Building the Erbe-Verb: Extending the Feedback Delay Network Reverb for Modular Synthesizer Use

Tom Erbe
UC San Diego Department of Music
tre@ucsd.edu

ABSTRACT

The paper describes the design of a Eurorack Synthesizer reverb module that extends the feedback-delay network reverb in several ways. The extensions are oriented around the needs of live electronic music: 1) the creation of abstract resonant spaces which may or may not have a resemblance to acoustic spaces, 2) quick modulation of all parameters – modeless morphing between all states of the reverb. In this short paper I will open up the design of the Erbe-Verb, show the techniques used, and how they combine to make a very flexible performance reverb processor.

1. INTRODUCTION AND DESIGN CONSIDERATIONS

The Erbe-Verb was designed for use in the Eurorack synthesizer system, a modular synthesizer format created by Doepfer in 1995 [1]. The goals for this processor were to create a reverb that would integrate into the ecosystem of a patchable synthesizer, both functionally and sonically.

Most reverb processors are used as the last effects device, the room that one places an instrument in. Although the Erbe-Verb supports this use, I wanted it to be more fully integrated into a synthesizer voice. For this reason, all parameters are fully modulatable, designed to be controlled by common control-rate signal generators such as envelope generators, low frequency oscillators and envelope followers. Each note, moment, or gesture can have it's own resonant character. In addition, all time-related parameters (modulation speed, pre delay time) can to be clocked to external pulses. And finally, when possible, the parameters can be modulated at the audio-rate to allow audio effects not usually associated with reverb like FM sidebands on the internal delay or absorption filters. By allowing all parameters to be freely interconnected and modulated by any other synthesizer module, the synthesist is given flexibility to create new ways to use and incorporate the reverb in a larger patch.

Although the sonic goal of the Erbe-Verb is to emphasize synthetic or unusual resonant spaces, its basic characteristics are the same as any other reverberation device or performance hall. It provides sustained sound after the source is silent. It shapes the timbre, possibly smoothing attacks, adding distinct echoes or diffusion. Finally, the reverb gives a strong sense of location, even when the resonant space is completely unrealistic. To achieve these basic goals, I studied many previous designs, incorporating elements of each. With this solid foundation, it gave me freedom to experiment with extensions to the design.

2. BASIC DESIGN

The Erbe-Verb is based on a 4-delay feedback delay network reverb (FDN) as proposed by both Michael Gerzon [2], and Puckette and Stautner [3]. This model was chosen after investigating several topologies: the early experiments of Schroeder, the multi-tap feedback delay line of Christopher Moore's Ursa-Major Spacestation [4], and David Greisinger's figure-8 loop [5].

The feedback delay network was chosen for several characteristics. First, with a unitary feedback matrix it is a lossless gain design. This gives it the ability to sustain indefinitely when the feedback gain is 1.0, and also decay exponentially when the gain is below 1.0. This ability to sustain indefinitely is of importance to frozen or infinite reverb. Second, it allows the use of several short delays to quickly build up echo density. Finally, early reflections can be easily inserted as multiple taps in the delay network. This prototype FDN makes for a very compact and efficient reverb that can be easily modulated and modified.

2.1 Added Diffusion - All Pass Filters

Percussive signals require even higher initial echo density than the FDN can provide. For this reason, allpass filters are placed before each delay in the FDN. The gain on the allpass filters can be varied from 0.0 to 0.8 so that the density of early reflections can be increased. It should be noted that although allpass filters have a flat frequency response over time, the tail will ring. For this reason, the gain is kept low enough to avoid audible ringing, and the relative delay times of each allpass are kept mutually prime so that there are no common resonances. The delay time for each allpass is short enough so that individual echoes are not heard, they are kept under 15 milliseconds to maintain fusion with the original sound [6]. To finish the basic design, first order low pass filters are placed in each delay branch. The filter frequencies (absorption) and the feedback gain (decay) are set uniformly for all branches of the network so that the reverb is kept balanced without any one delay time becoming dominant.

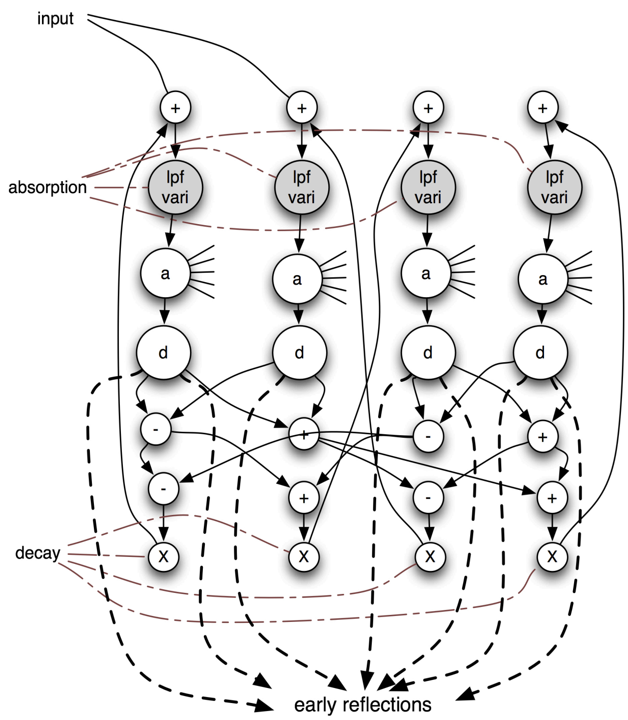

Figure 1. Basic reverb structure of Erbe-Verb showing all-pass filters (a), delay lines (d), low-pass filters, feedback matrix and early reflection taps.

3. EXTENSIONS TO THE FDN.

3.1 Parameter Range and Modulation

The first extension of this basic reverb design comes from simply giving the control parameters a large range, and attaching control voltage modulation to these parameters. For instance, the delay time is widely variable, and can be changed by signals as fast as 500Hz. The total delay variation ranges from as large as a football stadium to as small as a closet. The internal delay times and all early reflection times are continuously variable and scaled with the same parameter. This modulation will create a mass of coordinated doppler shifts within the reverberated signal for a dramatic change when changed slowly. When the reverb delay time is modulated quickly, percussive effects can be created, and when modulated at audio rate, FM effects can be heard. Recently I saw a synthesist (Alessandro Cortini) using the Erbe-Verb in just this way, creating cymbal-like sounds with fast changes of the delay time. Other parameters can be modulated as well, adding doppler shift, tremulous sustain, etc.

3.2 Infinite Reverb

Creating long and infinite sustain is one of the primary goals of the Erbe-Verb. The feedback gain (decay) is given a maximum gain of 1.25, which can quickly put the reverb network into infinite sustain. This feedback is controlled by simple saturation functions (based on a 3rd-degree chebyshev polynomial), placed before the feedback matrix. The harmonic generation of the saturation stages is balanced by the lowpass filters. Consequently, because these stages counteract each other, varying the decay and absorption controls can create a large variety of sustained and infinite reverb effects. The Erbe-Verb also generates a control voltage corresponding to the overall level of the reverb, which can be used to negatively modulate the decay (like an audio compressor with a feedback topology), and control the sustain without saturation.

3.3 Periodic time modulation

Modulation of the delay time and allpass filter time has been used in many reverb designs to break up resonances from the ringing of the allpass filters or to stop the pulsing that comes from the combined overall delay time. Usually this time modulation uses a sine oscillator or random function to avoid the constant doppler shift which results from modulating by a constant slope waveform. The depth of modulation is kept low to minimize chorusing effects. In my design, I make available two types of modulation, with speed and depth completely under control of the user. The first type is a multiphase sine modulation section, with separate sine modulation for each of the four delay lines. The frequency is variable from 30 to .1 Hz., but can be synchronized to an external clock for a wider range, or to connect the modulation to the tempo of a piece. The modulation depth varies from 0 to 100% of the delay time.

3.4 Random time modulation

The Erbe-Verb also contains a random modulation scheme which is inspired by Gordon Mumma's piece [7] "Stressed Space Palindromes." In this piece, the room size and shape is changing randomly and quickly from sound to sound. My scheme uses four layer granular/raised cosine envelopes on each delay line, with each grain selecting a random delay time, constrained by the modulation depth. The modulation speed is the grain creation speed. This type of modulation is effective at breaking up resonances and pulsing, without adding any chorusing or doppler shift. At extreme settings (when the modulation depth is high) it imparts it's own characteristic sound. At a slow modulation rate one can easily hear the space change size, and at a high modulation rate it quickly adds noise sidebands to all harmonics.

3.5 Input and output processing

The remaining additions are basic input and output processing. At the input of the reverb is a simple delay that acts as a pre delay – the time before you hear the first reflections in a space. It can be synchronized to an external clock, and the delay buffer can also be reversed (with a slight overlap to avoid clicking). At the output of the reverb is a three-band shelf filter, which can quickly make the reverb brighter or darker. The input and output processing both help separate the reverberated signal from the unaffected signal.

Figure 2. Front panel of Erbe-Verb, showing controls, control voltage parameter inputs and outputs.

4. HARDWARE IMPLEMENTATION

There are several things to consider to implement this reverb as a Eurorack module. Naturally, the processor must be fast enough to run the algorithm, and contain enough memory for the delays, tables and filters. The reverb also requires multiple DC-coupled analog to digital convertors so that all parameters can be controlled, and a high fidelity audio codec. We chose ARM Cortex M4 processor, because of its floating point processor, its multiple 12-bit ADCs for control voltage, and its easy interface to external memory and audio codecs. Also, there is GCC support for this chip, as well as a community of audio developers using it for their projects. Porting the code from the Pure Data prototype to the ARM chip was as simple as recoding the abstractions to C functions. It should be noted that all math library functions (cos(), atan(), pow(), etc.) were replaced with interpolated table lookup specfic to the range of the controls and the algorithm.

The only difficulty in the transition from the prototype Pure Data patch to the final hardware reverb is the stabilization of the control voltage inputs. The 12-bit ADCs on the ARM processor have a fair bit of noise and spurious input samples [8]. If values from these ADCs are used to directly control delay-time related parameters (size and pre-delay), the convertor noise and jitter can quickly become audible. This problem is addressed by running the ADCs at the highest possible sample rate, and downsampling with several cascaded low-pass filters, while at the same time rejecting outlier sample values. This eliminates the convertor noise, and leaves a high enough bandwidth to quickly modulate any parameter.

5. CONCLUSION AND FUTURE WORK

With the release of the Erbe-Verb, I feel that I achieved most of my goals: a reverb processor that is capable of a wide variety of standard reverb sounds, but that also goes beyond these to create more abstract resonant spaces. I would like to expand upon this in a further software and PD abstraction release, illustrating all of these techniques in a patchable environment, but also including other reverb topologies.

Acknowledgments

I would like to thank Anthony Rolando and Matthew Sherwood of Make Noise Music for designing the Erbe-Verb hardware. I would also like to thank Anthony Rolando, Walker Farrell, Richard Devine, Miller Puckette, Anthony Burr and the many beta testers for all their valuable feedback, listening, and ideas.

.

6. REFERENCES

[1] Wikipedia contributors, “Doepfer A-100,” [Online]. Avaliable: http://en.wikipedia.org/wiki/Doepfer_A-100.

[2] Michael Gerzon, “Synthetic Stereo Reverberation,” *Studio Sound*. 1971.

[3] John Stautner and Miller Puckette, “Designing Multi-Channel Reverberators” *Computer Music Journal, Vol. 6, No. 1*, Spring, 1982, pp. 52-65.

[4] Christopher Moore, U.S. Patent 4,268,717, *Time-Modulated Delay System and Improved Reverberation using Same.* 1979.

[5] Jon Dattorro, “Effect Design: Part 1 Reverberator and Other Filters,” *J. Audio Eng. Soc. Vol. 45, No. 9, p. 660.*

[6] H. Wallach, E.B. Newman, & M.R. Rosenzweig, “The precedence effect in sound localization,” *The American Journal of Psychology, 62*, 1949, pp. 315-338.

[7] Gordon Mumma, “Stressed Space Palindromes,” *Brain in the Wire*, Brainwashed, 2002.

[8] STMicroelectronics, “How to improve ADC accuracy when using STM32F2xx and STM32F4xx microcontrollers,” App Note AN4073, 2013.