# DIFFERENTIABLE MIMO FEEDBACK DELAY NETWORKS FOR MULTICHANNEL ROOM IMPULSE RESPONSE MODELING

*Riccardo Giampiccolo* [*] *, Alessandro Ilic Mezza, 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
riccardo.giampiccolo@polimi.it | alessandroilic.mezza@polimi.it | alberto.bernardini@polimi.it

## ABSTRACT

Recently, with the advent of new performing headsets and goggles, the demand for Virtual and Augmented Reality applications has experienced a steep increase. In order to coherently navigate the virtual rooms, the acoustics of the scene must be emulated in the most accurate and efficient way possible. Amongst others, Feedback Delay Networks (FDNs) have proved to be valuable tools for tackling such a task. In this article, we expand and adapt a method recently proposed for the data-driven optimization of single-input-single-output FDNs to the multiple-input-multiple-output (MIMO) case for addressing spatial/space-time processing applications. By testing our methodology on items taken from two different datasets, we show that the parameters of MIMO FDNs can be jointly optimized to match some perceptual characteristics of given multichannel room impulse responses, overcoming approaches available in the literature, and paving the way toward increasingly efficient and accurate real-time virtual room acoustics rendering.

## 1. INTRODUCTION

The market of consumer electronics has lately experienced an increase of the number of headsets and goggles for Augmented Reality (AR) and Virtual Reality (VR). For instance, we can mention Meta Quest 3, HTC VIVE, HTC Cosmos Elite, Sony PlayStation VR, and the recently introduced Apple Vision Pro. Hand in hand, the number of AR/VR applications has grown as well, providing the user increasingly immersive experiences [1]. Although, for many years, the success of such applications has fallen heavily on the shoulders of computer vision, acoustics modeling is lately being put on the same level as deemed necessary to level up the quality of the virtual scene [1]. Hence, there is an urgent need for highly-efficient and accurate algorithms able to make the virtual scene navigation as coherent and natural as possible [2, 3].

Acoustics rendering in AR/VR applications is typically associated with the mere navigation of the virtual scene [4, 5], but other applications recently met the interest of the research community. For instance, the space-time rendering of musical performance in concert halls is becoming a trendy topic, as it potentially allows the users to attend a concert from remote by precisely choosing the seat, with all the spatial characteristics that come with it [6, 7]. Room acoustics is usually tackled following either model-based or convolution-based approaches [8]. The latter are the most straightforward as they entail the convolution between the impulse response (IR) acquired by means of a microphone and the audio signal itself. However, such methods are typically not tabled in this scenario, due to the high computational cost that prevents multi-dimensional convolutions to be executed in real-time [8]. As a matter of fact, in order to grasp the spatial characteristics of an acoustic environment, it is desirable to acquire multichannel Room Impulse Responses (RIRs) by means of, e.g., higher-order microphones positioned in different points in space, and still take into account Head-Related Transfer Functions (HRTFs), enabling an even higher level of personalization and realism for the listener [9]. Hence, in this work, we are interested into methodologies that allow us to render multichannel RIRs in a cost-effective fashion.

Introduced by Gerzon in the 70s [10], Feedback Delay Networks (FDNs) are among the most known and used systems for artificial reverberation. FDNs are digital filters characterized by $N$ absorbing delay lines whose outputs are first fed back through a scalar matrix and then mixed together to provide the reverberated signal. Moreover, FDNs are characterized by a low number of parameters and low storage requirements, being thus well-suited for real-time applications. The FDN parameters are usually set analytically in order to obtain certain acoustic characteristics, such as a certain reverberation time [11, 12] or echo density [13, 14]. With the aim of removing human intervention, genetic algorithms have been recently considered for optimizing said parameters and matching measured RIRs [3, 15, 16]. Then, with the advent of differentiable digital signal processing, new methodologies have been introduced, either using FDNs as part of a pipeline involving convolutional neural networks [17], or optimizing some FDN parameters in a reference-free fashion [18]. No fully-differentiable FDN has been proposed until the work in [19], where all the FDN parameters are optimized and learned through automatic differentiation in order to match some perceptual characteristics of target RIRs. Neither said approach nor other approaches of the literature are, however, applied to learn all the parameters of multiple-input-multiple-output (MIMO) FDNs. In fact, the method proposed in [19] entails only single-input-single-output (SISO) time-domain FDNs.

In this article, we extend the approach presented in [19] for SISO FDNs to the MIMO case by jointly optimizing the parameters of MIMO FDNs to match perceptual qualities of multichannel RIRs. We show that by employing a cost function that combines two measures of perceptual features, i.e., the energy decay curve and the echo density profile, we are able to learn the afore-

[*]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.")

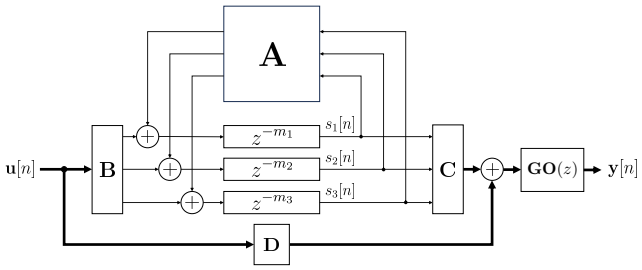


Figure 1: *MIMO FDN with three delay lines ($N = 3$).*

mentioned parameters such that the FDN matches said features for each channel $j$ of the target $J$-channel RIR. It is worth stressing that we limit the scope of this article to matching the frequency-independent decay in the time domain, and we leave for future work the extension to time-frequency decays [20]. We evaluate the proposed methodology taking into account measured RIRs sampled from two different datasets, namely the AIR dataset [21] and the HOMULA-RIR dataset [22], and we compare our results with those obtained by extending the method shown in [16] to the MIMO case. Our approach turns out to be characterized by the best performance paving the way towards increasingly efficient and accurate methods for real-time multichannel RIR rendering.

## 2. MIMO FEEDBACK DELAY NETWORKS

Defined $N$ as the number of delay lines, $\mathbf{u}[n] \in \mathbb{R}^K$ as the vector of input signals, and $\mathbf{y}[n] \in \mathbb{R}^J$ as the vector of output signals, a MIMO FDN can be described by the following discrete-time equations [23]

$$\begin{aligned} \mathbf{y}[n] &= \mathbf{C}\,\mathbf{s}[n] + \mathbf{D}\,\mathbf{u}[n] \\ \mathbf{s}[n+\boldsymbol{m}] &= \mathbf{A}\,\mathbf{s}[n] + \mathbf{B}\,\mathbf{u}[n] \end{aligned}, \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{N\times K}$ is the input gain matrix, $\mathbf{C} \in \mathbb{R}^{J\times N}$ is the output gain matrix, $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the feedback matrix, $\mathbf{D} \in \mathbb{R}^{J\times K}$ is the direct gain matrix, and $\mathbf{s}[n] \in \mathbb{R}^N$ denotes the output of the delay lines at time index $n$. The lengths of the delay lines expressed in samples are described by $\boldsymbol{m} = [m_1, ..., m_N]^{\mathrm{T}}$, being thus $\mathbf{s}[n+\boldsymbol{m}] := [s_1[n+m_1], ..., s_N[n+m_N]]^{\mathrm{T}}$. It is worth noticing that, if $\boldsymbol{m} = \mathbf{1}_N$, $\mathbf{s}[n]$ corresponds to the vector of state variables at index $n$, and the MIMO FDN can be described by means of the state-space formalism. Hence, FDNs can be, in general, thought of as generalized versions of state-space systems with delays different than one. In addition, $\boldsymbol{m}$ are typically chosen to be co-prime in order to increase the echo density [13].

In this work, however, we consider as a prototype the MIMO FDN shown in Fig. 1, since we found it being more suited to be optimized by means of the proposed automatic differentiation framework; this will be better explained in Sec. 3. In particular, such a MIMO FDN can be described by means of the following equations

$$\begin{aligned} \mathbf{y}[n-\boldsymbol{\mu}] &= \mathbf{G}\left(\mathbf{C}\,\mathbf{s}[n] + \mathbf{D}\,\mathbf{u}[n]\right) \\ \mathbf{s}[n+\boldsymbol{m}] &= \mathbf{A}\,\mathbf{s}[n] + \mathbf{B}\,\mathbf{u}[n] \end{aligned}, \tag{2}$$

where $\mathbf{G} \in \mathbb{R}^{J\times J}$ is a diagonal matrix containing real scaling parameters and $\boldsymbol{\mu} := [\mu_1, \ldots, \mu_J]^{\mathrm{T}}$ is a vector containing the direct path delays. It follows that matrix $\mathbf{O}(z)$ in Fig. 1 is a diagonal matrix having on the main diagonal $[z^{-\mu_1}, \ldots, z^{-\mu_J}]$. Finally, in our work, we consider the delays $\boldsymbol{m}$ to be fractional [19].

Unitary orthogonal matrices, such as the Hadamard or the Householder matrices [24], are typically chosen as the prototype for the feedback matrix $\mathbf{A}$. In fact, being *unilossless*, they ensure stability regardless of the delays introduced in the FDN [25]. Then, losses are introduced by multiplying such a feedback matrix by a diagonal matrix containing scalar values designed to render a particular reverberation time [18].

Although designs of FDNs that include tone correction and attenuation filters are present in the literature [20, 26, 27], in this article, we focus on FDNs characterized by frequency-independent parameters, i.e., the entries of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are real-valued scalars. Finally, given the time-domain nature of parameters, the stability of the system can be easily guaranteed at training time, thanks to the reparametrization explained in the following section.

## 3. DIFFERENTIABLE MIMO FEEDBACK DELAY NETWORKS

In this section, we discuss how it is possible to learn and optimize the parameters of a differentiable implementation of the MIMO FDN shown in Fig. 1 by adapting the method proposed in [19] for the SISO case to the MIMO case.

The proposed method entails an iterative optimization procedure that relies on the same training engine employed for training deep neural networks [28]. We aim at minimizing a loss function $\mathcal{L}$ between the target $J$-channel RIR $\mathbf{h}[n] \in \mathbb{R}^J$ and the output of the differentiable MIMO FDN $\hat{\mathbf{h}}[n] \in \mathbb{R}^J$ at each time instant $n$, considering as input the set of Kronecker deltas $\boldsymbol{\Delta}[n] \in \{0, 1\}^K$.

At each iteration, the trainable MIMO FDN parameters $\theta$ undergo an optimization step involving the gradient $\nabla\mathcal{L}_\theta$ computed via reverse-mode automatic differentiation [19, 29]. There are no precise directions for the optimization of MIMO FDNs; however, typically, SISO FDNs are optimized taking into account just the late reverberation, as early reflections are handled differently [8]. Instead, we aim at jointly modeling early reflections and reverberant tails in order to fully take advantage of the efficient recursive structure characterizing MIMO FDNs.

As far as the training procedure is concerned, we first normalize the multichannel RIR. Although different normalization strategies can be considered, in this work, we divide each channel by the squared Frobenius norm of the multichannel RIR, i.e., $g$, and we store such a value in matrix $\mathbf{G}$ such that $\mathbf{G} = g\mathbf{I}_J$, with $\mathbf{I}_J$ being an $J \times J$ identity matrix. According to (2), this matrix will be later used to re-scale the output of the MIMO FDN.

We then remove the initial $\mu_j$ samples to ensure that the first sample of the $j$th channel always contains the direct path. We store such values in vector $\boldsymbol{\mu}$ since these will be re-introduced at inference time using matrix $\mathbf{O}(z)$ according to (2) (see Fig. 1). In order to get a consistent multichannel RIR, we zero-pad the channels such that they are all characterized by the same number of samples $L_x$. In particular, this value is set equal to the number of samples of the longest stripped RIR, i.e., $L_x = L_j - \mu_j$ with $j = j_{\min}$ s.t. $\mu_{j_{\min}} = \min_j\{\mu_j\}_{j=1}^J$, being $L_j$ the original length in samples of the $j$th channel. At this point, we compute the reverberation times $T_{60,j}$ for each channel $j$, and we define $T_{60}^{\max} = \max_j\{T_{60,j}\}_{j=1}^J$. As a further step, we trim all of the channels such that they have length $L_h = \lceil T_{60}^{\max} \cdot f_s \rceil$ with $f_s$ being the sampling frequency. In fact, after $T_{60}^{\max}$, we consider the multichannel RIR characterized by not-meaningful information, such as background noise, which, if taken into account, would badly condition the training process; in addition, the energy would be so low as to cause numerical er-

rors when using single-precision floating point numbers, leading to unwanted behaviors in the reverberant tails.

Finally, we optimize input, output, and direct gain matrices, the feedback matrix, and the delays $\boldsymbol{m}$ expressed as fractional delays. Then, we would like all the parameters but $\mathbf{A}$ to be nonnegative, i.e., we would like to use $\mathbf{B} \in \mathbb{R}_{\geq 0}^{N\times K}$, $\mathbf{C} \in \mathbb{R}_{\geq 0}^{J\times N}$, $\mathbf{D} \in \mathbb{R}_{\geq 0}^{J\times K}$, $\mathbf{A} \in \mathbb{R}^{N\times N}$, and $\boldsymbol{m} \in \mathbb{R}_{\geq 0}^{N}$. This is to let the gains not affect the polarity of the reflections, which is only determined by the feedback matrix. In order to enforce nonnegativity, we consider a nonlinear function $f_{\geq 0} : \mathbb{R} \to \mathbb{R}_{\geq 0}$; in particular, we choose $f_{\geq 0}(x) = |x|$ as it has shown to be effective in previous works [19, 28]. Thus, although we learn the parameters in an unconstrained fashion, what we actually use in the MIMO FDN are their nonnegative counterparts, e.g., $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_K]$ with entries $\mathbf{b}_k = [f_{\geq 0}(\tilde{b}_{k,1}), ..., f_{\geq 0}(\tilde{b}_{k,N})]^{\mathrm{T}}$, where the symbol $\tilde{(\cdot)}$ indicates the unconstrained learnable parameters.

### 3.1. Trainable Feedback Matrix

Let $\mathbf{W} \in \mathbb{R}^{N\times N}$ be an unconstrained real-valued learnable matrix and let matrix $\mathbf{U} \in \mathbb{R}^{N\times N}$ be defined as

$$\mathbf{U} = \exp\Big(\mathbf{W}_{\mathrm{Tr}} - \mathbf{W}_{\mathrm{Tr}}^{\mathrm{T}}\Big) \,, \tag{3}$$

where $\mathbf{W}_{\mathrm{Tr}}$ is the upper triangular part of $\mathbf{W}$. Given that $\mathbf{W}_{\mathrm{Tr}} - \mathbf{W}_{\mathrm{Tr}}^{\mathrm{T}}$ is skew-symmetric by construction and that matrix exponential maps skew-symmetric matrices onto orthogonal matrices [30], we can state that $\mathbf{U}$ is an orthogonal matrix. It follows that $\mathbf{U}$ is also *unilossless* despite the values assumed by $\mathbf{W}$. Thus, we may think to reparametrize the feedback matrix $\mathbf{A}$ exploiting $\mathbf{U}$ as [18]

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Gamma} \,, \tag{4}$$

where $\boldsymbol{\Gamma} \in \mathbb{R}_{[0,1)}^{N\times N}$ is a learnable diagonal attenuation matrix. In other words, instead of learning directly a unilossless matrix, we learn the entries of the unconstrained matrix $\mathbf{W}$, which then are mapped onto $\mathbf{A}$ by means of (4) [18, 19]. Losses are, instead, modeled entirely by matrix $\boldsymbol{\Gamma}$ that, in turn, is defined as

$$\boldsymbol{\Gamma} = \mathrm{diag}(g(\gamma_1), ..., g(\gamma_N)) \,, \tag{5}$$

where $\gamma_1, \ldots, \gamma_N$ are real unconstrained scalars, and

$$g(x) = \frac{1}{1 + e^{-x}} \tag{6}$$

is the sigmoid function. Equation (6) allows us to map the unconstrained attenuation coefficients $\gamma_n$ onto the sigmoid codomain such that they take values in the range $(0, 1)$.

Thanks to this reparametrization, the gradients are able to traverse the computational graph and reach the unconstrained entries of $\mathbf{W}$ and $\boldsymbol{\Gamma}$, thus enabling the optimization of $\mathbf{A}$.

### 3.2. Differentiable Delay Lines

Delay lines can be easily implemented in the discrete-time domain as shift operations applied to buffers that collect past samples. However, such an operation is not differentiable. Hence, in [19], we proposed a way for implementing such FDN blocks in the frequency domain. In particular, we first zero-pad and compute the Fast Fourier Transform (FFT) of the buffered signal, then we multiply the discrete spectrum by a conjugate symmetric fractional delay filter response. Finally, we obtain the shifted time-domain signal by computing the Inverse FFT. We refer the interested reader to [19] for further details about the implementation.

### 3.3. Loss Function

We adapt the loss function proposed in [19] for training time-domain SISO FDNs to the MIMO case. Such a loss function is composed of two terms: (i) an error for the energy decay curve (EDC) and (ii) an error for the echo density profile (EDP), which is meant just as a regularization term.

Let $\mathbf{h}[n] := [h_1[n], \ldots, h_J[n]]^{\mathrm{T}}$ be a multichannel IR of $L_h$ samples and $J$ channels at time instant $n$. We then define the multichannel EDC as $\boldsymbol{e}[n] := [\varepsilon_1[n], \ldots, \varepsilon_J[n]]^{\mathrm{T}}$, where the generic $\varepsilon_j[n]$ is computed via Schroeder's backward integration as follows

$$\varepsilon_j[n] = \sum_{\tau=n}^{L_h} h_j^2[\tau] \,. \tag{7}$$

The corresponding $L^2$-loss term is defined as

$$\mathcal{L}_{\mathrm{EDC}} = \frac{\sum_n \|\boldsymbol{e}[n] - \hat{\boldsymbol{e}}[n]\|_2^2}{\sum_n \|\boldsymbol{e}[n]\|_2^2} \,, \tag{8}$$

where $\hat{\boldsymbol{e}}[n] := [\hat{\varepsilon}_1[h], \ldots, \hat{\varepsilon}_J[n]]^{\mathrm{T}}$ with $\hat{\varepsilon}_j[n] = \sum_{\tau=n}^{L_h} \hat{h}_j^2[\tau]$ is the EDC of the predicted IR $\hat{\mathbf{h}}[n]$ with sum and subtract operations applied channel-wise.

Following what done in [19], the loss term in (8) is then regularized by means of an additional term named Soft EDP, which is meant to control the echo density of the predicted IR. In particular, the Soft EDP is derived as a differentiable approximation of the well-known normalized echo density profile [31]. Defined $\boldsymbol{p}[n] := [\eta_1[n], \ldots, \eta_J[n]]^{\mathrm{T}}$ as the multichannel Soft EDP, the generic channel Soft EDP $\eta_j[n]$ can be written as

$$\eta_j[n] = \frac{1}{\mathrm{erfc}(1/\sqrt{2})} \sum_{\tau=n-\nu}^{n+\nu} w[\tau] g_\kappa(|h_j[\tau]| - \sigma_n) \,, \tag{9}$$

where $\mathrm{erfc}(\cdot)$ is the complementary error function, $w[\tau]$ is a tapered window of length $(2\nu+1)$ such that $\sum_\tau w[\tau] = 1$, whereas $g_\kappa(x) := g(\kappa x)$ indicates the $\kappa$-scaled sigmoid function (with $\kappa \gg 1$) and $\sigma_n$ is the standard deviation of the IR values falling within the window centered at time index $n$. Unlike the original formulation, the Soft EDP defined in (9) is differentiable. Hence, it can be used in the learning procedure to regularize the echo density of the produced IR [19] with

$$\mathcal{L}_{\mathrm{EDP}} = \frac{1}{L_h} \sum_n \|\boldsymbol{p}[n] - \hat{\boldsymbol{p}}[n]\|_2^2 \,, \tag{10}$$

where $\hat{\boldsymbol{p}}[n] := [\hat{\eta}_1[h], \ldots, \hat{\eta}_J[n]]^{\mathrm{T}}$ is the Soft EDP of the predicted IR. By combining (8) and (10), we can finally write down the loss function

$$\mathcal{L} = \mathcal{L}_{\mathrm{EDC}} + \lambda \mathcal{L}_{\mathrm{EDP}} \,, \tag{11}$$

where $\lambda$ is a positive real hyperparameter.

### 3.4. Parameter Initialization

We implement the proposed differentiable MIMO FDN with $N = 16$ delay lines in Python using PyTorch. For each of the considered examples, the differentiable MIMO FDN is trained for a maximum of 1000 iterations. In particular, we employ a single Adam optimizer with learning rate of 0.1 acting on $\mathbf{W}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, and $\boldsymbol{m}$. Moreover, all the parameters of the FDN are initialized with no prior knowledge of the multichannel RIR $\mathbf{h}[n]$.

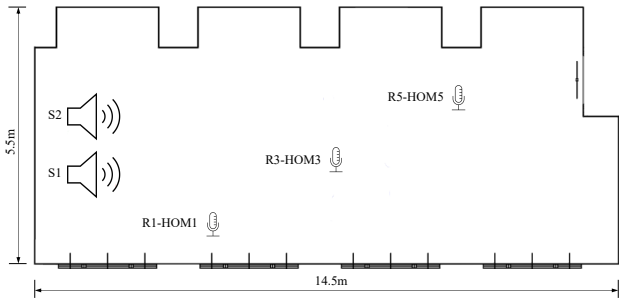


Figure 2: *Seminar room configuration of the multichannel RIR considered in Sec. 4.3.*

Then, for all $\imath\jmath$, we set $\tilde{\mathbf{B}}_{\imath\jmath}^{(0)} \sim \mathcal{N}(0, \frac{1}{N})$ and $\tilde{\mathbf{C}}_{\imath\jmath}^{(0)} = \frac{1}{N}$. We let $\tilde{\mathbf{D}}_{\imath\jmath}^{(0)} = 1$ and we initialize $\tilde{\mathbf{W}}^{(0)}$ and $\tilde{\mathbf{\Gamma}}^{(0)}$ such that $\tilde{\mathbf{W}}_{\imath\jmath}^{(0)} \sim \mathcal{N}(0, \frac{1}{N})$ and $\tilde{\gamma}_i^{(0)} \sim \mathcal{N}(0, \frac{1}{N})$. We set $\tilde{\boldsymbol{m}}^{(0)}$ so that $\tilde{m}_i^{(0)} = \psi \tilde{m}_i^\star$ with $\tilde{m}_i^\star \sim \text{Beta}(\alpha, \beta)$, for $i = 1, ..., N$, with $\alpha \geq 1$ and $\beta > \alpha$. We set $\psi = 1024$ as in [19], together with $\alpha = 1.1$ and $\beta = 6$ such that a maximum possible delay of 64 ms and a mean value of about 10 ms are ensured. We linearly vary $\kappa_n$, i.e., the parameter controlling the sigmoid scaling in the differentiable EDP loss term, from $10^2$ to $10^5$ with $n = 0, ..., L_h - 1$. Finally, we empirically set the hyperparameter $\lambda = 0.5$ as, in our experiments, it turns out to balance the two loss terms in (11).

## 4. EVALUATION

We evaluate the proposed methodology considering measured multichannel RIRs (resampled at 16 kHz) taken from two distinct datasets.

In particular, in Sec. 4.2, we consider the simpler case of $K = 1$ input and $J = 2$ outputs – thus, a single-input-multiple-output (SIMO) case – in the context of binaural rendering by employing the Aachen Impulse Response (AIR) dataset [21]. The AIR corpus comprises Binaural Room Impulse Responses (BRIR) acquired in four low-reverberant rooms (e.g., studio booth, meeting room, etc.) with and without a dummy head. Amongst others, we select one of the BRIRs acquired in the meeting room with dummy-head-source distance equal to 2.25 m. The IDs of the selected BRIRs are `air_meeting_1_1_4` (left ear) and `air_meeting_0_1_4` (right ear), which are then assigned to channel $j = 1$ and $j = 2$, respectively, to form the target multichannel RIR. As a further step, we trim the last part of the multichannel RIR by setting $L_1 = L_2 = 2\bar{T}_{60} = 2 \cdot 0.23$ s, i.e., twice the average $T_{60}$ reported in [21], in order to remove possible noise that would impair the training procedure.

Then, in Sec. 4.3 we take into account the HOMULA-RIR dataset [22], a corpus of multichannel RIRs acquired in a seminar room by means of Higher-Order Microphones (HOMs) and a uniform linear array. In particular, we select the signal coming from the V capsule (i.e., the fifth capsule of the microphone) of all the HOMs shown in Fig. 2, i.e., the first HOM of the first row, the third HOM of the third row, and the fifth HOM of the fifth row. We consider the two sources, $S_1$ and $S_2$, acting at the same time by summing the two acquired impulse responses for each of the three microphones. Hence, in this example, we consider a multichannel RIR with $K = 2$ inputs and $J = 3$ outputs. The IDs of the RIRs are:

- (`rir-S1-R1-HOM1`, `rir-S2-R1-HOM1`),
- (`rir-S1-R3-HOM3`, `rir-S2-R3-HOM3`),
- (`rir-S1-R5-HOM5`, `rir-S2-R5-HOM5`),

where each pair is assigned to channel $j = 1, 2, 3$ of the target multichannel RIR, respectively. The latter does not undergo further trimming since the signals are already provided with a length in seconds comparable to the $T_{60}$.

### 4.1. Baseline

As for the SISO case [19], to the best of our knowledge, there are no methods in the literature that can be directly compared to ours for learning all the parameters of a MIMO FDN. However, in [3], a genetic algorithm is used to optimize some parameters of a SIMO FDN with the aim of matching a simulated BRIR. Moreover, we used the genetic algorithm (GA) introduced in [16] as one of the baselines for the method presented in [19] to optimize a SISO FDN. Thus, we decide to select and adapt to the MIMO case the method employed in [16] since deemed to be a suitable method of the literature for drawing a comparison.

In this work, we implement the GA algorithm of [16] with $N = 16$, a population of 50 *individuals*, and a number of generations equal to 50 in order to improve the output of the optimization. The individuals are, therefore, MIMO FDNs characterized by $N(1 + K + J) + KJ$ mutable parameters, which, in turn, determine $\boldsymbol{m}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$; the feedback matrix is not affected by the algorithm and is set at the beginning of the optimization equal to a random orthogonal matrix. Scalar gains are constrained to take value in the range $[-1, 1]$, whereas delays in the range $[0.0002, 0.064]$ s. At each iteration, the attenuation filters are updated according to $\boldsymbol{m}$ and the target octave-band reverberation times [11]. We consider two fitness functions: (i) the first, proposed in [16], which is the mean absolute error between the MFCCs of the target multichannel RIR and the MIMO FDN output; (ii) the second, which is the same cost function considered in this work for optimizing the proposed differentiable MIMO FDN. It is worth pointing out that not all the parameters of the FDN considered in [16] are frequency-independent given that attenuation and tone-control filters are inserted in the processing pipeline, as well as graphic equalizers. Ultimately, this gives the FDN associated to the GA method a higher descriptive power with respect to our frequency-independent FDN. However, as we will show in the next subsections, this does not prevent the method to minimize the loss function in (11) with optimal results.

Finally, the baseline is implemented in MATLAB starting from the authors' codebase, which exploits the Feedback Delay Network Toolbox (FDNTB) [24], and we make use of MATLAB's Global Optimization Toolbox for finding $\boldsymbol{m}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$.

### 4.2. Binaural RIR

As a first example, we consider the BRIR with $K = 1$ input and $J = 2$ outputs (i.e., SIMO case) referenced at the beginning of this section. The FDN is trained with $\lambda = 0.5$ and, after 686 iterations, reaches the best model with an overall loss $\mathcal{L} = 0.0105$, two orders of magnitude less than the loss at iteration 0; this was obtained with the FDN randomly initialized as explained in Sec. 3.4.

Fig. 3 shows the EDCs of the BRIR ("Target") marked with a solid black line, the IRs obtained by means of our approach

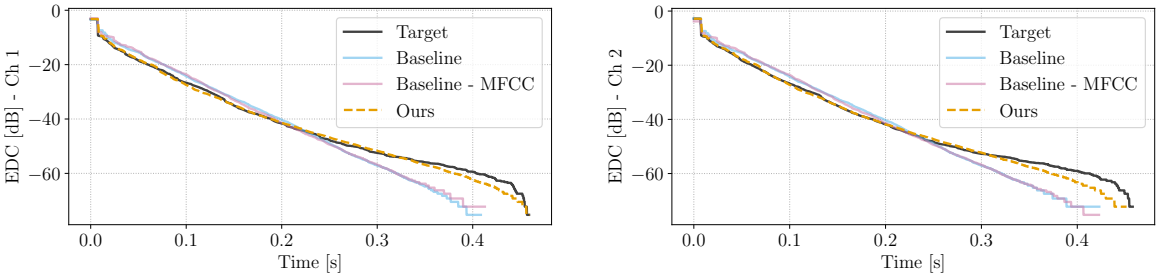


Figure 3: *EDC of the considered BRIR after* 686 *iterations.*

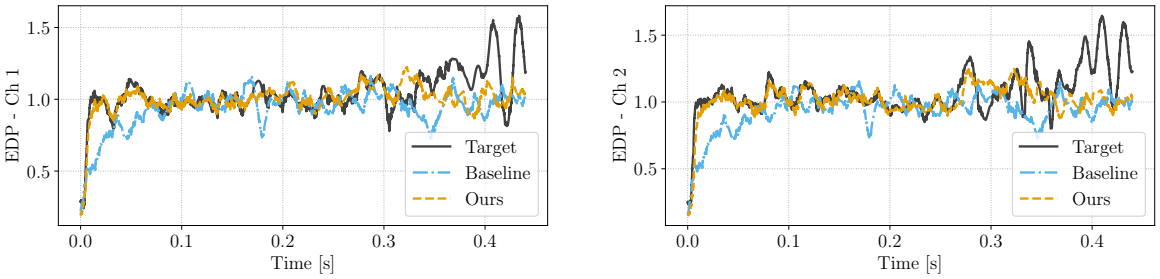


Figure 4: *EDP of the considered BRIR after* 686 *iterations.*

("Ours"), with dashed orange line, the baseline optimized considering the MFCC fitness function of [16] ("Baseline - MFCC"), with solid pink line, and the baseline optimized considering our fitness function ("Baseline"), marked with solid blue line; such a color convention will be used for all the plots analyzed in this section. First, we can state that the two different baselines are comparable as far as EDC is concerned, which leads to saying that the performance of the genetic algorithm does not depend much on which of the two fitness functions is considered. We performed further tests with other RIRs and they all led to the same result. Hence, from now on we will only consider the baseline optimized by means of (11) in order to provide a fair comparison with our method. Anyway, it is evident that our approach is able to better delineate the decay curve for both the two channels. This is confirmed looking at Figs. 5, where the IRs are compared to the target RIRs. In particular, the results of our approach are shown in Figs. 5(a) and (b), while those of the baseline method are shown in Figs. 5(c) and (d). The latter present peaks that are outside the target RIR envelope due to a wrong echo density. This is confirmed by looking at Fig. 4, where the EDPs are depicted. The orange curve nicely follows the black curve until the $T_{60}^{\max} = 0.316$ s since, we remind, we trained only for such a time span. The better agreement between the orange curve and the black curve in Fig. 4 is what makes the IR obtained with our approach closer to the target BRIR in Fig. 5. This, in turn, confirms the outcome of the ablation study presented in [19], corroborating the thesis that, for the proposed approach, the regularization term in (10) is instrumental for improving the matching between IRs and target RIRs.

Table 1 shows the reverberation times $T_{20}$ and $T_{60}$ and the errors $\Delta T_{20}$ and $\Delta T_{60}$ of the target RIR, the baseline IR, and our IR. The proposed approach performs better than the baseline for both channel 1 and 2, with an error $\Delta T_{20}$ of 2.2 ms and 19.3 ms, an error $\Delta T_{60}$ of 7.7 ms and 7.9 ms, respectively. In general, our approach led to errors one order of magnitude less than those of the baseline method, further demonstrating the good performance of the methodology.

Table 1: *Reverberation times for the considered BRIR.*

| | | $T_{20}$ [s] | $\Delta T_{20}$ [s] | $T_{60}$ [s] | $\Delta T_{60}$ [s] |
|---|---|---|---|---|---|
| Target | Ch 1 | 0.3156 | — | 0.4449 | — |
| | Ch 2 | 0.2963 | — | 0.4442 | — |
| Baseline | Ch 1 | 0.3308 | 0.0152 | 0.3578 | 0.0871 |
| | Ch 2 | 0.3270 | 0.0307 | 0.3558 | 0.0885 |
| Ours | Ch 1 | 0.3178 | **0.0022** | 0.4372 | **0.0077** |
| | Ch 2 | 0.3156 | **0.0193** | 0.4364 | **0.0079** |

### 4.3. Multichannel RIR

Let us now consider the HOMULA-RIR dataset and the multichannel RIR with $K = 2$ inputs and $J = 3$ outputs (i.e., MIMO case) obtained as mentioned above. The MIMO FDN is trained once again with $\lambda = 0.5$ and yields the best model after 776 iterations with a loss $\mathcal{L} = 0.0025$, three orders of magnitude less than the loss at iteration 0.

Fig. 6 shows the EDC of baseline and our IRs together with that of the target multichannel RIR. We can clearly spot that our method is able to learn the decay for each channel $j$, especially in the first 0.5 s, i.e., where most of the reflections happen to be. This can be evinced by looking at Figs. 7(a), (b), and (c), where the multichannel IR obtained by means of our approach is directly compared to the given RIR. It is worth pointing out once again the

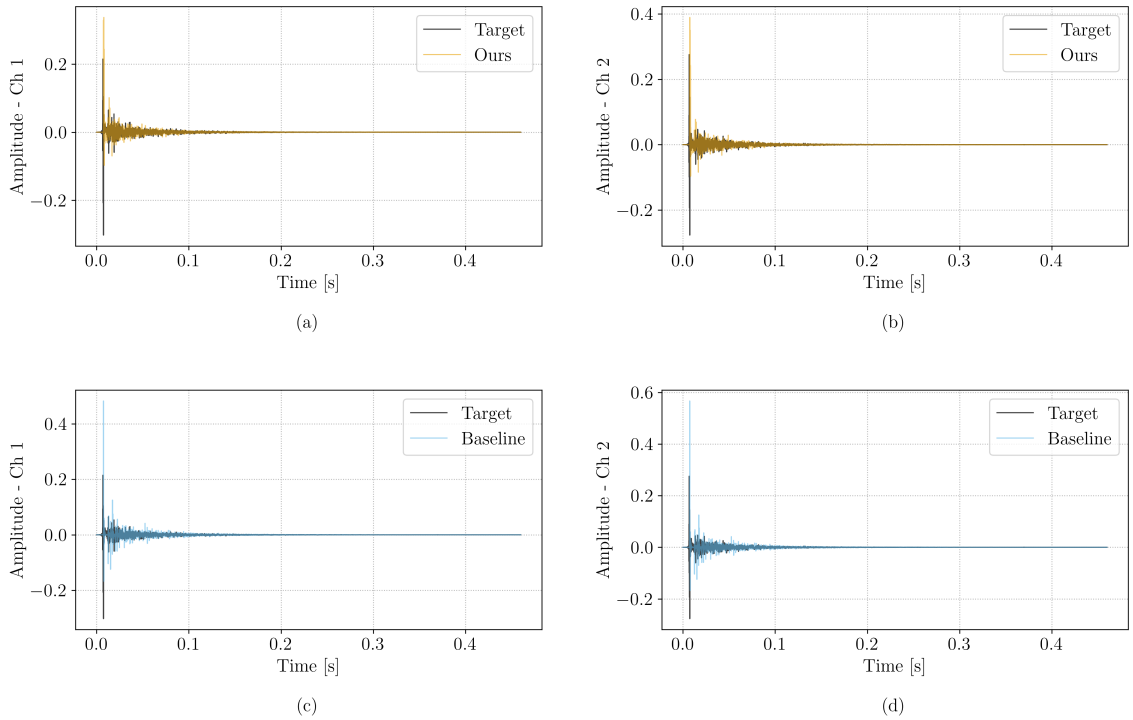


Figure 5: *IRs of the MIMO FDN approximating the considered BRIR optimized with the proposed approach (a), (b) and the baseline approach (c), (d).*

regularization brought by the EDP loss term that nicely makes the IR envelopes to match, which, unfortunately, due to constrained space we are not able to show. On the other hand, the baseline method is not able to jointly optimize the MIMO FDN parameters such that each channel decay curve matches the target. This is further corroborated by the results reported in Figs. 7(d), (e), and (f), where the IRs obtained with the baseline method are far from the desired goal. Finally, it is worth stressing that this happens despite the higher modeling capability of the FDN implemented in the baseline method, making our approach even more effective.

As far as the reverberation times reported in Table 2 are concerned, we can clearly state that once again the proposed approach is characterized by the lowest errors. In particular, we obtain an error $\Delta T_{20}$ of 23 ms, 9.2 ms, and 25.1 ms, and an error $\Delta T_{60}$ of 6.3 ms, 2.1 ms, and 17.1 ms, for channel $j = 1$, $j = 2$, and $j = 3$, respectively. Moreover, by looking at the baseline $T_{20}$ values we can state that, at least in the first milliseconds, the optimization driven by the genetic algorithm is not able to grasp the fine difference between the channels, whereas our approach does, showing its potential capability to model the little nuances that exist among channels.

Table 2: *Reverberation times for the considered multichannel RIR.*

| | | $T_{20}$ [s] | $\Delta T_{20}$ [s] | $T_{60}$ [s] | $\Delta T_{60}$ [s] |
|---|---|---|---|---|---|
| Target | Ch 1 | 0.7982 | — | 0.9302 | — |
| | Ch 2 | 0.8047 | — | 0.9244 | — |
| | Ch 3 | 0.8469 | — | 0.8969 | — |
| Baseline | Ch 1 | 0.9246 | 0.1264 | 0.8727 | 0.0576 |
| | Ch 2 | 0.9246 | 0.1198 | 0.8662 | 0.0581 |
| | Ch 3 | 0.9246 | 0.0776 | 0.8574 | 0.0395 |
| Ours | Ch 1 | 0.7746 | **0.0236** | 0.9366 | **0.0063** |
| | Ch 2 | 0.7956 | **0.0092** | 0.9265 | **0.0021** |
| | Ch 3 | 0.8218 | **0.0251** | 0.9139 | **0.0171** |

## 5. CONCLUSIONS

In this article, we proposed for the first time a differentiable MIMO Feedback Delay Network (FDN) for multichannel room acoustics simulation. Starting from previous work on SISO FDNs, we provided a reparametrization of the frequency-independent MIMO FDN that allowed us to learn all the parameters through automatic differentiation, with the aim of jointly modeling and rendering a given multichannel RIR. We did this by minimizing a perceptually-informed loss function that takes into account both the energy decay curve and the echo density profile. We evaluated our approach on items taken from two distinct datasets, tackling, in particular, the case of binaural RIR and multichannel RIR modeling. We showed that the proposed approach is able to overcome the selected baseline, i.e., a method for the FDN optimization via genetic algorithm, proving it to be suitable for jointly render IRs that match the considered perceptual characteristics of the target RIRs and paving the way towards efficient and lightweight algorithms for real-time multichannel acoustics simulation.

Future work may entail the extension of the proposed method to frequency-dependent MIMO FDNs for matching time-frequency decays, to MIMO FDNs characterized by a higher number of inputs/outputs, and the use of interpolative and/or regression techniques with the aim of obtaining the IRs in points in space that are not taken into account by the target multichannel RIR.

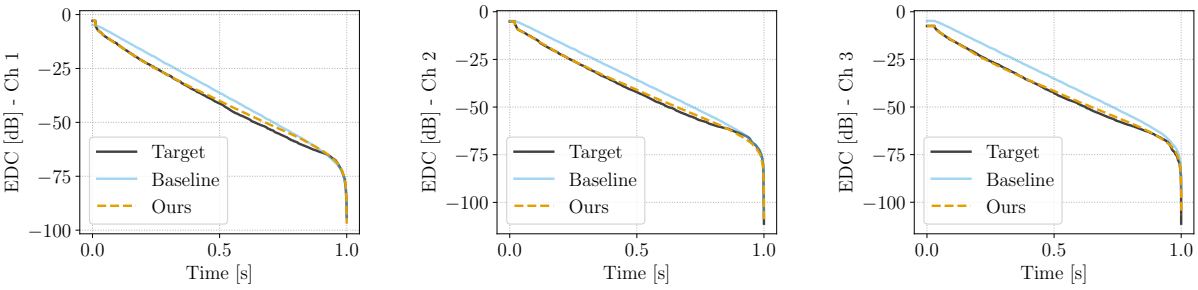


Figure 6: *EDC of the considered multichannel RIR after* 776 *iterations.*

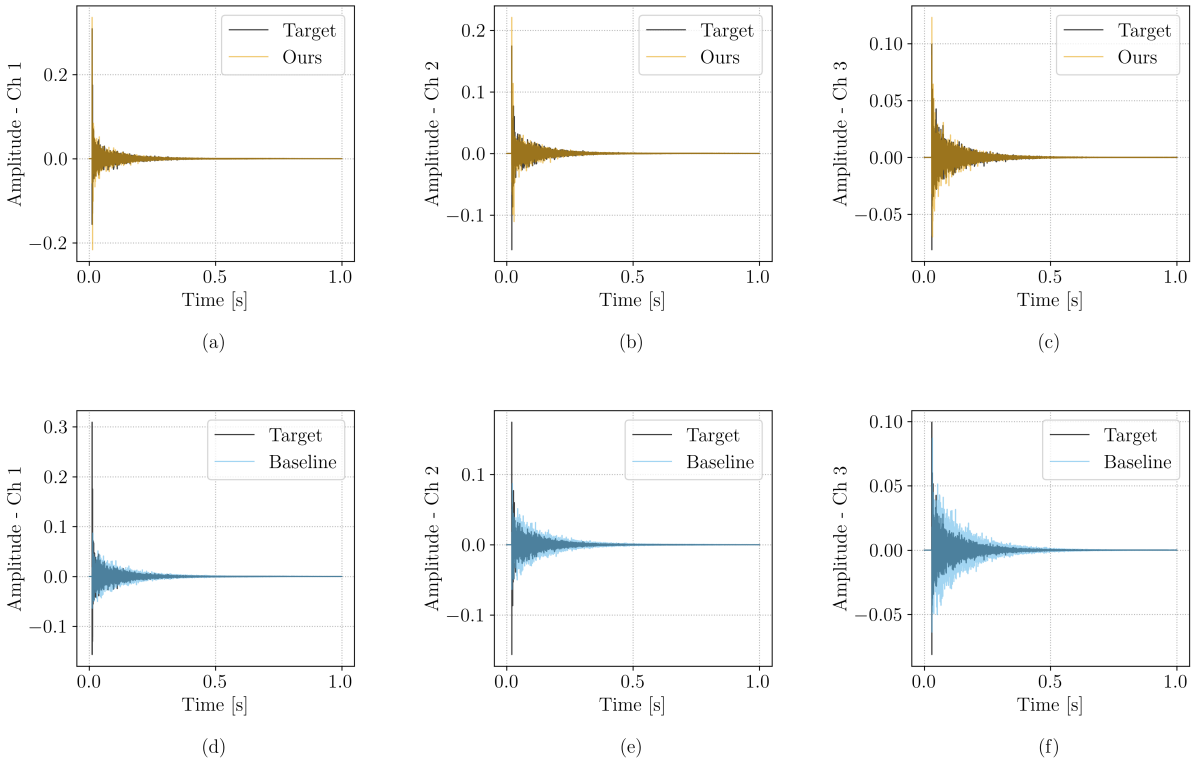


Figure 7: *IRs of the MIMO FDN approximating the considered multichannel RIR optimized with the proposed approach (a), (b), (c) and the baseline approach (d), (e), (f).*

## 6. REFERENCES

[1] John G. Apostolopoulos, Philip A. Chou, Bruce Culbertson, Ton Kalker, Mitchell D. Trott, and Susie Wee, "The road to Immersive Communication," *Proc. of the IEEE*, vol. 100, no. 4, pp. 974–990, 2012.

[2] 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.

[3] Justin Shen and Ramani Duraiswami, "Data-Driven Feedback Delay Network Construction for Real-Time Virtual Room Acoustics," in *Proc. of the 15th Int. Audio Mostly Conf.*, New York, NY, USA, Sept. 2020, pp. 46–52.

[4] Luis Fialho, Jorge Oliveira, Andre Filipe, and Filipe C. Luz, "Soundspace VR: Spatial Navigation using Sound in Virtual Reality," *Virtual Reality*, vol. 27, no. 1, pp. 397–405, 2023.

[5] Antonella Bevilacqua, Francesca Merli, and Lamberto Tronchin, "Development of MIMO technique for 3D Auralization," in *Proc. of the 2021 I3DA*, Bologna, Italy, 2021, pp. 1–5.

[6] Antonella Bevilacqua, Francesca Merli, Angelo Farina, Enrico Armelloni, Adriano Farina, and Lamberto Tronchin, "3DOF Representation of the Acoustic Measurements Inside the Comunale-Pavarotti Theatre of Modena," in *Proc. of the 2021 I3DA*, Bologna, Italy, 2021, pp. 1–4.

[7] Antonella Bevilacqua, Alessandro Martinetti, and Giacomo Tentoni, "The Workmanship of Luthiers in the House of Violin: The Auditorium G. Arvedi of Cremona and the Acoustic

Features suggested by Toyota and Nagata," in *Proc. of the 2023 I3DA*, Bologna, Italy, 2023, pp. 1–4.

[8] Vesa Välimäki, Julian D. Parker, Lauri Savioja, Julius O. Smith, and Jonathan S. Abel, "Fifty Years of Artificial Reverberation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 20, no. 5, pp. 1421–1448, 2012.

[9] Esrnee Henrieke Anne De Haas and Lik-Hang Lee, "Deceiving Audio Design in Augmented Environments : A Systematic Review of Audio Effects in Augmented Reality," in *Proc. of the IEEE Int. Symp. on Mixed and Augment. Reality Adjunct*, Singapore, Singapore, 2022, pp. 36–43.

[10] Michael A. Gerzon, "Synthetic Stereo Reverberation: Part one," *Studio Sound*, vol. 13, pp. 631–635, 1971.

[11] Sebastian J Schlecht and Emanuël A P Habets, "Accurate Reverberation Time Control in Feedback Delay Networks," in *Proc. of the 20th Int. Conf. on Digital Audio Effects*, Edinburgh, UK, 2017.

[12] Karolina Prawda, Sebastian J. Schlecht, and Vesa Välimäki, "Improved Reverberation Time Control for Feedback Delay Networks," in *Proc. of the 22nd Int. Conf. on Digital Audio Effects*, Birmingham, UK, 2019, pp. 1–7.

[13] Sebastian J. Schlecht and Emanuel 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, 2017.

[14] Davide Rocchesso, "Maximally Diffusive Yet Efficient Feedback Delay Networks for Artificial Reverberation," *IEEE Signal Process. Lett.*, vol. 4, no. 9, pp. 252–255, 1997.

[15] Michael Chemistruck, Kyle Marcolini, and Will Pirkle, "Generating Matrix Coefficients for Feedback Delay Networks using Genetic Algorithm," in *Proc. of the 133rd Audio Eng. Soc. Convention*, San Francisco, CA, USA, 2012.

[16] Ilias Ibnyahya and Joshua D Reiss, "A Method for Matching Room Impulse Responses with Feedback Delay Networks," in *Proc. of the 153rd Audio Eng. Soc. Convention*, New York, NY, USA, 2022.

[17] Sungho Lee, Hyeong-Seok Choi, and Kyogu Lee, "Differentiable Artificial Reverberation," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 30, pp. 2541–2556, 2022.

[18] Gloria Dal Santo, Karolina Prawda, Sebastian Schlecht, and Vesa Välimäki, "Differentiable Feedback Delay Network For Colorless Reverberation," in *Proc. of the 26th Int. Conf. on Digital Audio Effects*, Copenhagen, Denmark, 2023, pp. 244–251.

[19] 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*, 2023, arXiv:2404.00082.

[20] Alessandro Ilic Mezza, Riccardo Giampiccolo, and Alberto Bernardini, "Modeling the Frequency-Dependent Sound Energy Decay of Acoustic Environments With Differentiable Feedback Delay Networks," in *Proc. of the 27th Int. Conf. on Digital Audio Effects*, Guildford, UK, 2024.

[21] Marco Jeub, Magnus Schafer, and Peter Vary, "A Binaural Room Impulse Response Database for the Evaluation of Dereverberation Algorithms," in *Proc. of the 16th Int. Conf. on Digital Signal Process.*, Santorini, Greece, 2009, pp. 1–5.

[22] Federico Miotello, Mirco Pezzoli, Luca Comanducci, Alberto Bernardini, Fabio Antonacci, and Augusto Sarti, "HOMULA-RIR: A Room Impulse Response Dataset for Teleconferencing and Spatial Audio Applications Acquired Through Higher-Order Microphones and Uniform Linear Microphone Arrays," in *Proc. of the 2024 HSCMA*, Seoul, Korea, 2024.

[23] Michael A. Gerzon, "Unitary (Energy-Preserving) Multichannel Networks with Feedback," *Electron. Lett.*, vol. 12, no. 11, pp. 278–279, 1976.

[24] Sebastian J. Schlecht, "FDNTB: The Feedback Delay Network Toolbox," in *Proc. of the 23rd Int. Conf. on Digital Audio Effects*, Vienna, Austria, 2020, pp. 211–218.

[25] Hequn Bai, Gael Richard, and Laurent Daudet, "Late Reverberation Synthesis: From Radiance Transfer to Feedback Delay Networks," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 23, no. 12, pp. 2260–2271, 2015.

[26] Huseyin Hacihabiboglu, Enzo De Sena, and Zoran Cvetkovic, "Frequency-Domain Scattering Delay Networks for Simulating Room Acoustics in Virtual Environments," in *Proc. of the 7th Int. Conf. on Signal Image Tech. & Internet-Based Syst.*, Dijon, France, 2011, pp. 180–187.

[27] Vesa Välimäki, Karolina Prawda, and Sebastian J. Schlecht, "Two-Stage Attenuation Filter for Artificial Reverberation," *IEEE Signal Process. Lett.*, vol. 31, pp. 391–395, 2024.

[28] 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.

[29] Atılım Gunes Baydin, Barak A. Pearlmutter, Alexey Andreyevich Radul, and Jeffrey Mark Siskind, "Automatic Differentiation in Machine Learning: a Survey," *J. Mach. Learn. Res.*, vol. 153, pp. 1–43, 2018.

[30] Mario Lezcano-Casado and David Martinez-Rubio, "Cheap Orthogonal Constraints in Neural Networks: A Simple Parametrization of the Orthogonal and Unitary Group," in *Proc. of the 36th Int. Conf. on Machine Learning*, Long Beach, CA, USA, 2019, pp. 3794–3803.

[31] Jonathan S. Abel and Patty Huang, "A Simple, Robust Measure of Reverberation Echo Density," in *Proc. of the 121st Audio Eng. Soc. Convention*, San Francisco, CA, USA, 2006.