UNIVERSITY OF MIAMI

AUTOMATIC DESIGN OF FEEDBACK DELAY NETWORK REVERB PARAMETERS FOR PERCEPTUAL ROOM IMPULSE RESPONSE MATCHING

By

Jay Sterling Coggin

A THESIS PROJECT

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Master of Science in Music Engineering Technology

Coral Gables, Florida

April 19, 2015

UNIVERSITY OF MIAMI

A Thesis Project submitted in partial fulfillment of
the requirements for the degree of
Master of Science in Music Engineering Technology

AUTOMATIC DESIGN OF FEEDBACK DELAY NETWORK REVERB PARAMETERS FOR PERCEPTUAL ROOM IMPULSE RESPONSE MATCHING

Jay Sterling Coggin

Approved:

Prof. William C. Pirkle Assistant Professor of Music Engineering	Dr. Shannon de l'Etoile Associate Dean of Graduate Studies
Dr. Colby N. Leider Associate Professor of Music Engineering	Dr. Christopher L. Bennett Research Assistant Professor of Music Engineering

COGGIN, JAY S. M.S., Music Engineering Technology
April 19, 2015

Automatic Design of Feedback Delay Network Reverb Parameters for Perceptual Room Impulse Response Matching

Abstract of a Master's Research Project at the University of Miami

Research Project supervised by Prof. William C. Pirkle
Number of Pages in Text: [80]

The parameters controlling feedback delay network (FDN) reverberators have historically been tuned through laborious trial and error, so an automated process is of interest. In this work, a system is designed to perceptually match the reverberation of a monaural target room impulse response (IR) using a 16-channel FDN reverberator structure. The early reflections portion of the target IR is kept to capture the onset character of the room, and a Hadamard matrix is used in the feedback path. The genetic algorithm (GA) is used to iteratively design the delay line lengths as well as the input and output scaling factors. IIR filters are designed at each iteration in order to achieve the target reverberation time curve, $T_{60}(f)$, and a novel approach is used to design an FIR filter to match the tonal character of the target room. The time domain envelope of the target IR is compared to that for the FDN output at each step, and the maximum absolute difference is used as the error value for the GA. Other fitness criteria are also investigated. The algorithm is used to match three IRs covering a range of reverberation times. ABC/HR listening tests performed show that the

target IR and FDN approximation can be difficult to distinguish when mixed at typical wet/dry levels for music production. This perceptual reverb compression can approximate a room sound in about 1-5% of the memory of the target IR, as well as allow for significant CPU cycle savings over convolution reverb in low latency environments. Using the algorithm developed, all parameters for an FDN reverberator can be determined automatically in order to match a target sound so that time consuming hand-tuning is not needed.

ACKNOWLEDGMENTS

Taking on a thesis while working a full-time job 3000 miles from school has not been the easiest way to go about this project. There was a period—especially just after being pulled away to the industry—when I considered not finishing this at all. First, I want to thank everyone along the way who told me that was a dumb idea. Dan Klingler, Adam Kriegel, Baptiste Paquier (and all below)—thank you for holding me to finishing what I started.

To my parents, thank you for all your support—educationally and otherwise—over the years. You both worked so hard so that my brother, sister, and I could pursue educations without financial stress; and you helped keep me on my path by believing in my potential. To Drew and Claire, thanks for being the best brother and sister around—getting to spend time with both of you throughout our Florida educations was a great middle-child privilege. I also want to thank my friend, Gina, for paper edits and for being a great friend throughout this thesis process.

Maybe the best side effect of taking four years to complete a two year degree is that I've gotten to know five years worth of Music Engineering students—I haven't met a bad one. And to my original classmates—Jordan Whitney, Michael Everman, and Matan Ben-Asher—it was especially a pleasure banging our heads against the whiteboard together.

To our department head, my professor, Colby, thank you for your commitment to growing this great program's sphere of influence. There's no place I would have rather gone for this education—becoming a MuE changed the trajectory of my life and I know that type of effect doesn't happen without careful planning and craft. Thank you for your childlike enthusiasm for everything—it's contagious.

To my advisor and professor Will, you've been one of the greatest teachers of my career—and an all-around great guy to boot. Thank you for the original kernel of this thesis idea and your advice throughout the process. You know, if you wouldn't have prepared me as well for the industry as you did, I could have finished this two years ago.

DEDICATION

To my parents, for their constant support.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

1

Introduction

When a sound is created in a room, the pressure waves emitting from the sound source eventually bombard physical obstacles such as walls, ceilings, or chairs. At each of these collisions, part of the incident wave's energy is reflected away from the obstacle, propagating away in a new direction until the process repeats. Imagine a point source on a theater stage emitting a single pulse, creating one expanding spherical wavefront. Different areas on this sphere will encounter obstacles at different times and from different incident angles. After some period of time, the sphere of sound is completely dispersed by the obstacles, and the once easily visualized expanding wavefront disintegrates into a hugely complex web of sound waves traveling in a myriad of directions. As time progresses, this web only grows more dense and complex as the waves continue to reflect around the room.

Now imagine sitting in the theater and witnessing this phenomena. Depending on your position, your ears act as an obstacle for some subset of these waves darting about, which push on the eardrums to create a perceived sound in the brain. This subset of waves encounter the ears at different times and with varying intensities depending on the path that led each from some location on the original spherical wavefront, around the room, and to the ear. What was once a single pulse arrives at your ears along with a large number of *reflections*

over a short interval of time. This sequence of reflections created from the pulse fired is called the *impulse response* of the room. The perceptual affect it produces is called *reverberation*, or *reverb*.

While it may seem that this time-smearing affect would muddle the sound and thus, be undesirable, this is not necessarily the case. A well-designed room's reverberations can add a pleasing sense of spaciousness to musical performances. Furthermore, the human brain is constantly and subconsciously using this spacial *fingerprint* to aid our physical awareness throughout the day. Not surprisingly then, as recording technology has been pushed to provide a higher fidelity and more visceral listening experience since its inception, our love for natural room reverberation has necessitated means by which recording engineers can add this effect artificially.

To artificially "place" a recorded sound in a room, the precisely correct method is to record the desired room's impulse response (IR), and then apply that IR to the recorded sound through a mathematical operation called *convolution*. The resulting audio clip will ideally sound *exactly* as if the sound was created in that room.

However, this method, simply called *convolution reverb*, can be computationally expensive. Because of this, much research over the last half-century has been devoted to developing less computationally expensive algorithms which produce a perceptually similar effect [Valimaki et al., 2012]. This is done by using signal processing methods that have, at best, a vague

relation to the acoustic process by which natural reverberation occurs. These algorithms—aptly named *artificial* (or *perceptual*) *reverb algorithms*—attempt to trick the ear into thinking it is hearing natural reverberation.

One such perceptual reverb topology, the *feedback delay network* (FDN) reverberator, has been a well-studied topology over the last 30 years. While the structure is known to be able to produce a wide range of convincing reverb sounds, the parameters controlling its character behave in complex, coupled ways. Hand-tuning these parameters in order to achieve some specific room sound is impractical at best due to the giant search space of the 12-48 parameters. This time-consuming tuning process is the problem that is addressed by this work.

The goal of this thesis is to automate the tuning process so that no trial-and-error hand-tuning is needed. An algorithm is designed to find the FDN parameters that produce a reverb as perceptually similar to some *target* IR as possible. The iterative FDN design procedure proposed herein uses as many analytic design methods as possible to get close, in a perceptual sense, to the target IR's sound. The FDN parameters which cannot be designed analytically are optimized using a genetic algorithm approach. This search heuristic requires the design of a *fitness function* to indicate how well a set of parameters generated by the genetic algorithm fit our desired outcome (to sound like the target IR), so the quality of this fitness function plays a crucial role in the search for optimal parameters. Several fitness functions are investigated.

This thesis is laid out in 5 chapters. Chapter 2 will discuss the relevant

background knowledge and prior research in the field. Chapter 3 will explain the proposed optimization system, Chapter 4 will evaluate its effectiveness, and Chapter 5 will discuss future work and conclusions.

2

Background

To understand the approach to the present design algorithm, some theoretical background and prior research should be considered. We begin with room IR structure and convolution reverb, then follow the development of perceptual reverb theory through the years. The feedback delay network architecture and theory are of particular interest to this chapter. Some common objective features of IRs will also be presented as these can be used as part of a fitness function steering the iterative optimization procedure.

2.1 Convolution Reverb

Perhaps most fundamental to reverberation theory is the convolution operation and its application to acoustics. Convolution is a mathematical operator, denoted with a "$*$", that operates on two functions. The convolution of two continuous time-varying signals, $x(t)$ and $h(t)$, is defined as

$$y(t) = h(t) * x(t) \triangleq \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau \tag{1}$$

Conceptually, the operation flips one of the signals in time and slides it across the other. The *convolved* output, $y(t)$, is the area under both functions along the way. The process is depicted in Figure 1.

In the context of reverberation, the IR of a room, $h(t)$, "slides" across the sound emitted from a source in the room, $x(t)$, as time progresses. The output,

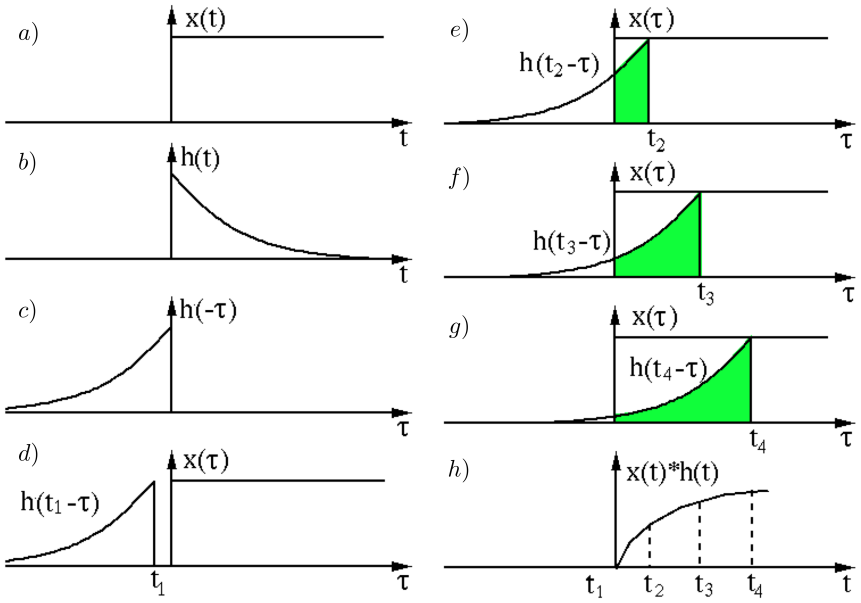

Figure 1: Convolution of (a) $x(t)$ with (b) $h(t)$ over the interval $t = \{t_1, t_4\}$. (c) $h(t)$ is reversed in time to form $h(-\tau)$. (d) Eq.1's lower integration bound is a negative time, t_1 in this case. (e) — (f) $h(-\tau)$ is slid across $x(\tau)$ and shown at three instants: t_2, t_3, and t_4. The total area under both functions is the convolution of the two signals at instant t. (h) The resulting convolution, $y(t) = x(t) * h(t)$ [Wang, 2013]

$y(t)$, is the sound perceived in the room. The discretized version of Eq. 1 is given by

$$y[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \tag{2}$$

Every output sample is simply a combination of current and past $x[n]$ values weighted by the sliding IR. We can also see now that for a length N IR, we will have N multiplies and N additions per output sample, making convolution a $O(N^2)$ operation. For a typical large acoustic hall IR that is five seconds in length at a sampling rate of 44.1 kHz, almost 10 billion multiplies and additions are required per second of audio. However, fast convolution methods exist that can lower that number to a degree [Gardner, 1994] [Garcia, 2002].

Regardless of the computational cost, another downside of convolution

reverb is the high memory cost associated with storing each IR. If each sample is stored as a four byte floating point value, the total memory cost, in bytes, for an IR of length T seconds at sampling rate f_s is

$$M = 4Tf_s \tag{3}$$

For the typical acoustic hall IR with $T = 5$ and $f_s = 44.1$ kHz, this is a cost of almost 900 kB for a single IR. In an application such as a reverb audio effect plugin where a large number of reverberation sounds should be available to the user, this cost can be too great in memory-constrained environments such as mobile devices.

2.2 Room Impulse Response Structure

We now turn to the commonly discussed features of a room IR to motivate their modeling and approximation.

2.2.1 Early Reflections & Late Reverberations

An IR breaks down into roughly 3 sections: *direct*, *early reflections*, and *late reverberation*. Figure 2 shows an idealized version of an IR plotted in dB magnitude over time.

If an impulse is emitted in a room at $t = 0$, the first event in the room IR happens T_0 seconds later when the impulse's wavefront—the *direct* sound—passes by the observer. Shortly after the direct sound, the early reflections arrive. These pulses of sound are the wavefronts passing over the observer from the first few reflections of the direct sound in the room. In this

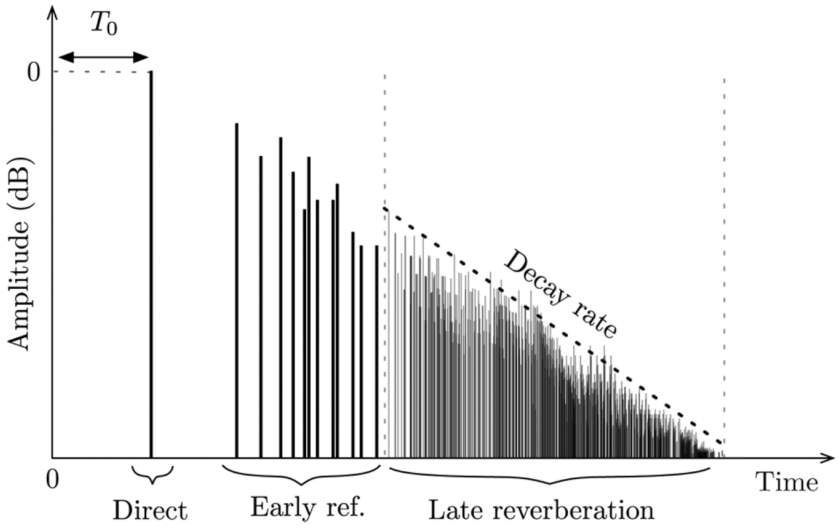

Figure 2: Idealized impulse response [Valimaki et al., 2012]

section of the IR, the pulses are scarce enough that they are perceived as distinct. The number of reflections in the room quickly grows to the point where individual pulses cannot be distinguished by the ear. This period of dense echoes is the late reverberation and is generally the longest audible portion. As the waves reflect about the room, the sound absorption of air and the absorption of the obstacles in the room eventually dissipate the acoustic energy to a point where we can no longer perceive the sound, which is where we would say the impulse response has reached its end.

The transition from the early reflections period to the late reverberation period is not as obvious in practice as Figure 2 indicates. Some efforts have been made to quantify this transition [Stewart and Sandler, 2007] [Primavera et al., 2010]. However, the first 80 ms of an IR is commonly referred to as the early reflections period [Moorer, 1979]. Despite its relatively short

length compared to late reverberation, the acoustic information in the early reflections plays an important role in our ear's perception of the space as well as the intelligibility of speech [Haas, 1972].

2.2.2 T_{60}, EDC, and EDR

Perhaps the simplest and most common quantitative measure of an IR is its T_{60}, sometimes simply called the *reverberation time*. This measurement encapsulates how long a reverberation lasts by measuring the amount of time taken for the sound pressure level in a room to decay by 60 dB after an excitation ends. The most common way to measure this from a recorded IR, introduced by Manfred Schroeder, is by use of *energy decay curves* (EDC) [Schroeder, 1965]. The energy decay curve of an IR shows how much energy remains in the signal at sample n and is given for continuous signals as

$$EDC(t) = \int_t^\infty h^2(t)dt \tag{4}$$

and for discrete signals as

$$EDC[n] = \sum_{m=n}^{N} h^2[n] \tag{5}$$

An EDC curve for a medium hall plotted in dB and normalized so that $EDC[0] = 0$ dB is shown in Figure 3.

The T_{60} for the hall can be gathered directly from Figure 3 by finding where the function passes -60 dB—about 1.2 seconds. Because the EDC must monotonically decrease in a well-defined way, it is favored over envelope-following methods for T_{60} calculations.

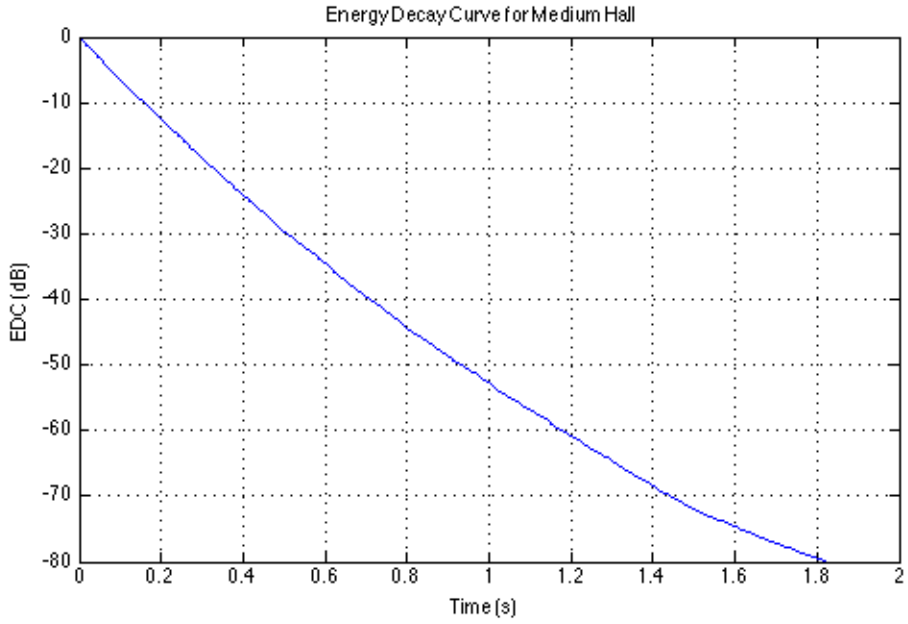

Figure 3: EDC for a medium hall

Jot took the EDC concept a step further and introduced *energy decay relief* (EDR) plots [Jot, 1992]. These 3D plots generalize EDC curves for multiple frequency bands, generally computed using the short time Fourier transform (STFT) magnitudes

$$EDR(t_n, f_k) \triangleq \sum_{m=n}^{M} |H(m,k)|^2 \tag{6}$$

where $H(m,k)$ is the kth frequency bin of the mth frame of the STFT. M represents the total number of STFT frames. A sample EDR plot for a large hall is shown in Figure 4. Figure 4 reveals some typical features of room reverberation. The gentle slope of the mesh in the lower frequency bands compared to in the higher bands indicates that the upper band energy dissipates quicker than in the lower bands. This is is due to the frequency-dependent acoustic absorption of materials in the room. Generally, higher frequency energy

is absorbed by a material more than low frequency energy, meaning more low frequency energy is reflected by the material and continues traveling through the room.

Just as EDR plots generalize energy decay curves for multiple frequencies, we can use EDR plots to generalize the T_{60} calculation for multiple frequencies to compute a $T_{60}(f)$ curve.

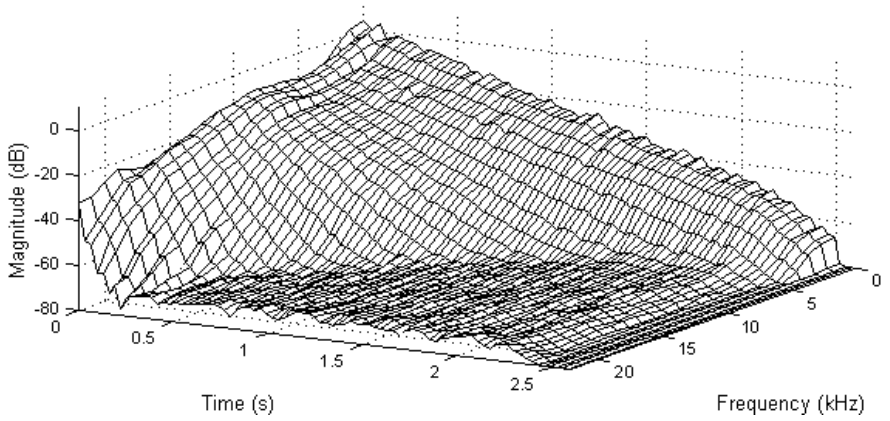

Figure 4: EDR plot for a large hall [Frenette, 2000]

To better match the critical bands of human hearing, bins of the STFT are sometimes warped to the Bark or Mel frequency scale prior to the summation. Time domain techniques such as third-octave or gammatone filter-banks can also be used to performa a similar perceptual weighting. All of these methods form bands of audio with increasing bandwidth as frequency increases.

2.3 Perceptual Reverb Theory

Manfred Schroeder pioneered the field of perceptual reverberation in the early 1960s. He saw the need for inexpensive algorithms that could emulate natural reverberation, and he published a set of postulates regarding the criteria

that these algorithms must satisfy [Schroeder, 1962]. In this initial work, he also explained two simple signal processing blocks that could be used to satisfy these criteria and then proposed the first perceptual reverberator architecture utilizing these blocks, famously known as Schroeder's Reverberator. The following summarizes his and other's work in perceptual reverb theory.

2.3.1 Echo and Modal Density

Schroeder postulated that for an artificial reverberation to be natural-sounding, it must have a large number of echoes and a flat frequency response. To speak about the number of echoes, Schroeder used the concept of *echo density*, which is the number of echoes per second experienced by an observer. In a room, the echo density due to an acoustic impulse is given statistically by

$$D_E(t) = \frac{4\pi c^3}{V} t^2 \tag{7}$$

where V is the volume of the room in m^3, c is the speed of sound in air in m/s, and t is time. The impulse is emitted at $t = 0$. Eq. 7 quantifies the increase in echo density with time that was noted in the study of the room impulse response. Schroeder stated that the echo density must be at least 1,000 per second. Later, Griesinger increased this to 10,000, especially for percussive instruments [Griesinger, 1989]. If the density is lower, distinct echoes can be differentiated by the ear leading to an unnatural fluttering sound [Schroeder, 1962].

Schroeder noted that his second criteria—that a reverberator have a flat frequency response—is not actually true of a room, which has various standing

wave modes between walls. However, in a rectangular room, the modal density, $D_M(f)$, (in resonances per Hertz) is given by

$$D_M(f) = \frac{4\pi V}{c^3} f^2 \tag{8}$$

where V is the volume of the room in m^3, c is the speed of sound in air in m/s, and f is the frequency in hertz. The modal density increases so quickly with the squared frequency term that it *sounds* flat to our ears at a relatively low frequency. As we'll see, we don't have the luxury of such a "resonance multiplier" and can only space resonances linearly between DC and Nyquist. Given this, Schroeder estimated that for a colorless frequency response, the modal density must be at least 0.15 resonances/Hz for a T_{60} of 1 second. Schroeder also showed that the average modal density in a room is proportional to the T_{60}, meaning that the desired modal density of our system, D_M, should be:

$$D_M \geq 0.15 T_{60} \tag{9}$$

2.3.2 Comb Filter

In designing an artificial reverberator structure that would satisfy his postulates, Schroeder proposed the use of a comb filter, shown in Figure 5. The comb filter is nothing more than a delay line with sample delay M and a feedback path with gain g. It creates a series of delayed echoes of the input signal, which is exactly what is needed for the artificial reverberator. The difference equation for the structure shown in Figure 5 is given by

$$y[n] = x[n - M] + gy[n - M] \tag{10}$$

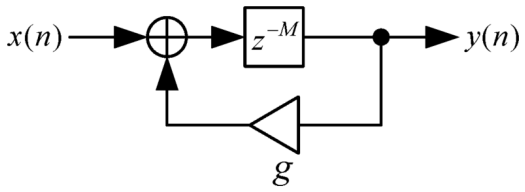

Figure 5: Comb filter structure [Valimaki et al., 2012]

which yields the transfer function

$$H(z) = \frac{z^{-M}}{1 - gz^{-M}} \tag{11}$$

The delayed feedback creates a linearly spaced set of poles around the unit circle (see Figure 6) that increase in magnitude as g is increased, creating sharper and sharper peaks in the frequency response, giving it its common name. These shapes are shown in Figure 7 for various values of g.

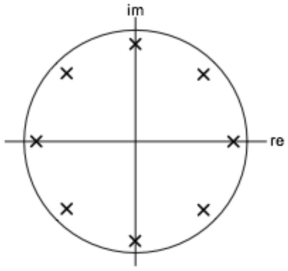

Figure 6: Comb filter poles in the z-plane [Valimaki et al., 2012]

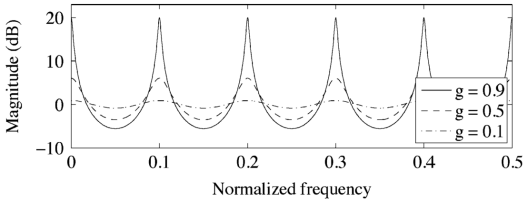

Figure 7: Comb filter magnitude response for various values of g [Valimaki et al., 2012]

So, the structure creates a series of echoes in the time domain and resonances in the frequency domain, but the resonances are linearly spaced

around the unit circle rather than piling up in the higher frequencies as Eq. 8 indicates for a room.. The density is instead given by:

$$D_m = \frac{M}{f_s} \tag{12}$$

where M is the comb filter delay and f_s is the sampling rate. This means that a comb filter with an M-sample delay will produce $M/2$ resonances between DC and the Nyquist frequency, $f_s/2$. Thus we see the fundamental trade-off in selecting the delay lengths when using the comb filter in a perceptual reverberator: a long delay length M produces high modal density, D_M, but poor echo density, D_E, and short delay lengths produce the opposite.

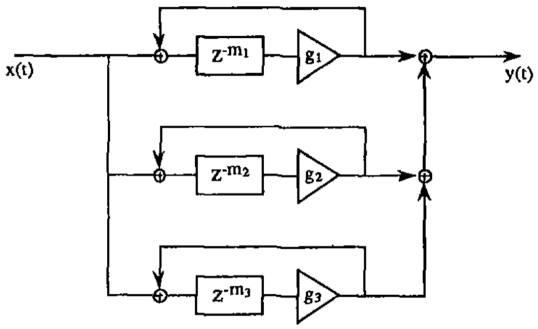

Figure 8: Bank of $N = 3$ parallel comb filters [Jot and Chaigne, 1991]

In order to get the sufficiently high modal density given by Eq. 9, we can place several comb filters with mutually prime delay lengths in parallel, as shown in Figure 8. For a bank of N parallel comb filters, this will produce $m_T/2$ resonances where

$$m_T = \sum_{i=0}^{N} m_i \tag{13}$$

and m_i is the delay length of the ith comb filter. If such a parallel bank of comb filters is used, the g_i values should be selected such that all resonances decay at

the same rate, which is to say that all of the system poles should have the same magnitude. Jot and Chaigne show that the g_i's should be exponentially proportional to the m_i's as

$$g_i = \gamma^{m_i} \tag{14}$$

where $0 < \gamma < 1$ is a tunable decay factor representing the system pole radii [Jot and Chaigne, 1991] . This can be written in decibels as:

$$\Gamma = \frac{G_i}{m_i} \tag{15}$$

where $\Gamma = 20\log(\gamma)$ and $G_p = 20\log(g_i)$. If T is the sampling period and $\tau_p = m_i T$ is the delay length in seconds, then

$$T_{60} = -60\frac{\tau_p}{G_p} \tag{16}$$

So given a desired delay length, m_i, we can compute the proper g_i for each comb filter so that as successive passes are made through the feedback path, the sound decays at the proper rate.

Satisfying Eq. 14 for all N banks guarantees that the spectral shape will not change during the decay, but does not address the relative strengths of the resonances between the comb filters, which varies inversely proportional to the delay length, m_i. If the strengths are different enough, the perceived modal density becomes less than the true modal density as certain comb filters' resonances are masked. The overall output of a comb filter is scaled by m_i^{-1}, so the comb filter outputs in Figure 8 should be multiplied by m_i prior to the summation [Jot and Chaigne, 1991].

Jot generalized his idea of control over T_{60} to be frequency dependent, which could be realized with filters on the outputs of the delay lines. He showed that given a target $T_{60}(\omega)$ curve, where $\omega = 2\pi f$, and a delay length, m_i, we can use Eq. 17 to find the target magnitude response $|h_i(\omega)|$ for each comb filter, then use an appropriate filter design method to create filters to match this curve.

$$|h_i(\omega)| = 10^{\frac{-3Tm_i}{T_{60}(\omega)}} \tag{17}$$

Jot relaxed the requirement that all the reverberator's poles be of the same magnitude and said, more generally, that the pole locus should be *continuous* so that over a small frequency range, the differences between resonance strengths are negligible. If we were designing to match an actual room's $T_{60}(\omega)$, this is a trivial. An example pole locus and corresponding EDR plot are shown in Figure 9. The poles near the unit circle at low frequencies create a slower decay, and thus, longer reverberation time, whereas the poles closer to the origin near the Nyquist frequency create a faster decay.

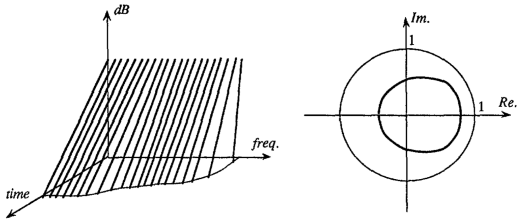

Figure 9: Example pole locus and corresponding EDR plot for a typical room [Jot and Chaigne, 1991]

When we use Eq. 17, we are designing the proper *decay rate* as a function of frequency, matching the slopes of the lines in Figure 9. These filters, of course,

color the sound as a side effect. To counter this effect, Jot showed that we should apply a tone correction filter $h_{tone}(\omega)$ at the system output which satisfies

$$|h_{tone}(\omega)|^2 \propto \frac{1}{T_{60}(\omega)} \tag{18}$$

2.3.3 Allpass Filter

Schroeder also introduced the allpass filter as a reverberator unit. This unit, as its name suggests, passes all frequencies without change in magnitude. What it does do is smear the phase and give us a series of decaying echoes like the comb filter, but with a faster decay. The structure includes the feedback comb filter from before, but with a feedforward comb filter with negated gain as well. This is shown in Figure 10. If we call the node entering the delay network

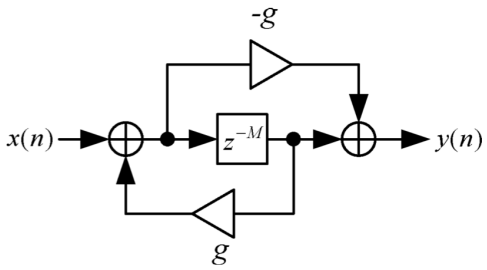

Figure 10: Allpass filter structure [Valimaki et al., 2012]

$w[n]$, the difference equation is

$$y[n] = -gx[n] + x[n-M] - g^2 w[n-M] + gw[n-2M] \tag{19}$$

giving a transfer function

$$H(z) = \frac{z^{-M} - g}{1 - gz^{-M}} \tag{20}$$

which has unit magnitude for all ω.

Schroeder noted that the echo density could be increased by connecting several in series—effectively multiplying the echo density while maintaining a flat frequency response.

2.3.4 Schroeder's Reverberator

Schroeder used his comb filter and allpass filter building blocks to create the first artificial reverb unit. He placed four comb filters in parallel feeding into two allpass filters in series, as shown in Figure 11. The comb filters produce the decaying echoes and high modal density, and the two allpass filters multiply the echo density without additional coloration. Schroeder noted that for best results, the feedback gains for the comb filters should not exceed 0.85, and the delay lengths should be mutually prime and span a range of about 1:1.5 (Schroeder covered the range of 30 to 45 msec) [Schroeder, 1962].

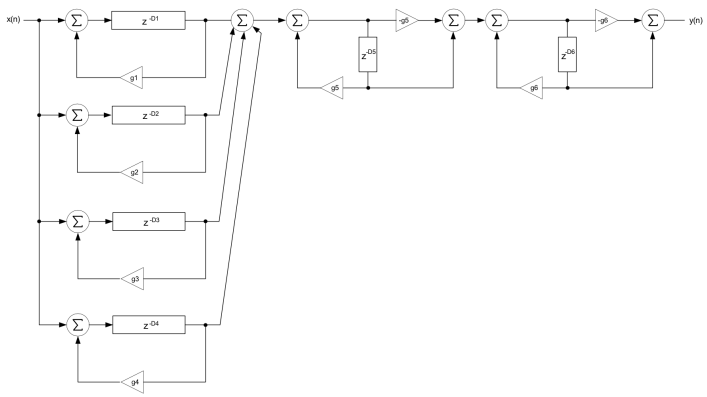

Figure 11: Schroeder's reverb structure [Pirkle, 2012]

2.4 Feedback Delay Networks

Over the next 2 decades, a few artificial reverb architectures were proposed involving various combinations of these comb and allpass filters

[Moorer, 1979]. We skip ahead now to the introduction of what is now called the *feedback delay network* (FDN) reverb architecture, the reverb structure used in the remainder of this thesis.

2.4.1 Gerzon's Work

In 1971, Michael Gerzon suggested a reverberator unit utilizing a feedback matrix connecting a bank of parallel comb filters [Gerzon, 1971] [Gerzon, 1972]. Gerzon noted that by allowing cross-coupling of feedback between a bank of comb filters, the echo density could build up much more quickly than if the comb filters' feedback paths were independent. This would help alleviate the known primary issue with Schroeder's Reverberator—insufficient echo density. He noted that one method to ensure the energy decay of the system would follow the proper exponential loss of energy in a room was to use an orthonormal feedback matrix and adjust only the overall loop gain of the system to be a value less than unity in order to control the reverberation decay.

Requiring an orthonormal feedback matrix $\mathbf{A} \in \Re^{NxN}$ meant that the overall energy of an input vector $\mathbf{u}$ was preserved when left-multiplied by $\mathbf{A}$. Since the columns of $\mathbf{A} = \{\mathbf{a}_1 \cdots \mathbf{a}_N\}$ form an orthonormal basis, the multiplication by $\mathbf{A}$ can be thought of as a norm-preserving rotation in $\Re^N$ such that $|\mathbf{v}| = |\mathbf{A}\mathbf{u}| = |\mathbf{u}|$.

A few years later, Gerzon showed that Schroeder's single-input, single-output allpass filter was generalized to an M input, M output allpass network by using a *unitary* feedback matrix [Gerzon, 1976]. A unitary matrix is

any complex matrix that satisfies

$$\mathbf{U}^{\dagger}\mathbf{U} = \mathbf{I} \tag{21}$$

where $\mathbf{U}^{\dagger}$ is the Hermitian conjugate of $\mathbf{U}$ and $\mathbf{I}$ is the identity matrix. If $\mathbf{U}$ is a real-valued matrix, the Hermitian conjugate is simply the transpose, $\mathbf{U}^{T}$, and the unitary matrix criteria from Eq. 21 reduces to the requirement for an orthonormal matrix

$$\mathbf{U}^{T}\mathbf{U} = \mathbf{I} \tag{22}$$

The term "unitary" is used almost exclusively in the literature when describing FDN feedback matrices, but the matrices are always real-valued, so orthonormal is really a more appropriate term.

2.4.2 Stautner & Puckette

In 1982, Stautner & Puckette—seemingly without knowledge of Gerzon's work—proposed a multi-channel reverberator to be used with a set of 4 speakers arranged around the listener, as shown in Figure 12
[Stautner and Puckette, 1982]. The authors noted that the network could be realized with a 4-channel version of the structure in Figure 13 using an orthonormal feedback matrix

$$\mathbf{G} = g \begin{bmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \end{bmatrix} \tag{23}$$

where

$$|g| < \frac{1}{\sqrt{2}} \tag{24}$$

guarantees that the system is stable.

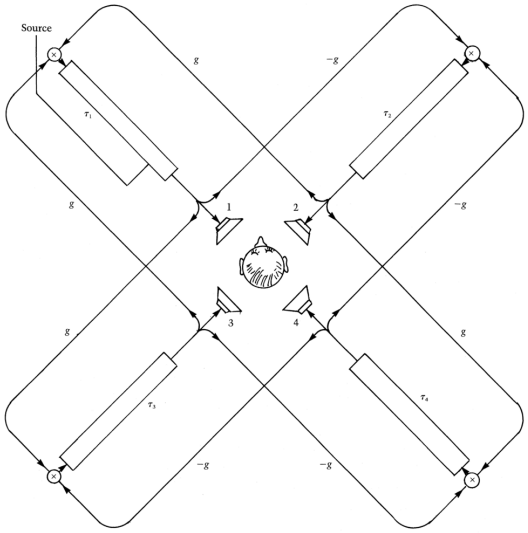

Figure 12: Quadraphonic inspiration for Stautner's delay network [Stautner and Puckette, 1982]

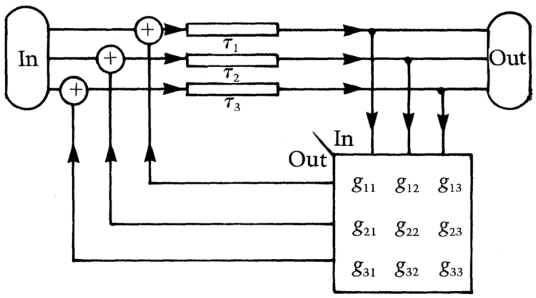

Figure 13: Stautner's delay network [Stautner and Puckette, 1982]

Stautner suggested inserting the sound source at various points in the structure from Figure 13 during the early reflection period to approximate the effect. They also suggested the use of lowpass or bandpass filters in the feedback path to simulate the frequency-dependent absorption qualities of room materials.

2.4.3 Jot's FDN

Jot and Chaigne introduced the term "feedback delay network" and the FDN reverberator structure still in use today [Jot and Chaigne, 1991]. The structure is shown in Figure 14 for a channel count of $N = 3$. The input, $x[n]$, is passed through a parallel bank of input scaling gains, $\mathbf{b} = \{b_1, \ldots, b_N\}$. The bank of delay lines of length $\mathbf{m} = \{m_1, \ldots, m_N\}$ have outputs $\mathbf{q}[n] = \{q_1[n], \ldots, q_N[n]\}$, which are multiplied by the feedback matrix, $\mathbf{A}$, and summed with $x\mathbf{b}$ before being sent into the delay lines. $\mathbf{q}[n]$ is also scaled by a set of output gains, $\mathbf{c} = \{c_1, \ldots, c_N\}$, which is then summed with the input, $x[n]$, scaled by gain d to form the output, $y[n]$.

The authors presented the FDN as a generalization to the comb filter bank that was discussed earlier (see Figure 8), allowing the feedback signals to cross-couple, and thus, the echo density to build up quicker. They also showed that allpass filters, as well as comb, could be implemented using the structure.

We can write the output of a general N-channel FDN in the z-domain as

$$y(z) = \mathbf{c}^T\mathbf{q}(z) + dx(z) \tag{25}$$

with $\mathbf{q}(z)$ given by

$$\mathbf{q}(z) = \mathbf{D}(z)[\mathbf{A}\mathbf{q}(z) + \mathbf{b}x(z)] \tag{26}$$

and $\mathbf{D}(z) = \text{diag}(z^{-m_1} \ldots z^{-m_N})$. Using Eq. 26 and Eq. 25 to eliminate $\mathbf{q}(z)$, the transfer function of the system can be derived as

$$H(z) = \frac{y(z)}{x(z)} = \mathbf{c}^T[\mathbf{D}(z^{-1}) - \mathbf{A}]^{-1}\mathbf{b} + d \tag{27}$$

The system poles are given by the solutions to the characteristic equation

$$\det[\mathbf{A} - \mathbf{D}(z^{-1})] = 0 \tag{28}$$

but this is not easily solved analytically.

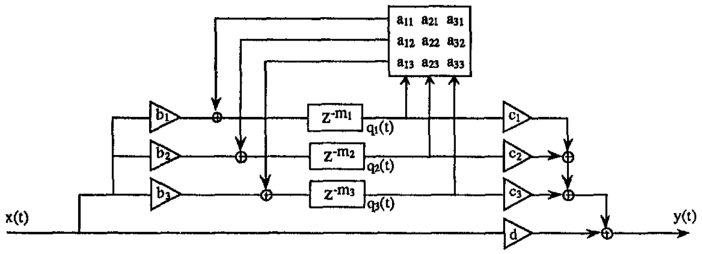

Figure 14: Jot's general feedback delay network structure [Jot and Chaigne, 1991]

Fortunately, we do not need to compute the system poles analytically. Eq. 14, 17, and 18 from the analysis of comb filter banks can be used to design frequency-dependent absorbent filters, $h_1(z) \ldots h_N(z)$, to be placed on the delay line outputs to achieve a target $T_{60}(\omega)$ curve. This design technique was used successfully by Jot to design an FDN reverberator to match a recorded room IR's reverberation curve [Jot, 1992].

The modified FDN structure to support this is shown in Figure 15. This figure also includes the tonal correction filter, $h_{tone}(z)$, from Eq. 18 (Jot called the filter $t(z)$).

2.4.4 Feedback Matrix Selection

Gerzon's idea to use unitary feedback matrices has become standard practice in FDN design. The idea is to design a *lossless prototype* FDN whose impulse response never decays due to the energy preservation guaranteed by the unitary matrix (all poles on the unit circle). In general, we want the output of

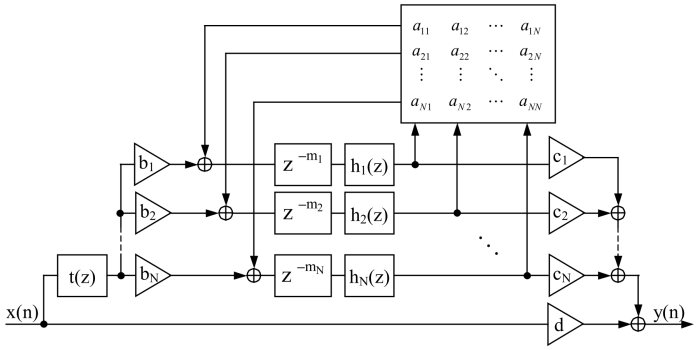

Figure 15: Modified FDN for frequency-dependent absorption and tone correction [Jot and Chaigne, 1991]

such a system to sound like white noise, meaning that it has a) sufficient echo density so that no fluttering is heard and b) sufficient modal density so that individual resonances aren't heard. If we can find unitary matrices that produce such a sound, we can use Jot's absorption filter design techniques to produce the appropriate decay rates and frequency response. We already introduced Stautner & Puckette's physically inspired unitary matrix in Eq. 23, but there are an infinite number of unitary matrices from which to choose. One commonly used unitary matrix family is the *Hadamard Matrix*. The second-order *Hadamard matrix* is defined as

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \tag{29}$$

with higher order versions created by recursive nesting. For example, a fourth-order Hadamard matrix is given by

$$\mathbf{H}_4 = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{H}_2 & \mathbf{H}_2 \\ -\mathbf{H}_2 & \mathbf{H}_2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{30}$$

The order of a Hadamard matrix must therefore be a power of 2. One interesting property of the Hadamard family of matrices is that an order-N Hadamard matrix has the maximum determinant of any $N \times N$ matrix whose entries are bounded by $|a_{ij}| \leq 1$. This produces maximal off-diagonal interactions, which can be thought of as a high level of acoustic scattering in our case [Smith, 2006]. Some computational improvements can also be gained when multiplying by a Hadamard matrix since all entries are either one (no multiplication needed) or negative one (simple negation).

Rocchesso and Smith suggested the use of unitary circulant feedback matrices having the form

$$\mathbf{A} = \begin{bmatrix} a(0) & a(1) & \cdots & a(N-1) \\ a(N-1) & a(0) & \cdots & a(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ a(1) & a(2) & \cdots & a(0) \end{bmatrix} \tag{31}$$

which is a matrix defined by the values in the first row [Rocchesso and Smith, 1997]. Each successive row is a circularly right-shifted version of the row above. The authors show that the discrete Fourier transform (DFT) matrix $\mathbf{T}$ can be used to diagonalize the circulant matrix $\mathbf{A}$ as:

$$\mathbf{D} = \mathbf{TAT}^{-1}, \tag{32}$$

which implies that we can design $\mathbf{A}$'s eigenvalues (resonances) directly, then find the corresponding $\mathbf{A}$ via an inverse DFT.

2.4.5 Delay Length and Gain Selection

Schroeder suggested using delay lengths for his reverberator that covered a range no larger than 1:1.5. In particular, he covered the range of 30 to 45 ms [Schroeder, 1962]. He also stated that the delay lengths should be mutually prime so that resonances from comb filters do not lie on top of each other. This maximizes the number of output samples that must be produced before the impulse response repeats. Jot used a range of 1:2.5 in choosing delay lengths for his absorbent FDN reverb [Jot, 1992]. 12 delay lines were used with a reported total delay length of about 1 second, meaning an average delay length of 83 ms.

Because the echo density builds up much quicker with the FDN as compared to Schroeder's Reverberator—especially with $N = 12$—the average delay length can be increased to help modal density without the echo density becoming too sparse. Rocchesso & Smith suggested a method to design the delay lengths, $\mathbf{m}$, as well as the gain vectors, $\mathbf{b}$ and $\mathbf{c}$, by starting with the geometric specifications of a room [Rocchesso and Smith, 1997]. Otherwise, relatively little is mentioned in the literature about the design of $\mathbf{m}$, $\mathbf{b}$, and $\mathbf{c}$.

2.5 Genetic Algorithm

Due to the complex and inter-coupled influence that $\mathbf{m}$, $\mathbf{b}$, and $\mathbf{c}$, and $\mathbf{A}$ have on the FDN reverb's aural character, its not so surprising that analytic methods hardly exist to map desired sonic aspects of the reverberator to appropriate parameters of the structure. The goal of this thesis though is exactly that: determine the optimal values of all FDN parameters to perceptually match

the sound of a target room IR. In such a situation, where we lack a closed-form mapping between a goal (perceptual indistinguishability of the synthetic and target IR) and parameters that influence how well we meet that goal (the FDN parameters), iterative search techniques often offer the best way forward.

One such search technique that has proven itself effective in an array of engineering optimization problems is a *genetic algorithm*. A genetic algorithm (GA) borrows ideas from evolutionary biology and mimics natural selection in its pursuit of an optimal solution. The algorithm itself has no knowledge of the problem it is optimizing. A common genetic algorithm user interface requires the user only to define a number of scalar parameters that the GA should optimize, a possible range for each parameter, and any other numeric constraints (such as forcing a parameter be an integer). Working within these constraints, the GA iteratively generates a set of these parameters, called an *individual*, and then requires that the user provide a *fitness value* for that individual. The fitness value is a single scalar value that describes how well that individual meets the optimization goal. The fitness value is computed using a *fitness function*, which is always problem dependent.

A flowchart of a genetic algorithm is shown in Figure 16. The algorithm first generates a random *population* (sometimes called a *generation*), which is a collection of individuals. The user then provides a fitness value for each of these individuals. If any of the individuals meet a user-specified fitness limit, the algorithm terminates, and the optimal individual is returned to the user as the

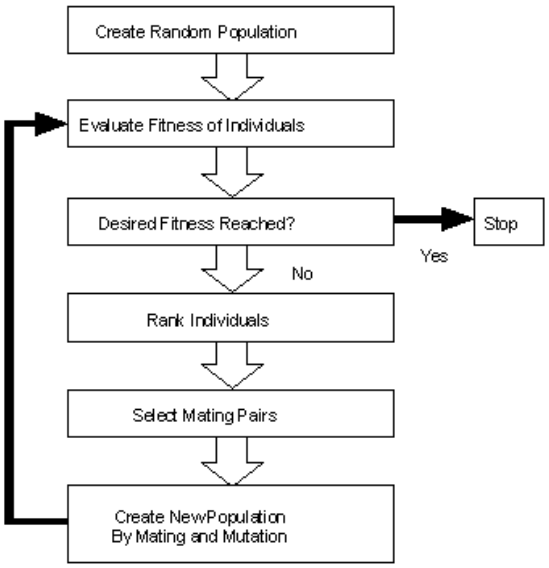

Figure 16: Iterative steps in a genetic algorithm [Noren and Ross, 2001]

solution. Otherwise, the algorithm selects a set individuals with the greatest fitness to "breed" a new generation of individuals. This breeding process is implementation-dependent, but involves combining parameters from fit individuals in various combinations. Random mutation is also modeled by introducing individuals whose parameters are at least partially random (not inherited from a parent). The fitness evaluation and selection process then repeats with this new population until the fitness limit is reached. Often, a time-based exit criteria is also implemented to prevent endless processing when a solution cannot be found.

2.6 Quantifying Reverbs

Several methods have been proposed over the years to describe IRs quantitatively. Since one of the cruxes of this work will be to derive a fitness function which produces a fitness value signifying the perceptual similarity of two

IRs, we should be familiar with prior art regarding quantitative and perceptual measurements of IRs. We've already seen how T_{60}, EDC, and EDR describe—in varying levels of detail—how the reverb energy decays in time. We now turn to measures which aim to capture other sonic aspects of room IRs.

Reverb measurements in the literature usually fall into one of two categories—those aimed at *quantifying* acoustic spaces and those aimed at *qualifying* artificial reverberations. The measures aimed at qualifying artificial reverbs are concerned more with verifying that the artificial IR has a believable sound (sufficient echo and modal density, colorless, etc.) whereas those for physical rooms are more geared to describe the behavior of acoustic phenomena (reverb time, clarity index, etc.). In developing a fitness function to match an artificial IR to a natural IR, both categories should be considered.

2.6.1 Echo Density

Griesinger outlined a method for measuring the echo density by sliding a 20 ms rectangular window across the signal, counting the number signal peaks within 20 dB of the largest peak [Griesinger, 1989]. More recently, Abel and Huang introduced a method based on the idea that the late reverberant field in a room has a Gaussian distribution, so echoes are counted as samples lying outside a standard deviation of the expected value [Abel and Huang, 2006]. Such a measure should be used to qualify an artificial reverberation, especially since insufficient echo density is known to be one of the pitfalls of artificial reverberators.

2.6.2 ISO 3382

An obvious source for objective measurements is the International Organization for Standardization's set of standard measurable quantities regarding room acoustics [ISO, 1997]. A few of the measures are explained here.

T_{10}, called the *Early Decay Time* (EDT), is, one of these measures. This measure of the time taken for an excitation in a room to decay by its first 10 dB is considered by some to correlate more with the perceived reverberance of a room than the more common T_{60} [Barron, 2005]. This is especially the case with music performance where the late reverberation tends to be masked by the performance itself, not allowing us to hear the details of the late decay. T_{20} and T_{30} are also on the standard, but both of these begin measuring after the signal has decayed 5 dB.

Clarity indices, also known as *early-to-late sound indices*, are another set of ISO 3382 measures. C_{50} is defined as

$$C_{50} \triangleq 10 \log \frac{\int_0^{50ms} h^2(t)dt}{\int_{50ms}^{\infty} h^2(t)dt} \tag{33}$$

with C_{80} similar. This gives the ratio of the energy in the early reflection period to the energy in the late reverberation. Clarity tends to be negatively correlated to EDT [Barron, 1995], and target C_{50} values for a good concert hall are between -1 and -3 dB. A common criticism of this measure is that it makes an immediate, fixed transition from the early to late portions, whereas our ear is a continuous integrator and does not interpret this sharp transition. *Definition* or D_{50}, is

similar, but measures the ratio of the energy in the first 50ms over the total energy, given by

$$D_{50} \triangleq 10 \log \frac{\int_0^{50ms} h^2(t)dt}{\int_0^{\infty} h^2(t)dt} \tag{34}$$

Other measures, such as *Early Lateral Energy Fractions* (LF) provide useful information about spatialization in the room, but require stereo IR recording so that we can compare energy as a function of incident angle, which is beyond the scope of this research [Barron, 2000].

2.6.3 Mel-Frequency Cepstral Coefficients

Heise et. al. attempted to design an optimization algorithm to adjust parameters on black box artificial reverb plugins in order to match a given IR [Heise et al., 2009]. The current research aims to accomplish a similar goal but with knowledge of the underlying reverb algorithm, which allows us to more intelligently adjust its parameters.

Heise et. al. note that Mel Frequency Cepstral Coefficients (MFCCs) are well-suited for describing noisy, non-tonal audio signals, which IRs generally are. This is because MFCCs ignore the fine structure of the frequency spectrum, instead identifying harmonic trends in it. Their system computes the 26 MFCCs over short intervals along the IR. A sampling frequency of 44,100 Hz, frame size of 2048 samples, and overlap of 50% was used. To compute a single value representing the difference between two IRs, they simply summed the Euclidean distances between time-aligned MFCC vectors for the two IRs. After conducting

listening tests, their results showed that their optimization algorithm was able to automatically design reverb patches at least as well as professional studio engineers could by hand.

2.6.4 Power Envelope

Chemistruck et. al. pursued the same goal as the current work, using the genetic algorithm to generate coefficients for a 4-channel FDN reverberator. The genetic algorithm was used to generate the feedback matrix and lowpass filter cutoffs for the $h(z)$ filters from Figure 15. The fitness value was computed as the difference in the target and synthetic IR's *power envelope*. An audio envelope detector was used to extract this envelope. The detector is a simple one pole lowpass filter with a difference equation

$$v[n] = \alpha p[n] + (1 - \alpha)y[n - 1] \tag{35}$$

where

$$p[n] = x[n]^2 \tag{36}$$

is the power of $x[n]$ and $v[n]$ is the envelope. The tunable α determines how closely the envelope tracks the instantaneous changes in $p[n]$, with $y[n] = p[n]$ for $\alpha = 1$. A sample IR with its envelope computed is shown in Figure 17. The authors computed the fitness error, ϵ, as the difference in the target's power envelope, $v_T[n]$, and the synthetic's, $v_S[n]$, scaled by the product of their respective lengths, N and M, with $N < M$.

$$\epsilon = \frac{1}{NM} \sum_{n=0}^{N-1} |p_{tar}[n] - p_S[n]| \tag{37}$$

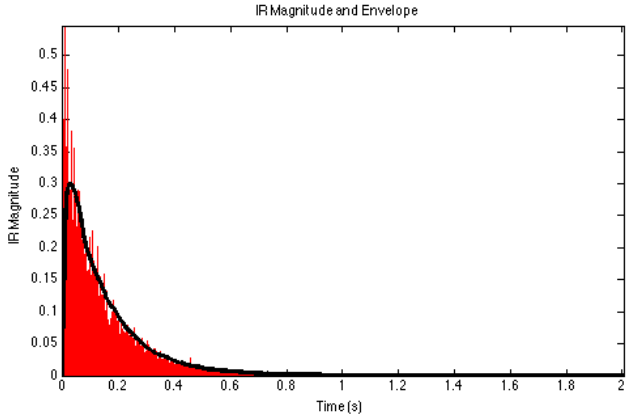

Figure 17: Sample room IR magnitude with traced power envelope

This error calculation produces a high error if there are ever periods of insufficient echo density, but ignores fine grained changes in the signals. It ensures that the energy in both signals is dissipated in approximately the same way.

3

Proposed Method

The ultimate goal is to remove the need for time-consuming hand-tuning of reverberator patches by automating the tuning process. To do this, a software system must be designed that accepts a target room IR, hereafter called $h_{tar}[n]$, and returns all gains, filter coefficients —generally, *parameters*—for a perceptual reverberator structure to produce a room sound indistinguishable from convolution with the target. Such a system allows a large quantity of room *fingerprints* to be modeled without requiring a human to sit and twiddle knobs.

3.1 Reverberator Structure

Given the controllability and flexibility of Jot's FDN reverberator, his basic structure was chosen—with a few modifications—as the perceptual reverberator. The structure used is shown in Figure 18. Processing begins on the left with the input, $x(z)$. Since the early reflections have a significant impact on the perception of a room, it was decided that this portion of the room IR should be kept and used. This set of FIR filter coefficients is obtained by truncating the target IR at sample S

$$h_{ear}(n) = \{h_{tar}[1], \ldots, h_{tar}[S]\}. \tag{38}$$

In most real-time applications of this work, processing will be performed on buffers of data at a time, making fast convolution techniques useful for any FIR that is longer than the audio buffer size. Buffer sizes are most often powers of 2

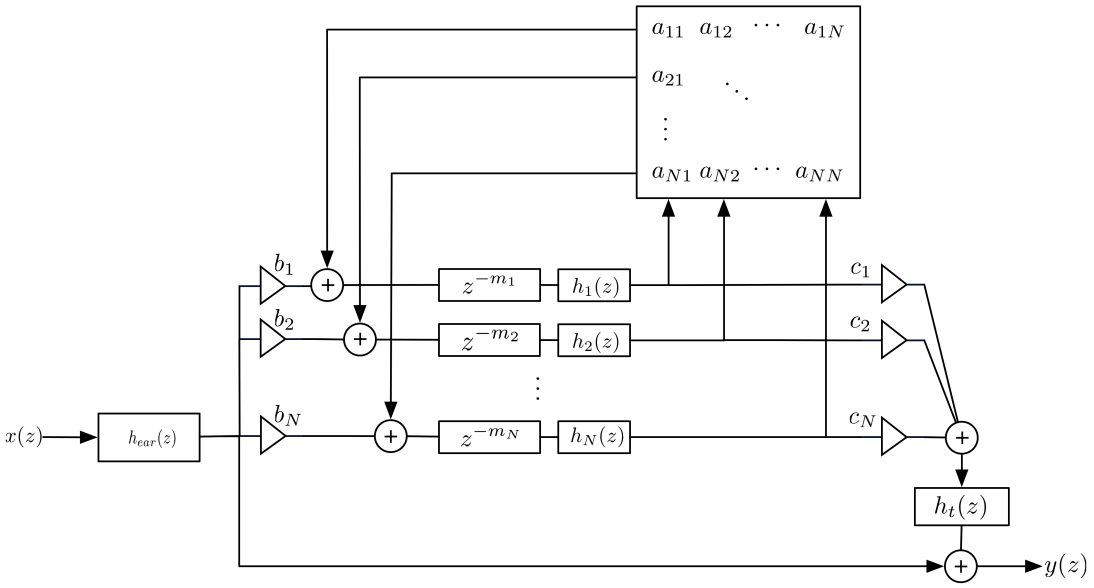

Figure 18: Feedback delay network structure used for perceptual reverberator

in length, so chopping the beginning of the target IR to a length that is a power of 2 allows us to maximize the amount of direct information we retain from the target while taking full advantage of any CPU cost. 4096 samples at 48 kHz equates to about 85 ms, which is just over the commonly quoted 80 ms "early reflections" length, so this length was chosen for S.

The output of the early reflections filter is scaled by the coefficients $b_1, ..., b_N$ ($\mathbf{b}$) and summed with the feedback matrix outputs, then fed into the delay lines of length $m_1, ..., m_N$ ($\mathbf{m}$). The delay outputs are fed into the decay filters, $h_1(z), ..., h_2(z)$ ($\mathbf{h}(z)$). The $\mathbf{h}(z)$ outputs are sent both into the feedback matrix and to the output scaling factors $c_1(z), ..., c_2(z)$ ($\mathbf{c}$). The sum of the outputs of the scaling by $\mathbf{c}$ is passed through a tonal corrective filter, $h_{tone}(z)$. The output of $h_{tone}(z)$ is summed with the unity gained early reflections output to form the system output, $y(z)$.

3.2 Parameter Design Algorithm

Since we chose to convolve with the early reflections, we only need to approximate late reverberation with the FDN. One of the prime objectives in this portion is to get as high an echo density as possible. If the early reflections truncation is late enough, then the late reverberation should not contain distinguishable reflections to the ear, so surely we cannot have too *many* reflections. In light of this lack of an upper bound, the Hadamard matrix, with its "maximal scattering" property, was chosen as the constant (not optimized) feedback matrix.

The number of channels for the FDN, N, also had to be decided, and using the Hadamard matrix meant it must be a power of two. 4, 8, and 16 channels were all tested throughout evaluation, with an expected increase in perceived diffusion as N increased. Since the current research is more concerned with matching a target IR than extreme computational efficiency, $N = 16$ was ultimately used, but the design procedure that follows can be used in any case.

The overall idea with the design algorithm is to analytically determine as many parameters related to the IR matching as possible, and use the genetic algorithm only to determine the values that behave in more complex ways. The final method is a genetic algorithm approach to the problem with some analytic design between passes. MATLAB was chosen as the prototyping environment and the available genetic algorithm solver was used.

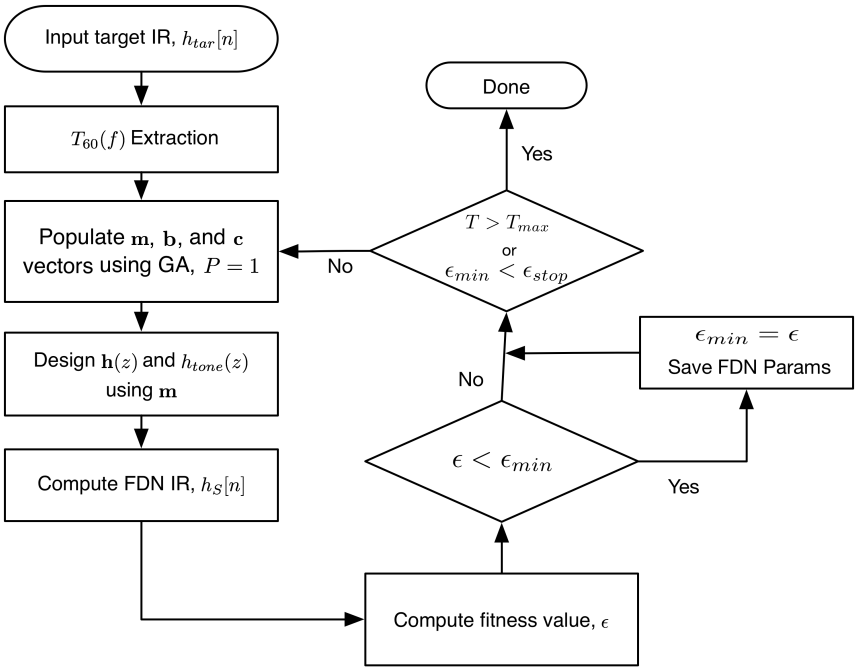

Figure 19: Initial design algorithm flow chart

3.2.1 Iterative Procedure

From Jot's work, we know we can design filters that—to our necessary accuracy– approximate the target IR's $T_{60}(f)$ curve with the filters $\mathbf{h}(z)$ controlling the frequency-dependent decay of the system. $h_{tone}(z)$ can be designed according to Eq. 18, so this leaves $\mathbf{m}$ and $\mathbf{c}$ to be optimized. Thus, the initial design flow was implemented as shown in Figure 19.

The system begins with extraction of $h_{tar}[n]$'s reverberation time curve, $T_{60}(f)$. Next, the genetic algorithm is used to create a population of $\mathbf{m}$ and $\mathbf{c}$ vectors. c_i values are constrained to be between $[-1, 1]$, and m_i values, given the prior art, are constrained to cover a scale of 1:2.5, with the longest value corresponding to 100 ms. m_i values were also constrained to be integer values in

the genetic algorithm.

After the population is created by the GA, we use each m_i value, our target $T_{60}(f)$ curve, and Eq. 17 to design each decay filter $h_i(z)$. The Yule Walker IIR design method is used for this task. Sixth order filters are used for a trade-off between computational complexity and fit quality. The same method is also used to generate a 12^{th} order corrective filter, $h_{tone}(z)$ using Eq. 18. Next, we use the generated $\mathbf{m}$ and $\mathbf{c}$, the $N \times N$ Hadamard matrix (Had(N)), and the generated filters, $\mathbf{h}(z)$ and $h_{tone}(z)$, to compute the IR of the FDN reverberator.

The fitness function, whose implementation is discussed later, is then used to compute a fitness error, ϵ, between the target and synthetic IRs. If the error is lower than any previous error, ϵ_{min}, this set of FDN parameters is saved off as the best set of parameters and ϵ_{min} is updated. If the fitness value is below the error required to stop, ϵ_{stop}, or the running time of the GA, T, exceeds the maximum time limit, T_{max}, the iterative procedure terminates. Otherwise, the genetic algorithm produces another individual for fitting.

This design technique produces promising results. However, the overall timbre of the reverb is poorly matched to the original. This is because Jot's design method for $h_{tone}(z)$ as a filter with a frequency response inversely proportional to $T_{60}(f)$ (Eq. 18) corrects the coloration introduced by $\mathbf{h}(z)$, but does not match the timbre of the original IR.

To improve the corrective filter, a new technique was used to design the corrective filter. This technique—one of the novel contributions of this

thesis—finds the average spectral error between the synthetic IR generated *without* a tonal correction filter and the target IR, then designs a filter to correct the difference. The period of comparison begins at the end of the early IR convolution and ends at the lower of the two IR's T_{60} time. The early reflections portion should match in timbre exactly, and after a 60 dB decrease in level, the timbre cannot be easily heard.

This portion is passed through a third octave band filter bank, and the per band RMS powers are calculated for both the target and synthetic IR. Frequency sampling FIR design is then used to design a 4096 tap FIR filter to correct the spectral error. With the h_{tone} filter designed, the final FDN IR for this GA iteration is computed, and the fitness calculation continues as before. These modifications are shown in the final design flowchart, Figure 20.

3.3 Fitness Functions

At every iteration, the genetic algorithm generates the desired number of parameters, and accepts in return a scalar value corresponding to the error in the system "fit" of those parameters it generated—without knowledge of the system. It merely attempts to minimize the fitness error value over multiple iterations by breeding the best sets of parameters and mutating to create new sets. Thus, the "fitness function" must be defined for each application, and it plays the central role in the search for optimal parameters.

Using the system described thus far, a handful of fitness functions were investigated to determine which performed best. Relative performance was

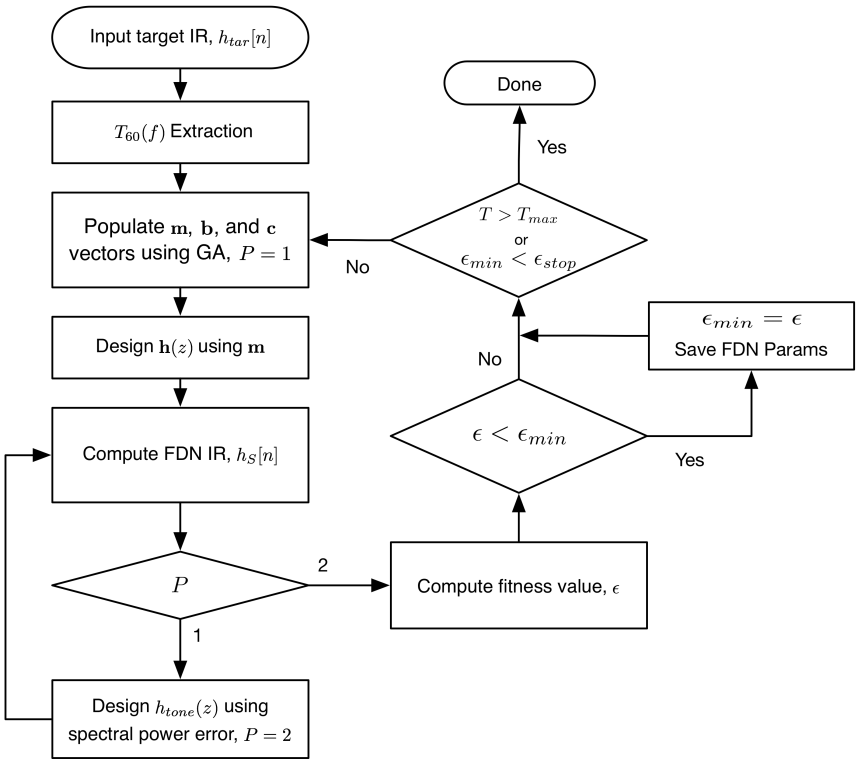

Figure 20: Final design algorithm flow chart

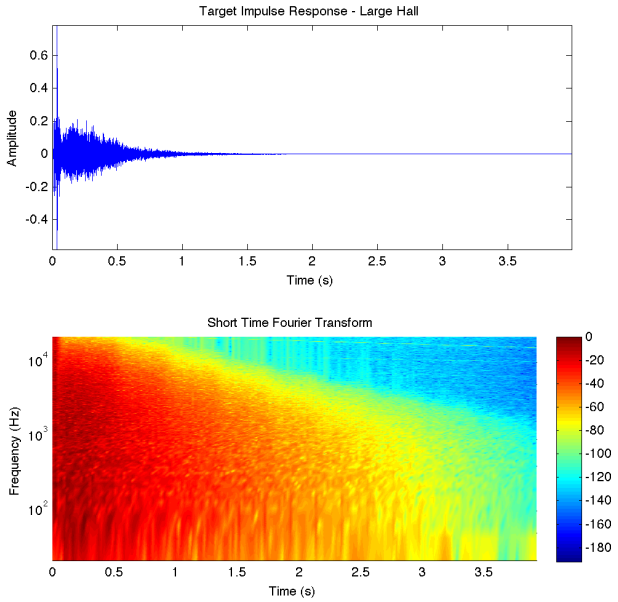

Figure 21: Large, dark hall IR chosen as the target IR for fitness function evaluation

judged by subjective comparison of sounds processed by the GA-designed FDN with convolution reverb using the target IR. Here, one exemplary target IR is chosen as a benchmark test against which to judge the fitness functions' performances. This target room IR, $h_{tar}[n]$, is a large, dark hall with a T_{60} time of 2.4 s, and represents a typical, but difficult-to-match room IR, due to its length. The IR and STFT are shown in Figure 21. In each case, the algorithm was allowed to run for 45 minutes in order to give sufficient time for minimization of the fitness function.

3.3.1 Human Reverb Perception Modeling

We are ultimately interested in matching a target IR in a way that *sounds* identical to human ears. Ideally then, we'd like the fitness scale to correspond to the perceptual similarity of the two IRs. This immediately throws out one

obvious, but certainly problematic fitness function: the sample-for-sample difference

$$\epsilon = \frac{1}{L} \sum_{n=0}^{L} |h_{tar}[n] - h_S[n]| \tag{39}$$

where L is the length of the shorter of the two IRs. Certainly, if the fitness error, ϵ, goes to zero, the IRs will sound identical, because they *are* identical. But if the synthetic IR was exactly the target IR but with its phase flipped, the error would be high even though the two would sound identical, so this is not a good fitness function. We are after a criteria that is robust to changes that aren't perceptible.

In search of such a measure, a novel experiment was designed to try and generate a mapping between differences in objective measurements of room IRs and their perceived differences. If such a mapping could be developed, then we could mimic a human listener for our fitness function by returning the mapped perceptual difference from differences in objective measurements between the target and synthetic IR.

A paired comparison listening test was used to generate this mapping. A set of M reverb IRs were selected, convolved with a stimulus to create a short processed audio clip, then paired with every other convolved clip so that M^2 ordered pairs of clips were generated. Thus, M pairings contained the same two clips, and all other pairings were redundant with an order swap. Test subjects would then listen to these pairings, and for each, rate the similarity of the room sounds on a five point scale from "very dissimilar" to "the same".

An early pilot study was run using $M = 7$ reverbs that had subjectively

varying characters. A 2-D multi-dimensional scaling was performed on the data, which confirmed that the similarity in T_{60} times of two reverbs has an overpowering effect on the perceived similarity. Since the $T_{60}(f)$ matching is essentially a solved problem from Jot's work, a new set of reverbs was selected with very similar $T_{60}(f)$ curves so as to remove this factor and uncover the second "tier" of similarity metrics.

The selection criteria was defined more rigorously for this iteration of the test. The idea would be to select IRs that were similar in $T_{60}(f)$ values but highly varying in a set of other objective measures. The objective measures whose variance was to be maximized were chosen based on the prior work from Chapter 2. They are:

1. Early Decay Time (EDT) - Time required for reverb to decay by initial 10 dB, as discussed in Section 2.6.2.

2. C_{50} - Clarity index as discussed in Section 2.6.2 and defined in Equation 33.

3. D_{50} - Definition index as discussed in Section 2.6.2 and defined in Equation 33.

4. Center Time (T_s) - The temporal center of gravity, or first moment, of the IR.

5. Onset Spectral Centroid (S_o) - The center of gravity of the power spectrum of the first 20 ms of the IR. This measure does not come from any

literature, but attempts to capture a measure of the "brightness" of an IR, since no other measure is concerned with timbre

6. T_{20} - Time required for reverb to decay from -5 dB to -25 dB

Over 900 reverb IRs were collected from libraries available on the internet. The optimal set of IRs was selected from this set for the paired comparison test. To minimize listener fatigue, only $M = 7$ IRs were to be selected, making a total of 49 comparisons—about a 20 minute test. More IRs would be better from a data-quantity point of view, but the squared increase in test time with M makes this test too long very fast, risking listener fatigue and skewed responses.

From the large set of IRs, a subset needed to be found with similar $T_{60}(f)$ curves. A wide-band T_{60} time of 1.1 seconds was first chosen as the target time for the selected reverbs since this is a typical target T_{60} time for music performance rooms. The set was narrowed to 176 IRs with T_{60} times between 1.0 and 1.25 seconds. From here, a search was performed to find the 50 IRs that minimized the total difference between all $T_{60}(f)$ curves

$$d = \sum_{i=0}^{N-1} \sum_{j=i+1}^{N} |\mathbf{T}_{60_i} - \mathbf{T}_{60_j}| \tag{40}$$

$$d = \sum_{i=0}^{S} \max(\mathbf{P}_{:,i}) - \min(\mathbf{P}_{:,i}) \tag{41}$$

$T_{60}(f)$ curves were first grouped into 26 bark bands to ignore fine-grained changes in the linearly spaced curve generated from the EDR. Each curve was then interpreted as a point in 26-D space, and the search was performed to find

Index	T_{60}	EDT	T_{20}	C_{50}	D_{50}	T_s	S_o (kHz)
1	1.22	0.19	0.35	-6.0	0.20	0.10	13.3
2	1.20	0.18	0.35	-0.8	0.46	0.08	0.14
3	1.15	0.17	0.30	-6.0	0.20	0.09	6.19
4	1.08	0.16	0.27	-2.7	0.35	0.08	8.87
5	1.07	0.05	0.37	9.4	0.90	0.03	7.47
6	1.11	0.23	0.34	-4.5	0.26	0.11	4.53
7	1.15	0.16	0.40	2.3	0.63	0.06	5.68

Table 1: Objective Features of the seven reverb IRs used

the combination of 50 points that minimized the total Euclidean distance between all pairs.

With the set narrowed to 50 IRs, the final narrowing was performed to maximize the variance in objective measurements for a set of seven IRs. The six objective measures outlined above were calculated for each of the 50 IRs and assembled into a vector, $\mathbf{p} \in \Re^6$, for each IR. To normalize the relative importance of each of the six measures in the maximization, the measures were all scaled into the range [0,1] where 0 represents the minimum value seen in any particular measure within the set, and 1 represents the maximum. After this step, all normalized vectors, $\hat{\mathbf{p}}_i$ for $i = 1 \ldots 6$, could be considered points in a 6-D unit box. Another search was then performed to find the seven IRs that maximized the total spread of values in all six dimensions. The final selected IRs are shown in Table 1.

These seven IRs were convolved with a short rock drum kit loop recorded in an acoustically dead environment and used as the clips for the paired comparison test. A total of ten subjects age 20 to 54 participated, and all subjects were familiar with audio and basic room acoustics. Results for each of

Index	1	2	3	4	5	6	7
1	4.4						
2	2.7	4.6					
3	2.8	2.45	4.6				
4	1.95	2.25	2.8	4.4			
5	1.05	1.2	1.5	2.3	4.9		
6	3.0	3.3	2.45	1.9	1.3	4.9	
7	1.55	1.55	2.2	2.8	2.95	1.85	4.5

Table 2: Average similarity scores, $\hat{\mathbf{S}}$, for the seven presented reverbs

the 49 pairings were averaged and compiled into a matrix, $\mathbf{S} \in \Re^{7\times7}$, then evaluated for validity . The greatest difference in average score between any given pairing and its flipped pairing (A then B versus B then A) was 0.8, or 20 % of the scoring range. This variance was higher than desired, but the high average scores for the identical pairings does help increase trust in the subjects judgement. Scores for each ordering were then averaged to create the final lower triangular score matrix

$$\hat{\mathbf{S}} = \frac{\mathbf{S} + \mathbf{S}^T}{2} \tag{42}$$

which is shown in Table 2. A least squares fit was then performed in order to map differences in objective measures between any two IRs to their perceptual difference. The perceptual difference matrix, $\mathbf{D}$, was first generated by mirroring the similarity matrix, $\hat{\mathbf{S}}$, in the scoring range such that

$$\mathbf{D} = 5 - \hat{\mathbf{S}} \tag{43}$$

which maps a similarity score of five, or "the same", to a corresponding difference score of zero. The objective measure row vectors, $\mathbf{p}$, were then combined into matrix $\mathbf{P} \in \Re^{7\times6}$ such that p_{ij} gives the value of objective measure j for IR i.

Finally, the problem was placed into the usual least squares format, $\mathbf{y} = \mathbf{A}\mathbf{x}$, where $\mathbf{y} \in \Re^{21}$ contains the perceived differences for each non-repeated pairing in $\mathbf{D}$

$$\mathbf{y} = \begin{bmatrix} D_{21} \\ D_{31} \\ D_{32} \\ D_{41} \\ \vdots \\ D_{76} \end{bmatrix} \tag{44}$$

and a_{ij} in $\mathbf{A} \in \Re^{21\times 6}$ gives the absolute difference in objective measurement j for the two IRs in the pairing for row i.

$$\mathbf{A} = \begin{bmatrix} |P_{21} - P_{11}| & |P_{22} - P_{12}| & |P_{23} - P_{13}| & \dots & |P_{26} - P_{16}| \\ |P_{31} - P_{11}| & |P_{32} - P_{12}| & |P_{33} - P_{13}| & \dots & |P_{36} - P_{16}| \\ |P_{31} - P_{21}| & |P_{32} - P_{22}| & |P_{33} - P_{23}| & \dots & |P_{36} - P_{26}| \\ |P_{41} - P_{11}| & |P_{42} - P_{12}| & |P_{43} - P_{13}| & \dots & |P_{46} - P_{16}| \\ \vdots & & & & \\ |P_{71} - P_{61}| & |P_{72} - P_{62}| & |P_{73} - P_{63}| & \dots & |P_{76} - P_{66}| \end{bmatrix} \tag{45}$$

Computing the least-squares solution $\mathbf{x}_{LS}$ as

$$\mathbf{x}_{LS} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y} \tag{46}$$

defines the linear weightings of the differences in objective parameters between two IRs to match most closely the perceived difference in the two. These weights are

$$\mathbf{x}_{LS} = \begin{bmatrix} -7.19 \\ 0.28 \\ -6.86 \\ 59.7 \\ 1.79 \times 10^{-4} \\ 14.85 \end{bmatrix} \tag{47}$$

where row i corresponds to the weighting for objective measurement i from the enumeration. With these weightings, the approximation error vector, $\mathbf{e}$ is

$$\mathbf{e} = \mathbf{y} - \mathbf{A}\mathbf{x}_{LS}. \tag{48}$$

The maximum absolute error is

$$\epsilon_{max} = |\mathbf{e}|_\infty = 1.03 \tag{49}$$

and the mean absolute error is

$$\epsilon_{avg} = \frac{1}{21}\sum_{i=0}^{21} |e_i| = 0.38. \tag{50}$$

This means that on average, by weighting the difference in the six objective measurements of two IRs by the coefficients in $\mathbf{x}_{LS}$, we can guess the average perceptual difference to within 9.5 %.

Given this reasonable success in estimation, a fitness function was written to compare a target and synthetic IR based on the weighted differences in these six measurements. Unfortunately, this method did not work as well as hoped as a fitness function alone. The best fit IR reported after 45 minutes of running is shown in Figure 22. The gap of silence just after the early response, caused by all delay lines being too long, gives a clearly delayed echo when applied. Optimizations run on shorter and longer IRs gave similar results, which were all poor. It seems the six parameters chosen did not specify the IR strongly enough to act as a guidance towards a perceptually identical match.

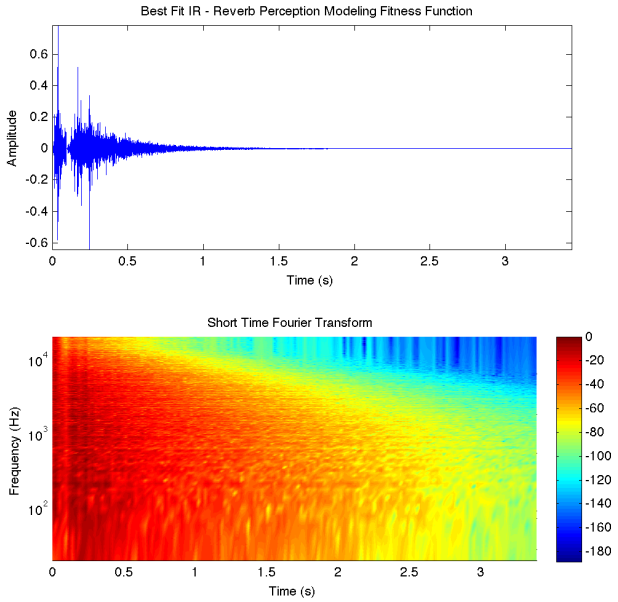

Figure 22: Final best IR generated using the human perception modeling fitness function

3.3.2 MFCC

Because this method was used with moderate success in [Heise et al., 2009], the frame-by-frame Euclidean distance between MFCC vectors was also investigated as a fitness function. The error is calculated as

$$\epsilon = \frac{1}{M} \sum_{m=0}^{M} |MFCC_{tar}(m) - MFCC_S(m)|, \tag{51}$$

where m is the frame number and M is the number of frames prior to some T_X reverberation time. -30 dB was used here is to keep the fitness focused on the most salient portion of the sound.

Results for this method were generally poor. The total distance was decreased over multiple generations, but the minimization did not correspond to an increase in perceptual similarity in any controlled way. Large gaps of silence

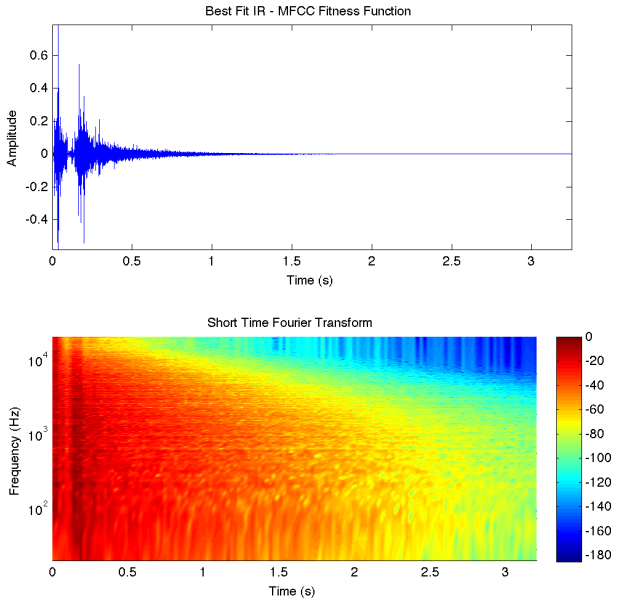

Figure 23: Final best IR generated using the MFCC-based fitness function

could be present in the output IR which helped to minimize the fitness function but had a distinctly adverse effect on the sound quality. This method might perform decently when working with reverb units that will always at least produce a passable reverb IR, as was the case in Heise's work on perceptual reverb unit tuning, but with an FDN, where large areas in the parameter-space will not produce such a sound, this method seemed to fall short.

3.3.3 EDC

Also considered were fitness functions based on the average absolute differences in the EDC's when both were normalized to a starting value of 1,

$$\epsilon = \frac{1}{L}\sum_{n=0}^{L} \left| \frac{EDC_{tar}[n]}{EDC_{tar}[0]} - \frac{EDC_S[n]}{EDC_S[0]} \right|. \tag{52}$$

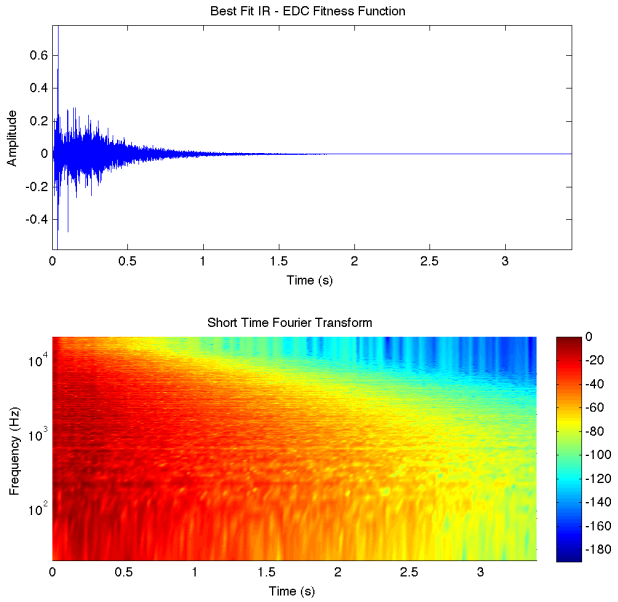

Figure 24: Final best IR generated using the EDC-based fitness function

where L is the shorter of the two IR's length. Unlike with MFCC's, this measure automatically favors the more relevant errors in the early decay since the late errors are necessarily small in magnitude. The best fit IR generated in with this is shown in Figure 24.

3.3.4 EDR

The EDC fitness function was extended to the frequency-dependent generalization, EDR, as well. Here, every frequency bin was normalized by its initial bin value to as to only compare decay in energy rather than constant offsets. The fitness error is given by

$$\epsilon = \frac{1}{KM} \sum_{m=0}^{M} \sum_{k=0}^{K} \left| \frac{EDR_{tar}(m,k)}{EDR_{tar}(0,k)} - \frac{EDR_S(m,k)}{EDR_S(0,k)} \right|, \tag{53}$$

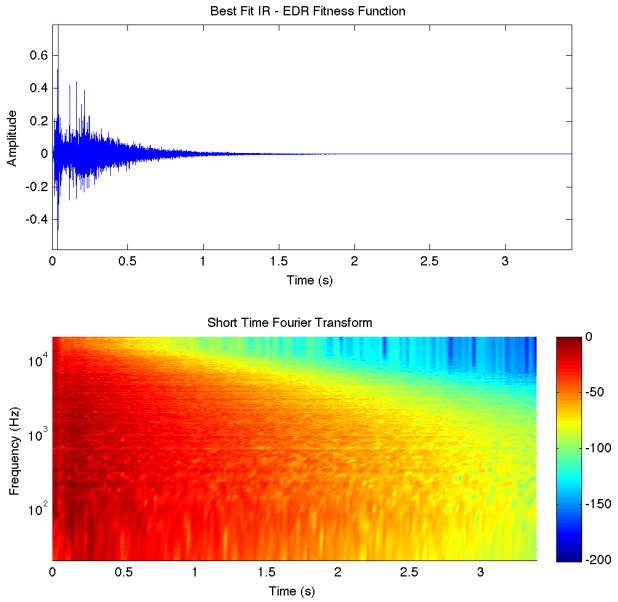

Figure 25: Final best IR generated using the EDR-based fitness function

where M is the total number of time frames and K is the number of frequency bins. However, these proved to be problematic as well. Because both EDC and EDR calculations involve a reverse-time summation, small errors in the very late reverberation propagate all the way to the beginnings of each calculation. While these differences in the very late reverberation may be hardly noticeable, they are capable of skewing the error calculation to be unjustly high. Thus, measurements involving cumulative summations were deemed inadequate.

3.3.5 Envelope

Chemistruck et. al. used a fitness function based on the average absolute error in the power envelope of each signal (Eq. 37) [Chemistruck et al., 2012]. This fitness method was also used with a few modifications. The first modification involved weighting the sample differences by an exponential decay.

This was used in order to steer the GA into caring more about errors in the early part of the IR than those in the late, where differences are less perceptible

$$\epsilon = \frac{1}{N} \sum_{n=m_{min}}^{N-1} e^{-\lambda n} |p_{tar}[n] - p_S[n]| \tag{54}$$

where N is the shorter of the IR lengths and m_{min} is the shortest delay length in $\mathbf{m}$. This lower bound was chosen because output sample m_{min} is the first sample where the target and synthetic IRs can have differing output value. λ was chosen so that

$$0.1 = e^{-\lambda N} \tag{55}$$

or

$$\lambda = \frac{-\ln(0.1)}{N} \tag{56}$$

meaning that errors at the end of the envelopes matter one tenth as much as errors at the beginning in the fitness.

Another variation tried was to simply return the maximum value (infinity norm) of the absolute differences in envelopes

$$\epsilon = \| \, p_{tar}[n] - p_S[n] \, \|_\infty \tag{57}$$

This fitness function, of all tried, and on many IRs, seemed to steer the GA to a better and better perceptual match in the most straightforward way, and was capable of producing results far better than any other fitness function. It's best case IR for the target is shown in Figure 26. The envelope fitness function was the clear winner and was chosen for further performance investigation in Chapter 4. The complete, proposed system flow chart is shown in Figure 27.

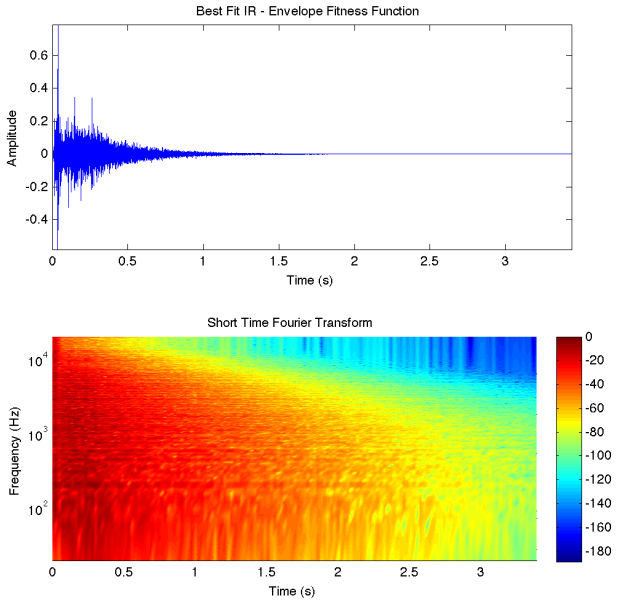

Figure 26: Final best IR generated using the envelope-based fitness function

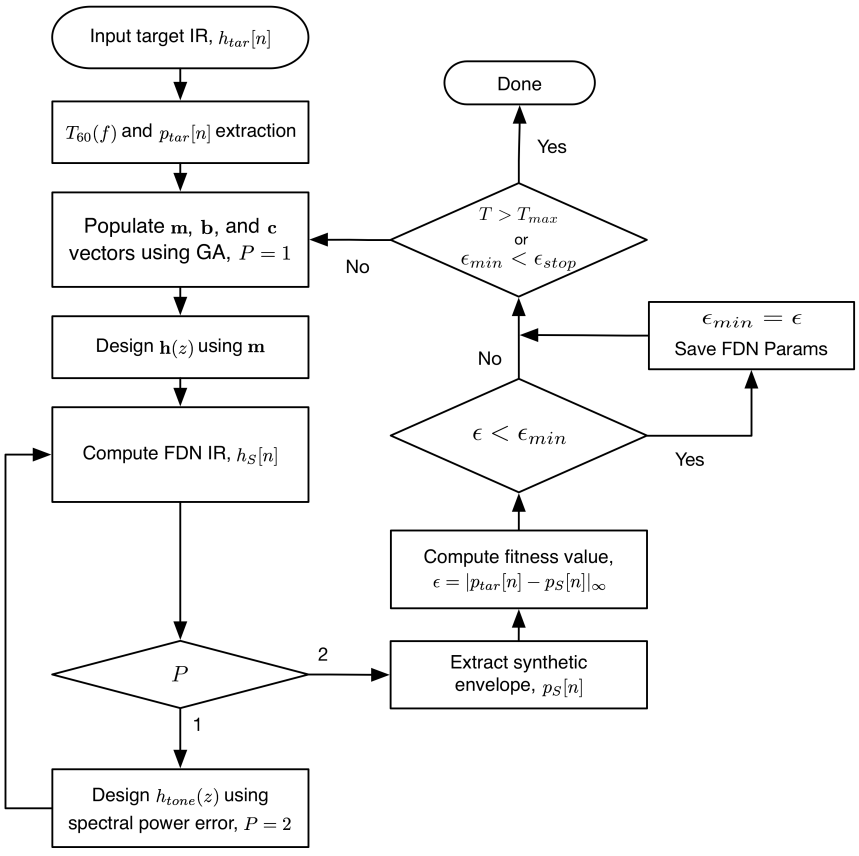

Figure 27: Final design flowchart with fitness calculation

4

Evaluation

The end goal of the proposed design system is to autonomously design an FDN reverberator which produces a reverb sound that *indistinguishable* from a given target reverb sound. Thus, this chapter first presents a user study that was designed and carried out to test this ability. The benefit of this approximation should be a CPU performance and memory savings when compared to convolution reverb, so this evaluation was also carried out.

4.1 Listening Test

To test the proposed system's performance, a listening test was created to measure test subjects' perceptibility of differences between convolution reverb and an FDN reverb designed to match the convolution IR. For this task an ABC/HR test was used. An ABC/HR test, where HR stands for *hidden reference*, is a testing method used to score the quality of an audio clip with respect to some *reference* recording. During each of these tests, 3 audio clips are presented to a user. One clip is the reference clip and is labelled as such. The two remaining clips are presented blindly. One of these is again the reference, and the other is an altered version. The user must rate these two clips on a sliding scale measuring quality or similarity with respect to the reference. Therefore, one of the two scores should always be fully in the "perfect" quality or "indistinguishable" similarity direction of the scale.

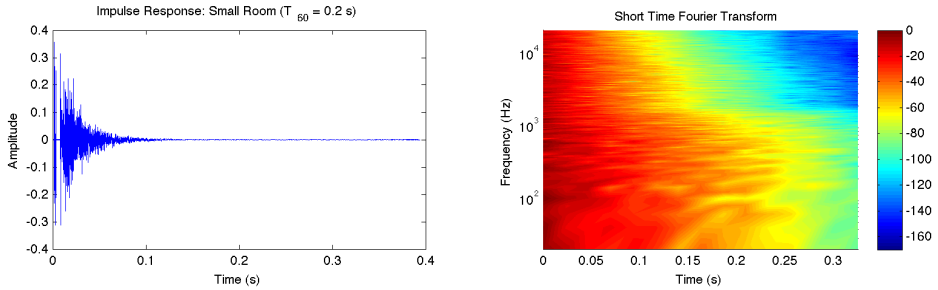

Figure 28: Small room target IR

4.1.1 Target IR Selection

Ideally, the reverberator structure and optimization system should be able to match a range of room types from small booths to arenas. Therefore three room IRs spanning a large range of T_{60} time were chosen as inputs to the system.

The shortest room IR was taken from an acoustic room impulse response library distributed by *Recording School Online* [Online, 2015]. The IR chosen is from the "House Rooms" collection of the library, and is called "RSO IR Room9". This clip was chosen not only to fulfill the small room end of the reverb spectrum with its T_{60} of 0.2 s, but also for its challenging, sharp spectral roll-off around 2 kHz, as shown in Figure 28. In the time domain, it appears that the entirety of the signal is captured in the 80 ms early reflections period, but when the STFT is viewed in the decibel power scale, the IR is clearly audible after this period, so the FDN approximation is relevant. The left side of the stereo IR recording was used since the current work only focuses on monaural IR matching. This IR will be referred to as the "small room" IR.

The remaining two IRs were taken from an IR library of the high-end Bricasti M7 perceptual reverb unit. The choice to use two IRs generated by a

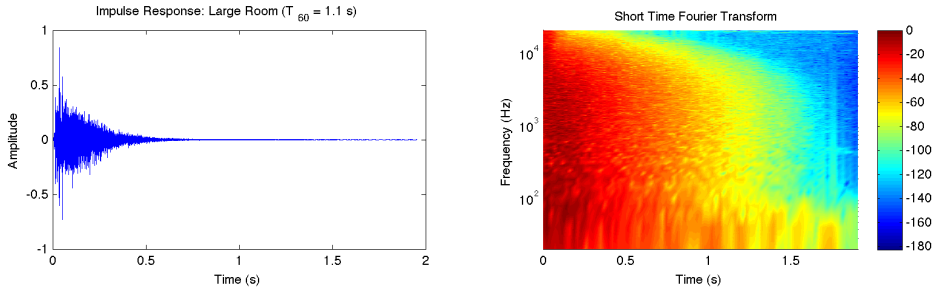

Figure 29: Large room target IR

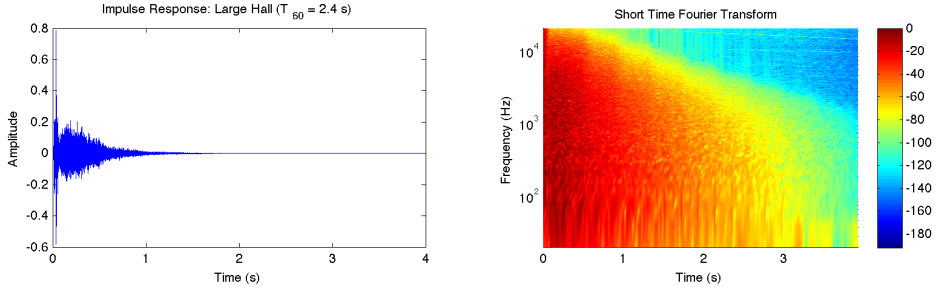

Figure 30: Large hall target IR

perceptual unit was made to show the design algorithm's ability to match a wide range of sonic characters, so matter how the target IRs were created. The first IR chosen from this collection represents a typical large room with its T_{60} of 1.1 s, and is shown in Figure 29. This preset on the M7 unit is called "08 - Music Room", and will be referred to as the "large room" henceforth.

The last IR represents a large hall with its long, 2.4 s T_{60}, as shown in Figure 30. This preset is called "24 - Troy Music Hall" and will be referred to as "large hall" here. The "mid" channel of a mid-side IR sampling of each of these Bricasti IRs was used.

4.1.2 Optimization Progression

The system was allowed to optimize the FDN parameters with a 45 minute time limit T_{max} and a fitness limit ϵ_{stop} of 0 for each IR, which, as

expected, was never reached. A population size of 100 was used.

Progress of the optimization is depicted for the small room in Figure 31. The top plot shows the best fit envelope after the first, middle, and final generation plotted against the target envelope in black. The bottom left graph shows how well the synthetic FDN reverb's $T_{60}(f)$ curve matches that of the target for the final best fit IR. Similarly, the bottom right plot depicts how well the average spectral shape of the final synthetic IR matches that of the target. The averaging here is performed over the same range of time used in the calculation of the tonal corrective filter, $h_{tone}(z)$.

The results for the small room are likely the least exciting of the results due to its relatively small section in time where we are beyond the perfect fit of the early reflections convolution and before the envelope is too near zero to see the optimization progression. Nonetheless, the envelope fit is clearly tightened—almost to a level of imperceptible deviation— between the end of the first and final generation. The apparent choppiness in the $T_{60}(f)$ curves is due to the short T_{60}, which exposes the temporal resolution of the STFT used to compute each IR's EDR. In general, both the $T_{60}(f)$ and spectral shapes match well, although high levels of deviation are noted in the sub 120 Hz band.

The same plots are shown for the large room optimization in Figure 32. Here, the optimization progress can be seen much more clearly. After the first generation, there is still a large dip in the envelope around 0.2 s which will certainly result in a perceived difference. After 17 generations, the mid-point, fit

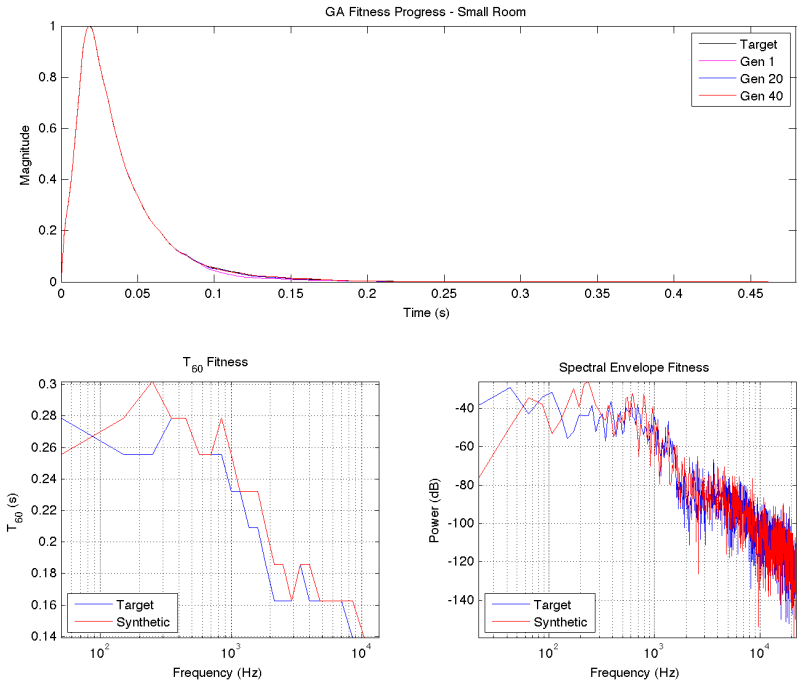

Figure 31: Small Room IR Fitness

is already much-improved, but the shape immediately following the early reflections is a bit exaggerated compared to the reference. After the final generation, this portion is tightened significantly, with the largest deviation around the 0.35 s mark. Of course, as expected, all envelopes converge to approximately the same line in sufficient time, as guaranteed by the analytic design of the decaying IIR filters in the FDN.

$T_{60}(f)$ fit is generally good, but shows a short-coming of the IIR design method used. In an effort to minimize the overall error for the given filter order chosen (six), the Yule-Walker design method creates filters with too high of a gain in the 150-7,000 Hz range so the extreme error in the sub 150 Hz range is lessened. This results in a T_{60} shortening of nearly half of a second around 50 Hz,

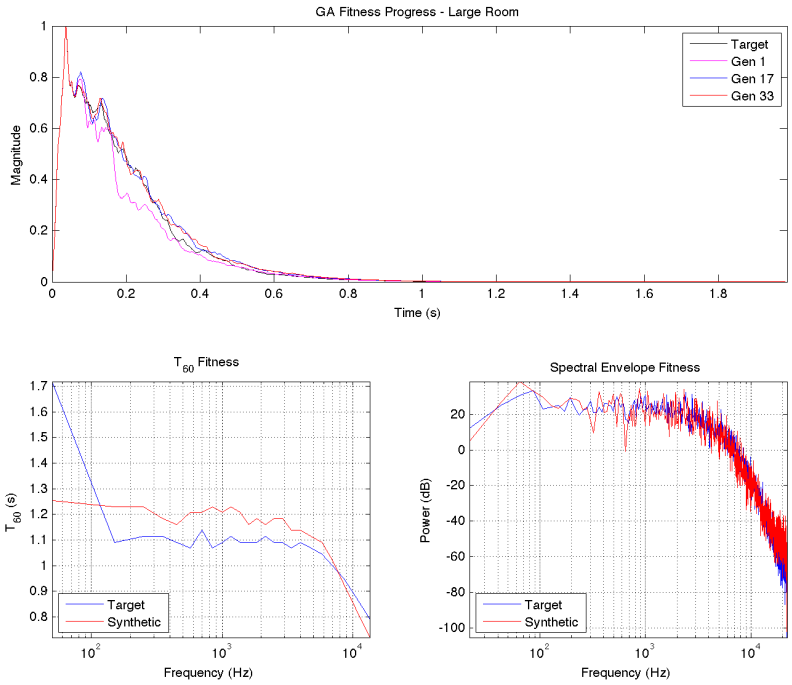

Figure 32: Large Room IR Fitness

and a lengthening of about 0.1 s throughout most of the musical content range. However, the spectral shape fit is very good across the entire frequency spectrum.

The optimization results for the large hall are shown in Figure 33. Due to the longer IR length, the testing of each GA individual takes longer to process and compare with the target. Whereas with the previous two IRs, 40 and 24 generations, respectively, were completed, a 45 minutes timeout only allowed 24 generations for this long IR, which left some fitness to be desired, especially since there was simply "more" envelope to fit. Therefore, the time limit for this IR was extended to three hours.

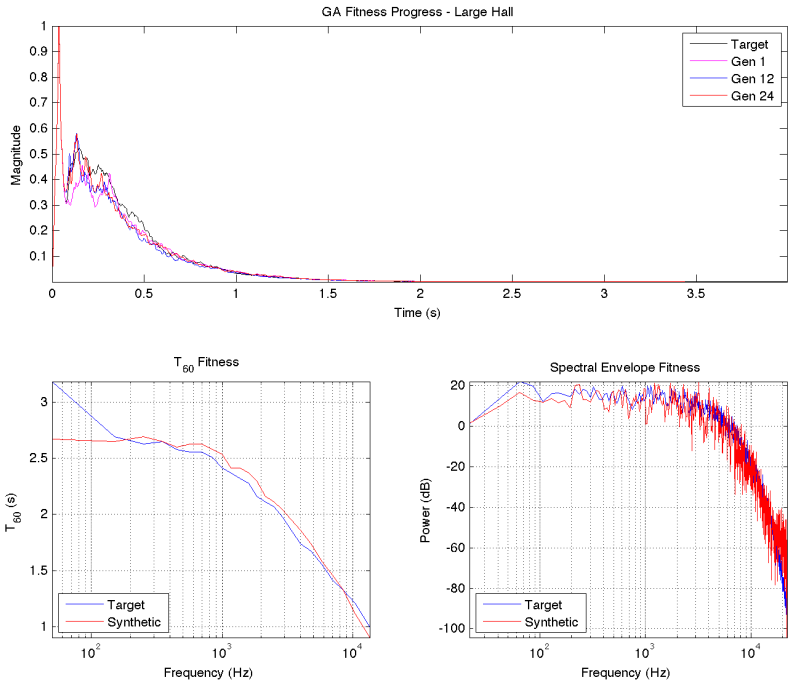

Figure 33: Large Hall IR Fitness

4.1.3 Test Procedure

Two stimuli audio tracks were chosen to which the three reverbs were applied. The prime objective in selection of the audio stimuli was to find high quality, typical use case musical tracks recorded in acoustically dry environments. The first stimulus is a six second drum kit loop recorded with close microphones in a dry environment. Percussion instruments, with their sharp transients and fast decay, are notorious exposers of artificial reverberator shortcomings such as ringing modes or insufficient echo density. The loop chosen is a short rock groove on the hi-hat with moderate use of a tight kick and snare drum. The second stimuli was chosen to test the performance on a more legato recording. A ten second soloed female pop vocal track recorded in a professional vocal booth was

used.

The three room reverbs under test were applied to the two stimuli, making for a total of six comparisons between convolution and FDN reverb. In musical applications, the reverberated, or *wet*, signal is rarely used alone. Rather, it is mixed with the un-processed dry signal to taste. In light of this, each reverberated signal was mixed with the dry signal to create the clips that test subjects heard. The mixes leaned on the wet side of typical applications so as not to falsely boost results. For the small room, the wet and dry signal were mixed in equal proportion. For the large room and hall, this mixture was considered too wet for typical applications. Instead, the wet signal was mixed -17 dB down with respect to the dry level.

Test subjects were presented with each comparison three times, making a total of 18 tests. The test interface, shown in Figure 34, was implemented in HTML and sourced from the internet [Kraft, 2014]. For each of the 18 tests, the subject was allowed to listen to the three clips any number of times before making a judgement of which blindly presented track was the reference and how perceptibility different the other track was from it. The sliders were not continuous in granularity, but split the scoring range into ten discrete segments. A score of 0 represented an "Imperceptible" difference from the reference, and a score of ten represented a "highly perceptible" difference.

A total of 12 subjects participated in the listening test. Test subjects ranged in age from 21-35 and worked in audio either professionally or

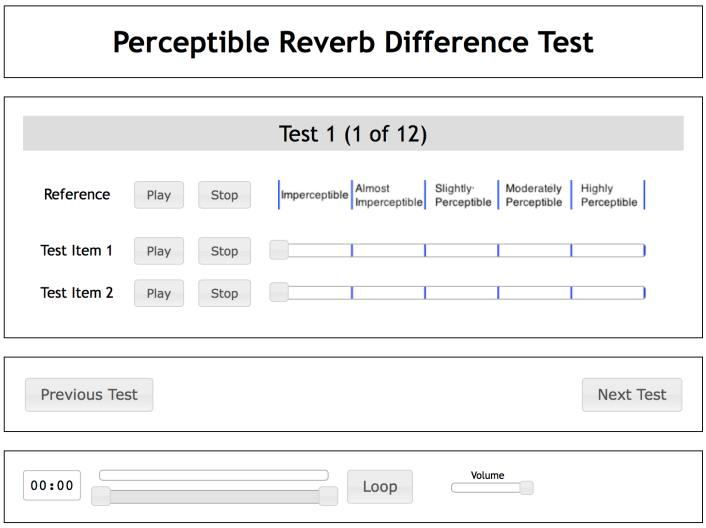

Figure 34: Listening Test user interface

scholastically. Testing was performed in an acoustically quiet environment using a consumer laptop driving Sennheiser HD-280 studio headphones. Subjects were allowed to adjust the output volume to their taste, but all presented tracks were normalized to equal perceptual loudness using Adobe Audition prior to the tests to mitigate perceptual differences due to simple loudness differences.

4.1.4 Results

After all tests were complete, individual scores for each of the six comparisons were grouped together and the mean score and standard deviation for each of these sets was calculated. These results are shown below, grouped so the scores for the reference and synthetic reverb applied to each of the two stimuli are plotted together for each of the three room IRs.

The small room IR approximation—not surprisingly—proved to be most difficult to distinguish. The results for the small room are shown in Figure 35.

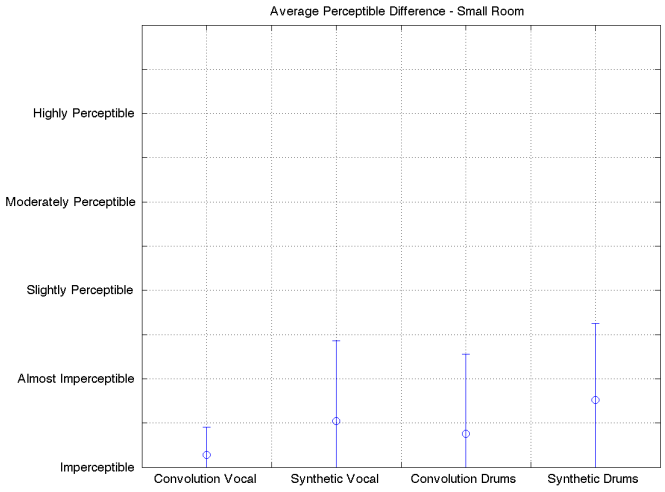

Figure 35: Scores - Small Room

On average, test subjects reported that the difference between both the vocal and drum recordings were between "imperceptible and "almost imperceptible', with several test subjects incorrectly identifying the reference, as seen by the non-zeros perceptible difference of the convolution reverbs.

The medium room results shown in Figure 36 again show that some test subjects were not able to correctly identify the reference—which is a good thing. On average, the subjects do place the perceptible difference for both the vocal and drum stimuli at higher levels than the reference, but even in the difficult case of the drum stimuli, the average difference was rated as "slightly perceptible".

Results for the large hall comparison are given in Figure 37. This proved to be the easiest of the three rooms to distinguish between convolution and synthetic FDN. Every test subject was able to correctly identify the reference drum track every time, which was not seen in any other test. Even so, the average perceptible difference here was rated as a little over "slightly

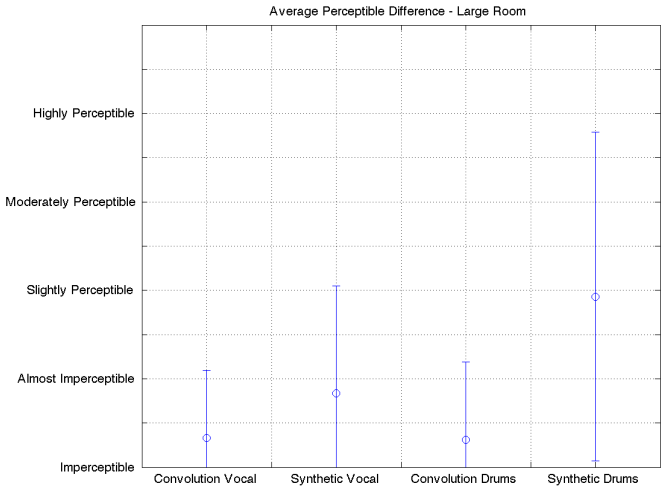

Figure 36: Scores - Large Room

perceptible". Incorrect identification was still present for the vocal stimuli. One interesting note from these results plots is that the standard deviation grows with the average perceived difference, which suggests that the upper end of the perceptual scale was more prone to individual interpretation.

4.2 Performance Evaluation

All of the previous results mean nothing if there are not benefits to using an FDN reverberator over convolution. The benefit, theoretically, is a CPU cycle and memory savings when using the FDN over convolution. This section presents a performance analysis of the two methods.

4.2.1 CPU Cycles

A performance test was designed to profile the relative processing power required to execute both convolution reverb and the FDN structure used in the current work in a realtime environment. The first consideration in a realtime

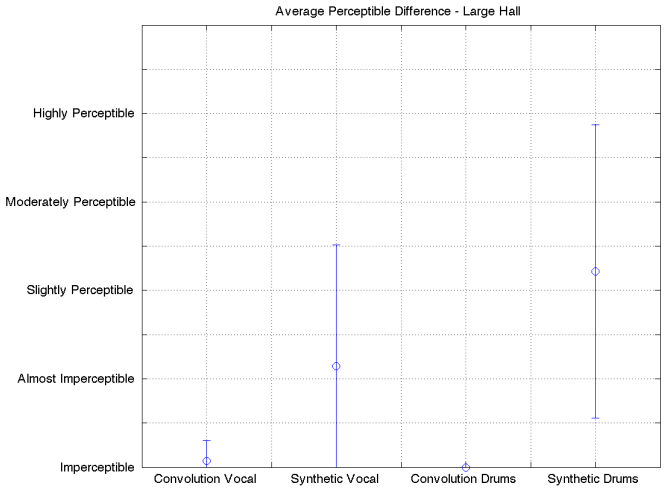

Figure 37: Scores - Large Hall

environment is that processing is generally done on a fixed-size buffer of data at a time—almost always a power of 2 in length. Smaller buffers equate to lower latency but generally higher CPU costs, and longer buffer lengths provide the opposite. The smallest buffer size in a modern digital audio workstation (DAW) is often 32 samples, but 64, 128, and 256 are commonly used. Longer buffer sizes such as 1024 or 2048 are usually used in situations where low-latency is not needed, such as in a digital playback system.

In a buffered audio system, fast convolution techniques exist to greatly speed up the convolution operation [Garcia, 2002] [Gardner, 1994]. Because of this, a comparison of the FDN method with direct time-domain convolution would be unfair. Therefore, an optimized FFT-based convolution module written in C++ was obtained on the internet and used [HiFi-LoFi, 2013].

A module for the FDN reverberator was written using Apple's Accelerate framework for vectorized math routines. The fast convolution module was used

for both FIR filters. A simple command line app was built around these modules to allow a user to pass in input/output audio file paths as well as the FDN parameters if necessary. The app processes the input audio file through the desired reverb algorithm in buffer sizes defined on invocation, measuring the total elapsed time spent in the process function, then prints this elapsed time on completion. The performance values are presented in terms of the CPU load percentage

$$\text{CPU load} = 100 \times \frac{\text{time spent processing}}{\text{realtime length of processed audio}} \tag{58}$$

which tells, in a realtime environment, the duty cycle that the CPU must devote to the processing task. For this test, a ten second mono audio file at a sampling rate of 44.1 kHz was used as test input.

This performance test was carried out for buffer sizes of 64, 128, 256, 512, and 1024. Three different length IRs were chosen to show the range of costs for the most common reverb IR lengths: one, three, and six seconds. Two FDN sizes were chosen, eight and 16, although 16 was used exclusively in the work until this point. Although performance gains can be had from the use of the Hadamard matrix, as discussed in Chapter 2, these were not taken advantage of; the standard matrix multiply operation from Apple's Accelerate framework was used to maintain generality. Early IR and tone correction FIRs were kept at 4096 samples as throughout this research.

The tests were carried out in succession on a MacBook Pro with a 2.4 GHz Intel i7 processor, and the results are shown in Figure 38. The most notable

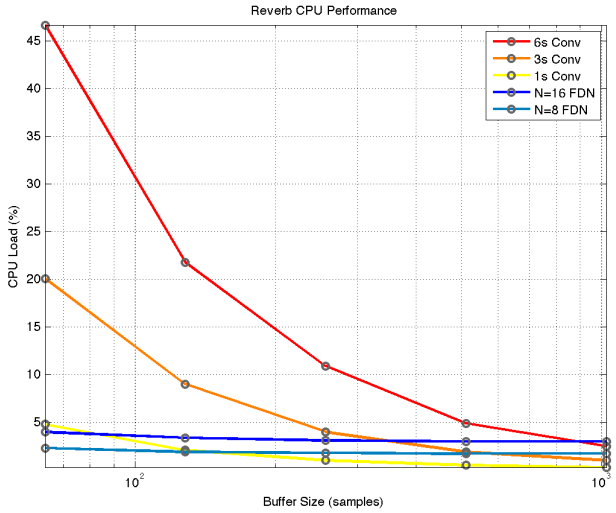

Figure 38: Comparison of CPU cost for convolution vs. FDN Reverb

difference between convolution and FDN reverberation is that convolution costs increase drastically as buffer size decreases, whereas the FDN cost remains nearly constant. This is due to the significant increase in frequency-domain multiply-accumulate operations incurred with fast convolution as buffer sizes decrease. On the other hand, the FDN structure is a recursive filter which must process on a sample-by-sample basis anyway, so the only benefit it gains when running at larger buffer sizes is the benefit its two FIR filters experience, just as with convolution reverb. However, these two filters are relatively short compared to convolution reverb lengths, so the decrease in frequency domain math with increased buffer size is small.

Clearly, FDN reverberation—even with $N = 16$—is the performance winner in low-latency situations for all but the shortest IRs. However, if the buffer sizes are sufficiently long, convolution does eventually match the FDN performance. If extremely long IRs are needed (greater than six seconds), an

FDN would be the best method since it's processing time does not depend on reverberation time.

4.2.2 Memory

The second way in which the FDN reverberator is more efficient than convolution reverb is in the memory size required to represent a single room's sound. If a large number of room sounds are to be available in a constrained memory environment, the FDN reverberator with the current design algorithm could be particularly useful.

It is assumed that the IRs for both the FDN and convolution reverbs are stored in 24-bit precision, and the remaining FDN parameter stored as 4-byte float values. Thus, the number of bytes required for convolution reverb with an IR of length T seconds is

$$M_{conv} = 3 f_s T \tag{59}$$

For the FDN, the **b**, **c**, and **m**, coefficients must be saved along with the decaying IIR filter coefficients, $h[n]$. For order k IIR filters, length L_e early reflections IR, and length L_{tc} tonal correction IR, the total number of bytes that must be stored is given by

$$M_{FDN} = 3(L_e + L_{tc}) + 4N(3 + 5\text{ceil}(\frac{k}{2})) \tag{60}$$

Figure 39 illustrates the difference in memory usage between convolution and the FDN structure used in this work ($k = 6$, $L_e = L_{tc} = 4096$). The savings is anywhere from 5x for a 1 s IR to over 50x for a 10s IR.

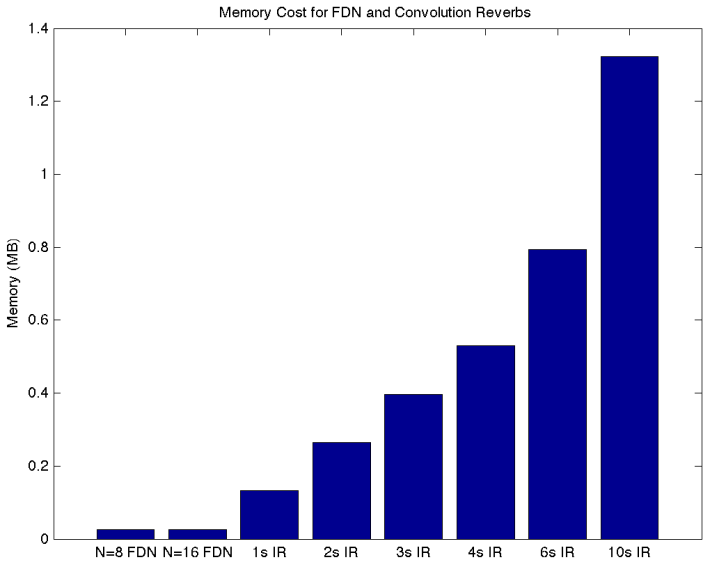

Figure 39: Memory Usage per room sound for FDN and convolution reverb at 44.1 kHz

5

Discussion

5.1 Listening Test Results

The listening test performed shows promising results. A few trends can be observed which were expected, and help validate the test. First, the drum kit stimuli really does expose aural differences more than the female vocal. For all three reverbs, the average perceived difference is a half point to two points higher for the drums than for the female vocals. Subjects often commented that they would focus on the kick and snare drum reverberation to pick out differences.

Another trend is that the average perceived difference increases with the reverberation time. This is straightforward to explain since the FDN is responsible for simply producing more sound for longer reverberation times, giving the listener a greater chance to notice differences along the decay. Even still, in the very worst case—the large hall FDN reverb on the drum kit—the average score was just "slightly perceptible". Ultimately, maybe the most impressive statistic is that only in that large hall/drum kit combination was every subject able to correctly identify the reference, which means that in all other cases, the target and synthetic reverbs sound similar enough to at least fool a handful of audio engineers listening critically.

5.2 Experimental Improvements

One weakness of any perceptually-based test is that the scoring rubric varies from subject to subject. How different do the target and synthetic have to

sound for the difference to be considered "moderately perceptible"? "Slightly perceptible"? "Almost imperceptible"? The judgement varies from subject to subject, and a look at the standard deviation of scores versus the average score for any given clip indicates that the variance in assigned scores grows with the score. In other words, test subjects tend to agree on what level of difference is "almost imperceptible", but a larger difference may be considered "slightly perceptible" by some and "highly perceptible" by others.

This points to the need for an anchor clip in the presentation, such as with a MUSHRA (Multiple Stimulus with Hidden Reference and Anchor) test. The anchor is a low-quality version of the reference that is blindly presented to the subject, and is usually a severely low-passed version of the reference. This helps to define the range in which subjects are to rate the clip under test—between the reference and anchor. Without this, only the lower bound of similarity is clearly defined.

Another shortcoming was exposed when one test subject took over twice as long to complete the test as the average, then reported scores that were significantly higher than any other test subject. The more time the subject has to scrutinize the recordings, the more obvious the differences may seem. A moderate time limit or play count limit on each test may better show how easily perceptible the differences really are.

5.3 Applications

The primary application of this work is in realtime audio processing environments such as music production or gaming—especially when low latency is desired. Reverb plugins used in DAWs generally ship with a bank of presets that can produce a range of sounds. A plugin could be developed around the current FDN reverberator and design method that could not only ship with a set of parameters optimized to various rooms, but also allow the user to easily run the design procedure on an IR of choice and generate a new preset.

Even more applicable though may be the use of this technique in gaming and virtual reality due to the large number of virtual spaces encountered. Designers of these environments only need to sample or find room IRs that are representative of the virtual space (or *are* the space if the space models a real location), then allow the design procedure to autonomously design parameters for the entire batch with no need for manual tuning. At runtime, switching between spaces could be as simple as fading between two instances of the FDN loaded with different parameters.

5.4 Future Work

One of the shortcomings of the $T_{60}(f)$ matching procedure that was noted for both the large room and large hall reverbs was the poor matching of T_{60} in the sub-150 Hz range. This was a result of the sharp increase in T_{60} for the target IRs in this range, which the least-squares based IIR design method could not match well. A possible improvement would be to design fourth order (as opposed

to sixth) IIR filters to the target $T_{60}(f)$ curve above 150 Hz, and design the remaining bi-quad as a low shelf filter, matching the gain, cutoff and Q to the low frequency bump using line-fitting techniques.

Optimization is also an obvious area for improvement. On average, an iteration of the GA—the work required to design and then compute the fitness for a single individual—is on the order of one to two seconds. This could be brought down significantly by leaving the MATLAB environment and using one the free genetic algorithm libraries available in a language like C++ [Wall, 2007]. Even within MATLAB, many additional features were computed and plotted at each iteration for debug purposes which aren't necessary to the core iterative procedure. The fact that good results are achieved after 20-40 generations of 100 individuals each, combined with the speed of the C implementation of the FDN, suggest that the iterative procedure could be performed in a much smaller time than 45-180 minutes per IR of the current work.

However, both of these improvements seem trivial in comparison to the obvious next step—stereo and multichannel reverberation. The monaural restriction of the current work helped to maintain focus, but in reality, monaural reverb lacks the sense of space created by stereo or multichannel reverb. One technique that has been used to attempt a stereo sound from an FDN reverberator has been to negate all the $\mathbf{c}$ coefficients on the right side with respect to the left. In this way, the increase in processing cost is minimal, but a certain level of decorrelation is gained, which widens the sound field

[Zahorik, 2009]. This method was tried as a final, post-iterative step in the current design. The sound was certainly no longer monophonic, but it did not produce a natural sound field. The effect was similar to listening to a pair of speakers or headphones with the polarity reversed on one side—it was difficult to localize the sound and was confusing to process.

However, the idea of only changing the **c** coefficients to create a stereo or multichannel reverberator is an appealing idea due to its small increase in memory and computational cost. One possible approach would be be to first compute the average IR sample-wise for all channels, run the current design method on that monaural IR to generate the usual parameters, then re-run just the GA portion for each channel's IR individually to generate the optimal **c** coefficients for that channel. The signal envelope fitness function may not be the best fitness measure for this portion and would need to be investigated. One possibility would be to optimize the two sets of **c** coefficients for a stereo reverberator simultaneously and use a fitness function based on the fit of the inter-aural cross correlation (IACC) of the two generated IRs to the target's IACC. This is based on the importance that IACC is known to have on the spatialization of a room [Zahorik, 2009].

5.5 Conclusion

The overarching goal of this work has been to capture the reverberation fingerprint of a space in a set of FDN reverberator parameters without the need for human intervention. Using this FDN representation of a room offers

significant memory and computational benefits over convolution reverb—*perceptual reverb compression*, if you will. Similar to perceptual audio compression techniques like mp3, a design procedure has been developed to take uncompressed room impulse response samples and capture the most salient portions into a compressed representation—a set of FDN reverberator parameters. But whereas coding techniques like mp3 incur a runtime cost (decoding) to gain a memory savings, the FDN reverb representation of an IR provides savings on both fronts.

We have designed as many parameters of the FDN as possible through analytic methods. The decaying IIR filters, $\mathbf{h}(z)$, are designed using Jot's work. The tonal corrective filter, $h_{tone}(z)$, is also designed analytically using the novel design technique proposed in this work. Other parameters that behave in complex and coupled ways—the scaling coefficients, $\mathbf{b}$ and $\mathbf{c}$, and the delay line lengths, $\mathbf{m}$—are found by using a genetic algorithm search. The optimal set is determined by using a fitness function which returns the maximum difference in the power envelopes of the target IR and the IR generated using the current set of genetic algorithm parameters. In this way, we steer the genetic algorithm to find sets of parameters which produce a similar time-domain evolution of the IR. Results have exceeded expectations and proven that the FDN structure and design procedure are capable of capturing room sounds in a way that is not only currently useful, but also highly extendible.

LIST OF REFERENCES

[Abel and Huang, 2006] Abel, J. S. and Huang, P. (2006). A simple, robust measure of reverberation echo density. In *Audio Engineering Society Convention 121*. Audio Engineering Society.

[Barron, 1995] Barron, M. (1995). Interpretation of early decay times in concert auditoria. *Acta Acustica united with Acustica*, 81(4):320–331.

[Barron, 2000] Barron, M. (2000). Measured early lateral energy fractions in concert halls and opera houses. *Journal of Sound and Vibration*, 232(1):79–100.

[Barron, 2005] Barron, M. (2005). Using the standard on objective measures for concert auditoria, iso 3382, to give reliable results. *Acoustical science and technology*, 26(2):162–169.

[Chemistruck et al., 2012] Chemistruck, M., Marcolini, K., and Pirkle, W. (2012). Generating matrix coefficients for feedback delay networks using genetic algorithm. In *Audio Engineering Society Convention 133*. Audio Engineering Society.

[Frenette, 2000] Frenette, J. (2000). *Reducing artificial reverberation algorithm requirements using time-variant feedback delay networks*. PhD thesis, University of Miami.

[Garcia, 2002] Garcia, G. (2002). Optimal filter partition for efficient convolution with short input/output delay. In *Audio Engineering Society Convention 113*. Audio Engineering Society.

[Gardner, 1994] Gardner, W. G. (1994). Efficient convolution without input/output delay. In *Audio Engineering Society Convention 97*. Audio Engineering Society.

[Gerzon, 1976] Gerzon, M. (1976). Unitary (energy-preserving) multichannel networks with feedback. *Electronics Letters*, 12(11):278–279.

[Gerzon, 1971] Gerzon, M. A. (1971). Synthetic stereo reverberation part i. *Studio Sound*, 13:632–635.

[Gerzon, 1972] Gerzon, M. A. (1972). Synthetic stereo reverberation part ii. *Studio Sound*, 14:24–28.

[Griesinger, 1989] Griesinger, D. (1989). Practical processors and programs for digital reverberation. In *Audio Engineering Society Conference: 7th International Conference: Audio in Digital Times*. Audio Engineering Society.

[Haas, 1972] Haas, H. (1972). The influence of a single echo on the audibility of speech. *J. Audio Eng. Soc*, 20(2):146–159.

[Heise et al., 2009] Heise, S., Hlatky, M., and Loviscach, J. (2009). Automatic adjustment of off-the-shelf reverberation effects. In *Audio Engineering Society Convention 126*. Audio Engineering Society.

[HiFi-LoFi, 2013] HiFi-LoFi (2013). Fftconvolver, audio convolution algorithm in c++ for real time audio processing. https://github.com/HiFi-LoFi/FFTConvolver.

[ISO, 1997] ISO (1997). 3382 acoustics—measurement of the reverberation time of rooms with reference to other acoustic parameters.

[Jot, 1992] Jot, J.-M. (1992). An analysis/synthesis approach to real-time artificial reverberation. In *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*, volume 2, pages 221–224. IEEE.

[Jot and Chaigne, 1991] Jot, J.-M. and Chaigne, A. (1991). Digital delay networks for designing artificial reverberators. In *Audio Engineering Society Convention 90*. Audio Engineering Society.

[Kraft, 2014] Kraft, S. (2014). mushrajs. https://github.com/seebk/mushraJS.

[Moorer, 1979] Moorer, J. A. (1979). About this reverberation business. *Computer music journal*, pages 13–28.

[Noren and Ross, 2001] Noren, K. V. and Ross, J. E. (2001). Analog circuit design using genetic algorithms. In *Second Online Symposium for Electronics Engineers*. Citeseer.

[Online, 2015] Online, R. S. (2015). Impulse responses. http://recordingschool.biz/homerecording/impulse-responses-p-87.html.

[Pirkle, 2012] Pirkle, W. (2012). *Designing Audio Effect Plug-Ins in C++: With Digital Audio Signal Processing Theory*. Taylor & Francis.

[Primavera et al., 2010] Primavera, A., Palestini, L., Cecchi, S., Piazza, F., and Moschetti, M. (2010). A hybrid approach for real-time room acoustic response simulation. In *Audio Engineering Society Convention 128*. Audio Engineering Society.

[Rocchesso and Smith, 1997] Rocchesso, D. and Smith, J. O. (1997). Circulant and elliptic feedback delay networks for artificial reverberation. *Speech and Audio Processing, IEEE Transactions on*, 5(1):51–63.

[Schroeder, 1962] Schroeder, M. R. (1962). Natural sounding artificial reverberation. *Journal of the Audio Engineering Society*, 10(3):219–223.

[Schroeder, 1965] Schroeder, M. R. (1965). New method of measuring reverberation time. *The Journal of the Acoustical Society of America*, 37(3):409–412.

[Smith, 2006] Smith, J. O. (2006). *Physical audio signal processing: For virtual musical instruments and digital audio effects*. \ url {http://ccrma. stanford. edu/jos/pasp/}.

[Stautner and Puckette, 1982] Stautner, J. and Puckette, M. (1982). Designing multi-channel reverberators. *Computer Music Journal*, pages 52–65.

[Stewart and Sandler, 2007] Stewart, R. and Sandler, M. (2007). Statistical measures of early reflections of room impulse responses. In *Proc. of the 10th int. conference on digital audio effects (DAFx-07), Bordeaux, France*, pages 59–62.

[Valimaki et al., 2012] Valimaki, V., Parker, J. D., Savioja, L., Smith, J. O., and Abel, J. S. (2012). Fifty years of artificial reverberation. *Audio, Speech, and Language Processing, IEEE Transactions on*, 20(5):1421–1448.

[Wall, 2007] Wall, M. (2007). Galib a c++ library of genetic algorithm components. http://lancet.mit.edu/ga/.

[Wang, 2013] Wang, R. (2013). Digital convolution – harvey mudd college e186 handout. http://fourier.eng.hmc.edu/e161/lectures/convolution/convolution.html. Accessed: 2015-2-12.

[Zahorik, 2009] Zahorik, P. (2009). Perceptually relevant parameters for virtual listening simulation of small room acoustics. *The Journal of the Acoustical Society of America*, 126(2):776–791.