, for binaural encoding, each source is individually processed using a `CListenerHRTFbasedModel` from the BRT library, which performs the corresponding convolution with HRTFs considering the needed interpolations, as described in Section 3.1. The ACN/N3D standard is used for Ambisonics encoding [41]. The implementation of this standard was adapted from the `getRSH_recur()` function in the Spatial Audio Framework[5]. Additionally, to save processing power, the required buffers are pre-allocated accounting for the highest Ambisonics order supported by the application.

4. EVALUATION TESTS

4.1. Reverberation time

4.1.1. Influence of room properties

The reverb duration is typically quantified by the reverberation time T_{60}, defined as the time it takes for the sound to decay by 60 dB after it ceases. To analyze the T_{60} of our SDN plugin, we replicated the tests performed by De Sena et al. [14] with equivalent settings to obtain comparable results. Thus, we considered a cubic room with side length l, and we measured T_{60} as the room size and absorption properties varied. The source and receiver were positioned at the room center with the line-of-sight component muted. RIRs were generated using the sine sweep approach [42]. The T_{60} values were calculated using the Aurora plugin[6] to obtain results compliant with the ISO 3382 standard [43].

As a first test, we measured T_{60} as the room size and frequency-independent absorption properties varied. Therefore, we set all absorption coefficients to the same value $\hat{\alpha}$. Fig. 3 shows the T_{60} values for room side lengths $l \in \{1, 2, \ldots, 10\}$ and absorption coefficients fixed to $\hat{\alpha} = 0.5$. On the other hand, Fig. 4 shows the T_{60} values for $\hat{\alpha} \in \{0.1, 0.2, \ldots, 1\}$ and room size fixed to $l = 5$ meters. The reported T_{60} values are consistent with those reported by De Sena et al. [14] and the Eyring and Sabine's equations.

A second test analyzed T_{60} by setting the frequency-dependent absorption coefficients of a cotton carpet [28] to each wall. Fig. 5 shows the obtained T_{60} for each octave-band between 125 Hz and 16 kHz. Some differences with the results by De Sena et al. [14,

[5] `https://leomccormack.github.io/Spatial_Audio_Framework/`

[6] `http://pcfarina.eng.unipr.it/Aurora_XP/index.htm`

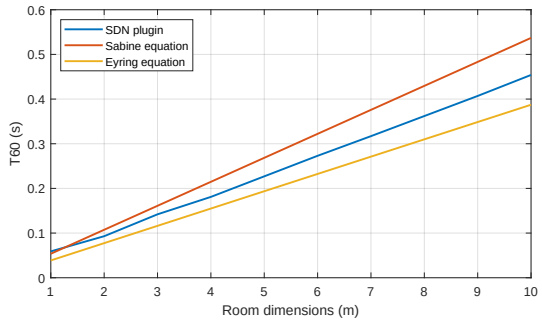


Figure 3: T_{60} *values as a function of the room side length* l *with fixed uniform absorption coefficient* $\hat{\alpha} = 0.5$.

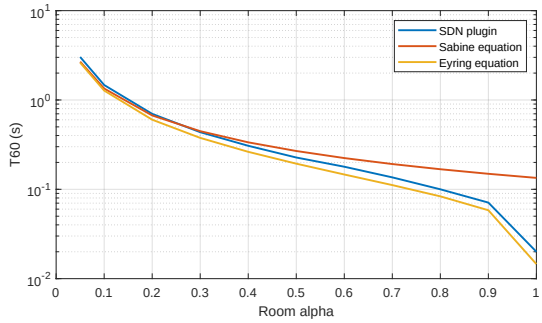


Figure 4: T_{60} *values as a function of the uniform absorption coefficient* $\hat{\alpha}$ *with fixed room side length* $l = 5$ *m.*

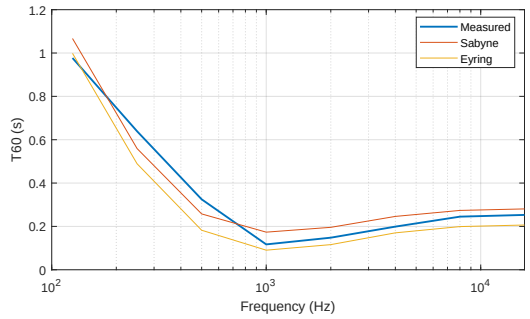


Figure 5: T_{60} *values per frequency band obtained with absorption coefficients of a cotton carpet and room size* $l = 5$ *m.*

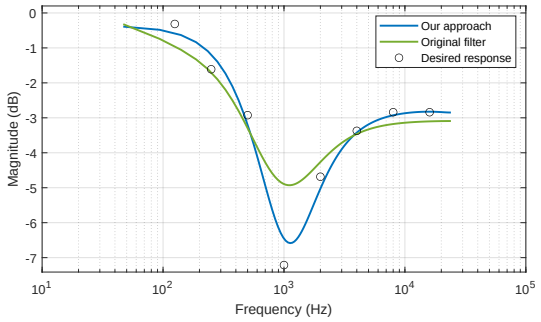


Figure 6: *Absorption coefficients converted to reflectance in dB of a cotton carpet (black circles) compared with corresponding frequency responses of the filter obtained with our design approach (blue) and with filter coefficients reported in [14] (green).*

Fig. 14] were observed and they can be attributed to our different filter design method. This is highlighted by Fig. 6 where the filter frequency response obtained with our approach is compared to the response obtained with the filter coefficients reported in [14].

4.1.2. Comparison with real RIRs

To further analyze the reverberation time of the SDN plugin, we compared the T_{60} of a real room with a room simulated by the plugin, constructed to have similar characteristics. The Benchmark for Room Acoustical Simulation (BRAS) [44] dataset was used as it contains all the required data, such as RIRs, room dimensions, source and receiver positions, and absorption coefficients of the surface materials. We chose the complex room 2 (CR2) in BRAS as it has a nearly shoebox shape and minimal furnishings. Fig. 7 depicts the simplified views of the CR2 shape. The RIRs measured in CR2 with the Genelec 8020c loudspeaker in the LS01 position and all the receiver positions were considered. The source and receiver positions in the SDN plugin were set to match those annotated in the dataset. Similarly, the absorption coefficients for each wall were set to the initial estimates of the coefficients reported in BRAS. The coefficients of the back wall were set to those of the windows because they cover almost the entire wall surface. The dimensions of the room simulated by the SDN have been set slightly smaller than its bounding box, as CR2 is not exactly a shoebox room. We computed the root mean squared error (RMSE) between the T_{60} values per frequency band measured for the RIRs of CR2 and the RIRs generated by our plugin with the corresponding parameters. The obtained average RMSE is 0.196 seconds. Fig. 8 shows an example of the T_{60} comparison for one of the RIRs of CR2. Similar values between the two conditions are observed, with only minor differences in lower frequency bands [7].

4.2. Filter design evaluation

The proposed filter design approach was evaluated using absorption coefficients measured on 61 real-world materials[8]. The absorption coefficients converted to reflectance (see Eq. (4)) were compared with the frequency response of the corresponding filters using spectral distortion (SD) [47]. The obtained average SD is 0.311 dB with a standard deviation of 0.561 dB. Fig. 9 shows four examples of this comparison with Figs. 9c and 9d being the best and worst match cases, respectively. We can observe in this figure that the filter curves closely match the coefficients.

The filter design approach was also evaluated using the BRAS dataset, which provides RIRs recorded for single reflections on given materials in its reference scene 1 (RS1). RS1 includes a RIR recorded in a hemi-anechoic chamber with the floor covered by a sheet of RockFon absorbing material. The SDN parameters were set to simulate a room with identical dimensions to the hemi-anechoic chamber. The receiver and source positions matched the LS01 and MP01 coordinates annotated in the dataset. The floor absorption coefficients were set to match those of RockFon at a

[7] Just noticeable difference (JND) for T_{60} values is not unanimously agreed upon from the literature [45, 46] and as such is not included in the evaluation.

[8] `https://www.acoustic-supplies.com/absorption-coefficient-chart/`

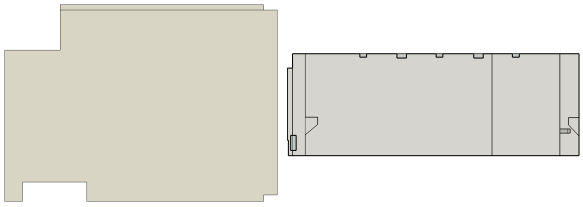


Figure 7: *Simplified view of the floor (left) and side wall (right) for the room CR2 of BRAS dataset [44].*

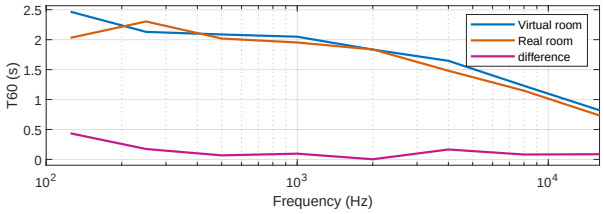


Figure 8: *Comparison of T_{60} per frequency band between the RIR of CR2 in the BRAS dataset [44] for LS01 and MP01 positions (pink) and the RIR of a virtual room with similar characteristics generated by SDN (blue).*

60° angle. Reflections from other walls in the SDN were muted. Fig. 10 compares the spectrum of the RS1 RIR with the corresponding RIR generated by the SDN. Although the real RIR exhibits larger variations in magnitude, particularly at high frequencies, the SDN RIR accurately follows the general trend of the real counterpart. The SD between the two RIRs is 2.868 dB.

4.3. Performance analysis

The percentage CPU load of the SDN plugin was measured in different real-time scenarios on a laptop equipped with an Intel Core i5-11400H processor. Table 1 shows the results for the VST3 plugin running in the Reaper digital audio workstation. The reported values were measured in the worst-case scenario with rapid and continuous changes of the involved parameters. Satisfactory results have been achieved as the CPU never exceeds 2.6%. In particular, for source and receiver transformation, CPU load remains below 2% even for 5th-order Ambisonics. Furthermore, the CPU load increment from stereo to 5th-order Ambisonics is negligible as it never exceeds 0.6%. The changes in wall absorption coefficients have a larger impact on the CPU load than any other parameter, although it is only slightly higher than 2%. Finally, the air absorption simulation only increases the CPU load by about 0.1%, while OSC messages for parameter control have no impact at all.

5. CONCLUSION

This paper presents a highly-parametrized and real-time implementation of an SDN plugin for artificial reverberation. The plugin enables interactive room auralization controlled by parameters tied to physical properties, such as source and receiver positions, room shape and size, and wall absorption properties. These features make our plugin suitable for applications in VR and AR settings, as well as video games, where interactive and real-time spatial audio rendering is essential. Evaluation tests showed that the plugin has minimal impact on CPU load. Additionally, the plugin accurately simulates reverberation time and wall absorption properties of the considered rooms, closely matching both theoretical references and real-world measurements. Future studies will include a more comprehensive evaluation using a larger sample of real-world RIRs. However, this evaluation is limited by the availability of RIR datasets containing all the needed information.

Table 1: *Percentage CPU usage (%) of the SDN plugin for different scenarios and output formats.*

Scenario	Static scene	Source and receiver transformation	Absorption coefficients changing
Mono	0.3	0.8	2.1
Stereo	0.3	1.0	2.2
Binaural	0.8	1.2	2.6
1st order Ambisonics	0.3	1.0	2.2
2nd order Ambisonics	0.3	1.0	2.2
3rd order Ambisonics	0.4	1.2	2.2
4th order Ambisonics	0.4	1.4	2.2
5th order Ambisonics	0.4	1.6	2.2

Future development could take various paths. Possible extensions for the SDN plugin include rendering multiple sources and receivers, modeling room shapes different from the shoebox, incorporating source directivity, and implementing other kinds of scattering matrix. A further evaluation of the computational performance will be conducted by comparing the SDN with other reverberation methods. Additionally, a perceptual evaluation of the plugin should be conducted to study concepts like "naturalness" and "envelopment" [19]. Further, the SDN spatialized reverberation could be investigated with ecological experiments exploring the Reality-Virtuality continuum, ranging from pure virtual reality to audio augmented reality and virtuality [48]. Finally, we plan to automatically tune the SDN parameters to match a target RIR by adapting our previously proposed approach [49].

6. ACKNOWLEDGMENTS

This work is part of SONICOM, a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017743.

7. REFERENCES

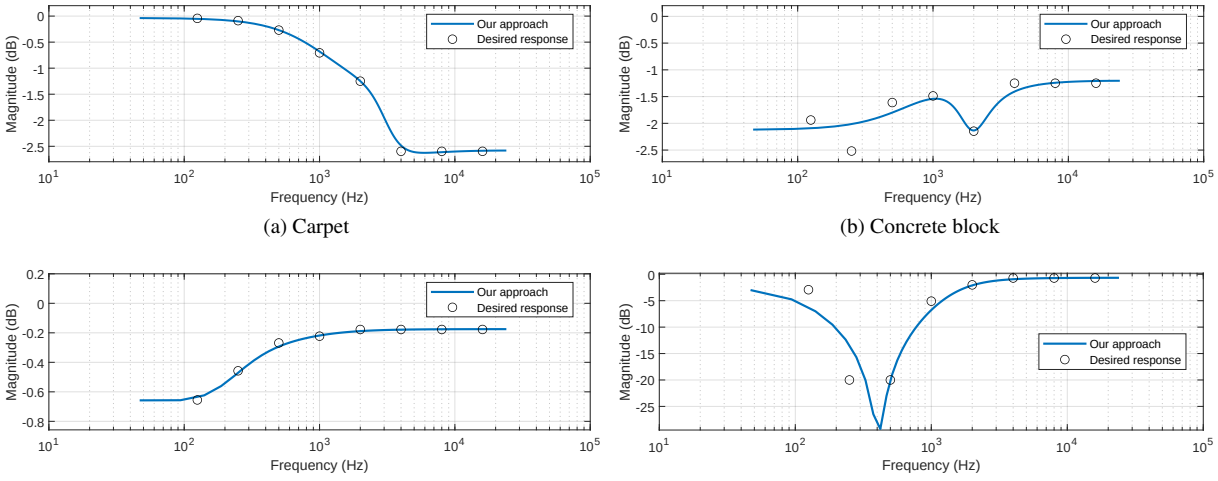
[1] Round Henry Joseph and West Arthur Gilbert Dixon, "Transmission and reproduction of sound," 1932, US Patent.

[2] Manfred R Schroeder and Benjamin F Logan, ""Colorless" artificial reverberation," *IRE Transactions on Audio*, vol. AU-9, no. 6, pp. 209–214, 1961.

[3] Vesa Valimaki, Julian D. Parker, Lauri Savioja, Julius O. Smith, and Jonathan S. Abel, "Fifty years of artificial reverberation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 5, pp. 1421–1448, 2012.

[4] Lauri Savioja and U Peter Svensson, "Overview of geometrical room acoustic modeling techniques," *The Journal of the Acoustical Society of America*, vol. 138, no. 2, pp. 708–730, 2015.

[5] Lauri Savioja, Timo Rinne, and Tapio Takala, "Simulation of room acoustics with a 3-D finite difference mesh," in *International Computer Music Conference*. 1994, pp. 463–466, International Computer Music Association ICMA.



(a) Carpet

(b) Concrete block

(c) Plaster (gypsum or lime, rough finish or timber lath)

(d) Pageboard over 75mm(3") fiberglass board

Figure 9: *Absorption coefficients converted to reflectance in dB of four real-world materials (black circles) compared with the frequency responses obtained with our filter design approach (blue line). The best (c) and worst (d) match cases from dataset are shown.*

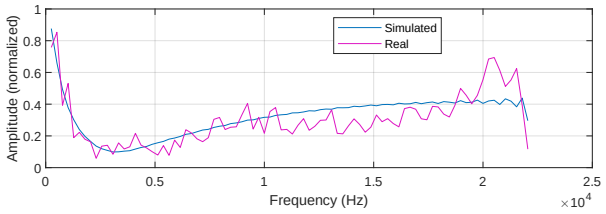


Figure 10: *Comparison of the spectra for the RIR of RS1 at positions LS01 and MP01 reported in the BRAS dataset [44] (purple) and the corresponding RIR generated by SDN (blue).*

[6] Asbjørn Krokstad, Staffan Strom, and Svein Sørsdal, "Calculating the acoustical room response by the use of a ray tracing technique," *Journal of Sound and Vibration*, vol. 8, no. 1, pp. 118–125, 1968.

[7] Jont B Allen and David A Berkley, "Image method for efficiently simulating small-room acoustics," *The Journal of the Acoustical Society of America*, vol. 65, no. 4, pp. 943–950, 1979.

[8] William G. Gardner, *Reverberation algorithms*, pp. 85–131, Springer, Boston, MA, 1998.

[9] Michael A. Gerzon, "Synthetic stereo reverberation: Part one," *Studio Sound*, vol. 13, no. 12, pp. 632–635, 1971.

[10] James A. Moorer, "About this reverberation business," *Computer music journal*, vol. 3, pp. 13–28, 1979.

[11] Jean-Marc Jot and Antoine Chaigne, "Digital delay networks for designing artificial reverberators," in *AES 90th Convention*. Audio Engineering Society, 1991.

[12] Enzo De Sena, Hüseyin Hacıhabiboğlu, and Zoran Cvetković, "Scattering delay network: An interactive reverberator for computer games," in *AES 41st International Conference*. Audio Engineering Society, 2011.

[13] Hüseyin Hacıhabiboğlu, Enzo De Sena, and Zoran Cvetković, "Frequency-domain scattering delay networks for simulating room acoustics in virtual environments," in *Seventh International Conference on Signal Image Technology & Internet-Based Systems*. IEEE, 2011, pp. 180–187.

[14] Enzo De Sena, Hüseyin Hacıhabiboğlu, Zoran Cvetković, and Julius O. Smith, "Efficient synthesis of room acoustics via scattering delay networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 9, pp. 1478–1492, 2015.

[15] Karolina Prawda, Silvin Willemsen, Stefania Serafin, and Vesa Välimäki, "Flexible real-time reverberation synthesis with accurate parameter control," in *Proc. of the 23rd Int. Conf. on Digital Audio Effects (DAFx-20)*, 2020, pp. 16–23.

[16] Damian T Murphy, Chris JC Newton, and David M Howard, "Digital waveguide mesh modelling of room acoustics: Surround-sound, boundaries and plugin implementation," in *Proc. of the COST G-6 Conf. Digital Audio Effects (DAFx-02)*, 2001.

[17] Christian Borß, "A VST reverberation effect plugin based on synthetic room impulse responses," in *Proc. of the 12th Int. Conf. on Digital Audio Effects (DAFx-09)*, 2009.

[18] Edward Ly and Julián Villegas, "Generating artificial reverberation via genetic algorithms for real-time applications," *Entropy*, vol. 22, no. 11, pp. 1309, 2020.

[19] Christopher Yeoward, Rishi Shukla, Rebecca Stewart, Mark Sandler, and Joshua D Reiss, "Real-time binaural room modelling for augmented reality applications," *Journal of the Audio Engineering Society*, vol. 69, no. 11, pp. 818–833, 2021.

[20] Davide Rocchesso and Julius O. Smith, "Circulant and elliptic feedback delay networks for artificial reverberation," *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 1, pp. 51–63, 1997.

[21] Julius O. Smith, "A new approach to digital reverberation using closed waveguide networks," in *Proc. Int. Computer Music Conf, Vancouver*, 1985, pp. 47–53.

[22] Matti Karjalainen, Patty Huang, and Julius O. Smith, "Digital waveguide networks for room response modeling and synthesis," in *AES 118th Convention, Barcelona, Spain*, 2005.

[23] Francis Stevens, Damian T Murphy, Lauri Savioja, and Vesa Välimäki, "Modeling sparsely reflecting outdoor acoustic scenes using the waveguide web," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 8, pp. 1566–1578, 2017.

[24] Timuçin Berk Atalay, Zühre Sü Gül, Enzo De Sena, Zoran Cvetković, and Hüseyin Hacıhabiboğlu, "Scattering delay network simulator of coupled volume acoustics," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 30, pp. 582–593, 2022.

[25] Matteo Scerbo, Orchisama Das, Patrick Friend, and Enzo De Sena, "Higher-order scattering delay networks for artificial reverberation," in *Proc. of the 25th Int. Conf. on Digital Audio Effects (DAFx20in22)*, 2022.

[26] Leny Vinceslas, Matteo Scerbo, Hüseyin Hacıhabiboğlu, Zoran Cvetković, and Enzo De Sena, "Low-complexity higher order scattering delay networks," in *2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2023, pp. 1–5.

[27] "Acoustics - measurement of sound absorption in a reverberation room," Standard ISO 354:2003, International Organization for Standardization, 2003.

[28] Jason E. Summers, "Auralization: Fundamentals of Acoustics, Modelling, Simulation, Algorithms, and Acoustic Virtual Reality," *The Journal of the Acoustical Society of America*, vol. 123, no. 6, pp. 4028–4029, 2008.

[29] Wallace Clement Sabine, *Collected papers on acoustics*, Harvard university press, 1922.

[30] Carl F Eyring, "Reverberation time in "dead" rooms," *The Journal of the Acoustical Society of America*, vol. 1, no. 2A, pp. 217–241, 1930.

[31] Daniel González-Toledo, Luis Molina-Tanco, M Cuevas-Rodriguez, Piotr Majdak, and Arcadio Reyes-Lecuona, "The Binaural Rendering Toolbox. A virtual laboratory for reproducible research in psychoacoustics," in *Forum Acusticum*, 2023.

[32] Piotr Majdak, Franz Zotter, Fabian Brinkmann, Julien De Muynke, Michael Mihocic, and Markus Noisternig, "Spatially oriented format for acoustics 2.1: Introduction and recent advances," *Journal of the Audio Engineering Society*, vol. 70, no. 7/8, pp. 565–584, 2022.

[33] María Cuevas-Rodríguez, Lorenzo Picinali, Daniel González-Toledo, Carlos Garre, Ernesto de la Rubia-Cuestas, Luis Molina-Tanco, and Arcadio Reyes-Lecuona, "3D Tune-In Toolkit: An open-source library for real-time binaural spatialisation," *PLOS ONE*, vol. 14, no. 3, pp. 1–37, 2019.

[34] Jyri Huopaniemi, Lauri Savioja, and Matti Karjalainen, "Modeling of reflections and air absorption in acoustical spaces a digital filter design approach," in *Proc. of Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 1997.

[35] EC Levy, "Complex-curve fitting," *IRE transactions on automatic control*, vol. AC-4, no. 1, pp. 37–43, 1959.

[36] John E. Dennis Jr and Robert B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1983.

[37] Julius O. Smith, "Introduction to digital filters with audio applications," `https://ccrma.stanford.edu/~jos/filters/`, 2007, online book, accessed 30/05/2024.

[38] Jyri Huopaniemi, "Virtual acoustics and 3-D sound in multimedia signal processing," Tech. Rep. 53, Helsinki Univ. of Tech., Lab. of Acoustics and Audio Signal Processing, 1999.

[39] James M Kates and Eugene J Brandewie, "Adding air absorption to simulated room acoustic models," *The Journal of the Acoustical Society of America*, vol. 148, no. 5, pp. EL408–EL413, 2020.

[40] Brian Hamilton, "Air absorption filtering method based on approximate green's function for stokes' equation," in *2021 24th International Conference on Digital Audio Effects (DAFx)*. IEEE, 2021, pp. 160–167.

[41] Michael Chapman, Winfried Ritsch, Thomas Musil, Johannes Zmölnig, Hannes Pomberger, Franz Zotter, and Alois Sontacchi, "A standard for interchange of Ambisonic signal sets. Including a file standard with metadata," in *Proc. of the Ambisonics Symposium, Graz, Austria*, 2009.

[42] Angelo Farina, "Simultaneous measurement of impulse response and distortion with a swept-sine technique," in *AES 108th convention*. Audio Engineering Society, 2000.

[43] "Acoustics–measurement of the reverberation time of rooms with reference to other acoustical parameters," Standard ISO 3382:1997, International Organization for Standardization, 1997.

[44] Fabian Brinkmann, Lukas Aspöck, David Ackermann, Rob Opdam, Michael Vorländer, and Stefan Weinzierl, "A benchmark for room acoustical simulation. Concept and database," *Applied Acoustics*, vol. 176, pp. 107867, 2021.

[45] Constant CJM Hak, Remy HC Wenmaekers, and LCJ Van Luxemburg, "Measuring room impulse responses: Impact of the decay range on derived room acoustic parameters," *Acta Acustica united with Acustica*, vol. 98, no. 6, pp. 907–915, 2012.

[46] Matthew G Blevins, Adam T Buck, Zhao Peng, and Lily M Wang, "Quantifying the just noticeable difference of reverberation time with band-limited noise centered around 1000 hz using a transformed up-down adaptive method," 2013.

[47] Augustine Gray and John Markel, "Distance measures for speech processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 5, pp. 380–391, 1976.

[48] Davide Fantini, Giorgio Presti, Michele Geronazzo, Riccardo Bona, Alessandro Giuseppe Privitera, and Federico Avanzini, "Co-immersion in audio augmented virtuality: the case study of a static and approximated late reverberation algorithm," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 11, pp. 4472–4482, 2023.

[49] Riccardo Bona, Davide Fantini, Giorgio Presti, Marco Tiraboschi, Juan Isaac Engel Alonso-Martinez, and Federico Avanzini, "Automatic parameters tuning of late reverberation algorithms for audio augmented reality," in *Proc. of the 17th Int. Audio Mostly Conference*. 2022, pp. 36–43, ACM.