MODELING THE FREQUENCY-DEPENDENT SOUND ENERGY DECAY OF ACOUSTIC ENVIRONMENTS WITH DIFFERENTIABLE FEEDBACK DELAY NETWORKS

Alessandro Ilic Mezza *, Riccardo Giampiccolo, and Alberto Bernardini* †

Image and Sound Processing Lab
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano
Piazza L. Da Vinci, 32, 20133, Milan, Italy
alessandroilic.mezza@polimi.it | riccardo.giampiccolo@polimi.it | alberto.bernardini@polimi.it

ABSTRACT

Differentiable machine learning techniques have recently proved effective for finding the parameters of Feedback Delay Networks (FDNs) so that their output matches desired perceptual qualities of target room impulse responses. However, we show that existing methods tend to fail at modeling the frequency-dependent behavior of sound energy decay that characterizes real-world environments unless properly trained. In this paper, we introduce a novel perceptual loss function based on the mel-scale energy decay relief, which generalizes the well-known time-domain energy decay curve to multiple frequency bands. We also augment the prototype FDN by incorporating differentiable wideband attenuation and output filters, and train them via backpropagation along with the other model parameters. The proposed approach improves upon existing strategies for designing and training differentiable FDNs, making it more suitable for audio processing applications where realistic and controllable artificial reverberation is desirable, such as gaming, music production, and virtual reality.

1. INTRODUCTION

Feedback Delay Networks (FDNs) represent a versatile class of digital audio processing algorithms renowned for their applications in artificial reverberation. Originally proposed by Gerzon [1], FDNs are recursive filters featuring a bank of N absorbing delay lines whose outputs are mixed and fed back by a scalar feedback matrix. This way, FDNs can parsimoniously model the physical process of traveling sound waves being repeatedly reflected at the room boundaries, which ultimately results in acoustic reverberation. As such, delay-network models constitute an efficient alternative to the general representation of a room impulse response (RIR) as a finite impulse response (FIR) filter [2]. In fact, despite recent partitioned schemes [3], convolution still has a computational load incompatible with certain real-time applications, such as those pertaining virtual reality [4] and gaming [5].

In the past few years, significant efforts have been directed toward determining the optimal set of FDN parameters. Various strategies have been employed to address this challenge, with some adopting an analytical approach and designing the FDN so as to obtain certain desired acoustical characteristics, such as a target reverberation time [6, 7] and echo density [8, 9]. Others leverage optimization methods to adjust the parameters of a delay-network model in order to fit a target RIR, with genetic algorithms being the most common approach [10–14].

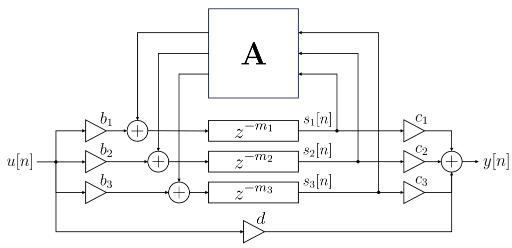


Figure 1: *General FDN ($N = 3$).*

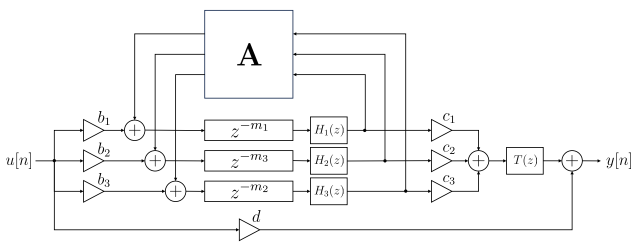


Figure 2: *Modified FDN ($N = 3$).*

Largely unaffected by the well-known limitations of gradient-free methods, differentiable machine learning techniques have also been recently introduced in the realm of FDN optimization. Lee et al. [15] estimate the parameters of a differentiable delay-network model using a convolutional-recurrent neural network trained in an end-to-end fashion. Dal Santo et al. [16, 17] recently proposed to optimize the model parameters directly within the digital structure of a differentiable FDN as a means to achieve colorless reverberation, i.e., having a flat magnitude response. However, [15] merely considers the artificial reverberator as a building block of the loss function, whereas [16, 17] define the loss function based solely on the characteristics of the FDN without targeting any real-world RIR, effectively resulting in a reference-free optimization scheme.

In a recent work [18], we proposed using automatic differentiation to find the parameters of an FDN so that its output matches some perceptual qualities of a target RIR. However, [18] considers a prototype FDN where gains and damping are modeled by instan-

* The authors wish to thank Enzo De Sena for the helpful discussions.

† This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 — program "RESTART")

taneous multiplications with learnable scalars. All FDN parameters are, therefore, frequency-independent,[1] and are optimized as so to minimize a likewise frequency-independent loss function. In this paper, we show that such an approach, although capable of accurately capturing the overall energy decay of the target RIR, fails to model the frequency-domain behavior that instead characterizes real-world room acoustics. Thus, we improve the training objective proposed in [18] by incorporating a frequency-dependent loss term based on the mel-scale energy decay relief (EDR) [20]. Furthermore, we extend the differentiable FDN prototype by including trainable finite impulse response (FIR) filters, and learn their taps along with the other FDN parameters. The proposed FDNs are shown to enhance the behavior of the energy decay at different frequencies compared to the state of the art.

2. FEEDBACK DELAY NETWORKS

Formalized by Stautner and Puckette [21], the single-input single-output (SISO) FDN shown in Figure 1 is characterized by [22]

$$\begin{aligned} y[n] &= \mathbf{c}^{\mathrm{T}}\mathbf{s}[n] + du[n] \\ \mathbf{s}[n+\boldsymbol{m}] &= \mathbf{A}\,\mathbf{s}[n] + \mathbf{b}u[n], \end{aligned} \tag{1}$$

where $u[n]$ is the input signal, $y[n]$ is the output signal, $\mathbf{b} \in \mathbb{R}^N$ is a vector of input gains, $\mathbf{c} \in \mathbb{R}^N$ is a vector of output gains, $(\cdot)^{\mathrm{T}}$ indicates the transpose operator, $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the feedback matrix, $d \in \mathbb{R}$ is the direct sound gain, and $\mathbf{s}[n] \in \mathbb{R}^N$ contains the delay lines output at time index n. The lengths of the delay lines expressed in (fractional) samples are denoted by $\boldsymbol{m} = [m_1, ..., m_N]$, while $\mathbf{s}[n+\boldsymbol{m}] := [s_1[n+m_1], ..., s_N[n+m_N]]^{\mathrm{T}}$.

If $\boldsymbol{m} = \mathbf{1}_N$, (1) corresponds to the measurement and state equations of a state-space model, and $\mathbf{s}[n]$ holds the state variables of the system at time n [22]. The delays, however, are commonly chosen to be co-prime integers to maximize echo density [9].

The feedback matrix $\mathbf{A}$ is often chosen to have unimodular eigenvalues and linearly independent eigenvectors [22]. This, however, is not enough to ensure that all the system poles of the resulting FDN lie on the unit circle [23]. A feedback matrix that guarantees critical stability regardless of the choice of delays $\boldsymbol{m}$ is said to be *unilossless* [23]. Notably, any orthogonal matrix is unilossless [23]. As such, Hadamard, Householder, and circulant matrices are widely used [19].

Starting from such a prototype, losses are then incorporated by multiplying $\mathbf{A}$ by a diagonal matrix of scalars designed to produce a specified reverberation time [16]. Alternatively, another classic approach is to extend every delay line with an attenuation filter [24]. Likewise, a *tone correction* filter can also be placed at the output of the FDN [24]. Different approaches for designing attenuation filters have been proposed in the literature. Existing designs include high-order octave-bands infinite impulse response (IIR) filters [25,26], graphic equalizers [6,7,27] and, more recently, two-stage filter structures [28]. Nevertheless, the optimal design of wideband attenuation filters based on a measured RIR remains an open challenge.

In the next section, we introduce a novel differentiable FDN architecture capable of capturing the frequency-dependent energy decay behavior of real-life acoustic environments thanks in part to the inclusion of learnable attenuation and output filters. Since every operation of the proposed FDN is differentiable, we are able to train its parameters via backpropagation, including filter coefficients, delay line lengths, scalar gains, and the feedback matrix.

3. DIFFERENTIABLE FEEDBACK DELAY NETWORKS

In [18], we showed that it is possible to optimize a differentiable implementation of the SISO FDN shown in Figure 1. In the present work, we focus on the prototype FDN depicted in Figure 2, which augments the N delay lines with N absorbent filters, $H_i(z)$, $i = 1, ..., N$, and features a tone correction filter, $T(z)$.[2]

In [24], Jot and Chaigne refer to the former as *general delay network*, and to the latter as *modified general delay network*. In the following, for the sake of clarity, we will use the same naming convention and call "general FDN" the one depicted in Figure 1, and "modified FDN" the one in Figure 2. In the next subsection, we will review the differentiable implementation of the general FDN presented in [18]. Sections 3.2 through 3.4 will then discuss the novelties of the proposed method.

3.1. Differentiable General FDNs

A general FDN can be implemented in such a way that it allows learning $\mathbf{A} \in \mathbb{R}^{N\times N}$, $\mathbf{b} \in \mathbb{R}^N_{\geq 0}$, $\mathbf{c} \in \mathbb{R}^N_{\geq 0}$, $\boldsymbol{m} \in \mathbb{R}^N_{\geq 0}$, $d \in \mathbb{R}_{\geq 0}$ via standard backpropagation [18]. In the following, we analyze each component of the differentiable general FDN one at a time.

Feedback matrix: Instead of learning the feedback matrix under unilosslessness constraints, we define an unconstrained real-valued matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$ and parameterize the lossy feedback matrix as $\mathbf{A} = \mathbf{U\Gamma}$, where $\mathbf{U} \in \mathbb{R}^{N\times N}$ is an orthogonal matrix, and $\mathbf{\Gamma} \in \mathbb{R}^{N\times N}_{[0,1)}$ is a learnable diagonal attenuation matrix. The matrix exponential maps skew-symmetric matrices onto orthogonal matrices [30]. Hence, we apply the following [16]

$$\mathbf{U} = \exp\Big(\mathbf{W}_{\mathrm{Tr}} - \mathbf{W}_{\mathrm{Tr}}^{\mathrm{T}}\Big)\,, \tag{2}$$

where $\mathbf{W}_{\mathrm{Tr}}$ is the upper triangular part of the unconstrained learnable matrix $\mathbf{W}$. As such, the argument of the matrix exponential $\exp(\cdot)$ is skew-symmetric by construction, and $\mathbf{U}$ is orthogonal, and thus unilossless, regardless of the values of $\mathbf{W}$. In turn, this implies that losses are entirely modeled by $\mathbf{\Gamma}$.

Differentiable reparameterization: Given N unconstrained real-valued scalars in $\boldsymbol{\gamma} = [\gamma_1, ..., \gamma_N]^{\mathrm{T}}$, the attenuation matrix is defined as [18]

$$\mathbf{\Gamma} = \mathrm{diag}(g(\gamma_1), ..., g(\gamma_N))\,, \tag{3}$$

where $g : \mathbb{R} \to (0, 1)$ is the logistic function $g(x) = \frac{1}{1+e^{-x}}$. Here, g is used to reparameterize $\boldsymbol{\gamma}$ so as to yield attenuation coefficients that take values in the codomain of said function.

Similarly, we use $f(x) = |x|$ to map d and the entries of $\mathbf{b}$, $\mathbf{c}$, and $\boldsymbol{m}$ onto $\mathbb{R}_{\geq 0}$. Akin to activation functions such as ReLU, f is differentiable almost everywhere. Hence, the gradients can flow up to the unconstrained learnable parameters, while we can use the output of g and f in every computation concerning the FDN.

Differentiable delay lines: We implement the delay lines in the frequency domain by evaluating their response on the unit circle at discrete frequency points. This is achieved by endowing each

[1] It is worth emphasizing that, although its parameters may indeed be frequency-independent, the FDN as a whole, belonging to a general class of recursive filters [19], is not.

[2] Extending the present study to differentiable MIMO FDNs [29] is left for future work.

delay line with a circular buffer collecting past samples. Thus, at each time step, we zero-pad the signal currently stored in the ith buffer, compute the Fast Fourier Transform (FFT), multiply the resulting spectrum by a conjugate symmetric fractional delay filter response, and go back to the time domain by computing the inverse FFT. This operation is carried out in parallel for $i = 1, ..., N$. For more details, we refer the reader to [18].

3.2. Differentiable Modified FDNs

A differentiable modified FDN allows us to learn the parameters of a general FDN plus the filter taps of $H_i(z)$ and $T(z)$. In practice, $\mathbf{b}$, $\mathbf{c}$, $\boldsymbol{m}$, and d are all parameterized as described in Section 3.1. The feedback matrix, instead, is given by $\mathbf{A} = \mathbf{U}$. Indeed, a modified FDN may forego the attenuation matrix $\mathbf{\Gamma}$ as its role is taken upon by the attenuation filters $H_i(z)$, $i = 1, ..., N$.

We implement $H_i(z)$, $i = 1, ..., N$, as time-domain FIR filters with p taps. Namely, we implement the entire attenuation filterbank as a single depthwise convolutional layer with kernel size p. By using a neural network block in lieu of alternative filtering implementations, the gradients of the loss function can efficiently flow through the convolutions, and we can train the kernel taps via standard backpropagation along with the other FDN parameters.

Depthwise convolutions here refer to a grouped 1D convolutional layer with N kernels and N groups. The output of each delay line is thus considered as a *channel* of the unbatched input tensor. Each channel is then processed by a dedicated kernel. By setting the number of groups equal to N, indeed, all cross-connections from input to output channels are blocked, and a dedicated FIR filter is applied to the output of each delay line independently of the others. In our implementation, the convolutional layers have no bias and no activation function. In principle, FIR filtering may be achieved with unit stride. In practice, however, the attenuation filters rely on p-sample circular buffers storing the output of each delay line. The input tensor is thus of size $N \times p$ and kernels have no stride. The ensuing filtering process is depicted in Figure 3.

Analogously, we implement $T(z)$ as a single-kernel 1D convolutional layer with no bias and activation.

In this work, we use FIR filters with $p = 63$ taps. Commonly, IIR filters are preferred as they require less taps compared to their FIR counterparts. However, updating the coefficients of an IIR filter via gradient descent may lead to instability problems during training. In contrast, FIR filters are always stable regardless of the values that taps may take on after each gradient update step.

3.3. Learning Objective

Frequency-independent objective: [18] introduced a composite loss function comprising two error terms: one for the energy decay curve (EDC) and one for the echo density profile (EDP).

Given a L_h-sample IR, $h[n]$, the EDC is defined via Schroeder's backward integration as [31]

$$\varepsilon[n] = \sum_{\tau=n}^{L_h} h^2[\tau], \tag{4}$$

and the corresponding L^2-loss term is given by

$$\mathcal{L}_{\text{EDC}} = \frac{\sum_n \left(\varepsilon[n] - \hat{\varepsilon}[n]\right)^2}{\sum_n \varepsilon[n]^2}, \tag{5}$$

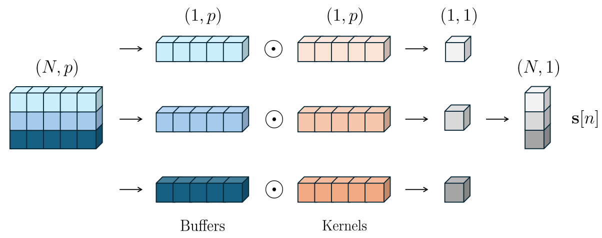


Figure 3: *FIR filtering via depthwise convolutions; ⊙ denotes the dot product between each input channel, i.e., a p-sample circular buffer storing the output of the corresponding delay line, and a dedicated p-tap kernel.*

where $\hat{\varepsilon}[n] = \sum_{\tau=n}^{L_h} \hat{h}^2[\tau]$ is the EDC of the predicted IR, $\hat{h}[n]$.

In [18], we introduced a regularization term, which we called Soft EDP, with the aim of better conditioning the IR's echo density. The Soft EDP, denoted by $\eta_\kappa[n]$, is a differentiable approximation of the normalized echo density profile introduced by Abel and Huang [32]. We define the Soft EDP as [18]

$$\eta_\kappa[n] = \frac{1}{\operatorname{erfc}(1/\sqrt{2})} \sum_{\tau=n-\nu}^{n+\nu} w[\tau] g_\kappa(|h[\tau]| - \sigma_n), \tag{6}$$

where $w[\tau]$ is a $(2\nu + 1)$-sample tapered window s.t. $\sum_\tau w[\tau] = 1$, $\operatorname{erfc}(\cdot)$ is the complementary error function, $g_\kappa(x) := g(\kappa x)$ indicates the κ-scaled logistic function, $\kappa \gg 1$, and σ_n is the standard deviation of the IR taps falling within the window centered at time index n. Contrary to the classic formulation [32], (6) is differentiable. Therefore, we can use the following loss term to regularize the echo density of the produced IR [18]

$$\mathcal{L}_{\text{EDP}} = \frac{1}{L_h} \sum_n \left(\eta_\kappa[n] - \hat{\eta}_\kappa[n]\right)^2, \tag{7}$$

where $\hat{\eta}_\kappa[n]$ is the Soft EDP of the predicted IR.

Combining the two terms, we obtain the following frequency-independent (FI) loss function [18]

$$\mathcal{L}_{\text{FI}} = \mathcal{L}_{\text{EDC}} + \lambda \mathcal{L}_{\text{EDP}}, \tag{8}$$

where $\lambda \in \mathbb{R}_{>0}$ is a positive hyperparameter.

Frequency-dependent objective: Frequency-dependent cost functions have been previously proposed for gradient-free automatic parameter tuning methods, based on, e.g., MFCCs [10, 14] and log-amplitude mel-spectrograms [33]. Likewise, [15] uses a multi-resolution spectral L^1-loss to train a neural network parameter estimator via backpropagation.

In this work, we introduce a new frequency-dependent loss term acting on the mel-scale energy decay relief (EDR). The EDR is typically defined via the backward integration of $|\mathcal{H}[\omega, m]|^2$, i.e., the squared magnitude of the short-time Fourier transform (STFT) of $h[n]$. Here, instead, we evaluate the EDR by integrating the mel-frequency spectrogram $\mathcal{H}_{\text{mel}}[k, m]$ to account for the nonlinear human perception of sound [34]. Namely, the mel-scale EDR is defined as

$$\mathcal{R}_{\text{mel}}^{\text{dB}}[k, m] = 10 \log_{10} \sum_{\tau=m}^{M} |\mathcal{H}_{\text{mel}}[k, \tau]|^2, \tag{9}$$

where $\mathcal{H}_{\text{mel}}[k, m]$ is obtained by filtering the 512-bin magnitude STFT of $h[n]$ with 64 triangular mel filters. The STFT is computed using a 320-sample Hann window (20 ms) with hopsize of 160 samples (10 ms). We define the corresponding L^1-loss term as

$$\mathcal{L}_{\text{EDR}} = \frac{\sum_k \sum_m \left| \mathcal{R}^{\text{dB}}_{\text{mel}}[k,m] - \hat{\mathcal{R}}^{\text{dB}}_{\text{mel}}[k,m] \right|}{\sum_k \sum_m |\mathcal{R}^{\text{dB}}_{\text{mel}}[k,m]|}, \quad (10)$$

where $\mathcal{R}^{\text{dB}}_{\text{mel}}[k,m]$ is the mel-EDR of the measured RIR in dB, and $\hat{\mathcal{R}}^{\text{dB}}_{\text{mel}}[k,m]$ is that of the IR of the optimized FDN.

The frequency-dependent (FD) training objective is obtained by linearly combining the EDC, EDR, and EDP terms, i.e.,

$$\mathcal{L}_{\text{FD}} = \lambda_1 \mathcal{L}_{\text{EDC}} + \lambda_2 \mathcal{L}_{\text{EDR}} + \lambda_3 \mathcal{L}_{\text{EDP}}, \quad (11)$$

where $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}_{>0}$.

The EDR generalizes the EDC to multiple frequency bands. Nonetheless, we argue that $\mathcal{L}_{\text{EDC}}$ and $\mathcal{L}_{\text{EDR}}$ are complementary rather than redundant. First, $\mathcal{L}_{\text{EDC}}$ has the same temporal resolution of the target IR, whereas $\mathcal{L}_{\text{EDR}}$, being defined in the time-frequency domain, has a coarser temporal resolution determined by the window stride. Second, we evaluate $\mathcal{L}_{\text{EDC}}$ on a linear scale, placing the focus on the first portion of the IRs, while $\mathcal{L}_{\text{EDR}}$ is defined on a dB scale, emphasizing errors in the reverberation tail due to the logarithmic compression. Notably, this approach is reminiscent of the well-established practice of combining linear-scale L^2-losses and log-scale L^1-losses that has been found beneficial in many audio signal processing tasks [35–38].

RIR length at training time: At training time, both (8) and (11) are evaluated limitedly to the span of time below the T_{60} of the target RIR [18]. In other words, L_h is trimmed to $\lceil T_{60} f_s \rceil$. Beyond that point, the residual energy of the target RIR is arguably negligible. Retaining such a late portion of the RIR would in fact overemphasize the contribution of the noise floor. This, in turn, could end up interfering with the learning process of the FDN, which, instead, exhibits a noiseless IR.

3.4. Learning Rates

We train every FDN model considered in the present study for a maximum of 650 iterations as follows. For general FDNs, we use a single Adam optimizer with learning rate of 0.1, acting on $\mathbf{W}$, $\boldsymbol{\gamma}$, $\mathbf{b}$, $\mathbf{c}$, $\boldsymbol{m}$, and d. As far as modified FDNs are concerned, instead, we follow [39] and invoke two Adam optimizers with different learning rates. The first acts on $\mathbf{W}$, $\mathbf{b}$, $\mathbf{c}$, $\boldsymbol{m}$, and d with a learning rate of 0.1. The second acts on the taps of the attenuation and output filters, and has a learning rate of 0.001. In both cases, we set $\beta_1 = 0.9$, $\beta_2 = 0.999$, and apply no weight decay.

3.5. Parameter Initialization

We initialize the differentiable FDNs with no prior knowledge of the target RIRs.

Scalar parameters: As in [18], we let $\mathbf{b}^{(0)} \sim \mathcal{N}(\mathbf{0}, \frac{1}{N}\mathbf{I}_N)$, $\mathbf{c}^{(0)} = \frac{1}{N}\mathbf{1}_N$, and $d^{(0)} = 1$, where $\mathbf{1}_N$ is a vector of N ones, and $\mathbf{I}_N$ is the $N \times N$ identity matrix. We initialize $\mathbf{W}^{(0)}$ so that $\mathbf{W}^{(0)}_{ij} \sim \mathcal{N}(0, \frac{1}{N})$. We initialize $\tilde{\boldsymbol{m}}^{(0)}$ so that $\tilde{m}^{(0)}_i = \psi \tilde{m}^\star_i$ with $\tilde{m}^\star_i \sim \text{Beta}(\alpha, \beta)$, for $i = 1, ..., N$, where $\alpha \geq 1$ and $\beta > \alpha$. We set $\psi = 1024$, $\alpha = 1.1$, and $\beta = 6$, such that, at $f_s = 16$ kHz, we ensure a maximum possible delay of 64 ms and a mean value of about 10 ms. We let the scaling term in (6) increase linearly from 10^2 to 10^5 as $n = 0, ..., L_h - 1$.

FIR filters: The p-sample buffers of each delay line are initialized with zeros. The kernels of the depthwise convolutional layers are initialized with a scaled Kronecker delta $\gamma^{(0)}_i \delta[n]$, where $\gamma^{(0)}_i = 0.9$, $i = 1, ..., N$. Hence, at the very first iteration, the attenuation in feedback loop is equivalent to what one would obtain by using $\boldsymbol{\Gamma} = \text{diag}(\gamma^{(0)}_1, ..., \gamma^{(0)}_N)$. Notice that, despite the name, convolutional layers cross-correlate input and kernels rather than performing a direct convolution. Contrarily to cross-correlation, in fact, direct convolution entails one of the functions to be time-reversed, i.e., reflected about the y-axis. Here, we model such a reflection by populating the circular buffers starting from the zeroth index, shifting the elements in a clockwise direction, and fixing the writing head location. For this reason, we initialize convolutional kernels without time-reversing their taps.

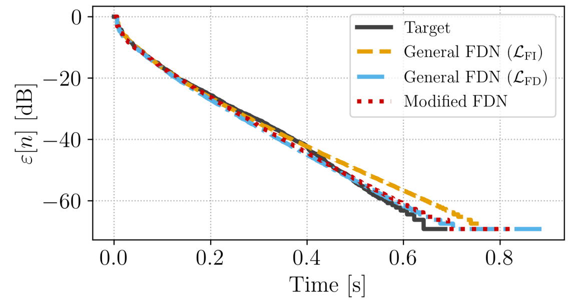


Figure 4: ***Test case 1:*** *Time-domain EDC* (4) *in dB.*

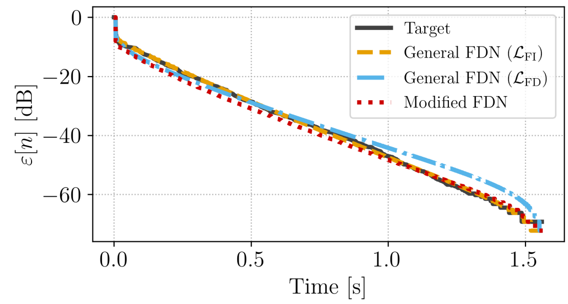


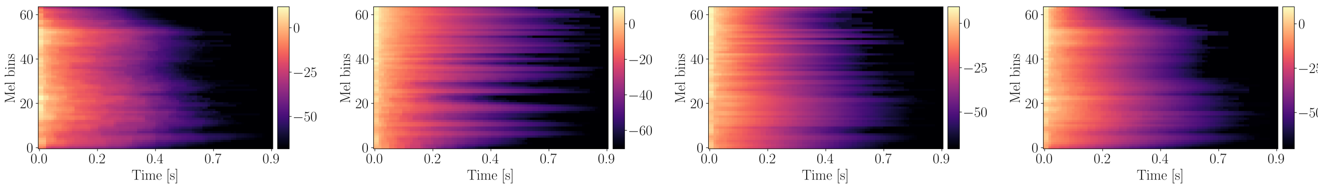
Figure 5: ***Test case 2:*** *Time-domain EDC* (4) *in dB.*

4. EVALUATION

We consider two RIRs measured in real-life acoustic environments taken from the 2016 MIT Acoustical Reverberation Scene Statistics Survey corpus [40]. The dataset contains 271 single-channel environmental IRs of both open and closed spaces, with reverberation times ranging from 0.06 s to 1.99 s.

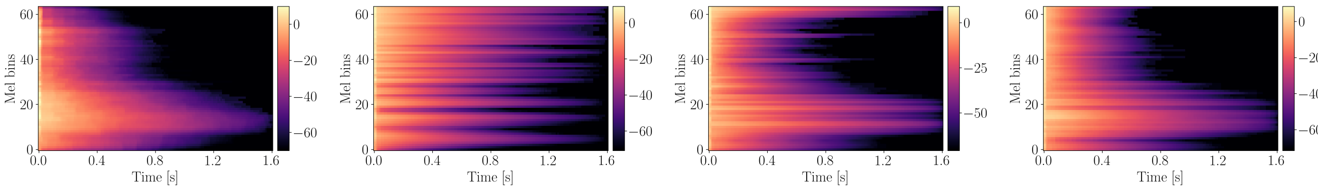
The first RIR, which we refer to as **test case 1**, was recorded in a hallway ($T_{60} \approx 0.6$ s) and has ID `h270`. The second RIR, which we refer to as **test case 2**, was recorded in a conference room ($T_{60} \approx 1.42$ s) and has ID `h060`.

For each test case, we train the general FDN described in Section 3.1 using the frequency-independent loss $\mathcal{L}_{\text{FI}}$ given in (8) and the proposed frequency-dependent loss $\mathcal{L}_{\text{FD}}$ given in (11). Additionally, we train the modified FDN presented in Section 3.2 using $\mathcal{L}_{\text{FD}}$ as learning objective. We set $N = 6$, $\lambda = 0.1$ in (8), and $\lambda_1 = 0.5$, $\lambda_2 = 1$, and $\lambda_3 = 0.1$ in (11).



(a) Target RIR (`h270`) (b) General FDN trained with $\mathcal{L}_{\text{FI}}$ (c) General FDN trained with $\mathcal{L}_{\text{FD}}$ (d) Modified FDN trained with $\mathcal{L}_{\text{FD}}$

Figure 6: ***Test case 1****: Mel-scale energy decay relief (EDR) in dB.*



(a) Target RIR (`h060`) (b) General FDN trained with $\mathcal{L}_{\text{FI}}$ (c) General FDN trained with $\mathcal{L}_{\text{FD}}$ (d) Modified FDN trained with $\mathcal{L}_{\text{FD}}$

Figure 7: ***Test case 2****: Mel-scale energy decay relief (EDR) in dB.*

4.1. Results and Discussion

Figures 4 and 5 show the time-domain EDCs, $\varepsilon[n]$, expressed in dB, for test case 1 and test case 2, respectively. Figures 6 and 7 depict the corresponding mel-scale EDRs. In particular, Figures 6a and 7a report the EDR of the target RIRs; Figures 6b and 7b show the EDR of the general FDNs trained using $\mathcal{L}_{\text{FI}}$; Figures 6c and 7c show the EDR of the general FDNs trained using $\mathcal{L}_{\text{FD}}$; Figures 6d and 7d show the EDRs of the proposed modified FDN. Furthermore, we report the EDCs of eight frequency bands corresponding to the center frequencies of the considered mel filters. Namely, we evaluate the EDCs at 58 Hz, 121 Hz, 264 Hz, 525 Hz, 988 Hz, 2027 Hz, 4075 Hz, and 7659 Hz. Figures 8 and 12 depict the EDCs of the baseline FDN. Figures 9 and 13 show the EDCs of the general FDN trained with the proposed frequency-dependent loss function. Figures 10 and 14 report the EDCs of the proposed modified FDN. For completeness, Figure 11 and 15 show the learned magnitude response of $H_i(z)$, $i = 1, ..., N$, and $T(z)$ pertaining to test case 1 and test case 2, respectively.

Figures 4 and 5 show that differentiable FDNs are able to accurately render the total energy decay of the target RIRs. However, Figures 6b and 7b reveal that the proposed loss function, $\mathcal{L}_{\text{FD}}$, is essential to capture the frequency-dependent behavior shown in Figures 6a and 7a, where low and high frequencies decay at noticeably different rates. Indeed, in both test cases, the general FDNs trained with $\mathcal{L}_{\text{FI}}$ yield a mel-EDR that is far from the target one. Ultimately, in fact, $\mathcal{L}_{\text{EDC}}$ and $\mathcal{L}_{\text{EDP}}$ do not inherently encourage the FDN to be aware of the desired energy spectral density.

Conversely, the differentiable FDNs trained with $\mathcal{L}_{\text{FD}}$ appear to produce an overall better energy decay. In particular, the general FDN trained with $\mathcal{L}_{\text{FD}}$ clearly outperforms the one trained with $\mathcal{L}_{\text{FI}}$, despite having the same architecture. This suggests that choosing the right learning objective is paramount in achieving the desired acoustical properties when training a differentiable FDN.

Whereas more closely resembling the target EDR, however, Figures 6c and 7c show two major drawbacks of general FDNs. First, the EDRs indicate a prominent comb-like frequency response, with several mel bands having noticeably less energy than the neighboring ones; this is a well-known problem affecting artificial reverberators employing delay loops, which, in turn, results in metallic sounding artifacts [41]. Second, we draw attention to the errors present in the high frequency range, where the energy appears to decay at a significantly lower rate than in the target EDR. The mismatch is particularly noticeable in Figures 9 and 13.

The proposed differentiable modified FDN improves both aspects. This is evidenced by Figures 10 and 14 where modified FDNs achieve good match at all test frequencies. Also, including the learned FIR filters appears to mitigate the comb effect in Figures 6d and 7d to some extent. This suggests that jointly using differentiable modified FDNs and the proposed loss function is beneficial when it comes to learning the frequency-dependent sound energy decay of real-life acoustic environments.

5. CONCLUSIONS

In this paper, we have showed that, unless explicitly regularized, current methods for training differentiable FDNs to match a target room impulse response fail to capture the frequency-dependent behavior of sound energy decay observed in real-life room acoustics. We thus proposed a novel loss function accounting for the mel-scale energy decay relief, along with a novel prototype FDN featuring differentiable attenuation and output filters. The proposed loss function proves crucial in rendering different decay rates across frequencies, while the integration of learnable FIR filters improves upon using a prototype FDN where delay line attenuation is modeled by scalar parameters.

6. REFERENCES

[1] Michael A. Gerzon, "Synthetic stereo reverberation: Part one," *Studio Sound*, vol. 13, pp. 631–635, 1971.

[2] Vesa Välimäki, Julian D. Parker, Lauri Savioja, Julius O. Smith, and Jonathan S. Abel, "Fifty years of artificial reverberation," *IEEE Trans. Audio Speech Lang. Process.*, vol. 20, no. 5, pp. 1421–1448, 2012.

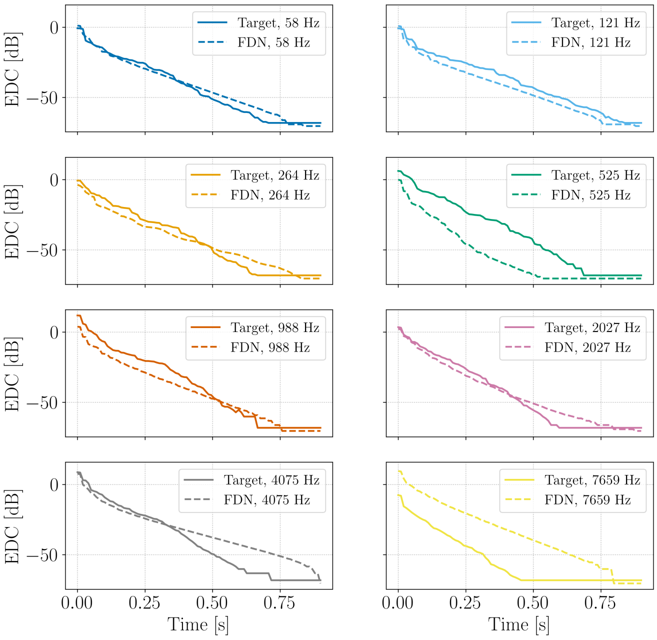


Figure 8: ***Test case 1:** EDCs of the general FDN trained with $\mathcal{L}_{FI}$.*

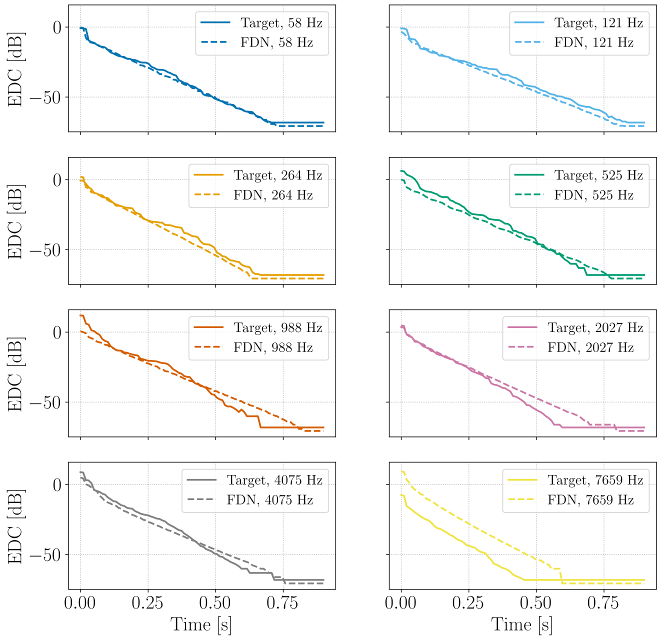


Figure 9: ***Test case 1:** EDCs of the general FDN trained with $\mathcal{L}_{FD}$.*

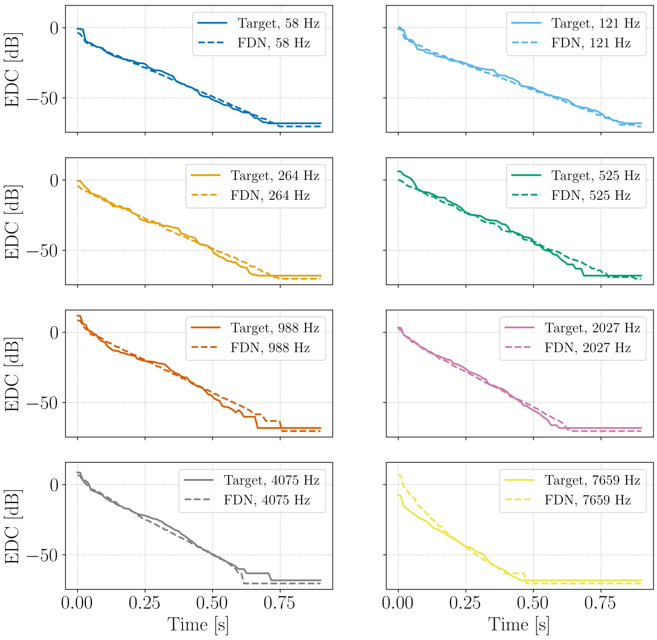


Figure 10: ***Test case 1:** EDCs of the modified FDN trained with $\mathcal{L}_{FD}$.*

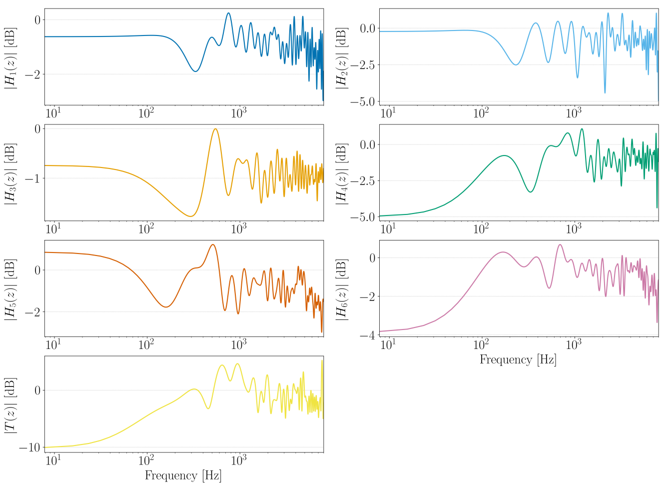


Figure 11: ***Test case 1:** Magnitude response of the learned filters.*

[3] Frank Wefers, *Partitioned convolution algorithms for real-time auralization*, vol. 20, Logos Verlag Berlin GmbH, Berlin, Germany, 2015.

[4] Thomas Potter, Zoran Cvetković, and Enzo De Sena, "On the relative importance of visual and spatial audio rendering on VR immersion," *Front. Signal Process.*, vol. 2, 2022.

[5] Enzo De Sena, Hüseyin Hacıhabiboğlu, and Zoran Cvetković, "Scattering delay network: An interactive reverberator for computer games," in *41st Audio Eng. Soc. Convention*, 2011.

[6] Sebastian J. Schlecht and Emanuël A. P. Habets, "Accurate reverberation time control in feedback delay networks," *Proc. Int. Conf. Digital Audio Effects*, pp. 337–344, 2017.

[7] Karolina Prawda, Sebastian J. Schlecht, and Vesa Välimäki,

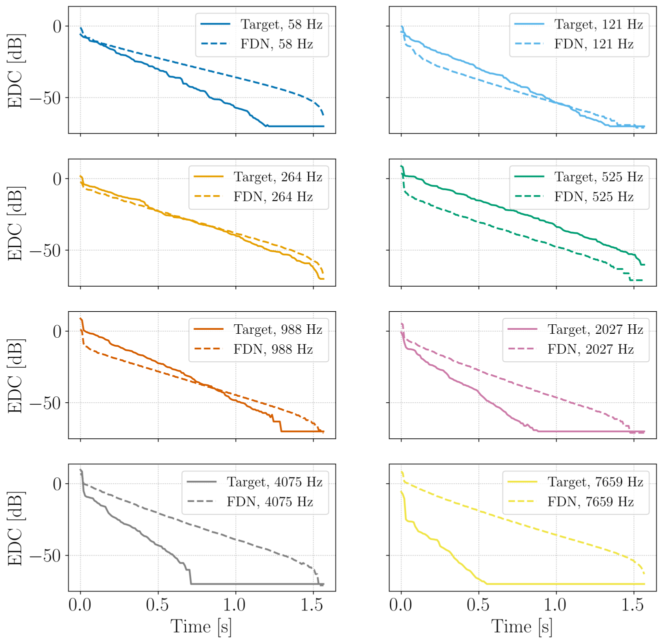


Figure 12: ***Test case 2:*** *EDCs of the general FDN trained with* $\mathcal{L}_{FI}$.

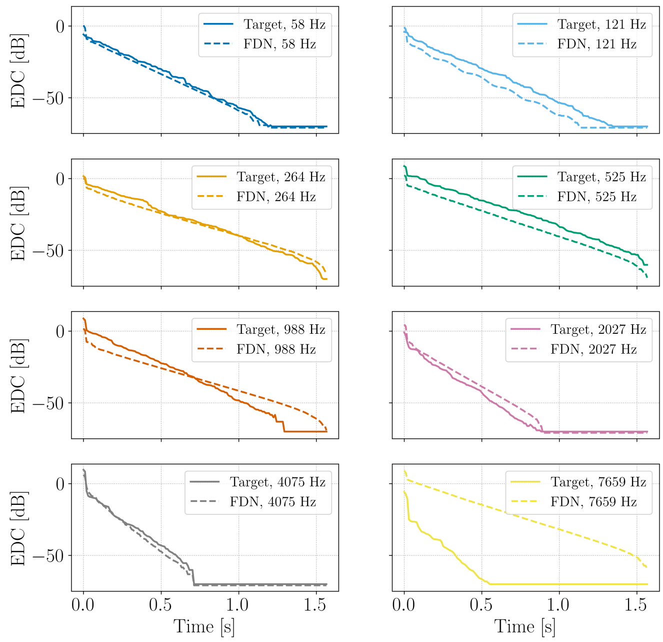


Figure 13: ***Test case 2:*** *EDCs of the general FDN trained with* $\mathcal{L}_{FD}$.

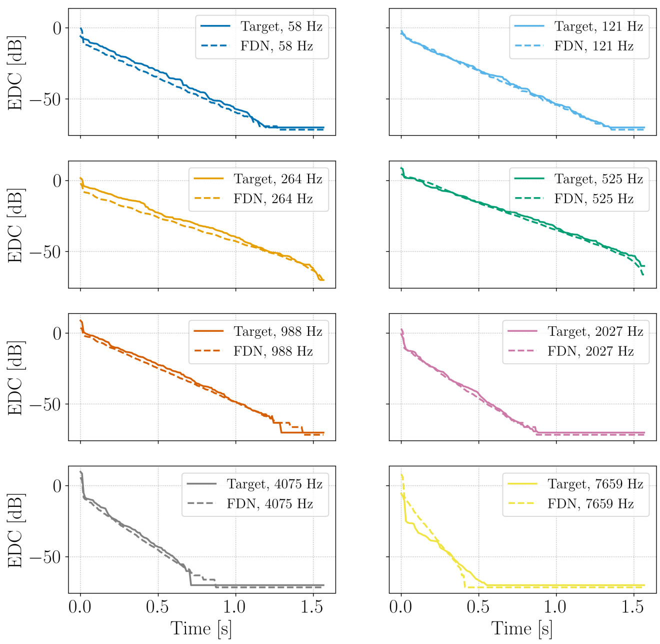


Figure 14: ***Test case 2:*** *EDCs of the modified FDN trained with* $\mathcal{L}_{FD}$.

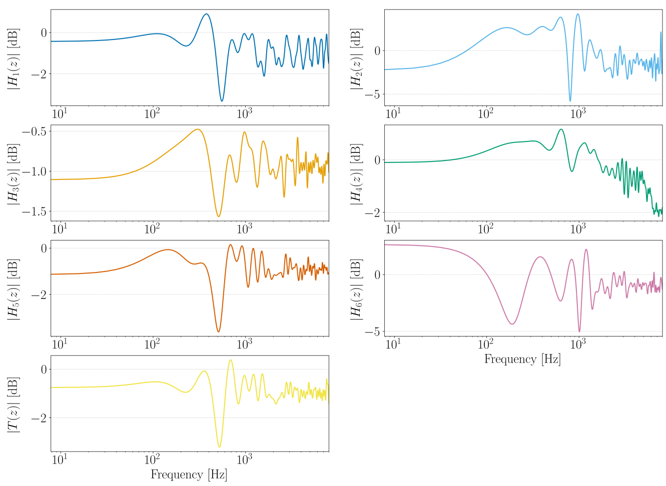


Figure 15: ***Test case 2:*** *Magnitude response of the learned filters.*

"Improved reverberation time control for feedback delay networks," in *Proc. Int. Conf. Digital Audio Effects*, 2019, pp. 1–7.

[8] Davide Rocchesso, "Maximally diffusive yet efficient feedback delay networks for artificial reverberation," *IEEE Signal Process. Lett.*, vol. 4, no. 9, pp. 252–255, 1997.

[9] Sebastian J. Schlecht and Emanuël A. P. Habets, "Feedback delay networks: Echo density and mixing time," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 25, no. 2, pp. 374–383, 2016.

[10] Sebastian Heise, Michael Hlatky, and Jörn Loviscach, "Automatic adjustment of off-the-shelf reverberation effects," in *126th Audio Eng. Soc. Convention*, 2009.

[11] Michael Chemistruck, Kyle Marcolini, and Will Pirkle,

"Generating matrix coefficients for feedback delay networks using genetic algorithm," in *133rd Audio Eng. Soc. Convention*, 2012.

[12] Jay Coggin and Will Pirkle, "Automatic design of feedback delay network reverb parameters for impulse response matching," in *141st Audio Eng. Soc. Convention*, 2016.

[13] Justin Shen and Ramani Duraiswami, "Data-driven feedback delay network construction for real-time virtual room acoustics," in *Proc. 15th Int. Audio Mostly Conf.*, 2020, pp. 46–52.

[14] Ilias Ibnyahya and Joshua D. Reiss, "A method for matching room impulse responses with feedback delay networks," in *153rd Audio Eng. Soc. Convention*, 2022.

[15] Sungho Lee, Hyeong-Seok Choi, and Kyogu Lee, "Differentiable artificial reverberation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 30, pp. 2541–2556, 2022.

[16] Gloria Dal Santo, Karolina Prawda, Sebastian J. Schlecht, and Vesa Välimäki, "Differentiable feedback delay network for colorless reverberation," in *Proc. 26th Int. Conf. Digital Audio Effects*, 2023, pp. 244–251.

[17] Gloria Dal Santo, Karolina Prawda, Sebastian J. Schlecht, and Vesa Välimäki, "Feedback delay network optimization," *arXiv preprint arXiv:2402.11216*, 2024.

[18] Alessandro Ilic Mezza, Riccardo Giampiccolo, Enzo De Sena, and Alberto Bernardini, "Data-driven room acoustic modeling via differentiable feedback delay networks with learnable delay lines," *arXiv preprint arXiv:2404.00082*, 2024.

[19] Sebastian J. Schlecht, "FDNTB: The feedback delay network toolbox," in *Proc. Int. Conf. Digital Audio Effects*, 2020, pp. 211–218.

[20] Jean-Marc Jot, "An analysis/synthesis approach to real-time artificial reverberation," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 1992, vol. 2, pp. 221–224.

[21] John Stautner and Miller Puckette, "Designing multi-channel reverberators," *Computer Music Journal*, vol. 6, no. 1, pp. 52–65, 1982.

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

[23] Sebastian J. Schlecht and Emanuël A. P. Habets, "On lossless feedback delay networks," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1554–1564, 2016.

[24] Jean-Marc Jot and Antoine Chaigne, "Digital delay networks for designing artificial reverberators," in *90th Audio Eng. Soc. Convention*, 1991.

[25] Hüseyin Hacıhabiboğlu, Enzo De Sena, and Zoran Cvetković, "Frequency-domain scattering delay networks for simulating room acoustics in virtual environments," in *7th Int. Conf. Signal Image Technol. & Internet-Based Syst.*, 2011, pp. 180–187.

[26] Torben Wendt, Steven Van De Par, and Stephan D. Ewert, "A computationally-efficient and perceptually-plausible algorithm for binaural room impulse response simulation," *J. Audio Eng. Soc.*, vol. 62, no. 11, pp. 748–766, 2014.

[27] Jean-Marc Jot, "Proportional parametric equalizers—Application to digital reverberation and environmental audio processing," in *139th Audio Eng. Soc. Convention*, 2015.

[28] Vesa Välimäki, Karolina Prawda, and Sebastian J. Schlecht, "Two-stage attenuation filter for artificial reverberation," *IEEE Signal Process. Lett.*, pp. 1–5, 2024.

[29] Riccardo Giampiccolo, Alessandro Ilic Mezza, and Alberto Bernardini, "Differentiable MIMO feedback delay networks for multichannel room impulse response modeling," in *Proc. 27th Int. Conf. Digital Audio Effects*, 2024.

[30] Mario Lezcano-Casado and David Martínez-Rubio, "Cheap orthogonal constraints in neural networks: A simple parametrization of the orthogonal and unitary group," in *Int. Conf. Mach. Learning*, 2019, pp. 3794–3803.

[31] Manfred R. Schroeder, "New method of measuring reverberation time," *J. Acoust. Soc. Am.*, vol. 37, no. 6, pp. 1187–1188, 1965.

[32] Jonathan S. Abel and Patty Huang, "A simple, robust measure of reverberation echo density," in *121st Audio Eng. Soc. Convention*, 2006.

[33] 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. 17th Int. Audio Mostly Conf.*, 2022, p. 36–43.

[34] David Howard and Jamie Angus, *Acoustics and psychoacoustics*, Routledge, London, United Kingdom, 2013.

[35] Sercan Ö. Arık, Heewoo Jun, and Gregory Diamos, "Fast spectrogram inversion using multi-head convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 26, no. 1, pp. 94–98, 2019.

[36] Ryuichi Yamamoto, Eunwoo Song, and Jae-Min Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2020, pp. 6199–6203.

[37] Geng Yang, Shan Yang, Kai Liu, Peng Fang, Wei Chen, and Lei Xie, "Multi-band MelGAN: Faster waveform generation for high-quality text-to-speech," in *IEEE Spoken Lang. Technol. Workshop*, 2021, pp. 492–498.

[38] Alessandro Ilic Mezza, Matteo Amerena, Alberto Bernardini, and Augusto Sarti, "Hybrid packet loss concealment for real-time networked music applications," *IEEE Open J. Signal Process.*, vol. 5, pp. 266–273, 2024.

[39] Alessandro Ilic Mezza, Riccardo Giampiccolo, and Alberto Bernardini, "Data-driven parameter estimation of lumped-element models via automatic differentiation," *IEEE Access*, vol. 11, pp. 143601–143615, 2023.

[40] James Traer and Josh H. McDermott, "Statistics of natural reverberation enable perceptual separation of sound and space," *Proc. Natl. Acad. Sci.*, vol. 113, no. 48, pp. 7856–7865, 2016.

[41] Manfred R. Schroeder and Benjamin F. Logan, ""Colorless" artificial reverberation," *IRE Trans. on Audio*, , no. 6, pp. 209–214, 1961.