FDNTB: THE FEEDBACK DELAY NETWORK TOOLBOX

Sebastian J. Schlecht

Acoustics Lab, Dept. of Signal Processing and Acoustics
Media Lab, Dept. of Media
Aalto University, Espoo, Finland
sebastian.schlecht@aalto.fi

ABSTRACT

Feedback delay networks (FDNs) are recursive filters, which are widely used for artificial reverberation and decorrelation. While there exists a vast literature on a wide variety of reverb topologies, this work aims to provide a unifying framework to design and analyze delay-based reverberators. To this end, we present the Feedback Delay Network Toolbox (FDNTB), a collection of the MATLAB functions and example scripts. The FDNTB includes various representations of FDNs and corresponding translation functions. Further, it provides a selection of special feedback matrices, topologies, and attenuation filters. In particular, more advanced algorithms such as modal decomposition, time-varying matrices, and filter feedback matrices are readily accessible. Furthermore, our toolbox contains several additional FDN designs. Providing MATLAB code under a GNU-GPL 3.0 license and including illustrative examples, we aim to foster research and education in the field of audio processing.

1. INTRODUCTION

If a sound is emitted in a room, the sound waves travel through space and are repeatedly reflected at the room boundaries resulting in acoustic reverberation [1]. Many artificial reverberators have been developed in recent years [2, 3], among which the feedback delay network (FDN), initially proposed by Gerzon [4] and further developed in [5, 6], is one of the most popular. The FDN consists of N delay lines combined with attenuation filters, which are fed back via a scalar feedback matrix $\boldsymbol{A}$. Thus, any filter topology of interconnected delays may be represented as an FDN in a delay state space (DSS) representation, similar to the general state space (SS) representations, which is an interconnection of unit delays. Therefore, FDN framework provide the means for a systematic investigation of a wide variety of filter topologies such as Moorer-Schroeder [7], nested allpasses [8], allpass and delay combinations [9], and many more. Equivalent structures are digital waveguides [10], waveguide webs [11], scattering delay networks [12] and directional FDNs [13].

Artificial reverberation can be alternatively applied by directly convolving the source signal with a room impulse response (RIR) [2]. Whereas the general representation as a finite impulse response (FIR) tends to imply higher computational costs, recent developments yielded partitioned fast convolution schemes with highly optimized implementation [14]. As any RIR, including the FDN impulse responses, can be applied by convolution, it might appear as a more general method. However, there are a few important advantages of FDNs. Where in principle, any combination of acoustic features can be applied by convolution as long as they are linear and time-invariant, in practice numerical generation or acoustic measurements are complex and involved topics [2]. Further, FDNs allow time-variation by modulating the input and output gains and delays for early reflections of a moving source [15, 12], and modulation of the feedback delays [16] and feedback matrix [17]. Also, synthesizing auditory scenes with multiple sources and multiple outputs for spatial reproduction scales computationally well with FDNs compared to individual source-to-receiver RIR convolution [18].

There are several central challenges in the design of FDNs, which are only partly addressed in the research literature. A significant challenge of FDN design is the inherent trade-off between three aspects: computational complexity, mode density, and echo density. Reduced modal density can lead to metallic sounding artifacts [19, 20], while reduced echo density can cause rough rattling sounds. A higher number of delays increases both modal and echo density, but also the computational complexity. Although attempts have been made [21, 22], it remains open how to achieve spectrally and temporally smooth FDNs with a minimal number of delays. A closely connected topic is the choice of delay length. While the co-prime criterium introduced by Schroeder [23] remains popular and extensions exists [24], actual delay lines choices are still open [25]. Recently, attempts have been made to quantify the spectral quality of FDNs by statistical measures [26] and based on modal decomposition [27], but perceptual verification and application examples need to be provided.

This work presents a Feedback Delay Network Toolbox (FDNTB) to support future research in this area. The toolbox contains a wide variety of conversion functions between different FDN representations such as delay state space, state space, modal, and rational transfer function. Further, matrix generation functions for feedback matrices such as Hadamard, Circulant, and, random orthogonal are provided. Some well-known structures such as the Moorer-Schroeder or nested allpasses are provided as well. Also, the toolbox provides additional code for various example applications. Such applications include time-varying feedback matrices, filter feedback matrices, proportional attenuation filters and reverberation time estimation. We believe that supplying an entire collection of different FDN approaches along with example applications within a unifying framework can be highly beneficial for both researchers as well as educators in the field of audio processing. The FDNTB can be found online[1] and is provided under the GPL-3.0 license.

The remainder of this work is organized as follows. In Sec-

[1] https://github.com/SebastianJiroSchlecht/fdnToolbox

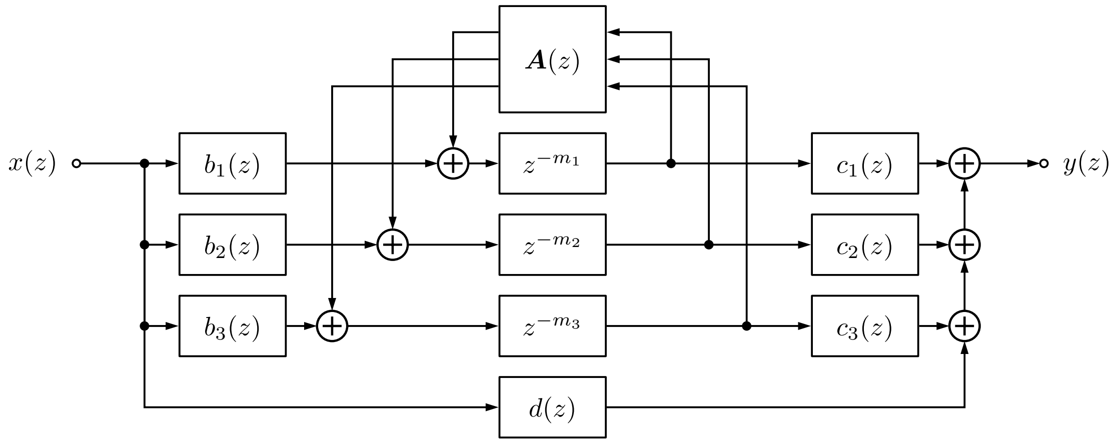


Figure 1: Filter feedback delay network (FFDN) with three delays, i.e., $N = 3$ and single input and output, i.e., $N_{\text{in}} = 1$ and $N_{\text{out}} = 1$, respectively. All input, output and direct gains are potentially filters as well as the filter feedback matrix (FFM) $\boldsymbol{A}(z)$.

tion 2, we review the general FDN structure, lossless and lossy systems as well as topological variations. In Section 3, we review a range of feedback matrix types and generation algorithms. Corresponding FDNTB function names are indicated as `function`.

2. FEEDBACK DELAY NETWORKS

In this section, we introduce FDN formally and review various representations of FDNs. Further, we consider lossless and lossy FDNs and the corresponding matrices and filters. This section is concluded with an overview of topology variations of the standard FDN.

2.1. Feedback Delay Networks (FDNs)

The single-input-single-output (SISO) FDN is given in time domain by the difference relation [28]

$$\begin{aligned} y(n) &= \boldsymbol{c}^\top \boldsymbol{s}(n) + d\, x(n) \\ \boldsymbol{s}(n+\boldsymbol{m}) &= \boldsymbol{A}\, \boldsymbol{s}(n) + \boldsymbol{b}\, x(n), \end{aligned} \tag{1}$$

where $x(n)$ and $y(n)$ are the input and output values at time sample n, respectively, and $\cdot^\top$ denotes the transpose operation. The FDN dimension N is the number of delay lines and we occasionally write N-FDN. The $N \times N$ matrix $\boldsymbol{A}$ is the *feedback matrix*, $N \times 1$ vector $\boldsymbol{b}$ of *input* gains, $N \times 1$ vector $\boldsymbol{c}$ of *output* gains and scalar d is the *direct* signal gain. The lengths of the N delay lines in samples are given by the vector $\boldsymbol{m} = [m_1, \dots, m_N]$. The $N \times 1$ vector $\boldsymbol{s}(n)$ denotes the delay-line outputs at time n. The vector argument notation $\boldsymbol{s}(n+\boldsymbol{m})$ abbreviates the vector $[s_1(n+m_1), \dots, s_N(n+m_N)]$. The system order of a standard FDN in (1) is

$$\mathfrak{N} = \sum_{j=1}^{N} m_j. \tag{2}$$

In Section 3, we discuss the time-varying feedback matrix $\boldsymbol{A}(n)$ as effective manipulation of the resulting reverberation. Any gain in Eq. (1) may consist of finite and infinite impulse response (FIR and IIR) filters, for instance, a filter feedback matrix (FFM) $\boldsymbol{A}(z)$ instead of a scalar feedback matrix $\boldsymbol{A}$ (see Fig. 1). The transfer function of the filtered FDN (FFDN) in the z-domain, corresponding to the difference relation in (1), is

$$H(z) = \frac{Y(z)}{X(z)} = \boldsymbol{c}(z)^\top [\boldsymbol{D_m}(z^{-1}) - \boldsymbol{A}(z)]^{-1} \boldsymbol{b}(z) + d(z), \tag{3}$$

where $X(z)$ and $Y(z)$ are the z-domain representations of the input and output signals $x(n)$ and $y(n)$, respectively, and $\boldsymbol{D_m}(z) = \operatorname{diag}\left([z^{-m_1}, z^{-m_2}, \dots, z^{-m_N}]\right)$ is the diagonal $N \times N$ delay matrix. We abbreviate the loop transfer function with $\boldsymbol{P}(z) = \boldsymbol{D_m}(z^{-1}) - \boldsymbol{A}(z)$. Alternatively, every gain and delay in (1) can be time-varying to adjust to changing acoustic scenes.

2.2. Representations

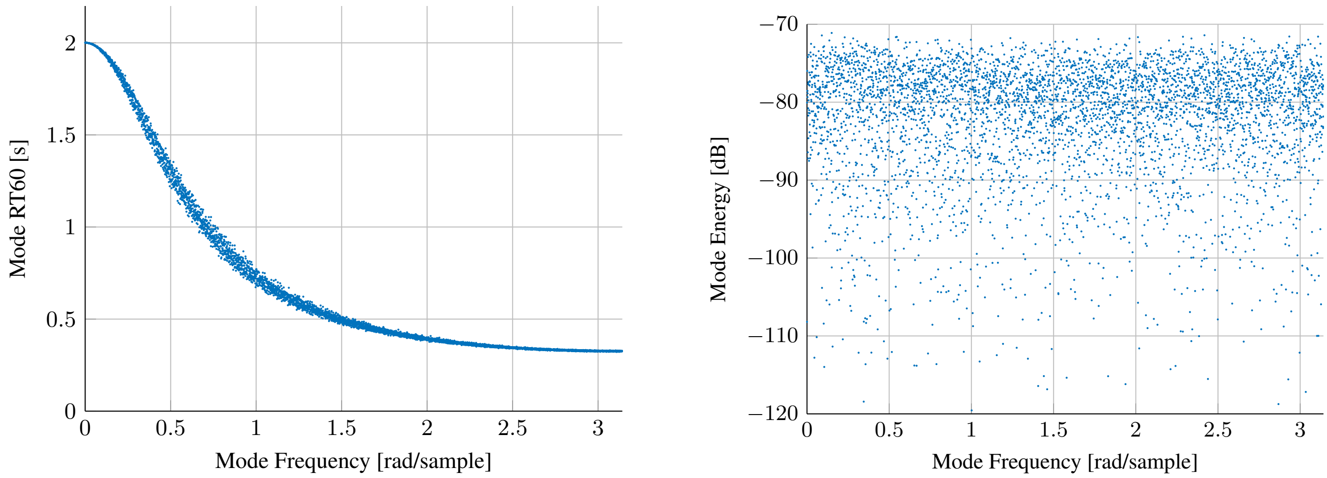
There are multiple useful representations of FDNs. The representation (1) is called a delay state-space (DSS) representation and (3) is the corresponding transfer function. Rocchesso [28] derived an equivalent standard state-space (SS) representation, i.e., with all delays equal to 1, see `dss2ss`. The matrix size of the equivalent SS is then equal to $\mathfrak{N}$ in (2). The transfer function is equivalently given by $H(z) = \frac{q(z)}{p(z)}$, where

$$q(z) = d(z) \det(\boldsymbol{P}(z)) + \boldsymbol{c}(z)^\top \operatorname{adj}(\boldsymbol{P}(z)) \boldsymbol{b}(z) \tag{4}$$

$$p(z) = \det(\boldsymbol{P}(z)), \tag{5}$$

where adj denotes the matrix adjungate. The transfer function form holds equally for the SS representation, see `dss2tf` and `dss2tfSym`. Please note that in the multi-input-multi-output (MIMO) case, $p(z)$ is a scalar function, where $q(z)$ is a matrix which describes the input-output relation of size $N_{\text{out}} \times N_{\text{in}}$. The polynomial coefficients are computed from the principal minors as given in [29, Lemma 1], see `generalCharPoly` and `generalCharPolySym`.

The roots of the polynomial $p(z)$ are the system poles, whereas the roots of each entry of $q(z)$ are the system zeros of the corresponding input-output combination. The modal decomposition of an FDN computes the partial fraction decomposition



(a) Reverberation time of poles

(b) Magnitude of residues

Figure 2: Poles and residues for an 8-FDN with a frequency-dependent decay. Each dot indicates one mode in (a) frequency and reverberation time and (b) frequency and residues magnitude. Delays are $\boldsymbol{m} = [2300, 499, 1255, 866, 729, 964, 1363, 1491]$ and $\boldsymbol{A}$ is a random orthogonal matrix.

of the transfer function in (3), i.e.,

$$H(z) = \sum_{i=1}^{\mathfrak{N}} \frac{\rho_i}{1 - \lambda_i\, z^{-1}}, \tag{6}$$

where ρ_i is the residue corresponding to the pole λ_i. Figure 2 shows the poles and residues of an FDN with frequency-dependent reverberation time. The FDNTB provides the polynomial Ehrlich-Aberth Method to compute the modal decomposition in (6), see `dss2pr`. Alternatively, the modal decomposition can be computed from the SS or transfer function representation, see `dss2pr_direct`. The residues in (6) can be computed in two ways, directly from the transfer function [27, Section II-D], `dss2res` or from the impulse response with a least-squares fit [30], see `impz2res`.

Typically, the efficient DSS representation (1) is used to implement the FDN, see `dss2impz`, but the impulse response can be produced based on each of the representations. Therefore, we provide also impulse responses from poles and residues as well as the matrix transfer function representation, see `pr2impz` and `mtf2impz`, respectively.

2.3. Lossless and Lossy Feedback

As a first step when designing, FDNs are commonly constructed as lossless systems, i.e., all system poles lie on the unit circle [31]. The lossless property of general unitary-networks, which in particular applies to the FDN with a filter feedback matrix $\boldsymbol{A}(z)$, was described by Gerzon [31]. An FDN is lossless if $\boldsymbol{A}(z)$ is paraunitary, i.e., $\boldsymbol{A}(z^{-1})^H \boldsymbol{A}(z) = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix and $\cdot^H$ denotes the complex conjugate transpose [31]. For real scalar matrices $\boldsymbol{A}$, the FDN is lossless if $\boldsymbol{A}$ is orthogonal, i.e., $\boldsymbol{A}^\top \boldsymbol{A} = \boldsymbol{I}$. However also non-orthogonal feedback matrices may yield lossless FDNs [32, 33], and we give some examples in Section 3.

Homogeneous loss is introduced into a lossless FFDN by replacing each delay element z^{-1} with a lossy delay filter $\gamma(z)z^{-1}$, where $\gamma(z)$ is ideally zero-phase with a positive frequency-response. The frequency-dependent gain-per-sample $\gamma(e^{\imath\omega})$ relates to the resulting reverberation time $T_{60}(\omega)$ by

$$\gamma(e^{\imath\omega}) = \frac{-60}{f_s T_{60}(\omega)}, \tag{7}$$

where f_s is the sampling frequency and ω is the angular frequency [5]. However, as substitution with lossy delays is impractical, the attenuation filters are lumped into a single filter per delay line. In standard FDNs with $\boldsymbol{A}(z) = \boldsymbol{U}\boldsymbol{\Gamma}(z)$, where $\boldsymbol{U}$ is lossless, the delay-proportional attenuation filters should satisfy [5]

$$|\boldsymbol{\Gamma}(e^{\imath\omega})| = \operatorname{diag}(\gamma(e^{\imath\omega})^{\boldsymbol{m}})m \tag{8}$$

where $|\cdot|$ denotes the absolute value. There are various absorption filters ranging from the computationally efficient one-pole shelving filter [5], see `onePoleAbsorption`, to highly accurate graphical equalizers [34], see `absorptionGEQ`, and FIR filters in `absorptionFilters`. With the modal decomposition (`dss2pr`), it is possible to demonstrate the lossless and lossy behavior of FDNs. In Fig 2, a pole-residue decomposition is depicted for a 8-FDN with one-pole shelving filters designed with $T_{60} = 2$s for the low and $T_{60} = 0.4$s for the high frequencies. Although, the reverberation time follows the specified values, due to the inaccurate magnitude and phase component, the reverberation time can deviate from the specification.

2.4. Topology Variations

The absorption filters can be placed also directly after the delay line such that the first pass through the delays are filtered as well. In a similar manner, the feedback matrix is occasionally placed on the forward path to increase the density on the first pass through. Another related variation are extra allpass filters in the feedback loop [35, 36]. Further, extra tap in and out points in the main delay lines were proposed to increase the echo density and reduce the initial time gap. More geometrically informed FDNs [12, 37, 13] may introduce extra filters and delays to account for source and receiver positions.

Please note that any of the mentioned topology variations can be represented and analyzed in formulation (3) by additional filtering of the input and output gains. Although, the computational complexity may differ from the optimal arrangements, in this work we prioritize the comparability and generalizability and leave the efficient implementation for the application scenario.

3. FEEDBACK MATRICES

In this section, we present a broad collection of feedback matrices which are useful in the context of FDN design.

3.1. Lossless Feedback Matrices

The most important class of feedback matrices in FDNs are the lossless matrices such that all system poles λ_i are on the unit circle. For the real matrices, we review special designs below, see `fdnMatrixGallery`. To test statistically properties of FDNs, we often choose random orthogonal matrices. Many random orthogonal matrix generators do not sample the space of possible matrices equally, e.g., Gram-Schmidt orthogonalization. For this reason, we provide the `randomOrthogonal` function, which samples the space of all orthogonal matrices uniformly. The orthogonal matrices can be diagonally similar such that the lossless property is retained (see `diagonallyEquivalent`) [32]. The reverse process is less trivial, i.e., determining whether a matrix $\boldsymbol{A}$ is a diagonally similar to an orthogonal matrix. Such a process might be necessary to determine whether a given matrix is lossless. The provided algorithm `isDiagonallySimilarToOrthogonal` is based on [41]. For instance, the allpass FDN matrix (shown below) can be shown to be lossless, although not orthogonal.

Alternatively, we can start with an arbitrary feedback matrix, e.g., inspired by a physical design [12, 37], and find the nearest orthogonal matrix. This so-called Procrustes problem solution is provided in `nearestOrthogonal`. However, often the specification does not necessary specify the phase (= sign) of the matrix entries as it results from an energy-based derivation. For instance, one might be interested in a feedback matrix which distributes the energy from each delay line equally. As the conventional Procrustes solution can give poor results, we have developed the sign-agnostic version in [42] given `nearestSignAgnosticOrthogonal`.

Further, it is useful to interpolate between two given orthogonal matrices. However, the linear interpolation between matrix entries does typically not yield orthogonal matrices. Instead, we proposed to perform the interpolation for the matrix logarithms [39]. The matrix exponentials map the antisymmetric matrices to orthogonal matrices. Because the linear interpolation between antisymmetric matrices remains antisymmetric, matrix exponential approach yields orthogonal interpolation matrices. We also provide a special implementation of the inverse function, the matrix logarithm, see `realLogOfNormalMatrix` based on [43]. For instance, `interpolateOrthogonal` allows to interpolate between a Hadamard matrix and an identity matrix to adjust the density of the matrix continuously and therefore the time-domain density of the resulting impulse response [39].

3.2. Scalar Feedback Matrices

Here, we review a number of important scalar feedback matrices. Table 1 lists many of the proposed matrices with the associated operation counts. The implementation cost of the matrix-vector-multiplication for a single time step vary from a conventional matrix multiplication N^2 down to linear number of operations N. Many of the examples are lossless matrices and loss is introduced by additional attenuation filters. Some feedback matrices, most notably from connections of allpass filters, do not have a lossless prototype as the poles and zeros would cancel out at this limit case.

Some of the presented examples result from translating well-known reverb topologies into a compact FDN representation. Feedforward-feedback allpass filters have been introduced with the delay lines to increase the short-term echo density [40, 7]. Alternatively, allpass filters may be placed after the delay lines [35, 36], which in turn doubles the effective size of the FDN [29]. Gardner proposed the nested allpass structure by [8], which recursively replaces the delay in the allpass with another allpass. As a unified representation, Fig. 4 depicts an overview of the present feedback matrices.

3.3. Filter Feedback Matrices

If the sound is reflected at a flat, hard boundary, the reflection is coherent (specular), while it is incoherent (scattered) when reflected by a rough surface. Towards a possible integration of scattering-like effects in FDNs, we introduced in [33] the delay feedback matrix (DFM), where each matrix entry is a scalar gain and a delay. In [44], we generalized the feedback matrix of the FDN to a filter feedback matrix (FFM), which then results in a filter feedback delay network (FFDN). As a special case of the FFM, we present the velvet feedback matrix (VFM), which can create ultra-dense impulse responses at a minimal computational cost [45].

FIR filter feedback matrices can be factorization as follows

$$\boldsymbol{A}(z) = \boldsymbol{D}_{\boldsymbol{m}_K}(z)\boldsymbol{U}_K \cdots \boldsymbol{D}_{\boldsymbol{m}_1}(z)\boldsymbol{U}_1\boldsymbol{D}_{\boldsymbol{m}_0}(z), \tag{9}$$

where $\boldsymbol{U}_1, \ldots, \boldsymbol{U}_K$ are scalar $N \times N$ unitary matrices and $\boldsymbol{m}_0, \boldsymbol{m}_1, \ldots, \boldsymbol{m}_K$ are vectors of N delays. In this formulation, the FFM mainly introduces K delay and mixing stages within the main FDN loop. A few examples of FIR FFMs are depicted in Fig. 3. `detPolynomial` provides an efficient FFT-based method for determining the polynomial matrix determinant $\det(\boldsymbol{A}(z))$ [46] for (5).

3.4. Time-Varying Feedback Matrix

Many FDN designs also introduce a time-varying component for enhanced liveliness, improved time and modal domain density as well as feedback stability in reverberation enhancement systems. The most prominent variations are delay line modulation [16], allpass modulation [47] and matrix modulation [48, 39]. The allpass modulation can be represented equally as a matrix modulation [39]. The matrix modulation is given by

$$\boldsymbol{A}(n+1) = \boldsymbol{A}(n)\boldsymbol{R}, \tag{10}$$

where $\boldsymbol{R}$ is an orthogonal matrix close to identity, see `tinyRotationMatrix`. A more robust and computationally efficient version can be implemented by performing the modulation in the eigenvalue domain, see `timeVaryingMatrix`.

Name	Definition	Operations Counts	Notes
Diagonal	$\mathrm{diag}(\boldsymbol{v})$	N	Corresponds to parallel comb filters Orthogonal for $\lvert v_i\rvert \equiv 1$ [38]
Triangular matrix Lower (or Upper)	$L_{ij} = 0$ for $i < j$	$N(N+1)/2$ or N	May correspond to series comb filters Lossless for $\lvert L_{ii}\rvert \equiv 1$ not orthogonal except diagonal [38]
Hadamard [6]	$\boldsymbol{H}_0 = 1$ $\boldsymbol{H}_{k+1} = \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{H}_k & \boldsymbol{H}_k \\ \boldsymbol{H}_k & -\boldsymbol{H}_k \end{bmatrix}$	$N \log N$ Fast Hadamard transform	Orthogonal $\lvert H_{ij}\rvert \equiv \frac{1}{\sqrt{N}}$, i.e., equal magnitude exists only for $N = 1, 2, 4, 8, 12, \ldots$
Anderson [21]	Sparse Block-Circulant matrix $K \times K$ blocks	NK	Orthogonal, $K = 4$ recommended Sparse structures allows larger sizes
Householder [35]	$\boldsymbol{I} - 2\boldsymbol{v}\boldsymbol{v}^\top$ for unit vector $\boldsymbol{v}$	$2N$	Orthogonal Symmetric
Circulant [28]	$\begin{bmatrix} v_1 & v_N & \ldots & v_2 \\ v_2 & v_1 & \ldots & v_3 \\ \vdots & \vdots & \ddots & \vdots \\ v_N & v_{N-1} & \ldots & v_1 \end{bmatrix}$ $\lvert\mathrm{DFT}(\boldsymbol{v})\rvert \equiv 1$	$2N \log N + N$ Fast Convolution	Orthogonal Convolution is across channels not time
Random Orthogonal	Uniform sampling of orthogonal group $\mathrm{O}(N)$	N^2	Orthogonal useful for statistical tests
Tiny Rotation [39]	$\boldsymbol{A} = \boldsymbol{Q}^{-1}\boldsymbol{\Lambda}\boldsymbol{Q}$ $\lvert\angle\Lambda_{ii}\rvert \approx \epsilon$ $\lvert\Lambda_{ii}\rvert = 1$	N^2	Orthogonal close to identity matrix $\boldsymbol{I}$ for small ϵ Allows small matrix modifications
Diagonally Similar Orthogonal [32]	$\boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{D}$ with diagonal $\boldsymbol{D}$ and orthogonal $\boldsymbol{U}$	N^2	lossless, but not orthogonal
Allpasses in FDN [35, 36]	Allpasses in a FDN of size $N/2$	$(N/2)^2 + N$	Equivalent to standard FDN of size N, lossless, but not orthogonal
Moorer-Schroeder [7] Reverberator	Series of $N/2$ parallel comb and $N/2$ series allpasses	$2N$	Moorer and Schroeder, Freeverb
Nested Allpass [8]	Allpasses nested within allpasses	N	SISO allpass characteristic, not lossless

Table 1: Special matrices. The operations count are for a single matrix vector multiplication and are rough estimates as there are many implementation details, e.g., the circulant matrix on a DFT implementation. Allpasses refer to Schroeder's feedforward-feedback comb filters [40, 36]. Some notation includes: $\lvert\cdot\rvert$ denotes the absolute value; $\equiv$ denotes a constant value for all parameters; and $\angle$ denotes the phase of a complex number.

4. CONCLUSIONS

In this paper, we have introduced the FDN toolbox (FDNTB), a unifying MATLAB framework which contains several FDN algorithms, various code examples for demo applications, as well as additional measures that have already been used for evaluating FDN algorithms. By doing so, we gave an overview on recent studies and open topics. We hope that this toolbox not only provides a solid code basis to work in the field of FDN, but also helps to direct attention to shortcomings of classical FDN algorithms, to foster the development of new FDN techniques, and to ease the design of listening experiments. Finally, we would like to encourage developers and researchers in the field of audio processing to use the toolbox to realize their innovative FDN ideas.

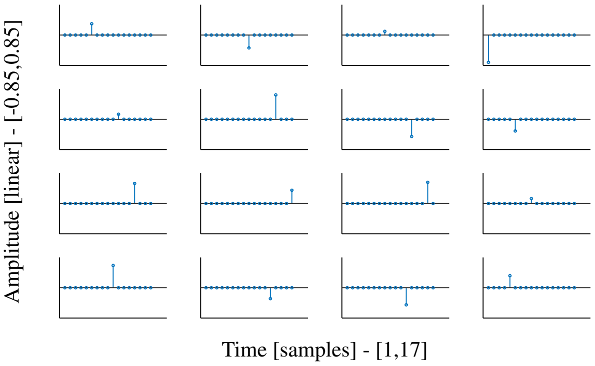
5. ACKNOWLEDGMENT

The author thanks the anonymous reviewers for extensive and helpful comments on the manuscript as well as the corresponding software toolbox.

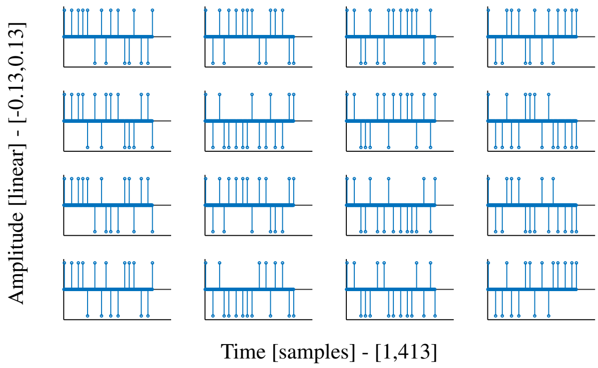
6. REFERENCES

[1] Heinrich Kuttruff, *Room Acoustics, Fifth Edition*, CRC Press. CRC Press, 2009.

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



(a) Delay feedback matrix (DFM) with $K = 1$.



(b) Velvet feedback matrix (VFM) with $K = 2$ and $\delta = \frac{1}{30}$.

Figure 3: Paraunitary filter feedback matrices $\boldsymbol{A}(z)$ with $N = 4$. The subplots depict the filter coefficients of the matrix entries $A_{ij}(z)$ with $1 \leqslant i, j \leqslant N$. The pre- and post-delays $\boldsymbol{m}_0$ and $\boldsymbol{m}_K$ are zero in Fig. 3b and non-zero for the delay feedback matrix in Fig. 3a.

[3] Vesa Välimäki, Julian D Parker, Lauri Savioja, Julius O Smith III, and Jonathan S Abel, "More Than 50 Years of Artificial Reverberation," 2016, Proc. Audio Eng. Soc. Conf., pp. 1 – 12.

[4] Michael A Gerzon, "Synthetic stereo reverberation: Part One," vol. 13, pp. 632 – 635, 1971.

[5] Jean Marc Jot and Antoine Chaigne, "Digital delay networks for designing artificial reverberators," Paris, France, 1991, Proc. Audio Eng. Soc. Conv., pp. 1 – 12.

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

[7] James Anderson Moorer, "About this reverberation business," *Comput. Music J.*, vol. 3, no. 2, pp. 13 – 17, 1979.

[8] William G Gardner, "A real-time multichannel room simulator," *J. Acoust. Soc. Am.*, vol. 92, no. 4, pp. 1 – 23, 1992.

[9] Jon Dattorro, "Effect Design, Part 1: Reverberator and Other Filters," *J. Audio Eng. Soc.*, vol. 45, no. 9, pp. 660 – 684, 1997.

[10] Julius O Smith III, "Physical modeling using digital waveguides," *Comput. Music J.*, vol. 16, no. 4, pp. 74 – 91, 1992.

[11] Francis Stevens, Damian T Murphy, Lauri Savioja, and Vesa Välimäki, "Modeling Sparsely Reflecting Outdoor Acoustic Scenes using the Waveguide Web," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 99, pp. 1 – 1, 2017.

[12] Enzo De Sena, Hüseyin Hacıhabiboglu, Zoran Cvetkovic, and Julius O Smith III, "Efficient synthesis of room acoustics via scattering delay networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 9, pp. 1478 – 1492, 2015.

[13] Benoit Alary, Archontis Politis, Sebastian J Schlecht, and Vesa Välimäki, "Directional feedback delay network," *J. Audio Eng. Soc.*, 2019.

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

[15] "Apparatus and method for generating output signals based on an audio source signal, sound reproduction system and loudspeaker signal," 2015.

[16] David Griesinger, "Improving Room Acoustics Through Time-Variant Synthetic Reverberation," 1991, Proc. Audio Eng. Soc. Conv., pp. 1 – 10.

[17] Sebastian J Schlecht and Emanuël A P Habets, "Reverberation enhancement systems with time-varying mixing matrices," 2015, Proc. Audio Eng. Soc. Conf., pp. 1 – 8.

[18] Sebastian J Schlecht, *Feedback delay networks in artificial reverberation and reverberation enhancement*, Ph.D. thesis, 2017.

[19] Matti Karjalainen and Hanna Järveläinen, "More about this reverberation science: Perceptually good late reverberation," 2001, Proc. Audio Eng. Soc. Conv., pp. 1 – 8.

[20] Natalie Agus, Hans Anderson, Jer-Ming Chen, Simon Lui, and Dorien Herremans, "Perceptual evaluation of measures of spectral variance," *J. Acoust. Soc. Am.*, vol. 143, no. 6, pp. 3300 – 3311, 2018.

[21] Hans Anderson, Kin Wah Edward Lin, Clifford So, and Simon Lui, "Flatter Frequency Response from Feedback Delay Network Reverbs," 2015, vol. 2015 of *Proc. Int. Comput. Music Conf.*, pp. 238 – 241.

[22] Orchisama Das, Elliot K Canfield-Dafilou, and Jonathan S Abel, "On The Behavior of Delay Network Reverberator Modes," *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 50–54, 2019.

[23] Manfred R Schroeder, "Natural Sounding Artificial Reverberation," *J. Audio Eng. Soc.*, vol. 10, no. 3, pp. 219 – 223, 1962.

[24] Sebastian J Schlecht and Emanuël A P Habets, "Feedback delay networks: Echo density and mixing time," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 25, no. 2, pp. 374 – 383, 2017.

[25] Fritz Menzer, "Choosing Optimal Delays for Feedback Delay Networks," 2014, DAGA Fortschritte der Akustik, pp. 564 – 565.

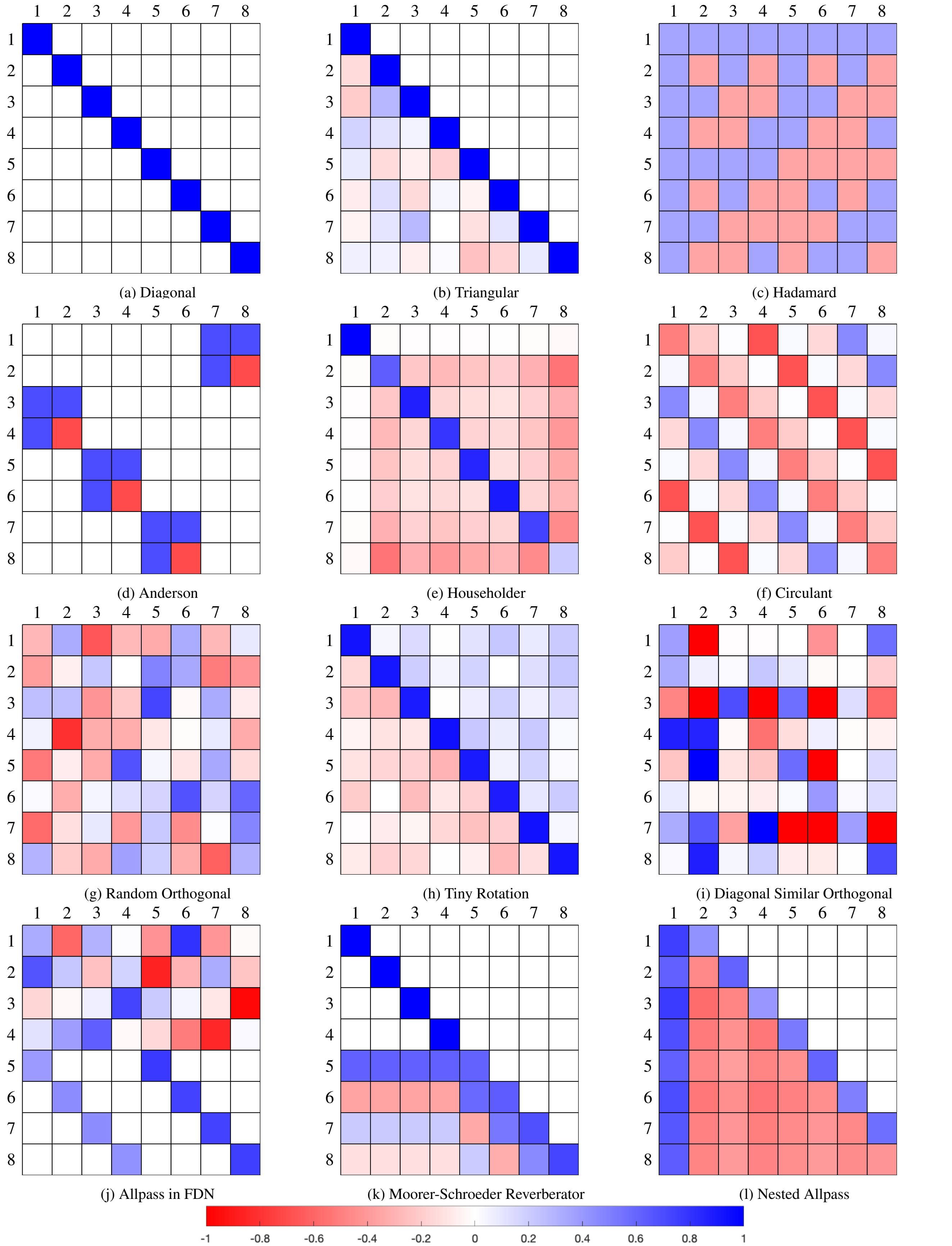


Figure 4: Scalar feedback matrix gallery of size 8×8. The color indicates a linear gain between -1 (red), 0 (white) and 1 (blue). More details are given in Table 1.

[26] Natalie Agus, Hans Anderson, Jer-Ming Chen, Simon Lui, and Dorien Herremans, "Minimally Simple Binaural Room Modeling Using a Single Feedback Delay Network," *J. Audio Eng. Soc.*, vol. 66, no. 10, pp. 791 – 807, 2018.

[27] Sebastian J. Schlecht and Emanuel A. P. Habets, "Modal Decomposition of Feedback Delay Networks," *IEEE Transactions on Signal Processing*, vol. 67, no. 20, 2019.

[28] D. Rocchesso and J.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.

[29] Sebastian J Schlecht and Emanuël A P Habets, "Time-varying feedback matrices in feedback delay networks and their application in artificial reverberation," *J. Acoust. Soc. Am.*, vol. 138, no. 3, pp. 1389 – 1398, 2015.

[30] Balázs Bank, "Converting infinite impulse response filters to parallel form [Tips & Tricks]," *IEEE Signal Process. Mag.*, vol. 35, no. 3, pp. 124 – 130, 2018.

[31] Michael A Gerzon, "Unitary (energy-preserving) multichannel networks with feedback," *Electronics Letters*, vol. 12, no. 11, pp. 278 – 279, 1976.

[32] Sebastian J Schlecht and Emanuël A P Habets, "On lossless feedback delay networks," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1554 – 1564, 2017.

[33] Sebastian J Schlecht and Emanuël A P Habets, "Dense Reverberation with Delay Feedback Matrices," 2019, Proc. IEEE Workshop Applicat. Signal Process. Audio Acoust. (WASPAA), pp. 1 – 5.

[34] Karolina Prawda, Sebastian J Schlecht, and Vesa Välimäki, "Improved Reverberation Time Control for Feedback Delay Networks," 2019, Proc. Int. Conf. Digital Audio Effects (DAFx), pp. 1 – 7.

[35] Riitta Väänänen, Vesa Välimäki, Jyri Huopaniemi, and Matti Karjalainen, "Efficient and Parametric Reverberator for Room Acoustics Modeling," Thessaloniki, Greece, 1997, Proc. Int. Comput. Music Conf., pp. 200 – 203.

[36] Luke Dahl and Jean Marc Jot, "A Reverberator based on Absorbent All-pass Filters," Verona, Italy, 2000, Proc. Int. Conf. Digital Audio Effects (DAFx), pp. 1 – 6.

[37] Hequn Bai, Gael Richard, and Laurent Daudet, "Late reverberation synthesis: From radiance transfer to feedback delay networks," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 12, pp. 2260 – 2271, 2015.

[38] Sebastian J Schlecht and Emanuël A P Habets, "Connections between parallel and serial combinations of comb filters and feedback delay networks," 2012, International Workshop on Acoustic Signal Enhancement (IWAENC), pp. 1 – 4.

[39] Sebastian J Schlecht and Emanuël A P Habets, "Practical considerations of time-varying feedback delay networks," 2015, Proc. Audio Eng. Soc. Conv., pp. 1 – 9.

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

[41] A Berman, B N Parlett, and R J Plemmons, "Diagonal Scaling to an Orthogonal Matrix," *SIAM Journal on Algebraic Discrete Methods*, vol. 2, no. 1, pp. 57–65, 1981.

[42] Sebastian J Schlecht and Emanuël A P Habets, "Sign-Agnostic Matrix Design for Spatial Artificial Reverberation with Feedback Delay Networks," 2018, AES Conference on Spatial Reproduction, pp. 1 – 10.

[43] Nagwa Sherif and Ehab Morsy, "Computing real logarithm of a real matrix," *International Journal of Algebra*, 2008.

[44] Sebastian J. Schlecht and Emanuël A. P. Habets, "Scattering in Feedback Delay Networks," *submitted to IEEE Transactions on Audio, Speech, and Language Processing*, 2020.

[45] Hanna Järveläinen and Matti Karjalainen, "Reverberation Modeling Using Velvet Noise," 2007, Proc. Audio Eng. Soc. Conf., pp. 1 – 9.

[46] Martin Hromčík and M Šebekt, "New algorithm for polynomial matrix determinant based on FFT," 1999, European Control Conference (ECC), pp. 4173 – 4177.

[47] Tapio Lokki and Jarmo Hiipakka, "A time-variant reverberation algorithm for reverberation enhancement systems," Limerick, Ireland, 2001, Proc. Int. Conf. Digital Audio Effects (DAFx), pp. 28 – 32.

[48] Sebastian J Schlecht and Emanuël A P Habets, "The stability of multichannel sound systems with time-varying mixing matrices," *J. Acoust. Soc. Am.*, vol. 140, no. 1, pp. 601 – 609, 2016.