HYBRID CONVOLUTION AND FILTERBANK ARTIFICIAL REVERBERATION ALGORITHM USING STATISTICAL ANALYSIS AND SYNTHESIS

By Rebecca Stewart

Supervised by Dr. Damian Murphy

Final Project Report Submitted in partial fulfillment of the requirements for the degree of Master of Science in Music Technology

> Departments of Electronics and Music The University of York York, England September 2006

ABSTRACT

The intent of this project was to create a hybrid artificial reverberation algorithm that uses elements of both convolution reverberation and filterbank reverberation. It was desired to have the accuracy and quality that a convolution reveberator provides, while introducing parametric controls and computational efficiency that have previously been limited to filterbank reverberators. This was accomplished by truncating an impulse response according to statistical analysis to contain only the early reflections, then combining the convolved audio with audio processed through a filterbank to simulate the late reflections. The parameters defining the filterbank were derived from an analysis of the impulse response being simulated. While a hybrid reverberator that is perceptually indistinguishable from a convolution reverberator was not created, the research shows that one is possible with further refinements in the derivation of filter coefficients for the filterbank.

ACKNOWLEDGEMENTS

I would like to thank my supervisors Damian Murphy and Jez Wells for their guidance and advice throughout my project. I am also grateful to my family, without whom I would not have had the strength and independence to travel across an ocean. Lastly, I want to thank Ben; your support improves the quality of my work and enhances my life.

I would also like to thank the Audio Engineering Society Educational Foundation for their financial support.

Table of Contents

| TABLE OF CONTENTS | | | | | | |
|-------------------|---|----|--|--|--|--|
| ТА | TABLE OF FIGURES II | | | | | |
| 1. | INTRODUCTION | 1 | | | | |
| 2. | OVERVIEW OF REVERBERATION | | | | | |
| , | 2.1 ACOUSTICS AND PSYCHOACOUSTICS | 3 | | | | |
| | 2.1.1 Accounted and 1 stenoneousness | 3 | | | | |
| | 21.1. Overview | 4 | | | | |
| | 213 Farly Reflections | 5 | | | | |
| | 2.1.4 Late Reflections | | | | | |
| | 2.1.5 Diffusion | | | | | |
| | 2.1.6 Absorption | 9 | | | | |
| , | 2.2 DIGITAL ARTIFICIAL REVERBERATION ALGORITHMS | 9 | | | | |
| | 2.2.1 Digital Filterbanks | | | | | |
| | 2.2.1.1 Schroeder Reverberator | | | | | |
| | 2.2.1.2 Moorer Reverberator | | | | | |
| | 2.2.1.3 Gardner Reverberator | | | | | |
| | 2.2.1.4 Feedback Delay Networks | | | | | |
| | 2.2.2 Impulse Response Convolution | | | | | |
| | 2.2.2.1 Collection of Impulse Responses | | | | | |
| | 2.2.3 Hybrid Algorithms | 20 | | | | |
| | 2.2.3.1 Early Research | | | | | |
| | 2.2.3.2 Browne | | | | | |
| | 2.2.3.5 Kadlord | | | | | |
| _ | | | | | | |
| 3. | REVERBERATION CHARACTERISTICS AND STATISTICS | | | | | |
| | 3.1 REVERBERATION TIME | 23 | | | | |
| | 3.2 EARLY AND LATE ENERGY RATIOS | 24 | | | | |
| | 3.3 INTERAURAL CROSS CORRELATION | 25 | | | | |
| | 3.4 ENERGY DECAY CURVES | | | | | |
| | 3.5 ENERGY DECAY RELIEFS | | | | | |
| | 3.6 LATE REFLECTIONS STATISTICS | | | | | |
| | 3.7 ECHO DENSITY ANALYSIS | | | | | |
| | 3.8 Assumptions and Faults | | | | | |
| 4. | HYBRID REVERBERATION ALGORITHM | | | | | |
| | 1 DESIGN CONSIDERATIONS | 33 | | | | |
| | 4.1 DESIGN CONSIDERATIONS | | | | | |
| | 4.1.1 Early Reflection Discrimination | | | | | |
| | 4.1.2 Lute Reflection Modelling | | | | | |
| | 4.1.5 System Input and Output | | | | | |
| | 4.1.4 Complete System | | | | | |
| | 4.2 ACOUSTIC MEASUREMENTS | | | | | |
| | 4.2.1 Selection of Acoustic Spaces | | | | | |
| | 4.2.2 Impulse Response Recording | | | | | |
| | 4.2.5 FOSI-FROCESSING | | | | | |
| | 4.2.4 Impulse Response Evaluation | | | | | |
| 4 | 4.5 EARLY KEFLECTION PARTITION | | | | | |
| | 4.3.1 1 IIFESHOW | | | | | |
| | 4.3.2 WITHOW SIZE | | | | | |
| | 4.3.3 I FUNCATION | | | | | |
| | 4.5.4 EValuation | | | | | |
| 4 | 4.4 FILUI STUDY WITH GARDNER KEVERBERATOR | | | | | |
| 4 | 4.5 ANALYSIS AND SYNTHESIS OF LATE KEFLECTIONS | | | | | |
| | 4.5.1 Derivation of Reverberation Times | | | | | |
| | 4.5.2 Derivation of Frequency-Dependent Reverberation Times | | | | | |
| | 4.5.5 LOW-PASS FILTER COEFFICIENTS | | | | | |

| 4 | 4.6 | FEEDBACK DELAY NETWORK | |
|----|--------|--|----|
| | 4.6.1 | Lossless FDN | |
| | 4.6.2 | Derivation of FDN Delay Lengths | |
| | 4.6.3 | Selection of Feedback Matrix | |
| | 4.6 | .3.1 Identity Matrix | |
| | 4.6 | .3.2 Stautner and Puckette Matrix | |
| | 4.6 | .3.3 Householder Matrix | |
| 4 | 4.7 | COMPLETE SYSTEM | |
| | 4.7.1 | Windowing and Delay | |
| | 4.7.2 | Stereo Output | |
| 5. | ANA | LYSIS OF RESULTS | 66 |
| | 5.1 | PILOT STUDY | |
| | 5.1.1 | Sir Jack Lyons Concert Hall | |
| | 5.1.2 | Trevor Jones Studio | |
| | 5.1.3 | Listening Comparisons | |
| | 5.1.4 | Summary of Pilot Study | |
| | 5.2 | COMPLETE SYSTEM | |
| | 5.2.1 | Sir Jack Lyons Concert Hall | |
| | 5.2.2 | Trevor Jones Study | |
| | 5.2.3 | Listening Comparisons | |
| | 5.2.4 | Summary of Complete System | |
| 6. | FURT | THER WORK | |
| 7. | CON | CLUSION | |
| RE | FEREN | CES | |
| AP | PENDIX | X A: ACCOMPANYING CD AUDIO FILE LISTINGS | 91 |
| AP | PENDIX | X B: HYBRID ALGORITHM MATLAB FILES | |
| AP | PENDIX | X C: FEEDBACK DELAY NETWORK SOURCE CODE | |
| | | | |

Table of Figures

| FIGURE 2.1 - AN IDEALISED IMPULSE RESPONSE OF A REVERBERANT SPACE. | 4 |
|--|------------|
| FIGURE 2.2 - THE PATH OF A DIRECT SOUND FROM THE SOURCE TO A LISTENER. | 4 |
| FIGURE 2.3 - A FEW POSSIBLE PATHS OF THE FIRST REFLECTIONS FROM THE SOURCE TO THE LISTENER | 6 |
| FIGURE 2.4 - A FEW POSSIBLE PATHS FOR LATE REFLECTIONS FROM THE SOURCE TO THE LISTENER | 7 |
| FIGURE 2.5 - BLOCK DIAGRAM OF AN ALL-PASS FILTER | 12 |
| FIGURE 2.6 - SCHROEDER REVERBERATOR. | 12 |
| FIGURE 2.7 - MOORER REVERBERATOR | 13 |
| FIGURE 2.8 BLOCK DIAGRAM OF THE STRUCTURE OF A GARDNER REVERBERATOR. ADAPTED FROM (BELTRÁN | J |
| and Beltrán, 1999) | 14 |
| FIGURE 2.9 - GENERAL FDN WITH J DELAY LINES | 15 |
| FIGURE 2.10 - JOT'S GENERAL FDN WITH ABSORBENT FILTERS HJ(Z) AND TONE CORRECTOR T(Z) | 16 |
| FIGURE 3.1 - STFT OF AN IMPULSE RESPONSE OF BERLIN PHILHARMONIC. FREQUENCY RESOLUTION 62.5 Hz, | |
| TIME RESOLUTION 16 MS. FROM (JOT, CERVEAU, AND WARUSFEL, 1997), FIG. 3 | 28 |
| FIGURE 3.2 - EDR OF BERLIN PHILHARMONIC WITHOUT NOISE REDUCTION. FROM (JOT, CERVEAU, AND | |
| WARUSFEL, 1997), FIG. 4 | 28 |
| FIGURE 3.3 - ILLUSTRATION OF THE GAUSSIAN NATURE OF LATE REFLECTIONS. FROM (ABEL AND BERNERS, 2004). | 29 |
| FIGURE 3.4 - THE TOP GRAPH SHOWS THE IMPULSE RESPONSE IN THE TIME DOMAIN. THE BOTTOM GRAPH IS THE | |
| CORRESPONDING ECHO DENSITY PROFILE. FROM (ABEL AND BERNERS, 2004) | 31 |
| FIGURE 3.5 - ECHO DENSITY PROFILE FOR THREE DIFFERENT SYNTHESISED IMPULSE RESPONSES WITH LESSENING | G |
| LEVELS OF 'DIFFUSION.' FROM (ABEL AND BERNERS, 2004). | 31 |
| FIGURE 4.1 - ON THE LEFT THE GENELEC S30D SPEAKER USED FOR THE SOUND SOURCE. ON THE RIGHT THE AH | KG |
| C414 AND THE FPC 900 USED TO RECORD THE EXCITATION SIGNAL IN THE SPACE. BEHIND THE | - |
| MICROPHONES ON THE WALL ARE THE ACOUSTIC PANELS | 38 |
| FIGURE 4.2 - T30 VALUES FOR SIR JACK LYONS ACCORDING TO FREQUENCY | 40 |
| FIGURE 4.3 - T30 VALUES FOR TREVOR JONES STUDIO ACCORDING TO FREQUENCY | 40 |
| FIGURE 4.4 - PROGRESSION OF HISTOGRAMS OF SAMPLE VALUES WITHIN A 20 MS WINDOW THROUGH THE SIR | |
| JACK LYONS CONCERT HALL IMPULSE RESPONSE | 42 |
| FIGURE 4.5 - THE SAME POINT IN THE SAME IMPULSE RESPONSE WINDOWED WITH RECTANGULAR AND HAMMIN | G |
| IN THE TOP GRAPH AND HANN AND BLACKMAN IN THE BOTTOM | 44 |
| FIGURE 4.6 - THE UPPER GRAPH SHOWS THE ECHO DENSITY PROFILE OF THE IMPULSE RESPONSE OF SIR JACK | |
| LYONS CONCERT HALL. THE UPPER LINE IS THE THRESHOLD AT 33.33% AND THE LOWER LINE IS 30%. T | HE |
| LOWER GRAPH SHOWS THE CORRESPONDING TIME DOMAIN. THE HIGHLIGHTED DATA POINT IS THE | |
| TRUNCATION POINT | 45 |
| FIGURE 4.7 - THE UPPER GRAPH SHOWS THE ECHO DENSITY PROFILE OF THE IMPULSE RESPONSE OF TREVOR JON | ES |
| STUDIO. THE UPPER LINE IS THE THRESHOLD AT 33.33% AND THE LOWER LINE IS 30%. THE LOWER GRA | PH |
| SHOWS THE CORRESPONDING TIME DOMAIN. THE HIGHLIGHTED DATA POINT IS THE TRUNCATION POINT | 46 |
| FIGURE 4.8 - ECHO DENSITY PROFILES FOR REVERB A REVERBERATION PLUG-IN. THE HIGHLIGHTED DATA POIN | VTS |
| ARE THE DETERMINED TRUNCATION POINTS. | 47 |
| FIGURE 4.9 - ECHO DENSITY PROFILES FOR ROOMWORKS REVERBERATION PLUG-IN IMPULSE RESPONSES. THE | |
| DETERMINED TRUNCATION POINT IS THE HIGHLIGHTED DATA POINT | 47 |
| FIGURE 4.10 - OVERALL STRUCTURE OF PILOT FUNCTION. | 49 |
| FIGURE 4.11 - EDC OF SIR JACK LYONS CONCERT HALL WITH THE LINE USED TO CALCULATE T_{30} | 50 |
| FIGURE 4.12 - STFT OF JACK LYONS IMPULSE RESPONSE USING 16 MS HANN WINDOWS. | 52 |
| FIGURE 4.13 - EDR OF JACK LYONS IMPULSE RESPONSE WITH 16 MS HANN WINDOWS. | 52 |
| FIGURE 4.14 - STFT OF TREVOR JONES IMPULSE RESPONSE WITH 16 MS HANN WINDOWS. | 53 |
| FIGURE 4.15 - EDR OF TREVOR JONES IMPULSE RESPONSE WITH 16 MS HANN WINDOWS. | 53 |
| FIGURE 4.16 - FREQUENCY-DEPENDENT T30 VALUES DETERMINED BY THE AURORA SOFTWARE | 54 |
| FIGURE 4.17 - T30 VALUES AGAINST FREQUENCY FOR SIR JACK LYONS CONCERT HALL AS DERIVED FROM THE | |
| EDR | 55 |
| FIGURE 4.18 - T30 VALUES AGAINST FREQUENCY FOR TREVOR JONES STUDIO AS DERIVED FROM THE EDR | 55 |
| FIGURE 4.19 - BLOCK DIAGRAM OF FDN FUNCTION | 57 |
| FIGURE 4.20 - OUTPUT FROM FOUR CHANNELS OF A 16 CHANNEL FDN WITH IDENTITY MATRIX | 61 |
| FIGURE 4.21 - COMPLETE SYSTEM OVERVIEW | 63 |
| FIGURE 5.1 - SIR JACK LYONS EARLY REFLECTIONS WITH GARDNER REVERBERATOR FOR THE LATE REFLECTION | NS |
| DEMONSTRATING THE EFFECTS OF INCREASED FEEDBACK GAIN | 66 |

| FIGURE 5.2 - SIR JACK LYONS EARLY REFLECTIONS WITH GARDNER REVERBERATOR FOR LATE REFLECTIONS |
|---|
| DEMONSTRATING INCREASING LEVELS OF GAIN FOR THE LATE REFLECTIONS |
| FIGURE 5.3 - FREQUENCY-DEPENDENT T_{30} values for pilot study on Jack Lyons impulse response. FB |
| DENOTES FEEDBACK GAIN WHILE LR DENOTES LATE REFLECTION GAIN |
| FIGURE 5.4 - FREQUENCY-DEPENDENT T_{30} values for second study of Jack Lyons impulse response69 |
| FIGURE 5.5 - COMPARISON OF C_{50} , C_{80} , and D_{50} in Pilot study on Jack Lyons impulse response70 |
| FIGURE 5.6 - FREQUENCY-DEPENDENT T_{30} values for Trevor Jones impulse response71 |
| FIGURE 5.7 - FREQUENCY-DEPENDENT T30 VALUES FOR SECOND STUDY OF TREVOR JONES IMPULSE RESPONSE72 |
| Figure 5.8 - Comparison of $C_{50},C_{80},$ and D_{50} in pilot study on Trevor Jones impulse response |
| FIGURE 5.9 - T30 VALUES FOR HYBRID SYSTEM WITH SIR JACK LYONS CONCERT HALL IMPULSE RESPONSE76 |
| FIGURE 5.10 - EDR of the impulse response created from the hybrid system with the Sir Jack Lyons |
| CONCERT HALL IMPULSE RESPONSE AND STAUTNER AND PUCKETTE MATRIX77 |
| FIGURE 5.11 – ACOUSTIC PARAMETERS FOR THE HYBRID SYSTEM WITH SIR JACK LYONS CONCERT HALL IMPULSE |
| RESPONSE |
| FIGURE 5.12 - T30 VALUES FOR HYBRID SYSTEM WITH TREVOR JONES STUDIO IMPULSE RESPONSE |
| FIGURE 5.13 - EDR OF IMPULSE RESPONSE OF THE HYBRID SYSTEM WITH THE TREVOR JONES STUDIO IMPULSE |
| RESPONSE AND STAUTNER AND PUCKETTE MATRIX79 |
| FIGURE 5.14 - ACOUSTIC PARAMETERS FOR HYBRID SYSTEM WITH TREVOR JONES STUDIO IMPULSE RESPONSE80 |

1. Introduction

Humans have a natural ability to perceive the geometry and size of a space without any visual cues. When the auditory cues that give an impression of space are absent, audio can sound undesirable. This was first experienced when microphones used to record the sound of an instrument or singer were placed near the sound source. The sound of the surrounding space was intentionally not captured as the signal could easily become muddled and loose clarity. In many situations, the recording venue was not an acoustically pleasing space, so the reverberant sound of the room was not wanted. This lead to recordings of 'dry,' non-reverberant sounds. Alternate ways were sought after of adding reverberation and a sense of space to the recordings after the initial recording session was finished.

There is perhaps no general effect used as consistently across all genres of recording and broadcasting as artificial reverberation. Engineering skills such as microphone placement can be refined to the point where equalisation is no longer needed, but no level of skill possessed by an engineer can change the architecture of a space. The capturing of that space can only be deliberately ignored if it doesn't meet the criteria of the recording.

As technology has evolved, techniques to simulate a space have been refined. Several acceptable techniques exist, and though they can use further refinement, they are considerably effective. Since the advent of digital reverberation algorithms, there have been two rather dichotomous methods. They each have specific benefits and are better fits for different applications, as one can accurately reproduce a particular space and the other can create a general impression of a space. However if an engineer or artist chooses to simulate a particular space, they cannot adjust the reverberation characteristics, and while general impressions can be controlled and varied with parametric controls. This project is working to refine and combine these two techniques in an effort to improve the quality, parameterisation, and computational efficiency of artificial reverberation.

1

This paper opens in Chapters 2 with an overview of the basic components of reverberation and the progression of algorithm development that has taken place over the past 50 years. Chapter 3 looks at various methods to predict and analyse the behaviour of sound in a room. Chapters 4 builds steps through the experimental design and implementation, Chapter 5 looks at the results of the developed system, Chapter 6 explores further work that can be done, and Chapter 7 draws together the conclusions.

2. Overview of Reverberation

When a sound occurs in an enclosed space, a listener experiences three different acoustical events with each having a specific effect on the perception of the space. The initial signal is the direct sound from the source, then the first set of reflections off of surrounding surfaces, and the third is the later set of reflections that are perceived as a single diffuse sound. All three parts, especially the latter two, are affected by the environmental factors of the space.

2.1 Acoustics and Psychoacoustics

The acoustics of an enclosed space refer to the physical geometry of the space and how sound propagates through it. Psychoacoustics refers to how the ear and brain interpret the resulting sound waves after they have passed through a space. The following sections discuss what information is conveyed by the sound propagation.

2.1.1 Overview

An idealised interpretation of the three sections of an impulse response of an acoustic space can be see in Figure 2.1. It should be emphasised that recorded impulse responses do not directly resemble the figure; it can be quite difficult to distinguish the various sections, as there are no clear boundaries, but rather gradual transitions.



Figure 2.1 - An idealised impulse response of a reverberant space.

2.1.2 Direct Sound

The direct sound is the first unadulterated wavefront that reaches the listener. Had the sound existed in a truly open space with no boundaries, it would be the only sound the listener would hear and could be measured and analysed as a free mean path (Angus and Howard, 2005:249). A free mean path is the idealised path of a sound without any interference such as reflections; it would exist in a completely open space with absolutely no boundaries in any direction, and could be compared to an outside open field.



Figure 2.2 - The path of a direct sound from the source to a listener.

The intensity of the direct sound can then be calculated using Eq. 2.1. If the direct sound is greatly altered timbrally or quieter than the subsequent reverberation, the source is perceived to be behind an obstacle and no longer existing in a free mean path.

$$I_{direct} = \frac{QW_{source}}{4\pi r^2}$$
 Eq. 2.1

where *I* is the intensity of the direct sound in watts m^2 , *Q* is the directivity of the source, *W* is the power of the source in watts, and *r* is the distance in m^2 .

The intensity and time difference between the direct sound and the following reflections gives the listener cues regarding the distance from the source. This is primarily due to the fact that the direct sound and early reflections are dependent upon location while the diffuse late field reflections are not (Howard and Angus, 2005:256). The ratio between the direct sound and the following reflected sound is one of the strongest spatial cues and can even override the cue from the intensity of the direct sound (Begault, 1994:89).

2.1.3 Early Reflections

The early reflections are the first set of reflections from the surrounding surfaces and differ in both time and direction from the direct sound (Howard and Angus, 2005:250). The differences are determined by the dimensions and materials that make up the space; they give a room the ability to alter the timbre of a sound as certain frequencies can be reinforced and filtered. The timbre and attack of musical instruments can be altered by the early reflections without affecting intelligibility (Begault, 1994:111), but musicians and others with discernible ears prefer a flat frequency response so that the timbre of the original signal is unaltered (Howard and Angus, 2005:321).



Figure 2.3 - A few possible paths of the first reflections from the source to the listener.

Though it is recognised as a fairly arbitrary number, it is generally accepted that the early reflections happen in the first 80 ms after the direct sound, while the late reflections occur afterwards (Begault, 1994:100). Ideally, a gap exists between the direct sound and the early reflections that does not exceed 30 ms; a bit of delay gives clarity, but a shorter one tends towards a feeling of 'intimacy.' If the reflections are spaced over 30 ms apart, then they may be heard at discrete echoes as opposed to reverberation (Howard and Angus, 2005:321).

Numerous sources give somewhat conflicting methods for distinguishing early from late reflections. The ISO standard defining acoustic measurements considers the early portion of the impulse response to be between 50 and 80 ms (ISO 3382) and Blesser defines the mixing time of a room to be three times the free mean path, while in the same article stating the mixing time is also the square root of the volume of the space (Blesser, 2001). Overall, most literature accepts the early reflections to end in the first 80 to 100 ms, but it is also conceded that this is a gross estimation. Certainly it can be concluded that there is not a time that is universal for all spaces; a number of variables are influential such as volume, surfaces, and geometry (Merimaa and Pulkki, 2005).

2.1.4 Late Reflections

The late field reflections are the diffuse portion of reverberation and are not dependent upon source or listener position. It has been suggested to use the same late reflection generator and separate early reflection generators to simulate several sources in one room (Jot, 1997) and has been implemented by (Merimaa and Pulkki, 2006).

Perceptually, the late reflections are directly related to the size of the room, but lack more specific spatial cues; the tail is statistically stochastic and does contain much perceptual information (Blesser, 2001).



Figure 2.4 - A few possible paths for late reflections from the source to the listener.

The late field reflections require a period of time to create enough density and intensity to be heard; this is determined by the size of the room (Howard and Angus, 2005:255). Once a reverberant field has been established, it can reach a steady state of reverberation, that is "[t]he steady state sound level, at a given point in the room, ...is an integrated sum of all the sound intensities in the reverberant part of the sound..." (Howard and Angus, 2005:257). This is a function of the amount of absorption in the room; louder rooms have a smaller amount of absorption (Howard and Angus, 2005:255).

The decay of the reverberant field is the amount of time for the reverberation to cease after there is no longer an input signal and is determined by the amount of energy absorbed at each reflection (Howard and Angus, 2005:256). In an idealised, modelled room, the late reflections have a smooth, exponential decay with no modal behaviour except for a slight boosting of bass frequencies, though this is not always observed in reality (Howard and Angus, 2005:321).

2.1.5 Diffusion

The geometry and surface materials of a room control the reflections that emanate throughout the space. Smooth, highly reflective surfaces absorb very little energy and reflect a large portion back into a space. If another smooth, reflective surface is parallel to the first, the sound waves bounce between the surfaces creating a sound called a flutter echo in which the echoes are spaced far enough apart that they don't colour the sound, yet close enough that a distinct pattern can be heard. Room modes also form from surfaces whose dimensions reinforce a frequency. Both flutter echoes and room modes are undesirable artefacts that can be placated with a proper amount of diffusion.

The late reflections in an impulse response are characterised by a diffuse sound that can be simulated as Gaussian white noise (Moorer, 1979). The denseness of the reflections and randomness of the energy gives a room a sense of size.

Both diffusion and reverberation are closely related to each other: the laws of reverberation can be formulated in a general way only for sound fields where all directions of sound propagation contribute equal sound intensities, not only in steady state conditions, but at each moment in decaying sound fields, at least in the average over time intervals which are short compared with the duration of the whole decaying process.

(Kuttruff, 1973)

Diffusion is the spreading of reflections throughout a space and it best done with irregularities. This can be irregularities in the architecture and dimensions of the space or in the materials on the surfaces. Larger spaces tend to be more diffuse as the sound waves do not lose energy as quickly as a small space where absorption from walls quickly deaden the reverberation.

2.1.6 Absorption

As discussed earlier, reverberation is a result of acoustic reflections off of surfaces defining a space. The absorption coefficient of a surface is the ratio of the energy of the reflection that is absorbed by the surface to how much is reflected.

Absorption coefficients are empirically measured and are represented by α . However, the absorption coefficient for a material is not constant across all frequencies, and since the reverberation is dependent on the absorption, reverberation is therefore frequency-dependent (Howard and Angus, 2005: 264).

Transmission through air also causes frequency-dependent attenuations. High frequencies are more susceptible to attenuation from humidity and impurities in the air than the lower frequencies (Howard and Angus, 2005:288). This attenuation is not related to any surface, but to the distance traveled in the air, so longer reverberation times tend to have more high-frequency attenuation.

It is important to model the effects of air absorption in order to create a more convincing artificial reverberator. Moorer (1979) first introduced the concept of using frequency-dependent filters to simulate the effects of air absorption and Jot (1991) further refined the technique.

2.2 Digital Artificial Reverberation Algorithms

Before the advent of digital technology, and especially before personal computers were made widely available, artificial reverberation was accomplished through electromechanical means. Initially, reverberation was recorded by placing a microphone inside the recording venue or in an adjacent room to record the natural reverberation of the space during the recording session. This could also be done at a later time by playing back a prerecorded signal over a loudspeaker in a room and recording the reverberation with a microphone at the opposite end

9

(Moura and Campos, 1957). This quickly became difficult to do when a large, acousticallypleasing space was not available; if such a space was available, it could not be altered in a significant way limiting the reverberation effect.

Smaller alternative devices with adjustable parameters were developed. While they varied in implementation, these devices also transmitted a signal across a resonant network with some kind of delay. The most popular method was using plates and springs (Moura and Campos, 1957), (Goodfriend and Beaumont, 1959). However, issues quickly arose when using metallic objects as sound is transmitted faster through metal than air. The resonances resulting from the metal are also further apart than what occurs in rooms, so the reverberation sounded 'metallic' and 'ringing' while the high frequencies did not decay as quickly without air absorption, so the reverberation sounded 'bright' (Howard and Angus, 2005:337).

Acoustic spaces can be described as a delay network and a possible effect of delay networks is comb filtering, but acoustically-pleasing rooms do not sound like comb filters. The difference is that the peaks created in a room are so dense and close to each in frequency, that they are no longer perceived as comb filters (Schroeder, 1962). While the delay network is the fundamental building block of artificial reverberation, it can be difficult to achieve the echo density needed to make a comb filter sound like a reverberant space.

These problems were largely overcome when digital systems became available. While there are several different approaches to artificial reverberation in the digital domain, there is not a single method unanimously preferred. Each approach has strengths and weaknesses and roughly falls into one of three categories: a mathematical model, a computer simulation, or a physical measurement of a real space (Blesser, 2001).

In a computer simulation such as image-source, a computer considers a space in terms of a finite number of surfaces. An algorithm can then be used to calculate all the possible images formed by rays extending from a source (Blesser, 2001). While this is not an efficient nor fast algorithm, it can become exceedingly complicated as the surfaces and their properties increase in complexity. It then does not model highly diffuse spaces very effectively, but can successfully imitate the early reflections and transition region to the late reflections. The ray-tracing method is very similar to the image-source method in that it follows the path of a sound wave through a series of reflections, however only a finite number of rays are calculated which end when they reach the listener (Blesser, 2001).

2.2.1 Digital Filterbanks

In the context of artificial reverberation, filterbanks are used to simulate reverberation usually based on perceptual models as opposed to physical models of a specific space. This makes them much more computationally efficient and flexible.

2.2.1.1 Schroeder Reverberator

Schroeder (1962) presented the first algorithm for overcoming the faults of electromechanical reverberators by using the all-pass filter. An all-pass filter consists of a combination of delayed and undelayed signal that results in a delayed version of the input signal without any change to the frequency response, but does greatly affect the phase response and adversely affects transient sounds (Moorer 1979). The zeros are the reciprocals of the poles as can be see in Eq. 2.2 and the block diagram can be seen in Figure 2.5.

$$h(t) = -g\delta(t) + (1 - g^2)[\delta(t - \tau) + g\delta(t - 2\tau) + ...]$$
 Eq. 2.2



Figure 2.5 - Block diagram of an all-pass filter.

Schroeder's method produced the needed echo density without colouration by placing parallel comb filters before the all-pass filters in series so that the resonances would be placed extremely close together like in a real room, as can be seen in Figure 2.6. Schroeder (1962) also noted that frequency-dependent gains could be added to his design to model the highfrequency attenuation found in large spaces like cathedrals.



Figure 2.6 - Schroeder reverberator.

In a later paper, Schroeder (1970) expanded his original design by using ray-tracing to create the early reflections and then used his earlier design for the late reflections.

2.2.1.2 Moorer Reverberator

In 1979, Moorer published a paper acknowledging the unique creation of synthesised music completely lacking environmental context. He refined and tuned Schroeder's model finding

the best combination of filters and altered the frequency response of the system by adding low-pass filters to simulate the air absorption. He intentionally did not boost the bass frequencies noting that it is not necessary in computer music (Moorer, 1979).

The network that Moorer (1979) found to work best for the late reverberation was six comb filters in parallel with first-order low-pass filters in the feedback loops summed and fed to one all-pass filter. This can be seen in Figure 2.7. He also decided that a nineteen-tap filter worked best for simulating the early reflections, though he was not completely satisfied with the quality (Moorer, 1979). The delay line was set up so that the last of the early reflections reached the output before the beginning of the late field reverberation for the all-pass and comb network (Zölzer, 2002:179).



Figure 2.7 - Moorer reverberator.

It is assumed that an all-pass filter has no audible effect on a signal, but this is only true for sufficiently short delay lines. Once the delay line is longer than the integration time of the ear, approximately 50 ms, the timbre can be affected (Zölzer, 2002:177). So a network of comb and all-pass filters can provide the necessary echo density and eigentone density, but may also add undesired colouration to the signal.

2.2.1.3 Gardner Reverberator

In 1992, Gardner proposed a reverberator structure using nested all-pass filters. It was found to greatly reduce the number metallic characteristic of previous implementations and use elements present in previous reverberator structures (Gardner, 1998:115). It was originally designed to be used as the reverberation engine in a virtual acoustic space, so it had three different implementations depending on the reverberation time: small, medium, and large (Gardner, 1992:55). Gardner recommends the small room simulation for 380 to 570 ms, medium room simulation for 580 ms to 1.29 s, and large room simulation for times greater than 1.30 s.

Figure 2.8 lays out the general structure of the reverberator, though there are slight variations in the small and medium room structures. The first nested all-pass filter, *Nested All-pass 1* in Figure 2.8, is a single nested all-pass filter with one all-pass filter followed by a delay inside an all-pass structure. The second, *Nested All-pass 2*, consists of two all-pass filters and one delay inside an all-pass structure (Beltrán and Beltrán, 1999).



Figure 2.8. - Block diagram of the structure of a Gardner reverberator. Adapted from (Beltrán and Beltrán, 1999).

The structure remains stable as long as the gain for all frequencies remains less than one. The output is not allpass, as there are cancellations in the summations from the taps, but the amplitude envelope can be controlled by the gain coefficients (Gardner, 1998:116).

2.2.1.4 Feedback Delay Networks

Stautner and Puckette were the first to fully develop a reverberation network composed of interconnected delay lines in a feedback loop (Stautner and Puckette, 1982). A generalisation of a network with *j* delays can be seen in Figure 2.9.



Figure 2.9 - General FDN with j delay lines

The network is highly flexible and allows for any number of channels. It also can be altered to incorporate filters to simulate air and surface material absorption (Stautner and Puckette, 1982). When delays of various lengths are used, the peaks and troughs are placed at irregular intervals, aiding in overcoming the 'metallic' characteristics of simple delays (Stautner and Puckette, 1982). Stautner and Puckette implemented this network for four channels using the feedback matrix G in Eq. 2.3.

$$G = \begin{vmatrix} 0 & 1 & 1 & 0 \\ -1 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \end{vmatrix} \cdot g \quad \text{where} \quad |g| < \frac{1}{\sqrt{2}} \qquad \text{Eq. 2.3}$$

The network can be extended to simulate early reflections and late reflections, however longer delay times can create some problems. The filters may resonate if pole density is not high enough and in general the colouration cannot be predicted; it can only be empirically tested (Stautner and Puckette, 1982).

In 1991, Jot proposed a method to further improve the reverberation techniques proposed by Schroeder and Moorer by reducing the resonances that give a reverberator its 'metallic' characteristic and building on Stautner and Puckette's concept of a feedback delay network (FDN). A general FDN with Jot's modifications can be seen in Figure 2.10.



Figure 2.10 - Jot's general FDN with absorbent filters hj(z) and tone corrector t(z).

Jot pointed out some weaknesses in earlier models and proposed solutions such as Schroeder's suggestion to have an echo density of 1000 echoes per second should be increased to 10,000 echoes per second. The problem of 'metallic' reverberation is eased by ensuring that all resonances in a narrow frequency band have the same decay time. A tone corrector that controls the decay characteristics and frequency response should be placed in series with the system (Jot and Chaigne, 1991).

Jot's earlier work was mostly interested in single-input-single-output systems such as in Figure 2.10, though worked later on spatialising outputs from a FDN (Smith, 2006). A FDN can also have multiple inputs and outputs.

With so many filters in the system, it is especially important for the system to be stable. The key to the stability is in the feedback matrix, a common design method for FDNs is to create a 'lossless' case by creating a system with an infinite impulse response similar to white noise. The individual frequency bands can then be adjusted with the desired reverberation time (Smith, 2006). It is then useful to implement a lossless matrix for the feedback matrix.

One of the most common matrices is the Householder Matrix developed by Jot in 1992. It has several unique features such as when the number of channels is not two, every channel is reflected into every other channel, and when the number of channels are a power of two, no multiplies need to be performed (Smith, 2006). A Householder Matrix is derived from Eq. 2.4 with an example of a 4x4 matrix in Eq. 2.5.

$$A_N = I_N - \frac{2}{N} u_N u_N^T \qquad \text{Eq. 2.4}$$

where $u_N^T = [1, 1, ..., 1]$ can be seen as the specific vector about which the input vector is reflected in *N*-dimensional space (Smith, 2006).

2.2.2 Impulse Response Convolution

When a system is linear and time-invariant (LTI), it can be represented accurately by its impulse response. Convolving an input signal with the impulse response of a LTI system has the equivalent output to the signal passed through that system. A room under most conditions can be approximated as an LTI system, meaning when an input signal is convolved with the impulse response of space, the resulting output sounds as if the signal had originally occurred in that space.

In Eq. 2.6, the output signal y(t) is the summation of random noise n(t) and the system F with input signal x(t). If F is a LTI system, then y(t) is equal to the convolution of the input signal and the impulse response of the system, h(t), plus the random noise n(t) as seen in Eq. 2.7.

$$y(t) = n(t) + F[x(t)]$$
 Eq. 2.6

$$y(t) = n(t) + x(t) \otimes h(t)$$
 Eq. 2.7

This method for artificial reverberation can overcome some of the issues present in other forms of artificial reverberation. The nuances, densities, and spatial cues of a space can all be accurately reproduced without a need to comprehend the details for synthesis. This can be especially useful for capturing the perceptually important early reflections of a space, but then limits the placement of the source and listener to precisely the locations used during the impulse measurement (Blesser, 2001). This is the main complaint and downfall of

convolution reverberation: the lack of parameterization. Methods based on digital filter banks allow for much more flexibility in choosing the sound of the reverberation; while convolution can accurately represent a space, it can only represent that single location within that space.

2.2.2.1 Collection of Impulse Responses

Convolution is only a valid approach when the requirements of a LTI system are satisfied. When non-linearities exist in the recording of the impulse response and the space, the requirements are no longer fulfilled.

The main two issues with recording impulse responses are dynamic range and timevarying sound velocity (Blesser, 2001). It is difficult to measure the reverberation time of a space because the reverberant sound can be quickly overwhelmed by ambient noise; this is true also when recording. The second problem exists because the speed of sound is highly dependent on air temperature, which is difficult to maintain throughout the length of the recording. This is often exasperated because multiple readings need to be taken in order to average out and eliminate ambient noise prolonging the time period (Farina 2000), (Blesser, 2001).

The speed of sound is proportional to the temperature, which can provide significant fluctuations in the speed of sound with relatively little temperature change. Blesser compares this effect to the visual observation of radiating heat waves on a hot summer day (2001). The system is simply not linear, particularly in large spaces with long reverberation times.

A second source of non-linearities is the sound source, most often a loudspeaker. A method developed by Farina (2000) has successfully overcome some of the non-linearities, particularly harmonic distortion, present in previous impulse measurement techniques. The technique has proved to be effective and popular in recent measurements of acoustical and historical spaces (Farina and Ayalon, 2003), (Ben-Hador and Neoran, 2004), (Murphy, 2005). Before Farina's work, the most common excitation signals for impulse response measurement

was Maximum Length Sequence (MLS) and Time-Delay Spectrometry (TDS) (Farina, 2000). The developed method uses a specific sine-swept signal to excite a space. Linear convolution with the inverse filter allows for the harmonics to be separated from the impulse response (Farina, 2000).

2.2.3 Hybrid Algorithms

Reverberation algorithms tend to either provide a highly accurate reproduction of a specific space, but do not allow for any adjustment to the reverberation, or they allow full control over the reverberation but lack the complete set of cues resulting in only a general impression of a generic space. Since in artistic settings a large numbers of easily manipulated parameters are desired, a more generic but highly adjustable algorithm is often preferred. This preference is further magnified by the immense processing power that is needed for convolution algorithms (Gardner, 1995). Initially, this prevented real-time processing as personal computers could not handle the load of convolution, but this has been gradually overcome with the combination of faster computers and more advanced convolution algorithms. A more current issue is the computational load created when multiple convolution engines are run inside a single program which can easily occur in digital audio workstation environments.

The following projects attempted to reconcile some of the faults of convolution reverberation algorithms.

2.2.3.1 Early Research

Before the advent of digital technology, the early reflections of a reverberant space were not simulated unless the reverberation was recorded from a room. Most electromechanical systems such as plates and springs only simulated the late reflections.

By 1979, Moorer had realized the significance of early reflections in reverberation and attempted to marry a method of creating early reflections to the recursive method he refined

from Schroeder. He used a basic image-source model to calculate the early reflections of a specific space after finding the multi-tap delay line lacking in any interesting spatial attributes (Moorer, 1979). However, the simplistic geometric models that were implemented did not sound appealing and the multi-tap delay line was resigned to be the best method until the modeling approach had evolved.

Some digitized impulse responses were studied, but they were not truncated and combined with recursive filters.

2.2.3.2 Browne

In 2001, Browne implemented a hybrid reverberation algorithm that combined impulse response convolution and filtering methods in order to decrease the computation time while creating a similar reverberation to convolution alone (Browne, 2001: 1). He found it advantageous to model a specific room by using early reflections from a impulse response as opposed to a multi-tap delay line as done in (Moorer, 1979).

A static truncation time of 150 ms was used for all impulse responses and a Moorer topology with empirically adjusted filters for air absorption was used for the late reflections. The results from ABX listening tests showed that the algorithm was transparent for some of the listeners, but not for most; it did, however, show viability for the concept (Browne, 2001: 73).

2.2.3.3 Radford

In 2003, Radford attempted to combine the strengths of both the perceptual approach using filterbanks and the physical approach using convolution and recursive filtering. He decided it was desired to match the output of the hybrid algorithm to the frequency response of the original impulse response, with a flat frequency response in the late field reflections. A feedback delay network was chosen to create the late reflections in order to provide maximum

flexibility and a high quality (Radford, 2003:41). The early reflections of a impulse response and a synthetic room were compared and used in combination with a recursive filterbank (Radford, 2003:58). The impulse response was windowed to approximately 80 ms using a Hanning window (Radford, 2003:61-62). The late reflections were created using a lossless FDN (Radford, 2003:43).

The most difficult problem that still needed development at the completion of the study was the transition from the impulse response containing the early reflections and the recursive filterbank. However, early reflections from the impulse response recorded from real spaces proved to be more successful that those created in a modelling program (Radford, 2003:81).

2.2.3.4 Merimaa and Pulkki

The hybrid technique developed by Merimaa and Pulkki differs greatly from the two previous projects, but is similar in that the overreaching goal was to give greater flexibility to an impulse response of a space. It does not involve any convolution, only synthesis, but the synthesis is driven by an energy analysis of an impulse response. The purpose of the algorithm is to use an impulse response to create a reverberant field over any arbitrary loudspeaker setup (Merimaa and Pulkki, 2005).

The authors noted that is it important to accurately reproduce the early reflections while the late reflections can be simulated by a statistically similar sound field. So an analysis distinguishes the highly directional early reflections from the diffuse portion and precisely synthesises the early reflections in a specific location using amplitude panning. A statistically similar diffuse field is simulated and distributed to all of the speakers (Merimaa and Pulkki, 2005). The amount of sound energy oscillating is then used to calculate the diffuseness, therefore allowing the directional, non-diffuse portion of the sound field to be extracted from the non-directional, diffuse portion (Merimaa and Pulkki, 2005).

22

3. Reverberation Characteristics and Statistics

Mathematical relationships between the dimensions of a space and the resulting reverberant field can be derived; while they are often idealized in comparison to real spaces, they can be used to model or predict the behaviour of sound in a space.

3.1 Reverberation Time

Several modifications have been made to the equations used to describe the reverberation time in a space since Sabine proposed his equation in the early 20th century, Eq. 3.1.

$$T_{60(\alpha<0.3)} = \frac{0.16V}{S\alpha}$$
 Eq. 3.1

where V is the volume, S is the surface area, and α is the average absorption coefficient of the space.

A more recently modified version of the calculation for reverberation time is the Norris-Eyring equation seen in Eq. 3.2.

$$T_{60} = \frac{-0.161V}{S\ln(1-\alpha)}$$
 Eq. 3.2

However, since reverberation is frequency dependent, separate calculations should be considered for differing frequency bands; the accepted bandwidth is the third octave, though an octave is the bandwidth commonly used for calculating absorption coefficients. Eq. 3.1 and Eq. 3.2 can be frequency dependent when an absorption coefficient for a specific bandwidth is used instead of the average absorption coefficient.

When an impulse response is taken of a space, the reverberation time can be derived even if the noise floor is above 60 dB. ISO 3382 measures reverberation time by finding the linear least-squares regression of the backwards integration of the squared impulse response. The time at -60 dB calculated from best fit line from 5 dB to 35 dB below the initial level is the measurement T_{30} . If the noise floor is too high, the regression can be taken from the 5 dB to the 25 dB points below the initial level which is then called T_{20} (ISO 3382).

3.2 Early and Late Energy Ratios

One of the most useful measurements to give an idea of the intelligibility of speech or music in a space is the clarity. Clarity is a ratio of early to late energy in an impulse response and is derived from Eq. 3.3. It was first proposed by Thiele as a measurement of clarity of speech in 1953 and later adapted for a measurement of clarity of music (Barron, 1993).

$$C_{t_e} = 10 \log \left(\int_{0}^{t_e} p^2(t) dt / \int_{t_e}^{\infty} p^2(t) dt \right) \text{ in dB}$$
 Eq. 3.3

where t_e is a time limit of either 50 or 80 ms and p is the impulse response. A time of 50 ms is usually used for speech and 80 ms for music. The ratio of the early energy to the total energy is 'Definition' or D_{50} and is directly derivable from C_{50} or by using Eq. 3.4. Definition is more commonly used for music, while clarity is used for speech (Barron, 1993).

$$D_{50} = \frac{\int_{0}^{0.050s} p^2(t)dt}{\int_{0}^{\infty} p^2(t)dt}$$
 Eq. 3.4

3.3 Interaural Cross Correlation

When two sounds arriving at each ear are coherent, the 'centre of gravity' of the source, where it appears to be located, is in a single event. When the sounds at each ear are not coherent at all, they appear to be two discrete events occurring in two locations (Blauert, 1997). The normalised interaural cross correlation function is a function of the correlation or cohersion between the signal at each ear and is defined in Eq. 3.5

$$IACF_{t1:t2}(\tau) = \frac{\left[\int_{t1}^{t^2} p_l(t) \cdot p_r(t+\tau)dt\right]}{\left[\int_{t1}^{t^2} p_l^2(t)dt\int_{t1}^{t^2} p_r^2(t)dt\right]^{1/2}}$$
Eq. 3.5

where $p_l(t)$ is the impulse response at the entrance to the left ear canal and $p_r(t)$ is the impulse response at the entrance to the right ear canal (ISO 3382). The interaural cross-correlation coefficients are the maximum of *IACF* +/- 1 ms to account for interaural time delay. They give a measurement of the spatial impression of a space and are found by Eq. 3.6.

$$IACC_{t1t2} = \max |IACF_{t1t2}(\tau)|$$
 for $-1 \text{ ms} < \tau < +1 \text{ ms}$ Eq. 3.6

The most general form of *IACC* is when $t_1=0$ and $t_2 = \infty$ and a wide frequency band, early reflections are measured with $t_1=0$ and $t_2 = 80$ ms, and late reflections $t_1 = 80$ ms and t_2 equals a time greater than the reverberation time. Monaural measurements usually are implemented in octave bands ranging from 125 Hz to 4,000 Hz (ISO 3382).

3.4 Energy Decay Curves

Schroeder developed a method to more accurately derive the reverberation time of a room (1965). Previous methods involved averaging multiple impulse responses together, but many problems still persisted, the randomness of the excitation signal could cause fluctuations in an impulse response. It was difficult if not impossible to detect multiple decay rates, and noise easily interfered with an accurate measurement (Schroeder, 1965).

Schroeder proved that the ensemble average of the squared noise decay is equal to an integral of the squared impulse response through a band-pass filter (Schroeder, 1965). The definition of an energy decay curve can be seen in Eq. 3.7. It shows that the $EDC_h(t)$ equals the remaining energy in the impulse response after time *t*.

$$EDC_{h}(t) = \int_{t}^{+\infty} h(\tau)d\tau$$
 Eq. 3.7

The reverberation time can then be found as discussed in Section 3.1 by finding the linear regression between two points on the curve. The chosen points are usually noted in the subscript of the value (e.g. T_{30} or T_{20}).

3.5 Energy Decay Reliefs

An energy decay curve can be created for a certain bandwidth of a signal, and is often analysed in octave or third-octave bandwidths. However, this is not enough frequency resolution for a thorough analysis, so Jot (1991) extended the energy decay curve into an energy decay relief (EDR).

A time-frequency representation of Schroeder's EDC can be extended from the impulse response h(t) into the distribution function ρ as seen below.

$$h(t) \rightarrow \rho_h(t, f)$$
 Eq. 3.8

An EDR is then subject to three definitions (Jot, Cerveau, and Warusfel, 1997). Definition 1:

If $\rho(\tau, f)$ is an energetic time-frequency representation of the signal h(t), EDR is defined by Eq. 3.9 (Jot, Cerveau, and Warusfel, 1997).

$$EDR_{h}(t,f) = \int_{t}^{\infty} \rho_{h}(\tau,f) d\tau \qquad \text{Eq. 3.9}$$

Definition 2:

The EDR is defined as the ensemble average of the time-frequency representation of the reverberation decay after an excitation signal, if the signal is stationary white noise (Jot, Cerveau, and Warusfel, 1997).

Definition 3:

The EDR describes the spectral energy density remaining after any time in the signal, and coincides with the future running spectrum.

$$EDR_{h}(t,f) = \left| \int_{\tau=t}^{+\infty} h(\tau) e^{-j2\pi f\tau} d\tau \right|^{2} = \int_{\tau=t}^{+\infty} P_{h}^{+}(\tau,f) d\tau \qquad \text{Eq. 3.10}$$

where P^+ is the Page distribution. If ρ is required to preserve frequency-domain marginal distribution, be anticausal, and preserve the causality and shifting of signals in the time domain, this distribution is the only solution ρ .

High frequencies and noise are present in a STFT of an impulse response as can be seen in Figure 3.1. Extracting data such as reverberation time is error-prone; taking the EDR of an impulse response provides a much smoother result to be fit with linear regressions to find the reverberation time. Notice the smoothness in the spectrum of Figure 3.2 as compared to the STFT in Figure 3.1. They are both measurements of the same impulse response.



Figure 3.1 - STFT of an impulse response of Berlin Philharmonic. Frequency resolution 62.5 Hz, time resolution 16 ms. From (Jot, Cerveau, and Warusfel, 1997), Fig. 3.



Figure 3.2 - EDR of Berlin Philharmonic without noise reduction. From (Jot, Cerveau, and Warusfel, 1997), Fig. 4.

3.6 Late Reflections Statistics

The late reflections of an impulse response fit the curve of a normal distribution, which is described by Eq. 3.11. The value μ is the mean of the population and σ is the standard deviation. These two values alone determine the unique normal curve.

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} -\infty < x < \infty$$
 Eq. 3.11

The late reflections of an impulse response result from a mixed room where energy is theoretically coming from every angle while the early reflections have energy that is concentrated from particular reflections. Figure 3.3 shows this progression towards a more normal distribution as time increases and the room becomes more mixed.



Figure 3.3 - Illustration of the Gaussian nature of late reflections. From (Abel and Berners, 2004).
Kurtosis is the upward peak in the distributions in Figure 3.3 that are less normal (Abel and Berners, 2004). One of the first to compare late reflections with Gaussian noise was Moorer (1979). Since then, it is a common engineering technique to approximate Gaussian noise in filterbank-based artificial reverberators before finely tuning the frequency response. Schroeder also did a large amount of study on the statistical nature of impulse responses.

When Schroeder derived a set of statistical properties about the frequency response of the random impulse response, he was extending Sabine's fundamental approach. The central limit theorem forces the composite sine and cosine components at any frequency to be a Gaussian distribution, and the magnitude to be a Rayleigh distribution. Because Gaussian statistics have been extensively studied, there is a large body of mathematical data that can be applied. Schroeder explored a few of the more useful metrics. (Blesser, 2001).

3.7 Echo Density Analysis

While most of the statistical research regarding impulse responses has been focused on the statistical nature of late reflections, these statistical characteristics can be used to distinguish the early reflections from the late as shown in Figure 3.4. In particular, the late reflections have a relatively normal distribution while the early tend towards a less normal distribution. The early reflections are more kurtoic (Abel and Berners, 2004).

A measure of the echo density of an impulse response over time is the number of samples per window that are outside one standard deviation (Abel and Berners, 2004). In a normal distribution, approximately 68% of the of population is within one standard deviation of the mean (Lapin, 1990). So a window of an impulse response that has less than about 33% of the samples inside one standard deviation is more likely to be a window of early reflections, while a window with about 33% of the samples outside is more likely to be a window of the mixed late reflections. This can be seen in Figure 3.4 where the bottom graph is the time is displayed versus the echo density profile in which about 30% is normalised to one (Abel and Berners, 2004).



Figure 3.4 - The top graph shows the impulse response in the time domain. The bottom graph is the corresponding echo density profile. From (Abel and Berners, 2004).

Figure 3.5 shows a comparison of three impulse responses generated from an artificial reverberator with progressively decreasing levels of diffusion. The first and second most mimic the echo density profile of an existing acoustic space.



Figure 3.5 - Echo density profile for three different synthesised impulse responses with lessening levels of 'diffusion.' From (Abel and Berners, 2004).

3.8 Assumptions and Faults

The previous equations and relationships are based on the assumption that the energy in the reverberant field is statistically random, however this can easily be false. If the shape of the room approaches that of a tube or if multiple rooms are connected together, reverberation fields can mix and these assumptions are no longer true (Howard and Angus, 2005:274). Eq. 3.1 and 3.2 are based on the assumptions that the sound field is highly diffuse and that the concept of a mean free path is valid (Howard and Angus, 2005:269), but real spaces can exist without these assumptions being fulfilled, for example in small, absorptive rooms.

Calculations for reverberation time also fail when the space is extremely small and with a fair amount of absorption; the early reflections become absorbed before the energy is dense enough to create diffuse late reflections (Howard and Angus, 2005:271). It can also be difficult to determine a single representative reverberation time when multiple spaces are coupled as different reverberation times will be manifested throughout the impulse response. The calculated value in the measurement technique used in (ISO 3382) would then be highly varied according to the points in time selected for a fitting linear regression.

4. Hybrid Reverberation Algorithm

4.1 Design Considerations

In hybrid reverberation algorithms, the early reflections are ideally preserved in an impulse response and recreated through convolution while the late reflections are synthesised through a filterbank. Previous implementations have made the assumption that any impulse response can be truncated at a single point in time, regardless of the characteristics of the space, without significantly compromising the integrity of the early reflections. This study explores the implementation of a truncation point that is determined by specific statistical characteristics of an individual impulse response.

It is also deemed important that the reverberation created with this hybrid system be perceptually indistinguishable from audio processed only by convolution with the original impulse response. A hybrid system can replace a convolution system only when it completely fulfills its functionality. Once that criteria is met, the flexibility of a filterbank with parameters can be applied to change the system response as deemed necessary.

4.1.1 Early Reflection Discrimination

Past studies have had difficulties implementing a hybrid algorithm that approaches the quality of either a convolution or filterbank reverberator (Browne, 2001), (Radford, 2003). Radford (2003) attributed this to the transition between the convolution portion and the filterbanks. However, in both studies a fairly arbitrary truncation point was used in order to optimise block convolution instead of considering the preservation of the early reflections. It was assumed that all early reflections occur within the first 80 ms, but as discussed in Chapter 2, this is not the case.

It is the hypothesis here that a selection of the truncation point based on statistical evaluation of the end of the early reflections will create a more perceptually accurate hybrid algorithm. By evaluating and then truncating, spaces with longer early reflections will not lose information and spaces with shorter early reflections will devote less computationally-expensive time to convolution.

As Blesser (2001) has noted, "[t]he use of stochastic metrics is not obvious to audio engineers because artificial reverberation is the deterministic output from a signal processing system." The novelty of this project is the use of statistics to determine the timing of early reflections and then applying parameters derived from the late reflections to a filterbank.. This divides the impulse response into a statistically deterministic portion to undergo convolution, and a statistically stochastic portion to be simulated through filterbanks.

4.1.2 Late Reflection Modelling

If great lengths are taken to accurately reproduce the early reflections of a space, the late reflections should not be disregarded by using a generic reverberator. Though the early reflections contain most of the specific cues needed for a listener to interpret a space, the late reflections also contain important information. The reverberation time over specific frequency bands contains most of the cues needed to give an impression of the size of a space.

Jot developed an analysis/synthesis system which analyses an impulse response, reduces the effects of noise in the recording, and derives a set of coefficients representing reverberation time for a set of frequency bands (Jot, 1997). The system is only valid for the mixed portion of the impulse response, which is precisely what it is needed for in this project. A similar system that analyses an impulse response and derives coefficients that can be applied to a filterbank is ideal.

4.1.3 System Input and Output

An infinite number of combinations of impulse response channels and dry audio channels can be combined to create any number of outputs. However, the simplest combination is a mono impulse response convolved with a mono dry recording, so this is the combination studied here. Testing and development will concentrate on one channel of input and impulse response with one channel of output, but will be expanded to support two output channels.

When a stereo or mono audio signal is convolved with a stereo impulse response, the outputs have the same relationship to each other as the inputs. That is the stereo image is determined by the convolution. When only the early reflections are used from an impulse response, the stereo image needs to be intentionally created for the late reflections. Each channel should not have the exact output from the filterbanks.

The decorrelation between the signals reaching the two ears increases the perception of spaciousness (Ben-Hardor and Neoran, 2004). The spaciousness can be controlled through IACC adjustments as first described by Schroeder and Jot and as summarised by Gardner (1998:111).

4.1.4 Complete System

A problem discussed in earlier hybrid algorithms is the connection between the convolved signal and the signal from the filterbanks. The main decision is between a parallel connection and a serial connection. Radford (2003) found that a parallel connection was more successful than a serial, as the integrity of the frequency response of the late reflections is maintained. Otherwise the convolved early reflections can colour the signal before it enters the filterbanks.

Echo density is also an issue. If the filterbank emulating the late reflections is not properly tuned, discrete echoes may be distinguished in the signal before a realistic density is built up. This can be compensated using appropriate amounts of delay between the processing.

35

4.2 Acoustic Measurements

4.2.1 Selection of Acoustic Spaces

Two spaces were chosen for measurement and modelling; both are acoustically-designed spaces within the Department of Music at the University of York. Trevor Jones Studio is a recording space within the Music Research Centre with adjustable acoustic panels. They can be opened to maximally cover the painted cinder-block walls, exposing absorptive material, or they can be closed exposing the wall and a reflective surface on the back side of the panel. They can be seen in the photograph on the right in Figure 4.1. Since it is a medium-sized room with carpeting, and a longer reverberation time is desired for modelling, the panels were closed.

The second space measured was Sir Jack Lyons Concert Hall. It is a larger, but still medium-sized performance space, that holds 350 seats. It is a highly diffuse room with many reflective surfaces including the seats, walls, and floors. It houses a pipe organ and is used for classical and jazz performances.

4.2.2 Impulse Response Recording

The swept-sine wave method developed by Farina (2000) was used to record the impulse responses. The method consists of recording an excitation in a space, then convolving that recording with the inverse filter of the excitation signal. As explained in (Farina, 2000), the harmonic distortions are separated from the impulse response and appear preceding the impulse response in time. They can then be removed by using a digital audio editor.

The excitation signal is a log-sweep created by Eq. 4.1. The sweep has the advantage of its inverse being the sweep in reverse order (Ben-Hardor and Neoran, 2004).

36

$$x(t) = \sin\left[\frac{w_1 T}{\ln\left(\frac{w_2}{w_1}\right)} e^{\frac{t \cdot \ln\left(\frac{w_2}{w_1}\right)}{T-1}}\right]$$
Eq. 4.1

T is the total time of the sweep and w_1 and w_2 are range the of the frequencies in the sweep. Previous measurements, (Farina, 2003), (Ben-Hardor and Neoran, 2004), had found that a length of 15 s works best for the length of the sweep, so the same length was used.

Two microphones were selected for the recordings, though only one impulse response recording from each space would be used for testing and development. Multiple takes in each space were also recorded so that a the best impulse response could selected.

The purpose of capturing the impulse responses of these two spaces was not to create a high-quality impulse response suitable for commercial use, but to create files that would be ideal for modelling and development. Therefore, single omni-directional microphones were chosen for the recording. All acoustical information from the room was needed for initial tests, having information rejected by a polar pattern was not desirable. An AKG C414 and a BBS Audio FPC 900 microphones were used along with a RME Fireface as the external soundcard plugged into an Apple PowerBook G4 running Nuendo 2.0 on OS X. The sound source played through the digital input on a Genelec S30D speaker. Both the speaker and microphones were 1.5 m off the floor, though in the Sir Jack Lyons Concert Hall the microphone was 1.5 m off the floor of the 5th row, which was raised above the stage level. All recordings were 96 kHz and 24 bit.



Figure 4.1 - On the left the Genelec S30D speaker used for the sound source. On the right the AKG C414 and the FPC 900 used to record the excitation signal in the space. Behind the microphones on the wall are the acoustic panels.

The positioning of the source and receiver in Sir Jack Lyons Concert Hall was to mimic a performance with an audience. The speaker was placed on-stage and the microphones were placed in the centre of the 5th row, approximately the middle of the seating area. Since there is not a designated source and listener position in Trevor Jones Studio as it is only a recording space, the speaker and microphones were placed at opposite ends of the room.

4.2.3 Post-Processing

The convolution of the recordings and the inverse filter were done in MATLAB. The MATLAB script used was modified from (Ponikowski, 2006). Some minor changes were made as the M-file was written for 72 impulse responses recorded in 360 degrees and one mistake was corrected in the normalising algorithm.

The harmonic distortions and excess silence at the beginning and end of the recordings were removed in Nuendo. The sample rate was down-sampled to at 44.1 kHz and the bitdepth was kept at 24 bits. Fade-ins and fade-outs were inserted at the beginning and end of the files. The direct sound was also removed from the impulse response. This is important for applications using a small amount of reverberation, there is then minimum influence on the input audio (Ben-Hador and Neoran, 2004). It also allows the user to choose a pre-delay time different from the original space.

4.2.4 Impulse Response Evaluation

Four impulse responses were recorded in Sir Jack Lyons Concert Hall on two microphones and three were recorded in Trevor Jones Studio on the same two microphones. One impulse response for each space was selected to find the most average impulse response that best represented the space. This was determined by comparing the T_{30} and T_{20} over octave bands for each impulse response as derived by the Aurora tools for Adobe Audition written by Farina. The T_{30} values for each space can be seen in Figure 4.2 and 4.2. This was done before the direct sound was removed; after an impulse response for each space was selected, the final editing was applied.

T30 for Acoustic Measurements in Sir Jack Lyons Concert Hall



Figure 4.2 - T30 values for Sir Jack Lyons according to frequency.



T30 for Acoustics Measurements for Trevor Jones Studio

Figure 4.3 - T30 values for Trevor Jones Studio according to frequency.

An impulse response selection for Sir Jack Lyons Concert Hall was fairly arbitrary as all of the measurements were extremely close to each other. The results for Trevor Jones were not so uniform, so a measurement that best fit the general curve was chosen. By chance, the second impulse response recorded on the FPC microphone in both spaces was chosen. These are the two impulse responses used in the development and evaluation of the hybrid reverberation system.

4.3 Early Reflection Partition

A MATLAB function was written to partition the early reflections from the late and then return a truncated array holding the data only pertaining to the early reflections. The impulse response was first windowed and then the standard deviation and mean of each window was calculated. The number of samples that were outside one standard deviation of the mean was tallied and stored in an array. The first window whose percentage of samples outside one standard deviation met a determined threshold and was initially selected as the truncation point distinguishing the early from late reflections.

In order for a point in time to be selected as the truncation time, a threshold needed to be selected. A truly normal distribution would have exactly 33 1/3% of the samples of a population outside one standard deviation. Any amount above or below that measure would not be completely normal, but as seen in Chapter 3 and in Figure 4.4, early reflections tend to be more kurtoic. This means the samples in early reflections tend to be within one standard deviation, lowering the percentage outside.



Figure 4.4 - Progression of histograms of sample values within a 20 ms window through the Sir Jack Lyons Concert Hall impulse response.

4.3.1 Threshold

(Abel and Berners, 2004) chose about 30% as their threshold, so both thresholds of 30% and 33 1/3% were investigated. The horizontal lines in Figures 4.6 and 4.7 mark both thresholds. It can be see that the threshold of 33 1/3% is seldom reached, while 30% seems to be reached at point relative to the end of the early reflections. But neither number seems to be universally appropriate, for while the different impulse responses have a similar shape and seem to curve upwards until reaching an asymptotic boundary, the exact values vary. A local maxima around the transition between the curve and asymptotic portion seems more appropriate.

Since a threshold of 30% was consistently reached, it was chosen for an initial threshold. Then the next 60 ms were examined and the local maximum found was chosen as the time of the end of the early reflections. This system allowed some flexibility between differences in rooms and acoustic measurements and errs on the side of including more late reflections rather than cut short the early reflections.

4.3.2 Window Size

Along with a threshold, an appropriate window size needed to be chosen. Larger window sizes allow for smoother curve by averaging out local variations, but smaller windows allow for a more exact examination of the progression of the distribution.

The smoother curve of the larger window size of 20 ms was chosen as favourable. A smaller window could cause the echo density plot to reach the threshold prematurely because of a small portion within the early reflections that tended towards a normal distribution. This was also the window size used by (Abel and Berners, 2004).

4.3.3 Truncation

Once a time at the end of the early reflections was determined, the impulse response needed to be truncated so that only the early reflections were used in convolution. However, truncating a signal can introduce unwanted frequencies into the spectrum of the truncated signal. A method of minimizing this effect is windowing. The most important consideration in windowing an impulse response in this project is the introduction of high frequencies to the truncated impulse that were not present in the original. It is granted that an amount of high frequencies will be added, but it is in the interest of the window type to select the window that minimises it.

An impulse response was windowed with the four window types: rectangular, Blackman, Hamming, and Hann. Only the falling edge of the window was applied to the impulse response, so only the last 32 points of the total 64 points. The frequency domain of the results can be seen in Figure 4.5.

43



Figure 4.5 - The same point in the same impulse response windowed with rectangular and Hamming in the top graph and Hann and Blackman in the bottom.

From Figure 4.5, it can be seen that the Hann and Blackman had the best results as the high frequencies were about 200 dB below the rest of the impulse response. The rectangular window had the worst response, while the Hamming had an undesirable response with large fluctuations in the highest frequencies. The equation describing the Hann window can be seen in Eq. 4.2 and for the Blackman window in Eq. 4.3.

$$w[k+1] = 0.54 - 0.46 \cos\left(2\pi \frac{k}{n-1}\right)$$
 where $k = 0, ..., n-1$ Eq. 4.2

$$w[k+1] = 0.42 - 0.5 \cos\left(2\pi \frac{k}{n-1}\right) + 0.08 \cos\left(4\pi \frac{k}{n-1}\right)$$
Eq. 4.3
where $k = 0, \dots, n-1$

As both windows gave a very similar result, the simpler window, the Hann window, was chosen for the impulse response truncation algorithm.

4.3.4 Evaluation

The echo density profiles and calculated truncation points for the measured acoustic spaces can be seen in Figures 4.6 and 4.7. The truncation point for Sir Jack Lyons Concert Hall at 141 ms was much greater than the estimated 80 ms usually assigned to early reflections. Trevor Jones Studio was much shorter at about 41 ms.



Figure 4.6 - The upper graph shows the echo density profile of the impulse response of Sir Jack Lyons Concert Hall. The upper line is the threshold at 33.33% and the lower line is 30%. The lower graph shows the corresponding time domain. The highlighted data point is the truncation point.



Figure 4.7 - The upper graph shows the echo density profile of the impulse response of Trevor Jones Studio. The upper line is the threshold at 33.33% and the lower line is 30%. The lower graph shows the corresponding time domain. The highlighted data point is the truncation point.

The truncation points of impulse responses of two artificial reverberators were explored. Both were VST plug-ins that are included with Steinberg products; they were used in Nuendo

3.

Reverb A is as basic of a reverberator as its name. It has only six parameters: mix, room size, predelay, reverberation time, filter highcut, and filter lowcut. Figure 4.8 shows the echo density profiles and truncation points of the impulse response Reverb A with all parameters equal except for room size.



Figure 4.8 - Echo density profiles for Reverb A reverberation plug-in. The highlighted data points are the determined truncation points.

RoomWorks is a more sophisticated reverberation plug-in with far more parameters than need to be listed here. The two of interest are the room size and diffusion. Figure 4.9 shows the echo density profiles and truncation points of the impulse response of RoomWorks with all parameters equal except for room size.



Figure 4.9 - Echo density profiles for RoomWorks reverberation plug-in impulse responses. The determined truncation point is the highlighted data point.

The truncation function is consistent in its determination of the end of the early reflections for the Reverb A impulse responses. As the plug-in increases what is to be perceived as room size, the truncation point is found to be later in response first at 82 ms for a size of 20, then 245 ms for 80, and 347 for a room size of 100. This was repeated with the RoomWorks reverberation plug-in. When the room size was increased from 20 m³ to 183 m³, the truncation time increased from 122 ms to 368 ms. The truncation algorithm performs as expected when impulse responses are adjusted to contain more early reflections.

4.4 Pilot Study with Gardner Reverberator

The first step towards creating a complete hybrid system that could be used in place of convolution reverberation was to combine the truncated impulse response containing the early reflections with a generic filterbank reverberator to create the late reflections. A Gardner reverberator was chosen because of its high quality despite a simple architecture and because source code was readily available, so implementation and testing could be quick. Beltrán posted M-files online that supplemented (Beltrán and Beltrán, 1999) and can be downloaded at (Beltrán, 2006).

The pilot function takes the impulse response, dry audio, wet/dry mix, and gain of the Gardner reverberator as input parameters and returns the processed audio. The pilot only supports mono channels input audio and returns only mono files of processed audio because the Gardner M-file only outputs one channel. The transition between the two was the focus of the study; developing a method to handle multiple channels was developed at a later point.

The wav files containing the impulse response and audio were first read into arrays. The impulse response was then processed and truncated with the before mentioned algorithm. The returned truncated impulse response was then convolved with the input audio and the original audio is passed through the Gardner reverberator. Since there was not a high echo density at the beginning of the output from the filterbanks, the output was muted for the length of the truncated impulse response. Then 32 points before the end of the convolved output, that is the moment in time that the windowing of the truncated impulse response occurs, the Gardner reverberator is unmuted. The first 32 points are windowed with the rising edge of a 64-point Hann window giving an extremely brief cross-fade between the two sections.



Figure 4.10 - Overall structure of pilot function.

While this process was not performed in real-time as it is implemented in MATLAB, the intent is to have the algorithm implemented at a later date in a plug-in and host environment. For this reason, the filterbank was allowed to build up density in the length of time that was needed to output the convolved audio, but no more. This limit was imposed so that the algorithm does not inherently inhibit real-time execution.

The pilot function had three parameters to control the output, other than the input files: the feedback gain, late reflection gain and mix. The feedback gain was the setting in the top-level loop of the Gardner reverberator used to control the feedback. The late reflection gain was the factor that the output of the Garner reverberation was multiplied by before being mixed to the output.

4.5 Analysis and Synthesis of Late Reflections

The analysis and synthesis of the late reflections consists of determining the frequencydependent reverberation times of an impulse, and then synthesising those reverberation times using a FDN. When the reverberation times are derived, they need to be translated into filter coefficients for the FDN, but then the filter-design method and precision of the filters need to be considered. A large variety of approaches could be applied to this general method; the following describes the methods that were chosen.

4.5.1 Derivation of Reverberation Times

The method described in (ISO 3382) was used to first derive the reverberation time of the broadband impulse response. The T_{30} value was calculated from the energy decay curve (EDC) of the file.

After the squared impulse response was integrated backwards over time, normalised, and translated into decibels, a line was found with the best fit to the -5 dB and -35 dB points on the curve. As the curve was normalised, the time at which the EDC was equal to -5 dB and -35 dB corresponded directly to the appropriate points to fit the curve. The time along the line that produces -60 dB was the T₃₀ value. An example of an EDC and best-fit line used to find the T30 value can be seen in Figure 4.11.



Figure 4.11 - EDC of Sir Jack Lyons Concert Hall with the line used to calculate T_{30} .

This method calculated the T_{30} time of Sir Jack Lyons Concert Hall to be 4427 ms and the Aurora software found it to be 4220 ms. The T_{30} value of Trevor Jones Studio was 1257 ms and the Aurora software calculated it to be 1470 ms. If the Aurora software is assumed to be the correct measure, then the respective errors are only 4.9% and 3.3%

4.5.2 Derivation of Frequency-Dependent Reverberation Times

Figures 4.12 and 4.13 and Figures 4.11 and 4.12 show the EDR's ability to smooth the signal in comparison with a STFT. The STFT and EDR were taken using a 16 ms Hann window on impulse responses with a sampling frequency of 44.1 kHz. While (Smith, 2006) recommends a window of 30-40 ms, (Jot, Cerveau, and Warusfel, 1997) used a 16 ms window. (Smith, 2006) did not discuss the EDR in context of analysis and synthesis of an impulse response, while (Jot, Cerveau, and Warusfel, 1997) did, so a window of 16 ms was used. This provided a greater time resolution as ultimately the filters shaping the frequency response of the FDN would only be first-order, so smaller nuances that would be gained in a greater frequency resolution would be lost in re-synthesis.



Figure 4.12 - STFT of Jack Lyons impulse response using 16 ms Hann windows.



Figure 4.13 - EDR of Jack Lyons impulse response with 16 ms Hann windows.



Figure 4.14 - STFT of Trevor Jones impulse response with 16 ms Hann windows.



Figure 4.15 - EDR of Trevor Jones impulse response with 16 ms Hann windows.

Finding the T_{30} time from an EDR differed in several ways from using an EDC. The main difference was that the entire EDR was normalised, so EDCs of individual frequency bands were not individually normalised. This meant that the -5 dB and -35 dB points were relative to the initial energy of the band, which in most cases did not coincide with the absolute values associated with -5 dB and -35 dB. Also, both points often occurred at the very end of the impulse response when all of the energy died away; this did not leave enough data points to calculate a well-fit line. Or, the energy in the frequency band could be so weak that the noise floor would overwhelm the signal, also preventing accurate -5 dB and -35 dB points to be found. Instead, if both points were found to be too close to each other for a linear regression to be calculated, the first 2 data points were used to fit a line.

The calculated T_{30} values along time for Sir Jack Lyons Concert Hall and Trevor Jones Studio can be seen in Figures 4.16, 4.17, and 4.18



T30 Values for Original Impulse Responses

Figure 4.16 - Frequency-dependent T30 values determined by the Aurora software.



Figure 4.17 - T30 values against frequency for Sir Jack Lyons Concert Hall as derived from the EDR.



Figure 4.18 - T30 values against frequency for Trevor Jones Studio as derived from the EDR.

The smooth decay across the frequency range of the Sir Jack Lyons Concert Hall impulse response shows the diffuseness of the room while the room modes and less smooth decay of the smaller and more absorbent Trevor Jones Studio. Since the T_{30} values calculated by Aurora are averaged over octave bandwidths, the resulting curve is much smoother than those in Figures 4.17 and 4.18, but the shapes are still similar. The high peak showing a room resonance between 4 kHz and 16 kHz appears in both Figures 4.16 and 4.17, while the small peak around 10 kHz in Figure is 4.18 is averaged out in Figure 4.16.

4.5.3 Low-Pass Filter Coefficients

The lossless FDN has all of its poles on the unit circle; to shape the reverberation time in a frequency-dependent manner, the pole need to be brought inside the unit circle. Consider Eq. 4.4.

$$z^{-1} \leftarrow G(z)z^{-1}$$
 Eq. 4.4

Let G(z) be the filtering per sample in the propagation medium. In order to set the reverberation time, a G(z) needs to be found that moves the poles to the desired locations and then a low-pass filter needs to be designed where $H_i(z) \approx G^{M_i}(z)$ (Smith, 2006). The relationship between the reverberation time and $H_i(z)$ derived in (Dahl and Jot, 2000) is shown in Eq. 4.5.

$$20 \log_{10} |H_i(e^{j\omega t})| = -60 \frac{\tau_i}{T_r(\omega)}$$
 Eq. 4.5

Let $T_r(\omega)$ be the desired reverberation time at radian frequency ω and H_i be the transfer function of the low-pass filter in delay line *i*. There will be compromises between the desired filter and the actual one as a result of the filter design process (Smith, 2006).

The H_i can be calculated by letting $T_r(\omega)$ be the curve of T_{30} times generated from the EDR of the impulse response. The filter coefficients for each delay line were then derived from the MATLAB function *invfreqz* and H_i .

4.6 Feedback Delay Network

Feedback delay networks give a large amount of control over the frequency response of the output and have been recommended to use in conjunction with derived parameters from an impulse response (Jot, Cerveau, and Warusfel, 1997). For this very feature, it was decided to use an FDN for the final step in the hybrid reverberation algorithm.

It had been advised by (Beltrán and Beltrán, 1999) to use a MEX file to implement a FDN in as a MATLAB function because to use the MATLAB filter function, the input, output, numerator and denominator of the transfer function would need to be matrices. Such a function does not currently exist in MATLAB. Also, several for loops would be needed throughout the process, which MATLAB does not handle well (MathWorks, 2006).

A MEX file was written that implements a C++ class creating a FDN. The C++ class, FDNObject was written using the Synthesis Toolkit (STK) classes and was loosely modelled after (Lebel, 2006). The STK classes are documented and can be downloaded at (Cook and Scavone, 2005).



Figure 4.19 - Block diagram of FDN function.

The FDN function takes the input audio, number of channels, feedback matrix, and filter coefficients as input parameters and outputs a matrix containing the same number of channels of audio as channels in the FDN. This is so that the mixing of the channels can happen independently of the FDN function. Any tone control or mixing with the direct sound also needs to be implemented outside of this implementation of an FDN.

4.6.1 Lossless FDN

As advised by previous papers, (Jot and Chaigne, 1991), (Dahl and Jot, 2000), (Blesser, 2001), the FDN was first designed to be a white noise generator. Several feedback matrices and numbers of channels were investigated.

4.6.2 Derivation of FDN Delay Lengths

At first the system was tested with four channels with delay lengths of prime numbers ranging from 350 to 4000 so that the system would be as diffuse as possible. Delay lengths that are not mutually prime, that is whose prime factorisation do not contain any common factors, cause delays to occur at the same time, decreasing the density. This maximises the number of samples the lossless FDN can run before repeating the same impulse response (Smith, 2006).

Delay lengths need to meet a number of criteria. The average delay length is equal to the mean free path of the space being modeled, represented in Eq. 4.6 and 4.7.

$$\frac{d}{cT} = \frac{1}{N} \sum_{i=1}^{N} M_i$$
 Eq. 4.6

$$d = 4\frac{V}{S}$$
 Eq. 4.7

where d is the free mean path, c is the speed of sound, T is the sampling period, M_i is the delay length of channel M, and V is the volume of the space and S the surface area.

Another measurement is the modal density of the system. The sum of the delay lengths is the total number of poles in the system, denoting the order. If the modes are uniformly distributed, the modal density can be described by MT modes per Hz where M is the sum of the delay lengths of the system and T is the sampling period (Smith, 2006).

Schroeder suggested that for a reverberation of 1 second, the modal density should be 0.15 modes/Hz (Schroeder and Logan, 1961). Schroeder's formula is Eq. 4.8; using this formula, then a reverberation time of 2.5 sec at 44.1 kHz should have a system delay of at least 16538 samples.

$$M \ge 0.15t_{60}F_s$$
 Eq. 4.8

Since the modal density is reliant upon the summation of the delay lengths, the modal density is easily increased when more delays are added to a system. The free mean path of two systems with 4 and 16 channels can remain the same, that is the space is not perceived as any bigger or smaller, but the 16 channel system has a much higher modal density. This makes a more realistic diffuse reverberation as the modes are closer together, like the modes of a real room.

Three sets of delays lengths for a 4, 8, and 16 channel FDN were chosen. The 4 channel FDN had delays of 4817, 3631, 2473, and 1667 samples. The 8 channel had lengths of 4409, 1733, 2213, 2687, 2903, 3181, 3413, and 3907 samples. The 16 channel had lengths of 4999, 4243, 3797, 3547, 3191, 2789, 2411, 2137, 1931, 1901, 1847, 1409, 853, 601, and 457 samples.

It was ideal to create a modal density of at least 0.15 modes/Hz for a reverberation time of 2.5 s. However, this was harder to do with fewer delay lines and the 4 channel reverberator

fell short, but did have the required modal density for up to 1.9 s. The 8 channel had a modal density of 0.15 up to 3.7 s and the 16 channel up to 6.2 s. The free mean path of each set of delays was 3147, 3055, and 2547 for the respective delays.

4.6.3 Selection of Feedback Matrix

Filterbank artificial reverberators often have problems with density, be it echo or modal, so it is important that no energy is lost in the feedback mixing process. The feedback matrix then needs to be lossless in order to maximise energy.

A feedback matrix is lossless if and only if its eigenvalues have modulus one and its eigenvectors are linearly independent. All unitary and orthogonal matrices containing only real numbers have unit-modulus eigenvalues and linearly independent eigenvectors (Smith, 2006). Therefore a feedback matrix should be unitary and orthogonal.

4.6.3.1 Identity Matrix

The identity matrix does not mix the outputs of the different delay lines. It only feeds back with equal gain exactly what was output from each channel into the corresponding input. When an identity matrix is used instead of a mixing matrix, the FDN is a series of parallel comb filters. This makes signal flow easy to track, so an identity matrix was used for early development and testing of the system such as determining delay lengths. A 4 channel identity matrix can be seen in Eq. 4.9.

$$A_{4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$
Eq. 4.9

There are no stability problems with an identity matrix of any size, but it does not use the FDN to its full capacity. The output of four channels from a 16 channel FDN with an identity matrix can be seen in Figure 4.20.



Figure 4.20 - Output from four channels of a 16 channel FDN with identity matrix.

Notice that each channel has a constant output relative to the delay length of the channel. While the delay lengths are not related to each other, each channel still has a periodic output.

4.6.3.2 Stautner and Puckette Matrix

Stautner and Puckette developed a 4 channel FDN in (1982) using the matrix discussed in Eq. 2.7. They stated that stability is guaranteed if the feedback matrix A is a product of a unitary matrix and a gain coefficient g, where |g| < 1 (Gardner, 1998). When g=1 the matrix is unitary.

The matrix with a gain coefficient of 1 in the FDN function produced a stable noise-like output with an infinite response.

4.6.3.3 Householder Matrix

The Householder matrix was discussed in section 2.2.1.4. It is a very popular matrix to use because it contains no non-zero elements. However, stability was an issue with the implementation of the Householder matrix.

The literature offers no support to why a Householder matrix is unstable in this system. The stability is dictated by the feedback matrix, and the Householder matrix meets all requirements. Brief investigations were done comparing the delay lengths and points of instability and a rough relationship was developed. The filter would first become unstable roughly ten to twelve times the longest delay length, but the delay length was never a factor of the points of instability. Time limits prevented further investigation, so a compromised system that produced acceptable results was adapted.

When the 4 channel matrix had a gain of A/N, the FDN exponentially grew. When lowpass filters were placed in the FDN, the instability of the system still gave an infinite impulse response with exponential growth. The gain factor was then decreased until stability was reached. This happened at 0.44 for a 4 channel FDN and 0.22 for a 16 channel FDN. The Householder matrix then only needed to be modified for the A_4 case then. The A_{16} case could be constructed like the 16 channel Householder matrix in Eq. 4.10, but using the modified A_4 .

$$A_{16} = \frac{1}{2} \begin{bmatrix} A_4 & -A_4 & -A_4 & -A_4 \\ -A_4 & A_4 & -A_4 & -A_4 \\ -A_4 & -A_4 & A_4 & -A_4 \\ -A_4 & -A_4 & -A_4 & A_4 \end{bmatrix}$$
Eq. 4.10

4.7 Complete System

A 4 channel FDN with a Stautner and Puckette feedback mixing matrix was chosen to simulate the late reflections as is the was the only matrix tested that created a stable impulse

response. A 4 channel and 16 channel FDN with a modified Householder matrix were also used to process audio. The Stautner and Puckette matrix created an ideal lossless prototype, but it only has four delay lines which can compromise the audio quality by low echo and modal densities. It also does not maximise the distribution of echoes as it has terms equal to zero. The modified Householder matrix is not an ideal matrix as it is not unitary, but it does have greater modal and echo densities as it can be expanded to sixteen delay lines. It does not contain any non-zero terms, increasing the initial echo density. By using all three, the affects of the number of delay lines and the feedback matrix used can be studied.

Figure 4.21 shows the overall signal flow through the hybrid reverberation algorithm.



Figure 4.21 - Complete system overview.

4.7.1 Windowing and Delay

The same windowing method as used in the pilot study was again used in the complete system. The length of the early reflections was also used to build up the reflections in the FDN, while not outputting the signal of the FDN. This helped minimise the effects of low echo density in the beginning of the FDN.

4.7.2 Stereo Output

A goal of multi-channel outputs of artificial reverberation systems, including FDNs, is to extract the same number of uncorrelated signals as outputs and feed them to the outputs. However, this can create undesired effects and make the reverberation appear to be emanating directly from the individual speakers (Smith , 2006). A way to control the stereo image is to directly control the level of correlation between the channels according to the IACC. This was implemented in (Radford, 2003) for a 16 channel FDN in a straightforward method as described by Gardner (1998:111).

A mixing matrix with the same number of columns as inputs and rows as outputs is used to mix the sixteen channels down to a stereo pair. The matrix needs to be orthogonal in each column in order to de-correlate the channels and the values should be +/- 1 in order to maximise the echo density while minimising the computation time (Gardner, 1998:111). The matrix used here can be seen in Eq. 4.11.

4 -

Γ. 4

| +1 | -1 | | |
|----|----|---------|---|
| -1 | +1 | | |
| +1 | +1 | | |
| -1 | -1 | | |
| +1 | -1 | Eq. 4.1 | 1 |
| -1 | +1 | | |
| +1 | +1 | | |
| -1 | -1 | | |
| Ŀ | : | | |

The two uncorrelated outputs y_1 and y_2 can then be mixed according to a desired IACC level using Eq. 4.12 and 4.13.

$$y_{L}(t) = \cos(\theta)y_{1}(t) + \sin(\theta)y_{2}(t)$$

$$y_{R}(t) = \sin(\theta)y_{1}(t) + \cos(\theta)y_{2}(t)$$

Eq. 4.12

where

$$\theta = \frac{\arcsin(IACC)}{2}$$
 Eq. 4.13
5. Analysis of Results

5.1 Pilot Study

The purpose of the pilot study was to test the combination of an impulse response truncated to contain only the early reflections with a recursive filterbank. Since the parameters for the filterbank were not derived from the original impulse response, they needed to be found empirically.

There were only two parameters to be controlled: the feedback gain within the Gardner reverberator and the gain factor multiplying the output of the Gardner reverberator. Figure 5.1 shows how the impulse response of the system was affected by increasing the feedback gain while the late reverberation gain was kept constant. Figure 5.2 shows the effects of the feedback gain kept constant and the late reverberation gain increased.



Figure 5.1 – Sir Jack Lyons early reflections with Gardner reverberator for the late reflections demonstrating the effects of increased feedback gain.

In all three impulse responses in Figure 5.1, the first portion, the section with the distinct reflections, of the late reflections from the Gardner reverberator is similar,. As the feedback gain is increased, those distinct reflections are not modified, but a longer, more noise-like tail appears.



Figure 5.2 – Sir Jack Lyons early reflections with Gardner reverberator for late reflections demonstrating increasing levels of gain for the late reflections.

The main difference between Figures 5.1 and 5.2 is the prominence of the discrete reflections from the Gardner reverberator. As gain increases, the discrete reflections also become larger in amplitude as the entire output is being scaled. The tail of the reverberator output becomes audible longer as the gain increases, but this also increases the amplitude of the discrete reflections. A balance between the feedback gain and late reflection gain is needed.

5.1.1 Sir Jack Lyons Concert Hall

Ringing or metallic sounding reverberation is a result of a non-uniform decay of the modes of the reverberation, so it is important that there is a smooth transition between decay times. Since the human ear is sensitive to differences in modes, a measure of similarity between two impulse responses is the reverberation time over frequency. Figures 5.3 compares the T_{30} values over octave frequency bands for the original impulse response and impulse responses from the pilot study with varying parameter values of Sir Jack Lyons Concert Hall.



Pilot Study: Jack Lyons T30 Values

Figure 5.3 - Frequency-dependent T_{30} values for pilot study on Jack Lyons impulse response. Fb denotes feedback gain while LR denotes late reflection gain.

In Figures 5.3, the values that are zero are measurements for that frequency band that the Aurora software could not compute. As can be seen in the time domain depictions of the impulse response in Figures 5.1 and 5.2, the later portion of the late reflections of the impulse response lack much power in the signal when either parameter are low, and the earlier portion

of the late reflections also is low when the late reflection gain is low. This lack of signal probably made it difficult to derive a reverberation time.

From Figure 5.3, the impulse response from a feedback gain of 0.3 and late reflection gain of 8 had T_{30} times that were a little high, but best followed the general curve of the original impulse response. A feedback gain of 0.4 and late reflection gain of 8 also mimicked the shape of the original impulse response, so three more impulse responses were taken with a constant late reflection gain of 8 and a varying feedback gain. The T_{30} time of the results can be seen in Figure 5.4.



Pilot Study - Jack Lyons T30 Values

Figure 5.4 - Frequency-dependent T_{30} values for second study of Jack Lyons impulse response.

The Aurora software also calculates a series of acoustic parameters defined by (ISO 3382) besides reverberation time. The measurements of interest here are those that compare the early energy to the late energy. Figure 5.5 compares the acoustic parameters of the first set of Sir Jack Lyons impulse responses with varying late reflection and feedback gains.



Comparison of Acoustic Parameters for Pilot Study with Jack Lyons

Figure 5.5 - Comparison of C_{50} , C_{80} , and D_{50} in pilot study on Jack Lyons impulse response.

Consistently in all three off the parameters, the original impulse response is significantly lower than all of the variations of the pilot impulse responses. C_{50} and D_{50} are a ratio of the energy in the first 50 ms of the backwards integration of the squared impulse response to the remaining energy and C_{80} is a ratio at 80 ms. In both time divisions the early portion contains only energy from the original impulse response as the truncation time is after 80 ms. So each ratio is a comparison of the original late reflections to those simulated by the Gardner reverberator.

The Gardner reverberator cannot achieve as high of an echo density as a real room easily, so it does not have as much energy. In each case, regardless of reverberator settings, the reverberator cannot create as dense of a reverberant field, hence the higher measurements.

5.1.2 Trevor Jones Studio

A pilot study was also done with the impulse response of the Trevor Jones Studio. The resulting T_{30} times for the initial set of impulse responses with varying feedback gain and late reflection gain can be seen in Figures 5.6.



Pilot Study: Trevor Jones T30 Values

Figure 5.6 - Frequency-dependent T_{30} *values for Trevor Jones impulse response.*

The impulse response that best followed the general curve of the original impulse response had a feedback gain of 0.4 and late reflection gain of 8. The impulse response with a feedback gain of 0.4 and late reflection gain of 4 followed the curve of the original above 250 Hz, but was lower overall. A second study of impulse responses from systems with the same feedback gain of 0.4 and varying late reflection gain from 4.5 to 7.5 was conducted; the resulting T_{30} values can be seen in Figure 5.7.

Pilot Study - Trevor Jones T30 Values



Figure 5.7 - Frequency-dependent T30 values for second study of Trevor Jones impulse response.

The scaling of the same output can be observed in Figure 5.7 as all of the pilot impulse responses have the same curve, but vary proportionately in reverberation time. None of the responses created a curve that was any more similar to the original than the others, and all had too long reverberation times. Even so, the impulse with the shortest reverberation times was not selected as most of the frequency bands could not be computed in Aurora. A compromise between the shortest time and stable response was chosen. The response with a feedback gain of 0.4 and late reflection gain of 6.0 was found to be the best.



Comparison of Acoustic Parameters for Pilot Study of Trevor Jones

Figure 5.8 - Comparison of C_{50} *,* C_{80} *, and* D_{50} *in pilot study on Trevor Jones impulse response.*

All three acoustic parameters for the original impulse response are greater for Trevor Jones Studio than Sir Jack Lyons Concert Hall. This is due to the nature of the spaces as Trevor Jones Studio is much drier than a concert hall, so there is less energy after 50 or 80 ms. The resulting ratio is then higher as more energy is in the early portion.

As was true for the pilot study with the Sir Jack Lyons Concert Hall impulse responses, the Gardner reverberator cannot create the same amount of energy as a real room. The acoustic measurements of the pilot are then greater than the original. Some of the measurements of the pilot impulse responses are quite close to the original impulse response, but they are the responses with high values in both parameters, for example a feedback gain of 0.8 and late reflections gain of 8.0. These are the responses that least resembled the original response.

5.1.3 Listening Comparisons

Filterbank reverberators are perceptual models and it is accepted that they will not mimic exactly the physical properties of reverberation. So even though numerical analysis will not

confirm a perfect replica, perceptual criteria may nevertheless be met. The best way to test this is to listen to audio processed through the system.

As this was only a pilot study, the author was the sole listener and contributor of subjective opinions about the audio. All listening was done in the Trevor Jones Control Room at the University of York. The comparisons were comprised of seven musical recordings consisting of a male singing voice, acoustic guitar, piano, brass horn section, flute, bongos, and a drumset. The dry mono recording, the recording processed through convolution alone, and the recording processed with the pilot hybrid algorithm were compared. Both reverberation algorithms had a 40% wet/dry mix.

The pilot algorithm passed the easier test of closely replicating the convolution reverberation when sustained music with little silence was processed, but it often failed when transient-laden music was processed. The worst performance was with the bongos recording. The transients exposed the low echo density of the Gardner reverberator leaving a tail of discrete echoes. This again could be heard with the drumset and the acoustic guitar. Even though the guitar recording was continuous strumming and had no silence, the transients from the onset of the strings being struck could be heard in discrete echoes in the upper frequencies.

The common artefact of a metallic sound appeared, ironically enough, with the brass horn section recording. Some colouration also occurred with the piano and guitar recordings.

Overall the Sir Jack Lyons Concert Hall was reproduced better than the Trevor Jones Studio. The reverberation time of the pilot algorithm was noticeably longer than the original impulse response. Trevor Jones Studio also has some audible artefacts from the space that were not recreated in the pilot algorithm, often causing it to sound better than the convolution algorithm such as in the case of horns.

In several cases such as the flute, piano, and voice, the pilot produced a more desirable reverberation than the convolution because the room modes were not exactly mimicked. The

ability to tune a room response to eliminate unwanted resonances is the goal of the algorithm, however it needs to more closely simulate the original space, including its faults, first.

5.1.4 Summary of Pilot Study

A system was developed that determines the end of the early reflections in an impulse response, truncates the response, convolves it with dry audio, and then combines the convolved audio with audio processed through a Gardner reverberator. The parameters for the late reverberation were empirically derived by taking a series of impulse responses of the hybrid system, comparing the frequency-dependent reverberation times to the original impulse response, and then repeating and refining the parameters once more.

The hybrid response was not able to fully recreate the original, though the Sir Jack Lyons Concert Hall was closer than the Trevor Jones Studio. In both cases the energy in the late reflections was not as high nor dense which was evident in acoustic measurements and in listening tests involving transients.

The algorithm was perceptually similar for several cases for the Sir Jack Lyons Concert Hall hybrid involving sustained music with little silence. The shortcomings of the algorithm became evident in other listening comparisons as discrete echoes could be distinguished in the late reflections and parameters affecting reverberation time were not tuned accurately.

The transition between the early and late reflections did not create problems as it had in previous studies, (Radford, 2003), and showed that the algorithm was viable and has the potential for success with further refinements.

5.2 Complete System

The major contribution to the complete system that differed from the system developed in the pilot study was the automatic analysis and extraction of parameters for the late reflection

synthesis. An impulse response was analysed and audio was processed with the user only specifying the files, wet/dry mix, and choice of mixing matrix.

5.2.1 Sir Jack Lyons Concert Hall

Impulse responses were taken of the hybrid system analysing and synthesising the Sir Jack Lyons Concert Hall impulse response with the three different feedback mixing matrices. The resulting frequency-dependent T_{30} values as calculated by Aurora can be seen in Figure 5.9.



T30 Values for Hybrid System with Sir Jack Lyons Concert Hall Impulse Response

Figure 5.9 - T30 values for hybrid system with Sir Jack Lyons Concert Hall Impulse Response

As Figure 5.9 illustrates, none of the impulse responses closely followed the reverberation times of the original impulse response. The Stautner and Puckette matrix did much better job at approximating the curve the Householder matrices, but not as closely as was hoped. The frequency curve was a flat; not much high-pass filtering occurred. Though, overall

reverberation time seems to be an average of the frequency-dependent reverberation times of the original impulse response.



Figure 5.10 - EDR of the impulse response created from the hybrid system with the Sir Jack Lyons Concert Hall impulse response and Stautner and Puckette matrix.

As can be seen in Figure 5.10, the impulse response using the Stautner and Puckette matrx did decay exponentially as expected, but it decayed uniformly over all frequencies. The higher frequencies needed to decay faster than the lower in order to create a convincing reverberation.

The two modified Housesholder matrices did not differ greatly in T_{30} values, especially in the mid and higher frequencies. However, their frequency curves were roughly reverses of the original with no low-pass filtering occurring at all.



Acoustic Parameters for Hybrid System with Sir Jack Lyons Concert Hall Impulse Response

Figure 5.11 – Acoustic parameters for the hybrid system with Sir Jack Lyons Concert Hall impulse response.

The 16 channel modified Householder matrix consistently had the most energy past 50 mx and 80 ms when compared to the 4 channel FDNs, and came close to the energy content of the original impulse response. The Stautner and Puckette matrix had more energy than the 4 channel Householder matrix. The Householder is modified and not lossless, so the lossless matrix with the same number of channels had more energy in the late reflections.

5.2.2 Trevor Jones Study

Impulse responses were taken of the hybrid system analysing and synthesising the Trevor Jones Studio impulse response with the three different feedback mixing matrices. The resulting frequency-dependent T_{30} values as calculated by Aurora can be seen in Figure 5.12.



T30 Values for Hybrid System with Trevor Jones Studio Impulse Response

Figure 5.12 - T30 values for hybrid system with Trevor Jones Studio impulse response.

Reverberation times of all the matrices were longer than the original when the 8 kHz peak is disregarded. The two Householder matrices had slightly longer times with 4 channels than with a 16 channel system.



Figure 5.13 - EDR of impulse response of the hybrid system with the Trevor Jones Studio impulse response and Stautner and Puckette matrix.

The Stautner and Puckette matrix did a better job approximating the curve with a firstorder filter. The higher frequencies did decay faster than the lower, as desired, but the reverberation time was overall much too long. The curve might have been distorted because of the peak resonance in the 16 kHz frequency band; the filter appears to be trying to split the lower T_{30} values with the peak the reconcile the difference.



Acoustic Parameters for Hybrid System with Trevor Jones Studio Impulse Response

Figure 5.14 - Acoustic parameters for hybrid system with Trevor Jones Studio impulse response.

A longer reverberation time means the impulse responses were longer, leading to more energy in the late reflections. As the C_{50} , C_{80} , and D_{50} values are a ratio of early to late energy, a lower reverberation time tends towards a lower values. This was demonstrated in the Trevor Jones Studio impulse responses. The Stautner and Puckette matrix had the most energy, which was not expected as the 16 channel Householder matrix had no non-zero values and more delay lines. However, since the Householder matrix has been modified and was no longer lossless, it had less energy.

5.2.3 Listening Comparisons

The auditioning of the processed audio was done in Trevor Jones Control Room by the author.

The system with numerical analysis that was the closest to the original impulse response did not create better sounding reverberation. The 16 channel Householder matrix was the best overall sounding reverberator as the density was high enough that discrete echoes could only be heard in extenuating excerpts like the bongos. The sound of the discrete echoes was altered from the sound created by the Gardner reverberator, though. The Gardner reverberator had such a low echo density that the echoes sounded like flutter echoes from existing architecture. However, the echoes resulting from the 16 channel FDN were too dense to sound like an existing physical space and were only slightly slower than noise. The result was a noise-like sound in the reverberation tail that did not resemble a real room.

The vocal recording was more difficult to reproduce with the complete hybrid system than in the pilot study. Metallic characteristics appeared and a high-pitched hissing noise could be heard in the Sir Jack Lyons impulse responses. The horns excerpt that had difficulties in the pilot study did much better with the complete system, though a 16 channel FDN was needed so that echoes were not heard in the transients.

The lack of low-pass filtering was obvious with the Stautner and Puckette matrix while the boost of the high frequency reverberation time in the Householder matrices was extremely unpleasant.

All of the audio processed with the Trevor Jones Studio impulse responses had reverberation times that were too long, and the audio processed with the Sir Jack Lyons Concert Hall impulse responses had reverberation times that were too short. Though some of the combinations, such as the Trevor Jones Studio impulse response and the flute recording, created a pleasant reverberation. However, the reverberation did not match the intended convolved recording. A 16 channel FDN was necessary for any recording that contained

enough silence for the full reverberation tail to decay, but a 4 channel FDN was adequate when a continuous signal was present.

5.2.4 Summary of Complete System

The relationship described in Eq. 4.5 is intended to describe the filter coefficients for a lossless FDN. When that FDN is not lossless, the poles of the filter are no longer on the unit circle but inside of it. A modified Householder matrix is not lossless and does not act as expected as the low-pass filters move the poles from an initial position that was not on the unit circle. The result was undesirable reverberation times and absolutely incorrect frequency responses.

However, the Householder matrices did show in the listening comparisons that more than four channels are needed for a smooth set of late reflections. The echo and modal density from the16 channel FDN was adequate for most of the recordings.

The Stautner and Puckette matrix is truly lossless, but still the filters did not act as expected. The translation between the curve generated by the EDR and the filters needs to be studied more. The EDR computed accurate results when compared with the values calculated by the Aurora software, but that curve did not appear in the FDN correctly. The analysis and synthesis of different impulse responses did alter the impulse response of the system, so an analysis and synthesis was attempted, but was not executed correctly.

Higher-order filters may help resolve some of the problems, as the impulse response of both systems using the lossless matrix had reverberation times that were relevant to the original. Neither was extremely high nor low, and in the case of the Sir Jack Lyons Concert Hall, the reverberation time for the entire frequency spectrum was roughly the average reverberation time of the original impulse response. Higher-order filters that match a curve more precisely may improve the response.

6. Further Work

While the general concept and implementation were successful, there is room for improvement. The individual analysis and synthesis methods could use more study for more accurate and efficient implementations. In particular the early reflection truncation algorithm could be improved by analysing overlapping windows so that a truncation point could be found with greater precision. It would also be ideal for a curve to be fit to the echo density profile, which would decrease the chance of a premature truncation point due to a window in the early reflections that happens to meet the threshold. A point would be selected at the beginning of the asymptotic portion of the curve.

The translation between the T_{30} curve produced by the EDR and the filter coefficients for the FDN deserves further investigation. Higher-order filters may improve results as the frequency response could be matched more precisely than a one-pole filter could provide.

Within the time allotted, the scope of this project needed to be limited to a study of the feasibility of the novel ideas. So this project has not produced a consumer-ready application, but showed that the ideas and methods could be used in conjunction with existing technology. Steps to create more consumer-friendly software would include basic impulse response processing such as noise reduction, and automatic removal of direct sound. Parameters like predelay and equalization of the entire impulse response would also be simple to implement. It could also accommodate multiple channels of input and create more than a stereo output.

As most digital audio effects are used as plug-ins within digital workstation environments, to be befitting to most music applications, the algorithm should be translated from MATLAB into C++. The FDN is already in C++ using the FDNObject class, and the convolution function in MATLAB uses the Fastest Fourier Transform of the West (FFTW), which is a free library available for a multitude of platforms.

The scope of this project did not entail investigating the parameters introduced by the FDN. More time could be devoted to exploring the mapping of perceptual effects to the

available parameters and creating additional ones. The ability to manipulate the late reflections of an impulse response could be mapped to several parameters including reverberation time and frequency response.

A further research project that could grow from this project is a hybrid reverberator that could pan around a space, allowing a user to move a sound within a space using the precise early reflections captured by an impulse response. Interpolation between early reflections of impulse responses could create the panning effect. Compatible impulse responses are currently being recorded with the turntable room acoustic measurement technique where a full 360 degrees of rotation are recorded at small intervals. Techniques being developed to smoothly transition between HRTFs could also be applied at a smaller scale to only the early reflections of an impulse response. When the same late reflections filterbank reverberator is used and only the early reflections are adjusted according to panning position, a computationally inexpensive algorithm could be implemented.

7. Conclusion

The initial goal of this project was to create a hybrid artificial reverberation algorithm that uses elements of both convolution reverberation and filterbank reverberation. It was desired to have the accuracy and quality that a convolution reveberator provides, while introducing parametric controls and computational efficiency that have previously been limited to filterbank reverberators. The hybrid algorithm should be perceptually indistinguishable from a convolution reverberator, since convolution reverberators can only be replaced by a system that offers the exact advantages.

A hybrid system consists of convolving the early portion of an impulse response with a signal and combining it with the same input audio processed through a filterbank. This ideally conveys the complex and difficult to replicate information of the early reflections and lets the more generic information conveyed in the late reflections to be approximated through filterbanks. The filterbanks can then be adjusted and altered affecting only the late reflections, unlike a convolved impulse response.

Previous algorithms had attempted to accomplish a similar goal, but for several reasons, had not met it. The main stumbling block was believed be the arbitrary truncation of an impulse response without considering the early reflections. An accepted standard time for early reflections is 80 ms, but it is acknowledged in literature that this is a poor approximation. So an algorithm was developed that determines the end of the early reflections before a truncation is point is chosen.

(Radford, 2003) had concluded that the transition between the convolved early reflections and later reflections produced from the filterbanks was a problem. This was overcome in a pilot study with the combination of selecting a truncation point after the early reflections and then windowing to transition between the two portions. The pilot study truncated an impulse response to contain only early reflections, windowed the end of the truncation, then convolved it with the input audio. The input audio was also processed with a Gardner

reverberator and the beginning of the audio was windowed and added to the end of the convolved audio.

The system was successful and showed that the technique of combining convolution and filterbank processing is viable. The early reflections from the original impulse response were present in the system, but the late reflections were still generic in the sense that they did not have a direct relationship to the impulse response. The user adjusting the parameters improved this, but automatic extraction and application of parameters to a filterbank was the next step. There were two iterations of trials and tests to find reasonable parameters and room for improvement still remained.

An impulse response then underwent further analysis to extract a curve that describes the reverberation time versus frequency of the stochastic portion of the impulse response. This curve was fit with filter coefficients which were applied to a FDN. The FDN then took the place of the Gardner reverberator in the pilot study.

The system proved to be partially successful as there were difficulties implementing the FDN. Matrices that mathematically meet all stability requirements for an FDN were still causing the system to become unstable. A compromise was made to elicit an acceptable response from a Householder matrix and systems with various matrices and channels were analysed.

A higher number of channels in the FDN were needed for audio with transients and/or silence as discrete echoes can be distinguished from low order systems. A system with only 4 channels were sufficient for musical excerpts without silence like the piano recording, but echoes could still be heard in the 16 channel system with transient-laden audio. They were not offensive for most excerpts; a 16 channel FDN still could be used successfully as a reverberator.

Issues arose with the use of modified Householder matrices as they were not lossless, so the relationship derived by Jot between the desired frequency response and the delay length

was no longer true. The Stautner and Puckette matrix followed the desired reverberation curve more closely than the two Householder matrices as it was a lossless matrix, but it still did not create a perceptually transparent system. The frequency response and frequencydependent reverberation time of the system did not correspond to the original impulse response being analysed. The majority of the problems are attributed to the filters used to shape the frequency response of the FDN. Higher-order filters and better algorithms for fitting coefficients to frequency responses need to be researched.

This project studied the feasibility of a hybrid artificial reverberation with automatic analysis and synthesis of an impulse response. While a successful implementation has not yet been created, the analysis of the results from the system implemented in this project suggests that one is possible. A more computationally efficient algorithm was created that introduces parametric controls to an impulse response, and while those parameters were not greatly explored here, they could be expanded to allow the user greater control over the reverberator. It has been shown that the statistically stochastic portion of an impulse response can be perceptually reproduced with a filterbank and combined with the more deterministic portion of the impulse response through convolution creating a hybrid reverberation algorithm that rivals previous convolution and filterbank reverberators.

References

J. S. Abel and D. P. Berners, "Reverberation: acoustics, analysis and synthesis," *Presented at Audio Eng. Soc. 117th Convention*, October 2004.

M. Barron, "The subjective effects of first reflections in concert halls – the need for lateral reflections," *J. of Sound and Vibration*, vol. 15 pp.475-94, 1971.

M. Barron, Auditorium acoustics and architectural design. London: E & FN Spon, 1993.

D. R. Begault, 3-D sound for virtual reality and multimedia. Boston: AP Professional, 1994.

J. R. Beltrán and F. A. Beltrán, "Matlab implementation of reverberation algorithms," *Proc. DAFX* '99, Trondheim (Noruega), pp. 91-96 Dec. 99.

F. A. Beltrán, "Matlab implementation of some reverberation algorithms," http://www.cps.unizar.es/~fbeltran/frame_inv.html, University of Zaragoza, Spain, accessed Aug. 2006.

R. Ben-Hador and I. Neoran. "Capturing manipulation and reproduction of sampled acoustic impulse responses," *Audio Eng. Soc. 117th Conv. Paper*, October 2004.

J. Blauert, *Spatial hearing: the psychophysics of human sound location*. 3rd Ed. Cambridge, MA: MIT Press, 1997

B. Blesser, "An interdisciplinary synthesis of reverberation viewpoints," J. Audio Eng. Soc., vol. 49, pp. 867-903, Oct. 2001.

S. Browne, "A hybrid reverberation algorithm using truncated impulse response convolution and recursive filtering," M.S. thesis, University of Miami, Coral Gables, FL, USA, 2001.

P. R. Cook and G.P. Scavone, *The synthesis toolkit in C++ (STK) 4.2.1.*, http://ccrma.stanford.edu/software/stk/, 2005, accessed July 2006.

L. Dahl and J. M. Jot, "A reverberator based on all-pass filters," *Proc. of DAFX '00*, Verona, Italy, Dec. 2000.

A. Farina, "Simultaneous measurement of impulse response and distortion with a swept-sine technique," *Audio Eng. Soc. 108th Conv. Preprint*, February 2000.

A. Farina and R. Ayalon, "Recording concert hall acoustics for posterity," *Audio Eng. Soc.* 24th Inter. Conf. on Multichannel Audio, June 2003.

W. G. Gardner, "The virtual acoustic room," M.S. thesis, MIT, Boston, MA, USA, 1992.

W. G. Gardner, "Efficient convolution without input-output delay," J. Audio Eng. Soc., vol. 43, pp. 127-135, March 1995.

W. G. Gardner, "Reverberation algorithms," in *Applications of Digital Signal Processing to Audio and Acoustics*. M. Kahrs and K. Brandenburg (Eds.), pp. 85-131, MA: Kluwer

Academic Pub., 1998.

L. S. Goodfried and J. H. Beaumont, "The development and application of synthetic reverberation systems," *J. Audio Eng. Soc.*, vol. 7, pp. 228-324, October 1959.

D. M. Howard and J. Angus, Acoustics and psychoacoustics. 2nd ed. London: Elsevier, 2001.

J.M. Jot and A. Chaigne, "Digital delay networks for designing artificial reverberators," *Audio Eng. Soc.* 90th Conv. Preprint, Feb. 1991.

J. M. Jot, L. Cerveau, and O. Warusfel, "Analysis and synthesis of room reverberation based on a statistical time-frequency model," *Audio Eng. Soc. 103rd Conv. Preprint*, Sept. 1997.

Inter. Standards Org., "Acoustics – measurements of the reverberation time of rooms with reference to other acoustical parameters," 2^{nd} ed., ISO 3382, June 1997.

H. Kuttruff, Room Acoustics. London: Applied Science Pub., 1973.

L. Lapin, *Probability and Statistics for Modern Engineering*. 2nd ed., Boston: WFS-Kent Pub., 1990.

D. Lebel, Real-time reverberation using feedback delay network. http://www.music.mcgill.ca/~lebel/reverb/, McGill University, accessed July 2006.

MathWorks, MATLAB 7 Documentation. can be found at http://www.mathworks.com/access/helpdesk/help/helpdesk.html, accessed Aug. 2006.

J. Merimaa and V. Pulkki, "Spatial impulse response rendering I: analysis and synthesis," *J. Audio Eng. Soc.*, vol. 53, pp. 1115-1127, Dec. 2005.

J. A. Moorer, "About this reverberation business," Computer Music J., vol. 3, 1979.

C. Moura and S. L. Campos, "Some notes on artificial reverberation," *J. Audio Eng. Soc.*, vol. 5, pp.182-186, October 1957.

D. T. Murphy, "Multi-channel impulse response measurement, analysis and rendering in archaeological acoustics," *Audio Eng. Soc. 119th Conv. Paper*, October 2005.

D. Radford, "Hybrid artificial reverberation algorithms for VST," M.Eng. thesis, University of York, York, UK, 2003.

R. Ponikowski, "Spatial acoustics – the measurement and rendering of acoustic spaces," M.Eng. thesis, University of York, York, UK, 2006.

M. R. Schroeder, "Natural sounding artificial reverberation," J. Audio Eng. Soc., vol. 10, 1962.

M. R. Schroeder, "New method for measuring reverberation time," *J. Acoust. Soc. Am.*, vol. 37, pp. 409-412, 1965.

M. R. Schroeder and B. F. Logan, "Colorless artificial reverberation," *IRE Transactions*, vol. AU-9, pp. 209-214, 1961.

J. O. Smith, *Physical audio signal processing: for virtual musical instruments and digital audio effects*, online book at http://ccrma.stanford.edu/~jos/pasp/, Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, 2006, accessed Aug. 2006.

J. Stautner and M. Puckette, "Designing multi-channel reverberators," *Computer Music J.*, vol. 6, 1982.

U. Zölzer, DAFX : digital audio effects. Chichester : Wiley, 2002.

Appendix A: Accompanying CD Audio File Listings

The audio samples are arranged in a series of folders and subfolders. The top level folders are called Sir_Jack_Lyons and Trevor_Jones_Studio and the audio in each is processed with the respective impulse response. Within those folders are the subfolders Convolution, Pilot, and Hybrid. Each contains seven recordings processed by the respective system, whether full convolution, the pilot system, or the complete hybrid system. The recordings include bongos, drumset, acoustic guitar, male vocals, flute, piano, and brass horn section. All audio is recorded at 24 bit and 44.1 kHz.

Appendix B: Hybrid Algorithm MATLAB Files

The files are listed in the order that MATLAB would call them: hybrid.m, truncate.m, convolve.m, calcFilter.m and mixdown.m.

```
function [processedAudio] = hybrid(IRin, audioIn, mix, matrixChoice)
%[processedAudio with chanOut channels] = hybrid(IRin wav, audio in wav,
õ
            mix %, IACC 0.0-1.0, 1=stautner and puckette 2=4 chanel
õ
            Householder 3=16 channel Householder)
8
웡
% Reads in a wav files of IR and dry audio then adds reverberation,
% returning the mixed array. The returned array will have 2 channels.
% First the IR is truncated to the end of the early reflections
% then the truncated IR is convolved with the input audio. Then audio
% then passes through a FDN reverberator to recreate late relflections.
옹
% The M-files truncate.m, convolve.m, calcFilter, and mixdown and the mex-
% file FDN.dll need to be included in the search path with this M-file.
% Rebecca Stewart
% August 2006
응
%Function requires at least 3 inputs argument
if nargin < 3
  error('Not enough input arguments.');
end
%If matrix specified, default to householder
if nargin < 4
   matrixChoice = 3;
end
%FDN parameters
84 Channel with Stautner and Puckette Matrix
if matrixChoice == 1
   channels = 4;
   matrix = [0 1 1 0; -1 0 0 -1; 1 0 0 -1; 0 1 -1 0]/sqrt(2);
   delays=[4817 3631 2473 1667];
end
%4 Channel with modifiend Householder Matrix
if matrixChoice==2
   channels = 4;
    delays=[4817 3631 2473 1667];
   matrix = [1 -1 -1 -1; -1 1 -1 -1; -1 -1 1 -1; -1 -1 -1 1]*.44;
end
%16 Channel with modified Householder Matrix
if matrixChoice==3
   channels = 16;
   matrix = [1 -1 -1 -1; -1 1 -1 -1; -1 -1 1 -1; -1 -1 -1 1];
   matrix = [matrix -matrix -matrix; -matrix matrix -matrix -
matrix; -matrix -matrix matrix -matrix; -matrix -matrix matrix];
   matrix = matrix*.22;
   delays = [4999 4639 4243 3797 3547 3191 2789 2411 2137 1931 1901 1847
1409 853 601 4571;
end
```

```
IACC=0.5;
%Read in wavs into arrays
[IR, Fs,nbits1] = wavread(IRin);
[audio, Fs2, nbits2] = wavread(audioIn);
audioRow = rot90(audio);
%Sampling frequencies need to be equal in IR and dry audio
if (Fs > Fs2 || Fs < Fs2)
    error('Impulse response and audio need to have same Fs.');
end
%Truncate the IR to include only early reflections with 20 ms window
[truncTime, truncIR] = truncate(IR, Fs, 20);
%Convolve the IR with the dry audio
convd = convolve(truncIR, audio, 100);
%Find the FDN filter coefficients
[b a] = calcFilter(IR,Fs,delays);
%Process audio through FDN
output = FDN(audioRow, channels, delays, matrix, b, a);
%Mix to stereo
stereo = mixdown(output, IACC, channels);
%Window the late reflections
FDNLength = length(stereo);
startWind = (floor(truncTime/1000*Fs)-32);
window = hann(64);
stereo(1:startWind,:) = zeros(startWind,2);
stereo((startWind+1):(startWind+32)) =
stereo((startWind+1):(startWind+32)).*window(1:32)';
%Zero-pad so that matrix dimensions agree
audioLength = length(audio);
conLength = length(convd);
audio((audioLength+1):FDNLength) = zeros((FDNLength - audioLength),1);
convd((conLength+1):FDNLength) = zeros((FDNLength- conLength),1);
%Add the wet audio before mixing with dry
wet(:,1) = (convd + stereo(:,1));
wet(:,2) = (convd + stereo(:,2));
%Add the early and late reflections and mix with dry audio and normalise
processedAudio(:,1) = (mix/100)*(wet(:,1)) + ((100-mix)/100)*audio;
processedAudio(:,2) = (mix/100)*(wet(:,2)) + ((100-mix)/100)*audio;
            function [processedAudio] = convolve(IR, audio, mix)
R
옹
%[processedAudio] = pilot(IR array, audio array, mix wet%)
å
%Takes an array of an IR and an array of dry audio then convolves,
%returning the mixed array. Only supports a mono dry audio with
%mono IR.
```

```
ò
```

```
% Rebecca Stewart
% July 2006
옹
%Function requires at least 3 inputs argument
if nargin < 3
   error('Not enough input arguments.');
end
processedAudio = [];
%Check number channels on dry audio
[audioLength, audioChan] = size(audio);
%Check number of channels on IR
[IRLength, IRChan] = size(IR);
if(audioChan > 1)
   error('Too many channels in dry audio, only can be mono.');
end
if(IRChan > 1)
    error('Too many channels in IR, only can have mono.');
end
%Convolve
if(audioChan == IRChan)
   tempAudio = conv(audio, IR);
end
%Zero-pad input audio to same size as convolved audio
audio((audioLength+1):(length(tempAudio))) = zeros((length(tempAudio)-
(audioLength)),1);
%Return mixed audio in array
processedAudio = (mix/100)*tempAudio + (1-mix/100)*audio;
            function [truncTime, truncIR] = truncate(IR, Fs, winSize)
옹
%[truncation time, truncated array] =
å
     truncate(array containing IR, sampling frequency, window size in ms)
옹
옹
%Determines the end of the early reflections and truncates
%the IR. The output time of truncation in msec and truncated
%array are output.
å
%Rebecca Stewart
%July 2006
%Function requires at least 3 inputs argument
if nargin < 3
  error('Not enough input arguments.');
end
truncTime = []; truncIR = [];
```

```
%Window wav
window = Fs * winSize/1000;
sizeInput = length(IR);
iterations = floor(sizeInput(1)/window);
time = linspace(0,sizeInput(1)/Fs,iterations); %time increments
for i=1:iterations;
    winIn = IR(window * (i-1)+1:window*i);
    %Find standard deviation and mean of window
    sigma = std(winIn);
    mu = mean(winIn);
    %Calculate percentage of samples outside 1 standard deviation
    out1 = 0;
    for n=1:window
        if winIn(n) < (mu - sigma) | winIn(n) > (mu + sigma);
            out1 = out1 + 1;
        end
    end
    %Store in array
    percent1(i) = out1/window;
end
%Check for first time over 30% in first 40 to 160 ms
for i=8:-1:3
    if percent1(i) >= .3
        met30 = i;
    end
end
%Find local max over the 60 ms after the first crossing of the threshold
[maxValue, truncIndex] = max(percent1(met30:met30+1));
truncTime = 1000*time(truncIndex+met30-1);
tempIR = IR(1:ceil(truncTime/1000 * Fs)+32);
%Hann window
w = hann(64);
%Window the end of the IR with the falling edge of window function
l = length(tempIR);
tempIR((1-31):1) = w(33:64) .* tempIR((1-31):1);
%Return truncated IR
truncIR = tempIR;
              * * * * * * * * * * * End of truncate.m* * * * * * * * * * * * * *
function[b, a] = calcFilter(IR, Fs, delays)
옹
%[T30 data, filter coefficients b, a] = calcFilter(array holding IR,
웡
                                             sampling frequency, delays)
%Finds the filter coefficients for a 1 pole IIR that match the T30 curve.
```

```
å
%Rebecca Stewart
%August 2006
응
lenIR = length(IR);
%window size
windowSize = 2^nextpow2(0.001*16 * Fs);
window = blackmanharris(windowSize);
%calculate number of windows
total = ceil(lenIR/windowSize);
%zero-pad the end of the IR
pad = zeros((total*windowSize) - lenIR,1);
IR((lenIR+1):(total*windowSize)) = pad;
%Loop through time domain and calculate STFT
for n = 1:total
    %Window a portion of the IR
    f = IR(windowSize*(n-1)+1:windowSize*n);
   f = f.*window;
    %Take FFT of window
   F=abs(fft(f,windowSize))/windowSize;
    z(n,:) = F(1:(windowSize/2));
end
%Loop through frequency domain and calculate EDC for each frequency
[sizeTime, sizeFreq] = size(z);
for n=1:sizeFreq
    iidc(sizeTime:-1:1) = cumsum(z(sizeTime:-1:1,n).^2);
    EDR(:,n) = iidc;
end
%Normalize and translate to dB
EDRdB = 10*log10(EDR/max(max(EDR)));
%Rotate matrix for processing
EDRdB = rot90(EDRdB);
t = linspace(1,lenIR/(Fs/1000),lenIR);
t = rot90(rot90(t));
8_____
                                ______
%Calculate the T30 values
for n=1:(windowSize/2)
    %Find -5dB point
    if(isempty(find(EDRdB(n,:) <= (EDRdB(n,1)-5),1)))
       minus5(n) = 1;
    else
       \min us5(n) = find(EDRdB(n,:) \leq (EDRdB(n,1)-5),1);
   end
    %Find -35dB point
    if (isempty(find(EDRdB(n,:) <= (EDRdB(n,1)-35),1)))
       minus35(n) = 1;
   else
       minus35(n) = find(EDRdB(n,:) <= (EDRdB(n,1)-35),1);</pre>
   end
    %If a value cannot be calculated from -5 to -35 dB
```

```
%Just fit a line to first 15 points
   if(minus35(n) \le minus5(n))
       fit = polyfit(t(1:15), EDRdB(n,1:15), 1);
   %Else fit a line to between -5 and -35 dB points
   else
       %Find best-fit line
      m5 = minus5(n);
      m35 = minus35(n);
       xx = t(m5:m35);
       yy = EDRdB(n,m5:m35);
       fit = polyfit(xx, yy, 1);
   end
   %T30
   T30(n) = (-60-fit(2))/fit(1);
end
8_____
                     _____
channels = length(delays);
Tsamples = T30/1000*Fs;
%Frequency in radians for 0 to PI
w = linspace(0,pi,windowSize/2);
%Find filter coeffecients for T30 frequency response
for n=1:channels
   exponent = -3*(delays(n)./Tsamples);
   desired(:,n) = 10.^(exponent);
   [bb aa] = invfreqz(desired(:,n),w, 0, 1, [], 30);
   b(n,:) = bb;
   a(n,:) = aa;
end
           function[audioOut] = mixdown(audioIn, IACC, chanIn)
옹
%[stereo output audio] = mixdown(array holding audio, IACC value, number of
웅
                               input channels)
õ
%Mixes 16 channels to 2 uncorrelated channels, then mixes them according
%to the IACC value. Returns a stereo array of audio.
R
%Rebecca Stewart
%August 2006
å
if chanIn == 16
   1 -1 1 1 -1 -1 1 1 -1];
end
if chanIn == 4
   mixArray = [1 -1 1 -1; -1 1 1 -1];
end
mixedAudio = mixArray * audioIn;
theta = asin(IACC)/2;
audioOut(:,1) = cos(theta)*mixedAudio(1,:) + sin(theta)*mixedAudio(2,:);
audioOut(:,2) = cos(theta)*mixedAudio(2,:) + sin(theta)*mixedAudio(1,:);
```

Appendix C: Feedback Delay Network Source Code

The MEX-file is the compiles into an dll which is called from MATLAB. The dll needs to be placed in the MATLAB search path.

The FDNObject class was written using the STK classes that can be found at http://ccrma.stanford.edu/software/stk/.

```
C file to be compiled to a MEX file for MATLAB 7.0
  Written by Rebecca Stewart
  July 2006
#include <math.h>
#include <stdio.h>
#include "mex.h"
#include "FDNObject.h"
void mexFunction( int nlhs, mxArray *plhs[],
             int nrhs, const mxArray*prhs[] )
{
     double *in, *outArray, *matrix, *lengthIn, *sampleOut, *channels,
*filtersB, *filtersA;
     int i,j;
     int inRowLen, inColLen, outRowLen, outColLen;
     StkFloat *lengthStk;
     /* Check for proper number of arguments */
     if (nrhs != 6) {
           mexErrMsgTxt("Six input arguments required.");
     } else if (nlhs > 1) {
           mexErrMsgTxt("Too many output arguments.");
     }
     //Get input parameters
     in = mxGetPr(prhs[0]);
     channels= mxGetPr(prhs[1]);
     lengthIn = mxGetPr(prhs[2]);
     matrix = mxGetPr(prhs[3]);
     filtersB = mxGetPr(prhs[4]);
     filtersA = mxGetPr(prhs[5]);
     inRowLen = mxGetN(prhs[0]);
     inColLen = mxGetM(prhs[0]);
     /* Check for proper dimensions of arguments */
     if(inColLen != 1) {
           mexErrMsgTxt("First argument (input audio) must be a single row
vector.");
     }
     if((mxGetM(prhs[1]) != 1) || (mxGetN(prhs[1]) != 1)){
           mexErrMsgTxt("Second argument (channels) must be a scalar.");
     }
     if((mxGetN(prhs[2]) != channels[0]) || (mxGetM(prhs[2]) != 1)){
           mexErrMsgTxt("Third argument (delays) must be matrix with
dimensions 1 X CHANNELS.");
     }
```

```
if((mxGetM(prhs[3]) != channels[0]) || (mxGetN(prhs[3]) !=
channels[0])){
            mexErrMsgTxt("Fourth argument (feedback matrix) must be matrix
with dimensions CHANNELS X CHANNELS.");
      }
      if((mxGetM(prhs[4]) != channels[0]) || (mxGetN(prhs[4]) != 1)){
            mexErrMsgTxt("Fifth argument (B coefficients) must be matrix
with dimensions CHANNELS X 2.");
      }
      if((mxGetM(prhs[5]) != channels[0]) || (mxGetN(prhs[5]) != 2)){
            mexErrMsgTxt("Sixth argument (A coefficients) must be matrix
with dimensions CHANNELS X 3.");
      }
      /* Calculate dimensions of output array */
      outRowLen = lengthIn[0]*12 + inRowLen;
      outColLen = channels[0];
      lengthStk = (StkFloat*)lengthIn;
      //Allocate memory and assign output pointer
      plhs[0] = mxCreateDoubleMatrix(outColLen, outRowLen, mxREAL);
      //Get a pointer to the data space in our newly allocated memory
      outArray = mxGetPr(plhs[0]);
      sampleOut = new double[channels[0]];
      /* Set up FDN */
      FDNObject myFDN(channels[0]);
      myFDN.updateDelays(lengthIn);
      myFDN.updateFilters(filtersB, filtersA);
      myFDN.clear();
      myFDN.updateMatrix(matrix);
      for(i=0;i<inRowLen;i++)</pre>
      {
                  /* Calculate the output of each channel */
                  sampleOut = myFDN.computeSample(in[i*inColLen]);
            for(j=0;j<outColLen;j++)</pre>
            {
                  /* Copy each output from channel to corresponding channel
in output array */
                  outArray[i*outColLen+j]= sampleOut[j];
            }
      }
      /* Flush out rest of delay */
      for(i=inRowLen;i<outRowLen;i++)</pre>
      {
            /* Calculate the output of each channel */
            sampleOut = myFDN.computeSample(0);
                        for(j=0;j<outColLen;j++)</pre>
            {
                  /* Copy each output from channel to corresponding channel
in output array */
                  outArray[i*outColLen+j]= sampleOut[j];
            }
      }
      return;
}
```

```
* FDNObject Class
* Written by Rebecca Stewart
* August 2006
* A fully configurable template for a feedback delay network
* User defines the number of delay lines, all values for delay
* lengths, all coefficients for absorbent filers, and values
* for the feedback matrix.
^{\star} An array equal to the number of delay lines is output when
* samples are computed. This is so the user can choose
* desired number of channels and mix(down) output as desired.
#ifndef _FDNOBJECT_H
#define _FDNOBJECT_H
#include "Stk.h"
#include "Filter.h"
#include "OnePole.h"
#include "OneZero.h"
#include "Delay.h"
#include "DelayL.h"
class FDNObject
public:
     /* Constructor and destructor */
     FDNObject(int numDelays);
     ~FDNObject();
     /* Member functions */
     void updateMatrix(double *values);
     void updateDelays(StkFloat *values);
     void updateFilters(StkFloat *b, StkFloat *a);
     int getNumLines();
     void clear();
     double* computeSample(double in);
protected:
     /* Members */
     int numLines;
     DelayL *delayLines;
     OnePole *lowpass;
     double *matrix;
private:
     StkFloat *temp;
     double *out;
};
#endif //FDNObject
           * FDNObject Class
* Written by Rebecca Stewart
* August 2006
```

```
* A fully configurable template for a feedback delay network
* User defines the number of delay lines, all values for delay
* lengths, all coefficients for absorbent filers, and values
* for the feedback matrix.
* An array equal to the number of delay lines is output when
* samples are computed. This is so the user can choose
* desired number of channels and mix(down) output as desired.
#include "FDNObject.h"
#include <math.h>
/* Creates a blank FDN of declared number of lines
  No coefficients have been entered in anything*/
FDNObject :: FDNObject(int numDelays)
      int i;
     numLines = numDelays;
      delayLines = new DelayL[numDelays];
     lowpass = new OnePole[numDelays];
     matrix = new double[numDelays*numDelays];
     out = new double[numDelays];
      temp = new StkFloat[numDelays];
      //Set maximum delay for delay lines
      for(i=0; i<numDelays; i++) {</pre>
            delayLines[i].setMaximumDelay(5000);
      }
}
/* Deletes all dynamically allocated members */
FDNObject::~FDNObject()
      delete [] delayLines;
     delete [] lowpass;
     delete [] matrix;
     delete [] out;
      delete [] temp;
}
/* Returns the number of delay lines in current instace */
int FDNObject::getNumLines()
{
     return numLines;
}
/* Updates the feedback matrix with the values */
void FDNObject::updateMatrix(double *values)
{
      int i, j;
      for(i=0; i<numLines; i++) {</pre>
            for(j=0; j<numLines; j++) {</pre>
                 matrix[(i*numLines)+j] = values[(i*numLines)+j];
            }
      }
}
/* Updates the delay lengths of the delay lines */
void FDNObject::updateDelays(StkFloat *values)
```
```
{
      int i;
      for(i=0; i<numLines; i++) {</pre>
             delayLines[i].setDelay(values[i]-1);
      }
}
/* Updates the filter coefficients */
void FDNObject::updateFilters(StkFloat *b, StkFloat *a)
{
      int i,j;
      for(i=0; i<numLines; i++) {</pre>
             for(j=0; j<1; j++) {</pre>
                    lowpass[i].setB0(b[j*numLines+i]);
             }
             for(j=1; j<2; j++) {</pre>
                    lowpass[i].setA1(a[j*numLines+i]);
             }
      }
}
/* Clears all values in the delay lines and filters */
void FDNObject::clear()
{
      int i;
      for(i=0;i<numLines; i++) {</pre>
             delayLines[i].clear();
      }
}
/* Computes the next sample through the system */
double* FDNObject::computeSample(double in)
{
      int row,col,i,j;
      double t;
      StkFloat stkIn;
      stkIn = (StkFloat)in;
             for(i=0; i<numLines; i++)</pre>
       {
             temp[i] = 0;
             for(j=0;j<numLines; j++)</pre>
             {
                    temp[i] += matrix[numLines*i+j]* delayLines[j].lastOut();
             }
             out[i] =
(double)delayLines[i].tick(lowpass[i].tick(temp[i])+stkIn);
      }
      return out;
}
```