Schlecht, Sebastian J.; Fierro, Leonardo; Välimäki, Vesa; Backman, Juha

# Audio peak reduction using a synced allpass filter

*Published in:*
Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2022

*DOI:*
10.1109/ICASSP43922.2022.9747877

Published: 01/01/2022

*Document Version*
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

*Please cite the original version:*
Schlecht, S. J., Fierro, L., Välimäki, V., & Backman, J. (2022). Audio peak reduction using a synced allpass filter. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2022* (pp. 1006-1010). (IEEE International Conference on Acoustics, Speech and Signal Processing; Vol. 2022-May). IEEE. https://doi.org/10.1109/ICASSP43922.2022.9747877

# AUDIO PEAK REDUCTION USING A SYNCED ALLPASS FILTER

*Sebastian J. Schlecht*[1,2], *Leonardo Fierro*[1], *Vesa Välimäki*[1]* *Juha Backman*

[1]Acoustics Lab, Dept. of Signal Processing and Acoustics
[2]Media Lab, Dept. of Art and Media
Aalto University, Espoo, Finland

AAC Technologies Solutions
Turku, Finland

## ABSTRACT

Peak reduction is a common step used in audio playback chains to increase the loudness of a sound. The distortion introduced by a conventional nonlinear compressor can be avoided with the use of an allpass filter, which provides peak reduction by acting on the signal phase. This way, the signal energy around a waveform peak can be smeared while maintaining the total energy of the signal. In this paper, a new technique for linear peak amplitude reduction is proposed based on a Schroeder allpass filter, whose delay line and gain parameters are synced to match peaks of the signal's autocorrelation function. The proposed method is compared with a previous search method and is shown to be often superior. An evaluation conducted over a variety of test signals indicates that the achieved peak reduction spans from 0 to 5 dB depending on the input waveform. The proposed method is widely applicable to real-time sound reproduction with a minimal computational processing budget.

***Index Terms—*** Audio systems, digital filters, music, optimization, sound enhancement.

## 1. INTRODUCTION

Dynamic range reduction of a signal via peak amplitude limiting is an established process in modern audio signal processing, as it can be used to restrict the dynamics of a sound and, consequently, maximize its loudness [1, 2]. A common compressor used for this purpose is called a limiter, which prevents a signal from exceeding the available dynamic range. In sound engineering and music production, limiting is applied in combination with a gain element to increase the perceived loudness by reducing the signal's peak–to–RMS (Root Mean Square) ratio [1, 3].

Traditional dynamic range reduction involves nonlinear techniques, which introduce new frequency components [3] and, consequently, harmonic distortion that can negatively affect the sound quality [2]. A linear approach to limiting was proposed by Parker and Välimäki by using an allpass filter (APF) chain [4]. An APF features a flat magnitude response and a nonlinear phase response. The APF has numerous applications in audio processing, such as artificial reverberation [5, 6, 7, 8], spectral delay filtering [9], delay equalization [10, 11], decorrelation [12, 13, 14], and a variety of audio effects [15]. The phase processing introduced by an appropriately designed APF chain provides distortion–free peak reduction by smearing the energy of transients in a short, limited range around their peak. The peak amplitude of the impulse response of the APF is minimized when the allpass coefficient is equal to the inverse of the golden ratio, or $\pm 0.618$ [16, 4].

A major drawback of Parker and Välimäki's technique is the large parameter space and the consequential high computational cost inherent to the optimization of the filter chain. Belloch et al. [17] studied the same problem extensively by performing an exhaustive search using a GPU accelerator to find optimal solutions over a wide space. However, a GPU may not be available in common audio equipment.

In this paper, a novel method for peak reduction using a Schroeder APF is proposed, introducing a computationally efficient approach to linear compression of the audio waveform. The delay of the APF is synced to the dominant peaks in the autocorrelation function, while the gain is derived from a fast local optimization based on the gradient of the filter response. Both parameters can be estimated quickly for a local peak in the signal waveform, thus making the method suitable for real-time processing.

This paper is structured as follows. Section 2 revisits the APF implementation. Section 3 introduces the new technique, describing the delay synchronization and the gain optimization. Section 4 reports a comparison with previous methods and a performance evaluation over a selection of test samples. Section 5 concludes the paper.

## 2. SCHROEDER ALLPASS FILTER FOR PEAK REDUCTION

The basic Schroeder APF [5, 8] is defined as

$$H_{\mathrm{S}}(z) = \frac{g + z^{-m}}{1 + gz^{-m}}, \tag{1}$$

*This research has been financed in part by AAC Technologies through the Planer project (Aalto Univ. project no. 410970). This work belongs to the activities of the NordicSMC network (NordForsk project no. 86892).

where $g \in [-1, 1]$ is the feedback gain and $m > 0$ is the delay in samples. A comprehensive overview of Schroeder APF properties can be found in the appendix of [8].

As the structure is sparse, the non-zero values of the Schroeder APF impulse response are placed on a regular grid. In turn, the Schroeder APF can generate large group delays with minimal computational effort. The feedback gain and delay parameters shape the group delay of the filter. Depending on the parameter choices, the filter can break up phase coherence and reduce the peak value of a signal.

The peak reduction method determines the filter parameters to solve the following problem:

$$\arg\min_{m,g} \max_{n} |y_{m,g}(n)|, \tag{2}$$

where $y_{m,g}(n) = (h_S * x)(n)$ is the processed signal with APF time–domain response $h_S$ and $x(n)$ is the input signal. The term $\max_n |y_{m,g}(n)|$ is the absolute peak value of the signal. We name the objective function

$$Y(m, g) = \max_{n} |y_{m,g}(n)| \tag{3}$$

the *absolute peak map*. The absolute peak map can be computed with an exhaustive grid search used for the ground truth and plotting of the map. Fig 1c presents an example.

In previous work, the solution of the allpass peak reduction problem required an exhaustive grid search on all parameter values [17]. In the following, we present a fast method to determine viable parameters.

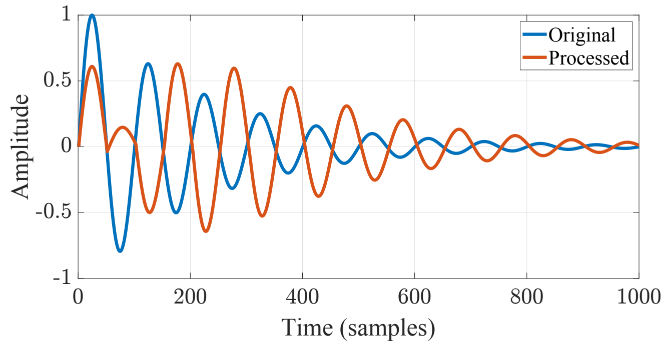
## 3. PROPOSED METHOD

This section introduces the novel peak reduction method. A systematic way to locate favorable values for parameters $m$ and $g$ of the Schroeder APF is described. All examples are at a sample rate of 44.1 kHz.

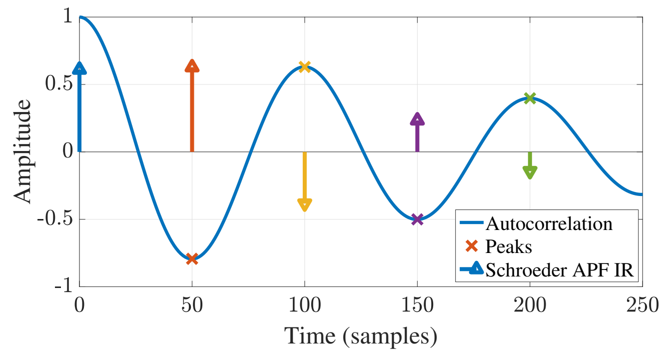
### 3.1. Delay choice

A motivating example of an exponentially–decaying sine wave is depicted in Fig. 1a. If the Schroeder delay $m$ is chosen to match the sine semiperiod (Fig. 1b), the phase smearing produces destructive interference over the first two cycles, reducing the overall peak level, as demonstrated by the red curve in Fig. 1a. The absolute peak map $Y(m, g)$ for the decaying sine is shown in Fig. 1c, overlayed with the normalized autocorrelation function $R_{xx}(m)$, where $n$ indicates the time lag. The lowest autocorrelation peak at $m = 50$ samples corresponds to a suitable delay for peak reduction with $g = 0.67$. The other autocorrelation peak time lags also lead to minima in the absolute peak map $Y(m, g)$, namely $m = 150, 250, \ldots$ for positive $g$ and $m = 100, 200, \ldots$ for negative $g$.

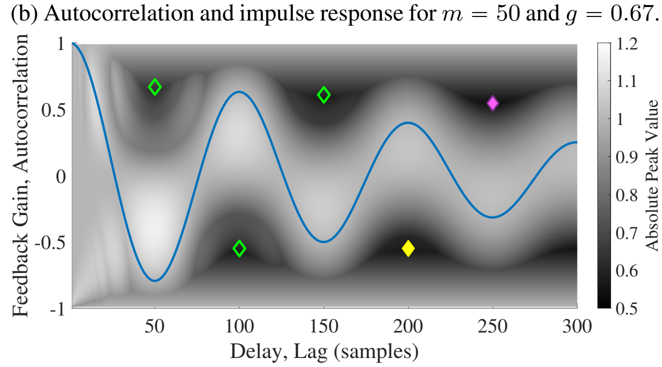
The absolute peak map highlights the initial challenge of locating globally optimal solutions, and how a poor choice




(a) Original and processed sounds for $m = 50$ and $g = 0.67$.




(b) Autocorrelation and impulse response for $m = 50$ and $g = 0.67$.




(c) Absolute peak map $Y(m, g)$ (grayscale) and autocorrelation (blue line). The y-axis indicates both APF feedback gain $g$ and normalized autocorrelation function $R_{xx}(m)$, where the x-axis indicates both APF delay $m$ and autocorrelation time lag. The filled diamonds indicate the solution for the grid search (pink) and the proposed synced APF (yellow), respectively; empty diamonds mark the discarded candidate solutions.

**Fig. 1**: Application of the proposed synced APF method to an exponentially–decaying sine wave.

of parameters leads to an undesired peak augmentation. On the bright side, there are many viable local optima with good results and the peak reduction varies smoothly for small changes in delay and gain. Overall, a good choice is found in syncing the delay to one among the two most dominant negative peaks and the two most dominant positive peaks of the autocorrelation function. A Schroeder APF following this design is named *synced APF* (SyncAPF). This design further reduces the search space, without significantly affecting the reduction performance.

It is good practice to limit the delay to avoid the perceptual degradation that typically follows excessive temporal distortion [4]. An empirically–found value for the maximum delay $M$ is 7 ms, or about 300 samples at the sample rate of 44.1 kHz [13]. It may be that neither a negative nor a positive peak is found in the autocorrelation below the defined maximum delay. In such a case, the first negative peak above $M$ is detected and a candidate delay is derived as $m = m_\mathrm{L}/k$, where $m_\mathrm{L}$ is the autocorrelation peak lag above $M$ and $k = 2, 3, \ldots$ is the minimum integer value for which $m_\mathrm{L}/k < M$.

### 3.2. Gain optimization

Let us now assume that delay $m$ is fixed and only gain $g$ is optimized. The feedback choice is solved with a gradient descent (GD) on the gain $g$, where the gradient is computed by a filter derivative and the optimal step size of the GD can be optimally solved with linear programming.

Given a processed signal $y_{m,g}$ with an initial gain $g$, the gain gradient at time $n$ is:

$$y'_{m,g}(n) = \frac{\partial y_{m,g}}{\partial g}(n) = (h'_\mathrm{S} * x)(n), \tag{4}$$

where $h'_\mathrm{S}$ is the impulse response of the Schroeder APF derivative. The corresponding transfer function is:

$$H'_\mathrm{S}(z) = \frac{\partial H_\mathrm{S}}{\partial g}(z) = \frac{1 - z^{-2m}}{1 + 2gz^{-m} + g^2 z^{-2m}}. \tag{5}$$

While the Schroeder allpass derivative is not allpass, it retains the sparsity and, consequently, the efficiency of the original filter: the total number of multiplications simply increases from one to two. Assuming the amplitude of $x(n)$ only spans the $[-1, 1]$ range, a conservative upper bound for the gradient is introduced based on (4):

$$\left|y'_{m,g}(n)\right| \leqslant \sum_{n=0}^{\infty} \left|h'_\mathrm{S}(n)\right|. \tag{6}$$

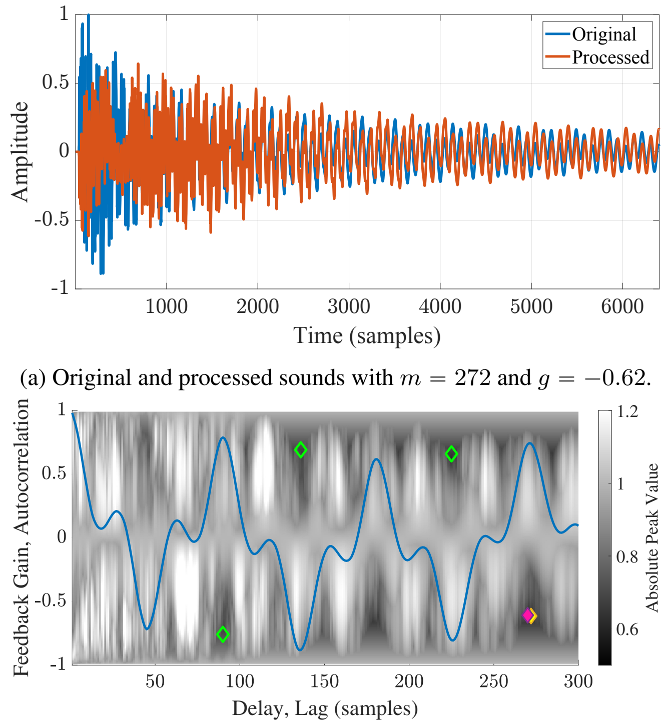
An important observation is that, for small changes of the feedback gain, signal samples carrying low values will not produce a relevant impact on the peak value, i.e. the peak amplitude variation is a smooth function of the gain.

A standard GD determines a suitable step size along the direction of the given gradient for the parameter update. The optimal step size $\gamma$ is given by stating the minimization problem (2) as a gradient descent:

$$\arg\min_{\gamma} \max_{n} \left|y_{m,g}(n) + \gamma\, y'_{m,g}(n)\right|, \tag{7}$$

which can be simplified by removing the absolute value:

$$\arg\min_{\gamma} \max_{n} \tilde{y}_{m,g}(n) + \gamma\, \tilde{y}'_{m,g}(n), \tag{8}$$




(a) Original and processed sounds with $m = 272$ and $g = -0.62$.

(b) Absolute peak map $Y(m, g)$ and autocorrelation, cf. Fig. 1c.

**Fig. 2**: Application of the SyncAPF method to a mallet sound.

where

$$\tilde{y}_{m,g}(n) = |y_{m,g}(n)|; \quad \tilde{y}'_{m,g}(n) = \frac{y_{m,g}(n)}{|y_{m,g}(n)|} y'_{m,g}(n) \tag{9}$$

are the sign–flipped amplitude and gradient lines. This is only accurate if $y_{m,g} + \gamma\, y'_{m,g}(n)$ does not cross the zero axis in the region of interest. Nonetheless, this simplification is quasi–optimal: as the gradient is relatively small, only large values of $\tilde{y}_{m,g}(n)$ contribute to the solution for small step sizes. This is equivalent to the following linear program:

$$\min_{\alpha} \text{ s.t. } \forall n : \tilde{y}_{m,g}(n) + \gamma\, \tilde{y}'_{m,g}(n) \leqslant \alpha. \tag{10}$$

Thus, we get the step size $\gamma$ and we can update the values for the new gain $g_{i+1} = g_i + \gamma$, where $i$ is the current iteration.

Linear programming can be sped up with a good initialization of $g$, as GD is bound to only find local minima and multiple initializations might be necessary. A statistical pre–evaluation determined that $g = \pm 0.7$ are strong candidates for gain initialization as those are the most consistently impacting gain values for the SyncAPF, see Figs. 2b and 1c for supporting examples.

Another element affecting performance and computational cost of GD is the number of iterations. Multiple experiments suggest the choice of three iterations for the differential allpass, as it has been shown to provide an acceptable trade–off between computational time and achieved peak reduction.

**Table 1**: Comparison of peak reduction (in dB) on five test samples achieved with different methods. The best result in each case is highlighted. SyncAPF is the proposed method.

| Method | Bass | Snare | Hihat | Piano | Mallet |
|---|---|---|---|---|---|
| Parker [4] | 0.5 | 2.3 | 1.4 | 1.3 | 1.2 |
| Belloch [17] | **2.6** | 3.2 | 2.5 | 2.3 | 2.7 |
| Grid Search | 2.1 | **4.9** | **3.6** | **3.1** | **4.3** |
| SyncAPF | 0.6 | 4.3 | 2.0 | 2.3 | 3.6 |

The cost of an iteration step is 1 multiplication (MUL) and 2 additions (ADD) (APF) plus 2 MUL and 3 ADD per sample (filter derivative), or a total of 3 MUL and 5 ADD per sample. The overall computational cost can be further reduced with fast linear programming [18] and by selectively updating peak signal and gradient values.

The exponentially–decaying sine example described above provides further insight on the delay choice algorithm. The peak reduction map for gains $g \in [-1, 1]$ and delays $m \in [0, 300]$ samples indicates a best possible peak reduction (Fig. 1c, pink diamond marker) of 5.2 dB with $g = 0.54$ and $m = 250$. SyncAPF achieves 4.8 dB with $g = -0.55$ and $m = 200$, which corresponds to the second–largest positive autocorrelation peak lag (yellow diamond marker). Discarded solutions are visualized with empty diamond markers.

Processing of a mallet percussion is shown in Fig. 2a and 2b, for the same gain and delay values. Here, the absolute peak map and SyncAPF provide similar solutions. The best peak reduction is 4.3 dB with $m = 270$ and $g = -0.61$, while SyncAPF achieves 3.9 dB with $m = 272$ and $g = -0.62$.

## 4. COMPARISON AND EVALUATION

This section compares the proposed methods with state–of–the–art methods [4, 17] and evaluates them on a dataset gathered for this study. Table 1 shows the reduction obtained for five test samples used in previous studies. The grid search for the Schroeder APF leads to the best reduction in four cases out of five. The method by Belloch et al. [17] remains slightly superior for the bass drum (Bass). The proposed method (SyncAPF) is also quite successful with less than 1 dB smaller reduction in most cases. Processed samples are available on the companion website for this paper [19].

The following computational costs are given for a signal frame of 2048 samples. The grid search exhaustively tests 300 delays and 100 feedback gain values. Each tested APF requires 3 flops (floating-point operations) per sample. Thus the grid search costs amount to 90,000 flops per sample. The costs of the SyncAPF include 1) autocorrelation of the signal frame and the peak finding which amounts to $5\log_2(2048)$ = 55 flops [20]; 2) gain optimization with three iterations for each of the four peaks, thus, $3\times4\times8$ flops = 96 flops per sample. In total, SyncAPF requires 151 flops per sample. In comparison, Belloch's method [17] requires 28,966,400 APFs, thus 86,899,200 flops per sample and Parker's method [4] requires 300 APFs, thus 900 flops per sample. The peak reduction costs only apply in general if a sufficiently high peak is present in the signal frame.

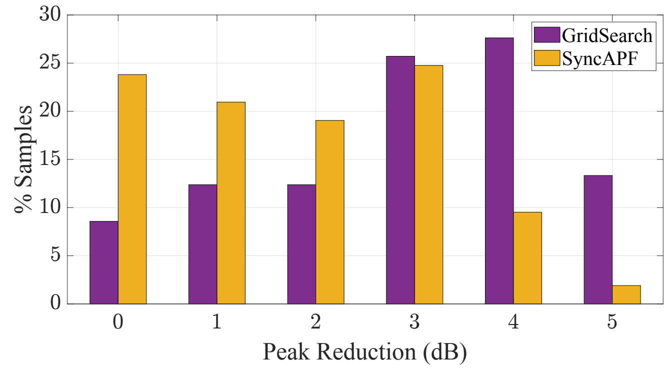



**Fig. 3**: Histogram of the peak reduction for 105 selected audio samples. Values have been rounded to the nearest integer.

Peak reduction performance was tested over 105 audio samples, containing different types of transients, e.g. percussions, speech excerpts, gunshots, beeps, and other impulsive sounds. Results are summarized in Fig. 3 for both the grid search and the SyncAPF. While the latter does not quite reach the ground truth set by the grid search (which provides at least 2.5 dB reduction for more than half the samples under test), it achieves at least 1 dB reduction for more than 65% of the samples under test. Sounds presenting a small peak–to–RMS ratio or pre-compressed audio are hard to reduce.

A secondary benefit achieved by SyncAPF is an improved symmetry of the audio waveform, as shown in Figs. 1a and 2a. This is particularly useful for small loudspeaker systems, such as modern amplifiers used in smartphones.

## 5. CONCLUSION

A novel method for linear audio compression by means of a Schroeder allpass filter was presented, whose delay is synced to the peaks of the autocorrelation function to inaudibly smear the energy around waveform peaks. The reduction is suitable for real–time use, given the affordable computational cost, and complements traditional nonlinear compression. In future work, the proposed method could be extended to frequency-dependent APF [7]. The method can be adapted to online processing by optimizing the APF for each signal frame. With this, continuity between frames requires additional attention to avoid signal discontinuities. A necessary trade-off is between frame size, look ahead, and perceptual quality.

## 6. ACKNOWLEDGEMENT

We thank Thomas McKenzie for his editorial suggestions.

## 7. REFERENCES


[1] R. A. Katz, *Mastering Audio: The Art and the Science*, Butterworth-Heinemann, 2003.

[2] E. Vickers, "The loudness war: Background, speculation, and recommendations," in *Proc. 107th Audio Eng. Soc. Conv.*, San Francisco, CA, USA, Nov. 2010.

[3] D. Giannoulis, M. Massberg, and J. D. Reiss, "Digital dynamic range compressor design—A tutorial and analysis," *J. Audio Eng. Soc.*, vol. 60, no. 6, pp. 399–408, June 2012.

[4] J. Parker and V. Välimäki, "Linear dynamic range reduction of musical audio using an allpass filter chain," *IEEE Signal Process. Lett.*, vol. 20, no. 7, pp. 669–672, July 2013.

[5] M. R. Schroeder and B. F. Logan, ""Colorless" artificial reverberation," *IRE Trans. Audio*, vol. 9, no. 6, pp. 209–214, Nov.–Dec. 1961.

[6] V. Välimäki, J. Parker, L. Savioja, J. O. Smith, and J. S. Abel, "Fifty years of artificial reverberation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 20, no. 5, pp. 1421–1448, July 2012.

[7] S. J. Schlecht, "Frequency-dependent Schroeder allpass filters," *App. Sci.*, vol. 10, no. 1, pp. 1–11, 2020.

[8] K. J. Werner, F. G. Germain, and C. S. Goldsmith, "Energy-preserving time-varying Schroeder allpass filters and multichannel extensions," *J. Audio Eng. Soc.*, vol. 69, no. 6, pp. 465–485, Jul./Aug. 2021.

[9] V. Välimäki, J. S. Abel, and J. O. Smith, "Spectral delay filters," *J. Audio Eng. Soc.*, vol. 57, no. 7/8, pp. 521–531, Aug. 2009.

[10] J. Rämö and V. Välimäki, "Graphic delay equalizer," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Brighton, UK, May 2019, pp. 8018–8022.

[11] J. Liski, A. Mäkivirta, and V. Välimäki, "Audibility of group-delay equalization," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 29, pp. 2189–2201, June 2021.

[12] E. K. Canfield-Dafilou and J. S. Abel, "Signal decorrelation using perceptually informed allpass filters," in *Proc. 19th Int. Conf. Digital Audio Effects (DAFx)*, Brno, Czech Republic, Sep. 2016, pp. 225–231.

[13] E. K. Canfield-Dafilou and J. S. Abel, "A group delay-based method for signal decorrelation," in *Proc. Audio Eng. Soc. 144th Conv.*, Milan, Italy, May 2018.

[14] C. Gribben and H. Lee, "The perception of band-limited decorrelation between vertically oriented loudspeakers," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 876–888, Mar. 2020.

[15] U. Zölzer (Ed.), *DAFX—Digital Audio Effects*, John Wiley & Sons, second edition, 2011.

[16] D. Griesinger, "Impulse response measurements using all-pass deconvolution," in *Proc. Audio Eng. Soc. 11th Int. Conf. Test & Measurement*, Portland, Oregon, USA, May 1992, pp. 308–321.

[17] J. A. Belloch, J. Parker, L. Savioja, A. Gonzalez, and V. Välimäki, "Dynamic range reduction of audio signals using multiple allpass filters on a GPU accelerator," in *Proc. 22nd European Signal Process. Conf. (EUSIPCO)*, Lisbon, Portugal, Nov. 2014, pp. 890–894.

[18] P. M. Vaidya, "Speeding-up linear programming using fast matrix multiplication," in *Proc. 30th Annual Symp. Foundations Comp. Sci.*, Research Triangle Park, NC, USA, Oct. 1989, pp. 332–337.

[19] S. J. Schlecht, L. Fierro, and V. Välimäki, "Audio peak reduction using a synced allpass filter: Companion webpage," `http://research.spa.aalto.fi/publications/papers/icassp22-peak/main.htm`, Accessed: 2021-10-06.

[20] S. G. Johnson and M. Frigo, "A modified split-radix FFT with fewer arithmetic operations," *IEEE Trans. Signal Process.*, vol. 55, no. 1, pp. 111–119, Jan. 2007.