
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Parker, Julian

Efficient Dispersion Generation Structures for Spring Reverb Emulation

Published in:
Eurasip Journal on Advances in Signal Processing

DOI:
[10.1155/2011/646134](https://doi.org/10.1155/2011/646134)

Published: 01/01/2011

Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Parker, J. (2011). Efficient Dispersion Generation Structures for Spring Reverb Emulation. *Eurasip Journal on Advances in Signal Processing*, 2011, 1-8. [646134]. <https://doi.org/10.1155/2011/646134>

Research Article

Efficient Dispersion Generation Structures for Spring Reverb Emulation

Julian Parker

Department of Signal Processing and Acoustics, Aalto University, 02150 Espoo, Finland

Correspondence should be addressed to Julian Parker, julian.parker@tkk.fi

Received 22 September 2010; Revised 21 December 2010; Accepted 9 February 2011

Academic Editor: Federico Fontana

Copyright © 2011 Julian Parker. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Spring reverberation is a sonically unique form of artificial reverberation, desirable as an effect distinct from that of more conventional reverberation. Recent work has introduced a parametric model of spring reverberation based on long chains of all-pass filters. Such chains can be computationally expensive. In this paper, we propose a number of modifications to these structures, via the application of multirate and multiband methods. These changes reduce the computational complexity of the structure to one third of its original cost and make the effect more suitable for real-time applications.

1. Introduction

Spring reverberation is an early method of artificial reverberation, first proposed by Hammond in the 1940s [1]. Standard spring reverberators consist of a configuration of one or more springs in parallel. Each spring has magnetic beads attached to its ends. Torsional motion is excited in the spring wire by applying an input signal to an electromagnetic coil, which then exerts a force on the magnetic bead at one end of the spring. At the opposite end of the spring, the movement of the magnetic bead induces a current in another electromagnetic coil, producing an output signal.

Spring reverberation gained popularity in the late 1950s and 1960s, in both studio applications and within instrument amplifiers, due to its relatively small size and inexpensiveness compared with the other methods available at the time. This method of reverberation has a unique sound, due to the highly dispersive nature of wave propagation through the spring. This sound became important to musicians as an effect distinct from standard reverberation [2], and, therefore, replication of this effect for use in a modern digital signal processing context is desirable.

Previous work on digital replication of spring reverberation has focussed mainly on physical modelling, using waveguide techniques [3] and finite difference techniques [2, 4, 5]. Recent work has introduced a spring reverberation

effect based on nonphysical modeling techniques. This model is structured similarly to a conventional Schroeder-style digital reverberator [6], with the addition of long chains of all-pass filters which produce the required dispersion [7]. This method approximates the main features of a spring reverberator well, and its parametric nature makes it highly flexible. However, the large chains of all-pass filters necessary can be more computationally expensive than is desirable for real-time applications, especially in the context of computer-based music production, where many audio synthesis and processing algorithms must run concurrently in real time. In this paper, we introduce a number of possible modifications which greatly reduce the overall computational cost of this structure, by application of multirate methods and by splitting the signal into multiple bands to exploit the relationship between the “stretch factor” and the impulse-response length of the “stretched” all-pass filters used [8].

In Section 2, we give a brief overview of the parametric spring reverberation model whose efficiency we are attempting to improve. In Section 3, we describe the principles of the multiband method used and how this method and others can be applied to the spring reverberation structure to produce a reduction in computational cost. In Section 4, we evaluate the total savings in computational cost provided by these modifications. In Section 5, we conclude.

2. Parametric Spring Reverberation with All-Pass Filters

The property of spring reverberation that gives it its special sound is its dispersivity—the way different frequencies take differing amounts of time to traverse the spring from one end to the other. In fact, the relationship between frequency and traversal time is not one-to-one, it is one-to-many. A particular frequency may travel through the spring at a number of different speeds simultaneously [4].

Like any reverberant system, the impulse response of a spring is made up of a series of echoes of the input impulse. However, this series of echoes has a number of qualities which distinguish it from, for example, a reverberant room. Firstly, the series of echoes produced by the spring is relatively regular and equally spaced, in contrast to the sparse set of irregularly spaced early reflections followed by a later diffuse tail which is exhibited by a room [6]. Secondly, individual echoes are altered more drastically by the system than is usual in a reverberant space—in addition to the usual attenuation and change in frequency content of the echoes, the dispersivity of the spring bends the echoed impulses into chirps.

Broadly speaking, the impulse response of a spring can be thought of as two series of these dispersive echoes or chirps. The main series of chirps is present only below a certain frequency, referred to as the transition frequency, f_C . This frequency is dictated by the geometry of the spring, and its material properties [4]. The group delay has a maximum at f_C . The second set of echoes is present over the whole audio band, but at much lower amplitudes than the lower-frequency echoes. This set of chirped echoes has its maximum group delay at low frequencies, decreasing as frequency increases.

The chirped echoes produced by a spring reverberation unit can be modelled in a number of ways. Välimäki et al. [7] propose the use of long cascades of all-pass filters, known as “spectral delay” filters [8]. In this case, the starting point is a first-order all-pass filter, whose transfer function is given by

$$A(z) = \left(\frac{a + z^{-1}}{1 + az^{-1}} \right), \quad (1)$$

where a is the all-pass filter coefficient. We then replace the unit delay within the filter with a longer delay element, giving the following equation:

$$H_{\text{single}}(z) = A(z^k) = \left(\frac{a + z^{-k}}{1 + az^{-k}} \right). \quad (2)$$

The length of this delay line, k , is known as the “stretch factor”. This stretching process aliases the curve describing the relationship between group-delay and frequency, squashing it and reflecting it around a number of points and hence resulting in several peaks of maximum group-delay. The stretching process also increases the overall group-delay of the structure multiplicatively, hence “stretching” the chirp which describes the filter’s impulse response. This structure can also be extended to allow noninteger values

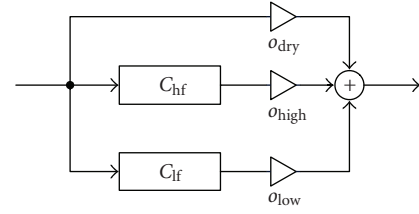


FIGURE 1: Block diagram showing high-level structure of the parametric spring reverberation model [7].

of k by replacing one of the internal unit delays with an interpolating first-order all-pass filter [7]. Finally, many of these stretched all-pass filters are cascaded, lengthening the chirp even further. This cascaded structure is given by

$$H_{\text{cascade}}(z) = A^M(z^k) = \left(\frac{a + z^{-k}}{1 + az^{-k}} \right)^M, \quad (3)$$

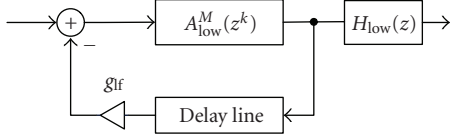
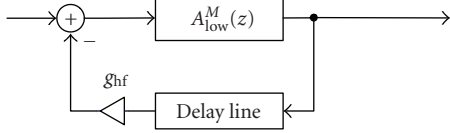
where M is the number of cascaded sections.

Välimäki et al. [7] propose a simple algorithm to produce the spring reverberation effect. It consists of two parallel delay lines with feedback, with an appropriately designed spectral delay filter placed within each feedback loop to produce the dispersion required. One feedback structure produces the chirped echo sequence below f_C . This structure is referred to as C_{lf} . This feedback structure is followed by a lowpass filter, in order to suppress any behaviour above f_C . The other feedback structure produces the wide-band chirped echoes and is referred to as C_{hf} . The parallel C_{lf} and C_{hf} structures are shown in Figure 1 whilst their internal structures are shown in Figures 2 and 3, respectively. The output of these structures is then mixed with a user-variable amount of the “dry” input signal to produce the final output of the effect.

Välimäki et al. [7] also include other features in the design of the effect, including random modulation of the delay lines, cross-coupling of the two feedback structures and an equalisation filter, in order to reproduce other qualities of the spring reverberator’s sound. These features are computationally light (on the order of 10 multiplications per sample period) compared to the dispersion generating all-pass filters, and, therefore, are ignored for the purpose of this work.

3. Efficient Dispersion Generating Structures

3.1. Chirp Straightening. In this section we describe a method for reducing the number of all-pass filters necessary to model the dispersion required by the spring reverberation effect. The genesis of this method lies in a basic property of the “stretched” all-pass filters used [8]—the strong dependence of the total group-delay provided by one all-pass section on its stretch factor, k . If we maximize the stretch factor of the filters used, we can reduce the number of cascaded sections needed to produce the required dispersion, and, hence, reduce the computational load. The problem then becomes finding a way of increasing the stretch factor of

FIGURE 2: Simplified block diagram showing C_{lf} structure.FIGURE 3: Simplified block diagram showing C_{hf} structure.

the filter whilst still producing a relationship between group-delay and frequency of approximately the right form.

In the case of the C_{lf} structure, it can be seen intuitively that if the stretch factor of the filters used is doubled and the all-pass coefficient changed from positive to negative, the cascade maintains a maximum group delay at the appropriate frequency, but also gains an additional maximum of group delay at DC. Shown in Figure 4(a) is the chirp before the coefficient negation and doubling of the stretch factor and, in Figure 4(b), the chirp produced after these alterations. Importantly, the modified structure has a minimum of group delay at a frequency below which the unmodified structure also has a very low group-delay. One can, therefore, produce a hybrid structure which approximates the unmodified chirp by splitting the signal with a crossover filter, presenting the upper band to the dispersion-generating all-pass chain and allowing the lower band to pass unfiltered.

Due to the feedback present in the system, a Linkwitz-Riley crossover filter [9] seems desirable due to its overall all-pass frequency response once the bands are recombined. The crossover frequency is chosen as the point of minimum group-delay of the all-pass chain, which is at half the chosen transition frequency f_c . The crossover order must be fairly high to suppress any cross-talk between the bands. An 8th-order crossover, where each crossover filter is produced by cascading two 4th-order Butterworth filters, proves to be a good compromise between steepness and the problems introduced by moving to a higher-order filter, such as extra group-delay around the crossover point.

In order to time-align the two bands, it is necessary to insert a delay line in the undispersed path, matched to the group-delay of the dispersion-generating all-pass chain at the crossover frequency. The group-delay of the all-pass chain can be calculated using the following expression [8]:

$$D = kM \frac{(1 - a^2)}{1 + 2a \cos(\omega k) + a^2}, \quad (4)$$

where D is the total delay in samples, k is the stretch factor, M is the number of cascaded all-pass stages, a is the all-pass filter coefficient, and ω is the angular frequency.

A fractional delay is necessary for precise time alignment of the two bands. A good fit for this purpose is a standard all-pass interpolated fractional delay line. However, in contrast to usual applications where a specific fractional delay time at DC is required, we need to match a particular fractional delay at an arbitrary frequency. The equation specifying the group-delay of a first-order all-pass filter is

$$D = \frac{(1 - a^2)}{1 + 2a \cos(\omega) + a^2}, \quad (5)$$

where D is the total delay in samples, a is the all-pass filter coefficient, and ω is the angular frequency. We can solve this equation with respect to a , producing the following equation:

$$a = \frac{-D \cos(\omega) \pm \sqrt{1 - D^2 + D^2 \cos^2(\omega)}}{1 + D}. \quad (6)$$

This expression allows us to design a first-order all-pass filter that possesses a specific fractional delay at a specific frequency. At angular frequencies of 0 and π , corresponding to DC and the Nyquist frequency, respectively, this equation collapses to give only one solution. At other arbitrary frequencies between these two values, two solutions are produced, one corresponding to a negative value of a and one to a positive value of a . In the case of the C_{lf} structure, it is more appropriate to use the solution with a positive a , as this places the maximum of group-delay within the region above the crossover frequency, where the fractional delay line is receiving little or no signal.

The response of this two-band chirp generation structure including the time alignment is shown in Figure 4(c). As can be seen, the group-delay of the modified structure is very similar to that of the unmodified structure, albeit with a slightly steeper transition between low and high group delay, and obviously no dispersion within the lower half. A generic block diagram of this two-band chirp generation structure is shown in Figure 5. We refer to this modification as “chirp straightening”, as it is based on discarding an area of the chirp where dispersion is low and replacing it with a straight impulse.

Another drawback of this structure compared to the original structure is a very small notch in the frequency response at the crossover frequency. This is very minimal for early chirps, but visible once echoes have cycled through the system a few times. This notch appears because the time-alignment delay line only matches the phase between the bands at exactly the crossover frequency. At frequencies close to the crossover but offset towards the dispersive region, the phases of the bands differ and cancellation occurs. The effect is relatively minor, but could be further suppressed by the use of a higher-order time-aligning all-pass filter to match both the group-delay of the upper band and its derivative(s).

3.2. Modifying C_{lf} . The structure C_{lf} can be made less computationally intensive via the chirp-straightening method described in Section 3.1. However, this is not the only saving

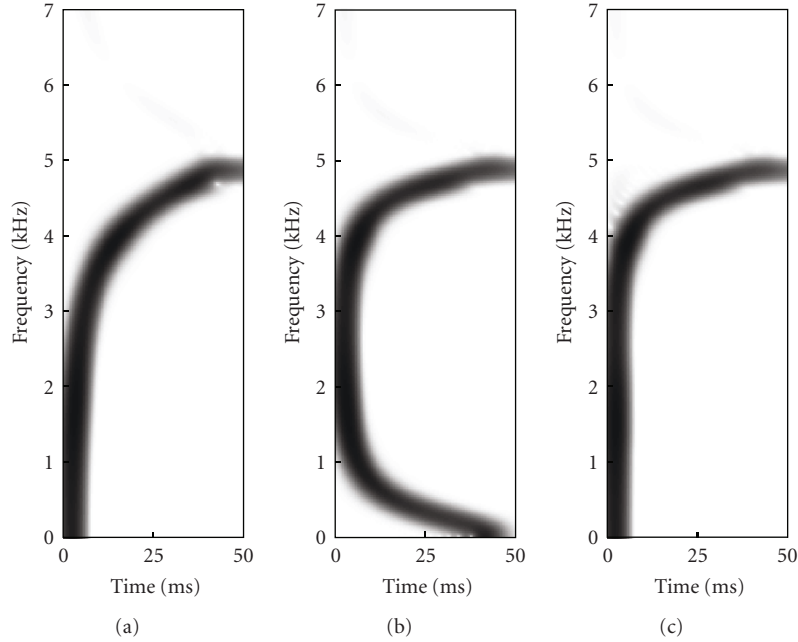


FIGURE 4: Spectrograms showing (a) a single C_{lf} chirp, produced by the original structure, (b) the modified chirp, and (c) the modified chirp after straightening of the bottom half. All chirps have been low-pass filtered to remove information above the desired f_c .

possible. C_{lf} is intended to only produce sound at frequencies below the transition frequency f_c , and, therefore, it can be run at a lower sample rate than the rest of the structure without degradation of sound quality. To avoid the overhead associated with an arbitrary change in sampling rate, it is best to downsample to a rate which is an integer division of the main sampling rate, rather than downsampling to f_c . An f_c of above 5 kHz is rarely seen in real spring reverberators, and, therefore, it makes sense to run the C_{lf} structure at $f_s/4$, where f_s is the master sampling rate, assuming a standard audio sampling rate of 44.1 kHz.

The downsampling and upsampling stages require an antialiasing and reconstruction filter, respectively. These should be kept relatively efficient in order to preserve the gains produced by the reduction in sampling rate. We already have an elliptic lowpass filter H_{low} at the output of the C_{lf} structure, whose cutoff is set below the new Nyquist frequency. We can, therefore, also use this filter to perform the reconstruction, at no extra computational cost, albeit losing the potential savings of running the elliptic filter at the lower sample rate. For the antialiasing filter, we take a standard approach and design a Type 1 Chebyshev lowpass filter [10] of order 10, with moderate (2 dB) pass-band ripple allowed. The cutoff frequency is set to be just below the Nyquist frequency of the downsampled structure. The application of multirate methods [11] could result in an antialiasing filter design with superior characteristics and lesser computational cost, but such an approach is beyond the scope of this paper.

A spectrogram of the impulse response of the modified C_{lf} structure can be seen in Figure 6, as well as the equivalent impulse response produced by the unmodified structure

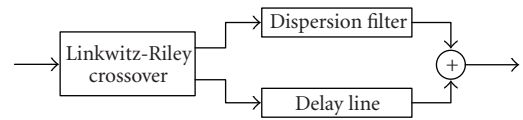


FIGURE 5: Block diagram showing generic structure used to produce straightened chirps.

for comparison. A block diagram describing the modified C_{lf} structure is given in Figure 7. Table 1 shows how the parameters of the original all-pass chain are changed by the straightening process, and to account for the downsampling of the structure.

3.3. Modifying C_{hf} . The structure C_{hf} can be made more efficient by applying the methods described in Section 3.1 in a slightly modified form. The wide-band chirped echoes produced by C_{hf} have a point of maximum group-delay at DC, and little dispersion above 10 kHz. Therefore, no change in all-pass coefficient is needed in this case. We can simply increase the stretch factor to $k = 2$, hence doubling the amount of dispersion produced by each cascaded stage. This produces another maximum in group-delay at the Nyquist frequency, with the minimum group delay present at $f_s/4$. We, therefore, place the crossover at this frequency, and allow only the lower band to be processed through the all-pass filter cascade. The dispersion is minimal in a large range around the crossover frequency, and, therefore, the order of the Linkwitz-Riley crossover can be reduced to 4, consisting of two cascaded second-order Butterworth filters for each band, without producing any adverse effects. The

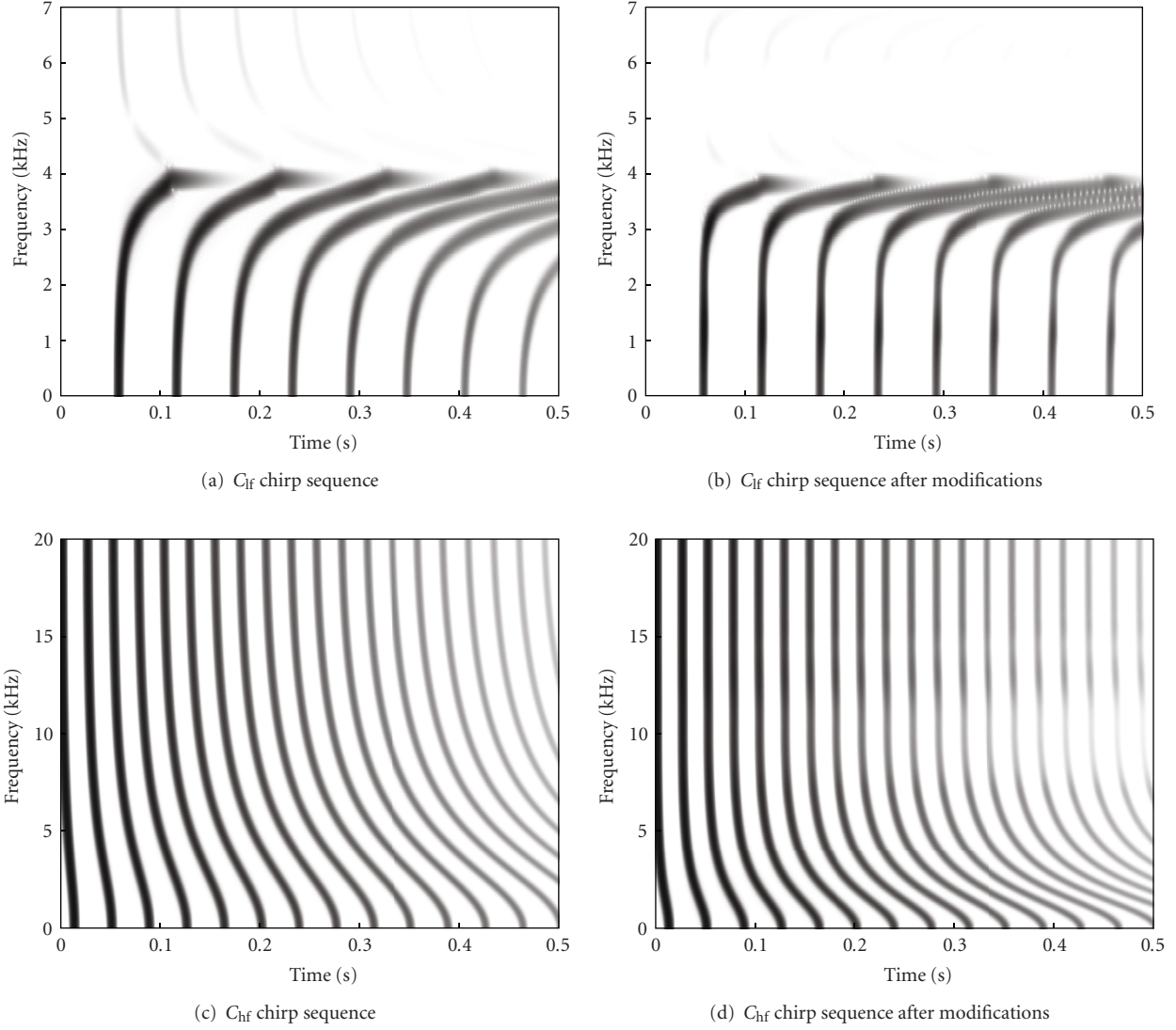
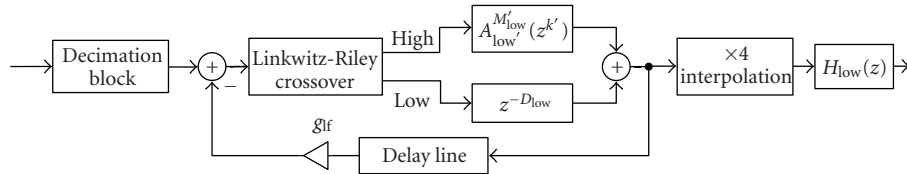


FIGURE 6: Spectrograms showing chirp sequences produced by the original and modified structures.

FIGURE 7: Block diagram showing C_{lf} with modifications.

time-alignment fractional delay is designed according to the method described in Section 3.1. However, this time the negative a solution is chosen in order to place the maximum of group delay of the interpolating filter below the crossover frequency. A block diagram describing the modified C_{hf} structure is given in Figure 8. Table 1 shows how the parameters of the original all-pass chain are changed by the straightening process.

4. Results

In order to properly evaluate the savings in computational cost which the modifications described above provide, it is necessary to quantify the computational cost in some way. We do this by counting the number of multiplications required by the effect before and after implementation of the suggested modifications. In order to count the number of

TABLE 1: Modifications required to the parameter values of the all-pass cascades in structures C_{lf} and C_{hf} , after chirp straightening and downsampling.

Structure	Parameter	Original	After straightening	After downsampling
C_{lf}	Stretch factor	k	$2k$	$k/2$
	All-pass chain coefficient	a_{lf}	$-a_{lf}$	$-a_{lf}$
	Chain length	M_{low}	$M_{low}/2$	$M_{low}/2$
C_{hf}	Stretch factor	1	2	n/a
	All-pass chain coefficient	a_{hf}	a_{hf}	n/a
	Chain length	M_{high}	$M_{high}/2$	n/a

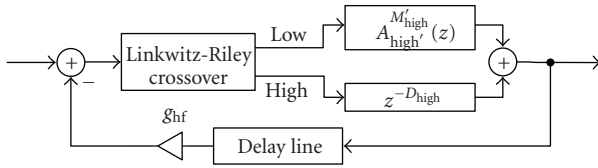


FIGURE 8: Block diagram showing C_{hf} with modifications.

multiplications necessary, we must make some assumptions about the implementation of the structures involved. Firstly, we assume that all filters are implemented in direct form II (DFII), and hence require $2(N + 1)$ multiplications, where N is the order of the filter in question. Some of the filters in the effect structure could potentially be implemented in a more efficient manner, particularly the all-pass filters [12]. However, assuming the use of DFII provides us with a reasonable baseline estimate of complexity in order to show the savings produced by the changes. Table 2 shows a comparison of the number of multiplications, broken down into sections of the structure, before and after the modifications are applied. Values shown are given as multiplications per sample period at full f_s , therefore, the sections which are undersampled can produce a fractional number of operations. In this example, we assume initial all-pass chain lengths before modification of $M_{low} = 100$ and $M_{high} = 200$, as used by Välimäki et al. [7]. According to this technique of evaluating the savings, the new structure possesses around one third of the computational complexity of the unmodified structure.

This evaluation of the efficiency of the spring reverberation structure raises an interesting question. In its modified state, the majority of the computational cost of the effect is produced by the C_{hf} structure. Given that the wide-band chirped echo sequence that this structure replicates is less important perceptually in real spring reverberation units due to its low amplitude (generally at least 30 dB below the other chirp sequence) [4], it may be acceptable to discard it completely and use only C_{lf} . This would produce an even lower cost spring reverberation effect at the cost of some realism.

Figure 9 shows spectrograms of the impulse response of the complete parametric spring reverb model including the additional features described in [7], such as random delay line modulation and a series of “pre-echoes” before each of

TABLE 2: Multiplications required (per sample period at full f_s) for effect structure, broken into sections and shown with and without the suggested modifications.

Section	Element	Original mult.	Efficient mult.
C_{lf}	$A_{low}^{M_{low}}$	600	75
	g_{lf}	1	0.25
	H_{low}	18	18
	L-R crossover	0	10
	Time-aligning delay	0	2
	Antialiasing filter	0	20
C_{hf}	$A_{high}^{M_{high}}$	400	200
	g_{hf}	1	1
	L-R crossover	0	24
	Time-aligning delay	0	2
General	Output mix gains	3	3
Total		1023	355.25

the main low-frequency chirps. Note that the downsampling of the C_{lf} structure means that care must be taken over the interpolation method used for the random delay line modulation, in order to avoid unwanted filtering of the signal. We, therefore, use low-order Lagrange interpolation instead of the linear interpolation originally suggested by Välimäki et al. [7]. The response produced by both the modified and original C_{lf} and C_{hf} structures are shown. The parameters of the effect are chosen as discussed in [7] and are of very similar value as the examples given in that work. The modified and unmodified structures appear to produce an output which is in reasonably close agreement, albeit with some changes in frequency-dependent reverberation time, especially around the crossover frequency in C_{lf} . Informal listening tests show that audibly the modified and unmodified model are reasonably similar. The notch in reverberation time introduced by the crossover is not easily audible. The straightening of the lowest part of the C_{lf} chirps is audible as a slight hardness of the echoes compared to the original structure, especially when presented with percussive material. However, the effect is not major, especially when listening to the effect applied to musical signals rather than examining the impulse response in isolation. If desired, the modified effect can be tweaked further in an attempt to more

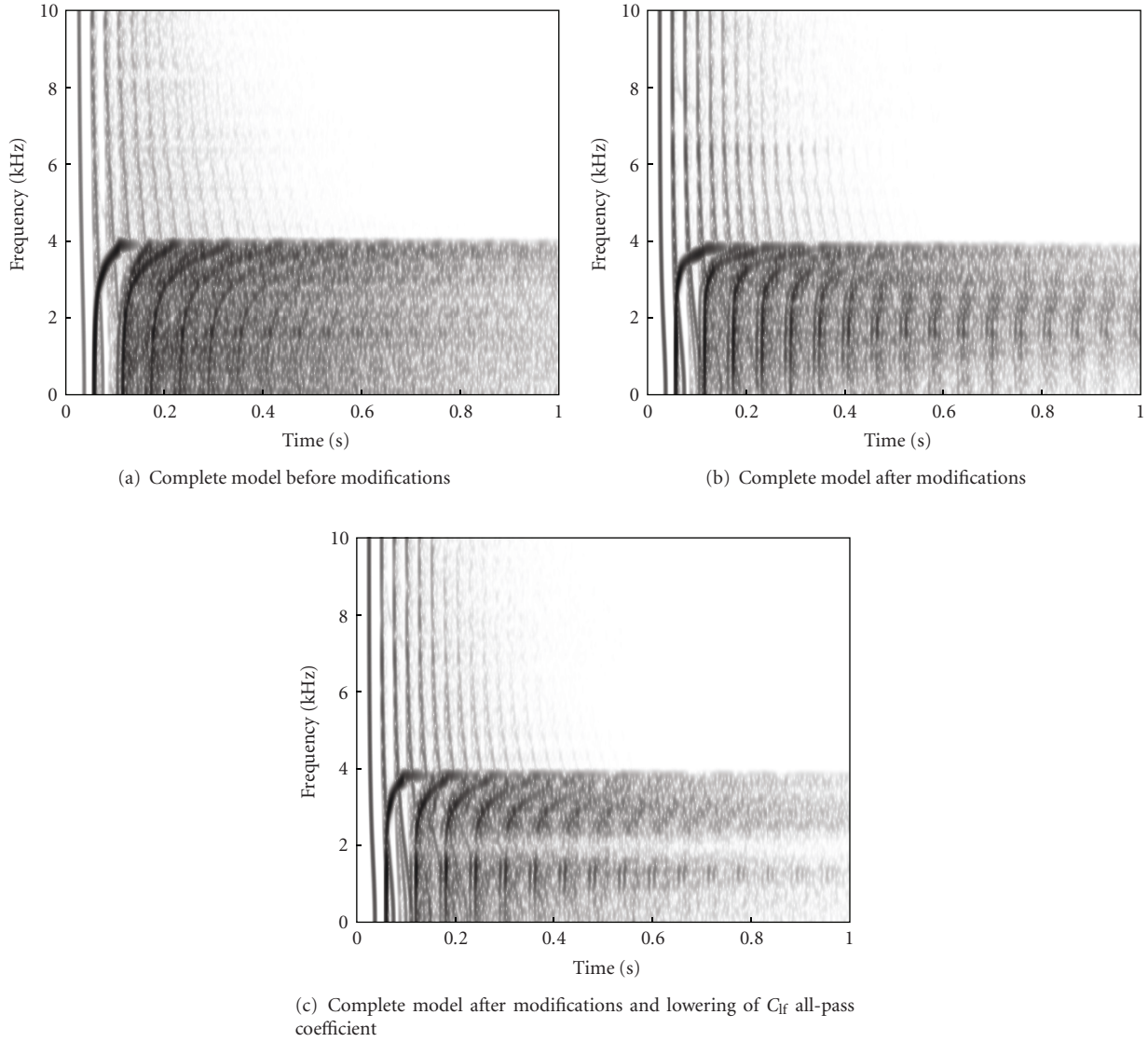


FIGURE 9: Spectrograms of the impulse response of the complete spring reverberation effect, before and after the efficiency modifications are implemented.

closely match the original effect. For example, the coefficient of the dispersion-generating all-pass filters used in C_{lf} may be lowered slightly, in order to match the original chirp shape a little more closely. The result of this alteration is shown in Figure 9(c). Unfortunately, this tweak comes at the cost of reducing the overall dispersion of the structure and exacerbating the notch in the frequency response around the crossover frequency, and, therefore, its appropriateness depends on the relative subjective importance of these qualities.

Sound examples can be obtained at the website associated with this work [13], and present the impulse responses of the original and modified structures, the impulse responses of the original and modified C_{lf} structures alone, and examples of the original and modified effect applied to a short vocal passage and a short synthesized drum loop.

5. Conclusion

In this work, we examined a simplified version of the parametric spring reverberation structure proposed by Välimäki et al. [7] and proposed a number of changes which greatly reduce its computational cost. These modifications are mainly based on splitting the signal into frequency bands which require dispersion and frequency bands which do not. We then design more efficient dispersion generation structures for the bands which do require dispersion, reducing the overall computational complexity. We also exploit the limited bandwidth of parts of the structure to apply multirate methods. The result is a structure that produces a similar spring-reverberation effect at approximately one third of the computational cost of the original structure proposed by Välimäki et al.

Acknowledgments

Many thanks are due to Vesa Välimäki for proofreading and guidance, and to the anonymous reviewers for improving this paper through their many useful comments. This paper was supported by the Academy of Finland (Project no. 122815).

References

- [1] L. Hammond, "Electrical musical instrument," US Patent no. 2230836, February 1941.
- [2] J. Parker, *Spring reverberation: a finite difference approach*, M.S. thesis, University of Edinburgh, 2008.
- [3] J. S. Abel, D. P. Berners, S. Costello, and J. O. Smith III, "Spring reverberation using dispersive allpass filters in a waveguide structure," in *Proceedings of the 121st Convention of the AES*, San Francisco, Calif, USA, 2006.
- [4] J. Parker and S. Bilbao, "Spring reverberation: a physical perspective," in *Proceedings of the 12th International Conference on Digital Audio Effects (DAFx '09)*, pp. 416–421, Como, Italy, 2009.
- [5] S. Bilbao and J. Parker, "A virtual model of spring reverberation," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 4, pp. 799–808, 2010.
- [6] W. G. Gardner, "Reverberation algorithms," in *Applications of Digital Signal Processing to Audio and Acoustics*, pp. 85–131, 2002.
- [7] V. Välimäki, J. Parker, and J. S. Abel, "Parametric spring reverberation effect," *Journal of the Audio Engineering Society*, vol. 58, no. 7-8, pp. 547–562, 2010.
- [8] V. Välimäki, J. S. Abel, and J. O. Smith, "Spectral delay filters," *Journal of the Audio Engineering Society*, vol. 57, no. 7-8, pp. 521–531, 2009.
- [9] S. P. Lipshitz and J. Vanderkooy, "In-phase crossover network design," *Journal of the Audio Engineering Society*, vol. 34, no. 11, pp. 889–894, 1986.
- [10] T. Parks and C. Burrus, *Digital Filter Design*, Wiley-Interscience, New York, NY, USA, 1987.
- [11] P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, New York, NY, USA, 1993.
- [12] S. K. Mitra and K. Hirano, "Digital all-pass networks," *IEEE Transactions on Circuits and Systems*, vol. 21, no. 5, pp. 688–700, 1974.
- [13] <http://www.acoustics.hut.fi/go/jasp-spring>.