

Dynamic Signal Phase Distortion Using Coefficient-Modulated Allpass Filters

JOSEPH TIMONEY,¹ *AES Member*, JUSSI PEKONEN², VICTOR LAZZARINI¹, AND
(jtimoney@cs.nuim.ie) (jussi@pekonen.cc) (victor.lazzarini@nuim.ie)

VESA VÄLIMÄKI,³ *AES Fellow*
(vesa.valimaki@aalto.fi)

¹*Maynooth University, Music Technology Research Group, Maynooth, Co. Kildare, Ireland*

²*Independent researcher, <http://pekonen.cc>, Espoo, Finland*

³*Aalto University, Department of Signal Processing and Acoustics, P.O. Box 13000, FI-00076 AALTO, Espoo, Finland*

Recent work has shown that if the coefficients of a first-order allpass filter are modulated over time, it behaves as a dynamic Phase Distortion (PD) device. A key advantage of this arrangement is that the filter input and the modulation signal can, within stability constraints, be arbitrary. Furthermore, no heterodyning or interpolation, as used for the techniques of Adaptive PD and Adaptive FM, respectively, is required. This now broadens the palette of sound effects that allpass filters are capable of generating and opens up a new perspective through which they can be configured for sonic modification purposes. However, a rigorous analysis of this dynamic device is not available and the purpose of this paper is to draw on the theory of time-varying filters to thoroughly understand its properties in the time and frequency domains. The significant influence of filter topology on the transient behavior of the output is also investigated. Finally, a procedure for mapping from a desired PD function to a modulation signal is also illustrated. This can be applied to derive modulation functions that impart particular timbral properties to the input, and it is used to create examples demonstrating the capabilities of this new technique.

0 INTRODUCTION

Allpass filters have a special place in musical signal processing, particularly in the area of digital audio effects. Their ability to delay an input signal by fractional sample values or in a frequency-dependent way means that they are key components of many well-known effects algorithms, such as reverberation [1–3], chorus [4], and flanger [5] units, or can be cascaded together to create phasers [5] or spectral delay effects [6]. Recent work has now added the ubiquitous distortion effect to the list [7]. It was shown in [7] that by modulating the coefficients of the allpass filter a time-varying phase delay is created that when applied to an input signal will result in the alteration of its waveshape, leading to the appearance of new spectral components in the output signal. This essentially creates a dynamic version of the sound synthesis technique known as Phase Distortion (PD) [8], which is a subset of phase or frequency modulation (FM) synthesis [9], and is a phase domain version of waveshaping [10]. It is an efficient way for using simple mono-component waveforms to generate others of a much more complex timbral, and consequently musically interesting, structure. This allpass filter-based approach is an alternative to that of Adaptive PD given in [11] and Adaptive FM [12]. It is wholly flexible because the input and the modulation

can be arbitrary signals but it does not require the heterodyning operation for Adaptive PD or the interpolation necessary for Adaptive FM, and thus is computationally more efficient to implement. Furthermore, this extends the usefulness of the allpass filter structure for audio effects so that it now indeed becomes a general-purpose signal processing component that is configurable to achieve a wide variety of sonic outcomes. It implies too that a typical sound effect unit that has distortion and time-delay based effects could be designed using allpass filter structures alone. This will inspire new perspectives for generating dynamic sound effects.

The relationship between the coefficient-modulated allpass filter and PD was developed in [13]. Following this, it was shown how to choose a modulation function to generate band-limited sawtooth and square waveforms [14]. The synthesis possibilities from chaining a number of time-varying allpass filters were discussed in [15] and for second-order allpass filters in [16]. The work in [11] discussed how it performed as an adaptive effect with arbitrary input signals, and moreover, also illustrated the influence that the implementation structure of the filter has on transients appearing in the output. Each of these contributions advanced the understanding of these filters. However, it was clear that a comprehensive study is required to fully grasp how to use them effectively.

These modulated allpass filters are time-variant filters and their analysis must be different to that for time-invariant filters for a number of reasons. Small variations in the filter parameters can change its characteristics dramatically [17], and changing parameter values also can give rise to transients [18, 19]. Additionally, topologies that are equivalent in the time-invariant case are not when the filters vary over time [20]. More importantly, a time-varying filter whose coefficients at any given time are those of a stable time-invariant filter will not necessarily be stable [21]. Thus, analyzing time-variant filters needs a different set of theoretical tools. The literature on time-varying filters is diverse. Approaches include impulse response analysis [17], state-space analysis [22], frequency domain analysis [23, 24], and approximation [25]. The most comprehensive work identified was [20], particularly for Periodic Linear Time-varying (PLTV) Systems. Recently, this theory was applied to Feedback Amplitude Modulation (FBAM) synthesis systems [26, 27]. The coefficient-modulated allpass filter is more complicated than the FBAM system but the principles are similar, and for the remainder of this article the term PLTV allpass filter will be used to refer to the coefficient-modulated first-order allpass filter. However, a caveat must be made, in that although the magnitude of the frequency response of the unmodulated LTI filter structure is allpass, it will be shown that this is not strictly the case for the time-varying version, but that it will be almost allpass.

The article is structured as follows. A brief illustration of the phase delay properties of the allpass filter is given first. This is followed by an explanation in Sections 1 and 2 of how it can be analyzed in the time and frequency domains using PLTV theory. In particular, Section 1 will explain constraints on the modulation function to ensure stability. Section 3 will then examine various implementation structures and show how each realization influences the presence of transients in the output. Section 4 will present two sound synthesis examples to illustrate its operation. Section 5 will then conclude the article.

0.1 The PLTV Allpass Filter

The transfer function of the standard first-order digital allpass filter can be written as:

$$H(z) = \frac{-a + z^{-1}}{1 - az^{-1}} \quad (1)$$

where a denotes the allpass coefficient [28].

The equation for the relationship between the value of the allpass coefficient and the phase shift introduced at a particular frequency in the input signal is given by [29]

$$\phi(\omega) = -2 \tan^{-1} \left(\frac{1+a}{1-a} \tan(\omega/2) \right) \quad (2)$$

where ω denotes the frequency in radians.

The upper panel of Fig. 1 shows a plot of the phase shift introduced by the allpass filter for coefficient values $a \in [0, 1]$ at frequencies corresponding to those of the musical note C at various octaves. The lower panel is a plot of the phase shift for coefficient values $a \in [-1, 0]$, again at

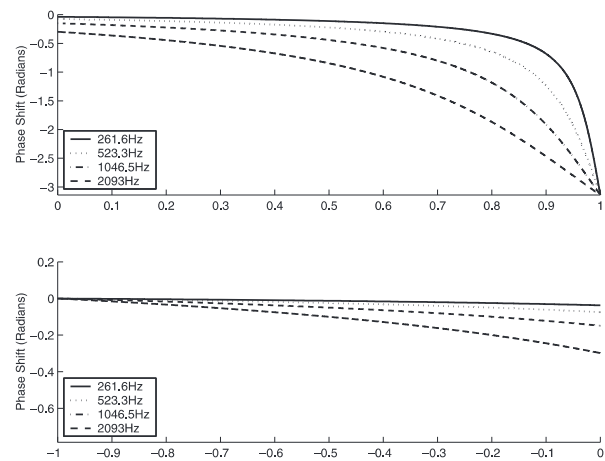


Fig. 1. Relationship between positive (upper panel) and negative (lower panel) allpass coefficient value and phase shift at different frequencies.

the frequencies of C. When the coefficient is positive the size of the phase shift increases nonlinearly with increasing coefficient value going from a value close to 0 up to $-\pi$. The relationship is more nonlinear at lower frequencies. However, when the coefficient value is negative the size of the phase shift is much smaller for all coefficient values, and is 0 at all frequencies for a coefficient value of -1 . Therefore, the effect of the coefficient-modulated allpass filter is most pronounced when the modulation function is positive.

Now that the fundamental relationship has been shown, Section 1 will give the relevant equations for analysis of the PLTV allpass filter in the time domain. The theoretical results will be illustrated with an example.

1 TIME DOMAIN ANALYSIS OF THE PLTV ALLPASS FILTER

In [7], the PLTV allpass filter was originally implemented as a time-varying difference equation with input $x(n)$, output $y(n)$ and a periodic time-varying modulation function $m(n)$, where n is the sample index and whose period is given as N :

$$y(n) = m(n) y(n-1) - m(n) x(n) + x(n-1) \quad (3)$$

The impulse response of a time-varying system is also known as Green's function [20]. Unlike linear and time-invariant systems, the output response depends on the moment of the observation as well as the moment of the signal application. Therefore, in a time-variant discrete system the impulse response, denoted as $h(p, n)$, is a function of two time variables p and n , representing the sample instants at which the impulse is applied and observed, respectively. As shown in [20], a time-varying IIR filter can be decomposed into a cascade of a time-varying FIR filter with a time-invariant IIR system whose overall impulse response can be written

$$h(p, n) = \sum_{k=p}^n h_{\text{FIR}}(p, k) h_{\text{IIR}}(k, n) \quad (4)$$

The impulse response of the non-recursive element of Eq. (3) can be expressed in matrix form as:

$$\mathbf{h}_{\text{FIR}}(p, n) = \begin{bmatrix} -m(0) & 1 & 0 & 0 \\ 0 & -m(1) & 1 & 0 \\ 0 & 0 & -m(2) & 1 & \dots \\ 0 & 0 & 0 & -m(3) & \dots \\ \vdots & & & & \end{bmatrix} \quad (5)$$

Also from [20], the impulse response of the recursive element of Eq. (3) can be written as:

$$h_{\text{IR}}(p, n) = \begin{cases} m(n) h_{\text{IR}}(p, n-1) + \delta(p, n), & \text{for } 0 \leq p \leq n, \\ 0, & \text{for } n < 0 \text{ \& } p > n. \end{cases} \quad (6)$$

Eq. (6) can also be written as [27]:

$$h_{\text{IR}}(p, n) = \begin{cases} \prod_{k=p+1}^n m(k) = \frac{g(n)}{g(p)}, & 0 \leq p \leq n, \\ 1, & p = n, \\ 0, & p > n, \\ 0, & n < 0, \end{cases} \quad (7)$$

where

$$g(k) = \prod_{i=1}^k m(i) \quad \text{for } k \geq 1 \quad \text{and} \quad g(0) = 1 \quad (8)$$

As the modulation function $m(n)$ is periodic with period N , the coefficients of the recursive filter are also periodic. This presents an advantage to the analysis as there is an explicit relationship between coefficients of successive periods. If we define the current sample time n as the combination of an integer number of periods and fraction thereof,

$$n = \mu N + \nu \quad (9)$$

where μ is an integer and μN represents an integer number of periods, and ν is also an integer representing the number of samples in the current incomplete period.

From Eq. (8), an expression can be written to take account of this,

$$g(n + N) = g(N) g(n), \quad (10)$$

where

$$g(N) = \prod_{k=1}^{N-1} m(k). \quad (11)$$

Returning to $h_{\text{IR}}(p, n)$ it is possible to use Eq. (8) to express it as:

$$h_{\text{IR}}(\eta N + \xi, \mu N + \nu) = g^{\mu-\eta}(N) h_{\text{IR}}(\xi, \nu), \quad (12)$$

where η is also an integer and ηN represents an integer number of periods, and ξ is an integer representing the number of samples in the current incomplete period.

Eq. (12) says that to compute the recursive impulse response at any point in time all we need to know is the impulse response for the first period only and the product of the filter coefficients. A further useful result is that we

can verify the stability of the system by Eq. (11). For system stability, the impulse response should decrease over time [20], and from Eq. (11) this will be ensured if

$$g(N) < 1 \quad (13)$$

Eq. (13) can be interpreted to mean that at any given time within the period the value of a filter coefficient can be greater than one and it will not disrupt system stability as long as the product of all the coefficients within the complete period is less than one.

Examining Eq. (12), it can also be understood that an expression for the transfer function of the recursive part in steady state mode could be written as:

$$H_{\text{IR}}(z, n) = \frac{\sum_{\xi=0}^{N-1} h_{\text{IR}}(\xi, \zeta) z^{-\xi}}{1 - g(N) z^{-N}} \quad (14)$$

where taking account of the periodicity of the impulse response, we can define

$$\zeta = n - N \lfloor \frac{n}{N} \rfloor, \quad (15)$$

and $\lfloor \cdot \rfloor$ represents the floor function

Eq. (14) shows that the transfer function is a combination of a non-recursive numerator and an inverse comb filter whose delay is N samples. This is a very interesting result because considering Eq. (3) in light of Eqs. (4) and (14), it means that a significant element of the signal shaping mechanism of the coefficient-modulated allpass filter is purely FIR. Eq. (12) can also be expanded in a similar manner to Eq. (5) and will result in

$$\mathbf{h}_{\text{IR}}(p, n) = \begin{bmatrix} 1 & m(1) & m(1)m(2) & m(1)m(2)m(3) \\ 0 & 1 & m(2) & m(2)m(3) \\ 0 & 0 & 1 & m(3) & \dots \\ 0 & 0 & 0 & 1 & \dots \\ \vdots & & & & \end{bmatrix} \quad (16)$$

Then, by applying the matrix descriptions of Eqs. (5) and (16), it is possible to outline in matrix form how the combined response of Eq. (4) will appear

$$\mathbf{h}(p, n) = \sum_{k=p}^n \mathbf{h}_{\text{FIR}}(p, k) \mathbf{h}_{\text{IR}}(k, n) = \begin{bmatrix} -m(0) & 1 & m(2) - \prod_{l=0}^2 m(l) & m(2)m(3) - \prod_{l=0}^3 m(l) \\ 0 & -m(1) & 1 - \prod_{l=1}^2 m(l) & m(3) - \prod_{l=1}^3 m(l) & \dots \\ 0 & 0 & -m(2) & 1 - \prod_{l=2}^3 m(l) \\ 0 & 0 & 0 & -m(3) \\ \vdots & & & & \end{bmatrix} \quad (17)$$

To illustrate using an example how an impulse response of the form of Eq. (17) evolves over time we first define a unipolar modulation as:

$$m(n) = 0.01 + 0.9((x(n) + 1)/2), \quad (18)$$

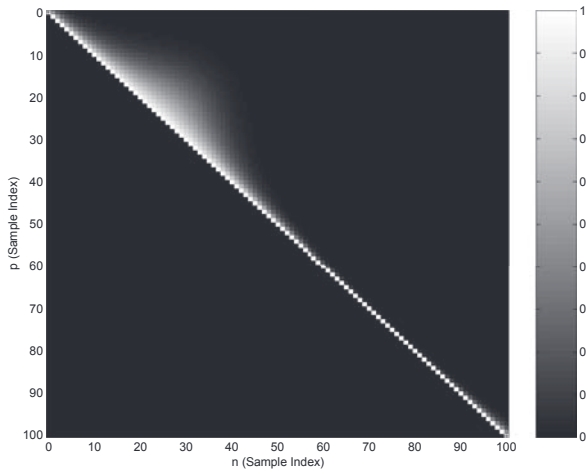


Fig. 2. One Period of the PLTV allpass impulse response for the modulation function in Eq. (18).

where the input to the PLTV allpass filter is

$$x(n) = \sin(0.01\pi n) \tag{19}$$

This input has a period of 100 samples, which corresponds to frequency 441 Hz when the sampling rate is 44.1 kHz.

The format of the modulation in Eq. (18) was used to ensure that the minimum value of the modulation was slightly greater than zero and thus prevent any divide by zero issues occurring in the calculations. The numerical evaluation of $g(N)$ from Eq. (11) for this modulation signal returns a value of approximately 0 which, from Eq. (13), indicates that the time-varying filter is stable.

Substituting the modulation function of Eq. (18) into Eq. (17), we can make a depth map image plot of the system impulse response over one period $N = 100$. This is shown in Fig. 2. The figure has a grayscale colorbar illustrating the mapping between the actual numerical value of the impulse response and the colors used, with white and black corresponding to the maximum (0.99) and minimum (-0.91) values, respectively. The time-varying nature of the impulse response can be observed in the variation of its numerical values between the maximum and minimum on the depth map. Also, looking along the diagonal its progression over time is highlighted.

The matrix in Eq. (17) grows over time. However, many of the entries of Eq. (17) will be formed from products of samples of the input modulation signal suggesting that it could be difficult to derive a tidy closed form expression for the impulse response in most cases. Thus, in general a computational approach should be used for its evaluation. Finally, if the input to the PLTV allpass filter is denoted $x(n)$, then the output $y(n)$ can be determined using [20]

$$y(n) = \sum_{p=0}^n x(p)h(p,n) \tag{20}$$

Using the input and modulation defined in Eqs. (18) and (19), along with the time-varying impulse response of Eq. (17), we can compute the output as shown by the solid line in

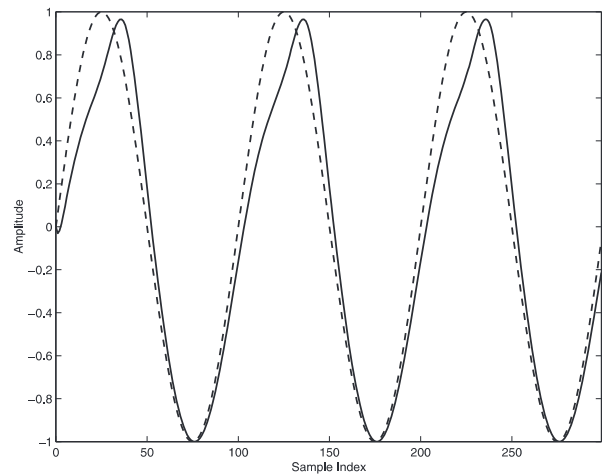


Fig. 3. Input (dashed line) and Output (Solid line) of the PLTV allpass filter.

Fig. 3. The distortion in its shape from that of the sinewave input, as given by the dashed line, can be seen in the plot confirming that this is the consequence of modulating the filter coefficient.

2 FREQUENCY DOMAIN ANALYSIS OF THE PLTV ALLPASS FILTER

Moving away from the time-domain, it is also useful to examine the PLTV allpass filter in the frequency domain. It is very worthwhile to find the time-varying magnitude and phase response of the filter. For actual synthesis applications, it is important to have an expression that relates the modulation function to the time-varying phase. It is also possible to draw on the theory of FM synthesis to have an expression for the spectrum of filter output given a time-varying phase function. This can be useful for evaluating the bandwidth of the filter output. Another tool is the Bifrequency function which is a visual representation that illustrates the mapping between frequency components in the input and output. All these are explained next.

2.1 General Analysis

In a similar manner to the previous section, the overall frequency response needs to be written as a combination of two cascaded PLTV systems. The general expression is also available from [20], and can be written for our case as:

$$H(\omega, n) = \sum_{p=0}^n H_{\text{FIR}}(\omega, p)h_{\text{IR}}(p, n)e^{j\omega(p-n)} \tag{21}$$

where the time-varying frequency response of the FIR part is simply

$$H_{\text{FIR}}(\omega, n) = -m(n) + e^{-j\omega} \tag{22}$$

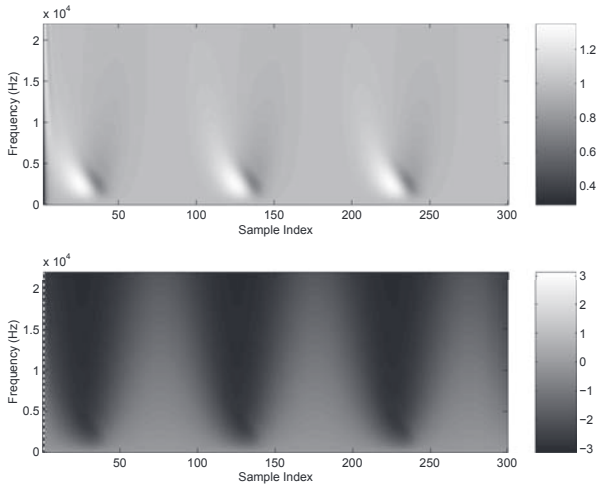


Fig. 4. PLTV allpass time-varying frequency response magnitude (upper panel) and phase (lower panel).

Substituting Eqs. (22) and (16) into Eq. (21) will lead to an expression for the overall PLTV allpass frequency response

$$H(\omega, n) = \frac{\sum_{p=0}^n [-m(n)e^{j\omega p} + e^{j\omega(p-1)}] h_{\text{IR}}(p, n)e^{-j\omega n}}{1 - \prod_{k=0}^{N-1} x(k)e^{-jN\omega}} \quad (23)$$

where $x(n)$ is the PLTV allpass filter input. It is worth noticing here that the input signal is an important element in the frequency response.

The frequency response in Eq. (23) is a complex quantity and can be represented as a time-varying magnitude and phase

$$H(\omega, n) = |H(\omega, n)| e^{j\angle H(\omega, n)} \quad (24)$$

Fig. 4 illustrates the time-varying magnitude and phase responses for our example. The number of frequency indices used to compute the frequency response was 1000, which represents from 0 Hz to half the sampling frequency. The upper panel shows the magnitude response and it appears to be almost flat with a magnitude of approximately 1, but with a peak periodically appearing in the low-frequency region. The phase response in the lower panel is changing over time and varies between $-\pi$ and π within each period as the modulation reaches its maximum and minimum value, respectively. The nonlinear shape of the time-varying phase response can be perceived from the plot.

To investigate further, it is worthwhile plotting the magnitude and phase responses at the same frequency as the input, that is, 441 Hz, for the example. These are given in Fig. 5. Here, although the magnitude response in the upper panel is centered around 1, there is a fluctuation that occurs as the phase response reaches its minimum value. Thus, the magnitude response of the PLTV allpass filter is not strictly allpass at every time instance when the coefficients are modulated. However, the RMS amplitude of this fluctuation in the magnitude response was found to be 0.0091 which is almost zero, and it can be concluded that

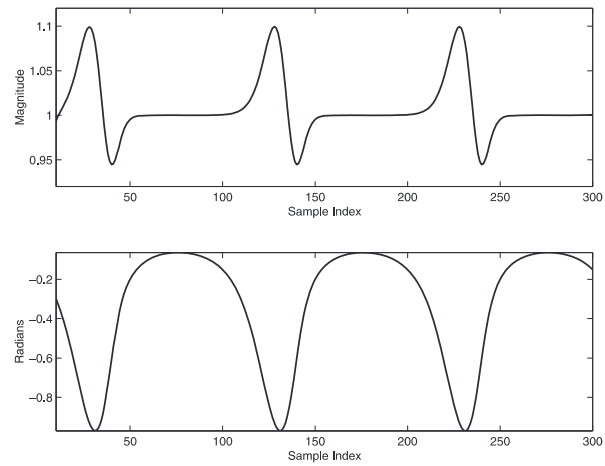


Fig. 5. Time-varying magnitude (upper panel) and phase response (lower panel) at 441 Hz only.

the modulation effects of the PLTV filter are primarily due to the time-varying phase response.

To establish the relationship between the modulation function and the time-varying phase, it is necessary to make some assumptions. Firstly, it is assumed that for the PLTV allpass that the time-varying magnitude in Eq. (24) is

$$|H(\omega, n)| \approx 1 \quad \text{for all } \omega, n \quad (25)$$

and, with respect to Eq. (2), the phase shift, denoted $\phi(\omega, n)$, introduced by the filter at a given frequency ω at time n is given by a time-varying version of the expression for the phase shift of a time-invariant allpass filter [13], that is,

$$\phi(\omega, n) \approx -\omega + 2 \tan^{-1} \left(\frac{-m(n) \sin(\omega)}{1 - m(n) \cos(\omega)} \right) \quad (26)$$

Additionally, the phase shift of Eq. (26) can be converted into a frequency deviation by differentiation with respect to time [13]

$$\dot{\phi}(\omega, n) \approx \frac{-2\dot{m}(n) \sin(\omega)}{1 - 2m(n) \cos(\omega) + m^2(n)} \quad (27)$$

where $\dot{m}(n)$ is the differential of the modulation function. Notice in Eq. (27) that if the modulation signal is not time varying, that is, $\dot{m}(n) = 0$, then the frequency deviation is also zero, that is, $\phi(\omega, n) = 0$.

The upper panel of Fig. 6 shows the actual time-varying phase from Fig. 5 (solid line) plotted with the approximation of Eq. (26) (dashed line). There is a good match between the two except that the approximation has deeper notches at the minimum points. A similar plot showing the instantaneous frequency deviations (centered around 441 Hz) is given in the lower panel. The actual instantaneous frequency deviation is found by differentiating the measured time-varying phase of Fig. 5 (solid line) while the approximation is from Eq. (27) (dashed line). They are broadly in agreement except most notably at the minimum points. The advantage of the instantaneous frequency deviation is that it can provide

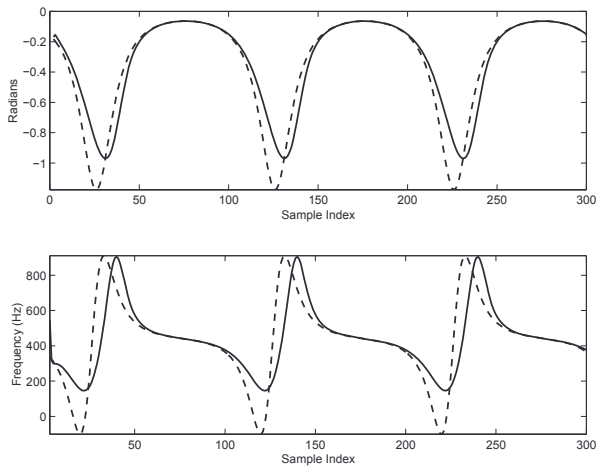


Fig. 6. Comparison of actual PLTV time-varying phase (solid line) and approximation of Eq. (26) (dashed line) in the upper panel, and PLTV instantaneous frequency deviation (solid line) with the approximation of Eq. (27) (dashed line) in the lower panel.

a more intuitive means of visualizing how the time-varying filter speeds up and slows down the signal distorting its waveshape and introducing new frequency components.

Complimenting Eq. (20), if the spectrum of the input to the PLTV allpass filter is denoted as $X(\omega)$, the time domain output signal can be written as [20]:

$$y(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) H(\omega, n) e^{j\omega n} d\omega. \quad (28)$$

2.2 Spectrum Analysis using Complex FM

To make use of the theory underlying the analysis of Complex FM signals, we assume that the output of the PLTV allpass filter for a mono-component input of frequency ω_0 is given by

$$y(n) = |H(\omega_0, n)| \sin(\omega_0 n + \angle H(\omega_0, n)) \quad (29)$$

To simplify the analysis, we assume again that the magnitude response of the allpass PLTV system is unity for all ω and n as per Eq. (25). Then, if the desired time-varying phase modulation of fundamental frequency ω_m , $\phi(\omega_m, n)$ is known and can be expressed as a combination of harmonically related sinusoids, the output $y(n)$ can be written as a Complex Frequency Modulated (FM) signal [30, 31]. The general expression is

$$y(n) = \sin(\omega_0 n + \phi(\omega_m, n)) = \sin\left(\omega_0 n + \sum_{k=1}^K I_k \sin(k\omega_m n + \varphi_k)\right) \quad (30)$$

assuming $|H(\omega_0, n)| = 1$ and $\angle H(\omega_0, n) = \phi(\omega_m, n)$.

In Eq. (30), the time-varying modulation is described by K harmonics each with magnitude I_k , and phase φ_k . From [32] then, the Complex FM signal described by Eq. (30) can be written as an additive equation in terms of its spectral

magnitudes

$$y(n) = \sum_{k_K} \dots \sum_{k_1} \left(\prod_{i=1}^K J_{k_i}(I_i) \right) \times \sin\left(\omega_0 n + \left(\sum_{i=1}^K k_i(k\omega_m n + \varphi_i) \right)\right) \quad (31)$$

where the $J_k(I)$ is a Bessel function of the first kind of order k with argument I . The indices k_1, \dots, k_K are each determined by the number of Bessel values required to describe each modulation spectrum. In theory these indices should be infinite but in practice they can be truncated to a limit such that the Bessel values are sufficiently small. Maximum values for the indices k_1, \dots, k_K , denoted $k_{1\max}, \dots, k_{K\max}$, could be determined such that the Bessel function terms $J_{k_{1\max}}(I_1), \dots, J_{k_{K\max}}(I_K)$ are less than -80 dB for example.

Eq. (31) has been described in [33] as illustrating that sidebands produced by the modulation of the carrier by the first modulating oscillator are modulated again as a carrier by the next modulation term. This process repeats for each successive modulation term. The spectrum of $y(n)$ described in Eq. (31) can also be written as a series of spectral convolutions [34–36] that can be conveniently rewritten as a single Fourier transform of a product,

$$Y(\omega) = \mathfrak{F}^{-1} \left\{ \mathfrak{F} \left\{ \mathfrak{F} \left\{ e^{j(\omega_0 n - \pi/2)} \right\} \right\} \times \mathfrak{F} \left\{ \mathfrak{F} \left\{ \sum_{k_1} J_{k_1}(I_1) e^{j(k_1 \omega_m n + \varphi_1)} \right\} \right\} \dots \times \mathfrak{F} \left\{ \mathfrak{F} \left\{ \sum_{k_K} J_{k_K}(I_K) e^{j(k_K(K\omega_m n + \varphi_K))} \right\} \right\} \right\} \quad (32)$$

where $Y(\omega)$ is the spectrum of $y(n)$, $\mathfrak{F}(\cdot)$ denotes the Fourier transform, and $\mathfrak{F}^{-1}(\cdot)$ denotes the inverse Fourier transform.

The component or term at the greatest distance in frequency from the carrier can be found from Eq. (32) to be

$$J_{k_{1\max}}(I_1) \dots J_{k_{K\max}}(I_K) e^{j(\omega_0 n - \pi/2 + k_1 \omega_m + k_{1\max} \varphi_1 + \dots + k_{K\max} K \omega_m n + k_{K\max} \varphi_K)} \quad (33)$$

By finding the complex magnitude of this furthest component it is possible to determine the actual bandwidth of the signal. Additionally, if

$$\omega_0 n + k_1 \omega_m + \dots + k_{K\max} K \omega_m < \pi f_s, \quad (34)$$

where f_s is the sampling frequency in Hertz, and if the magnitude

$$J_{k_{1\max}}(I_1) \dots J_{k_{K\max}}(I_K) \quad (35)$$

is sufficiently small, then it is certain that there will be practically no aliasing distortion present in the output of the allpass PLTV filter.

For output bandwidth assessment, there is a distinct advantage to employing this analytical approach over simply taking an FFT of the PLTV output. The FFT-based spectrum is constrained to be within the Nyquist limit and this obfuscates the true spectrum in the case where components

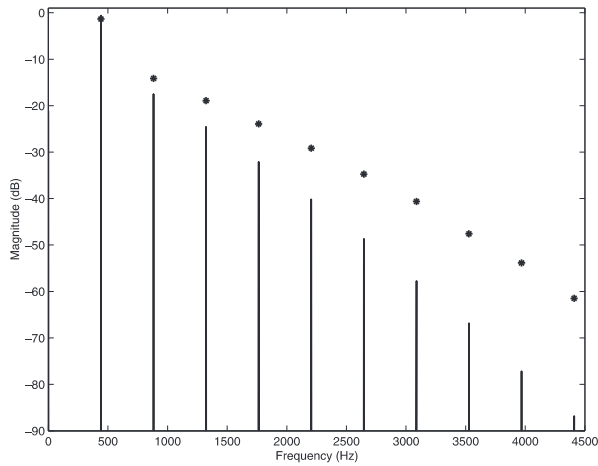


Fig. 7. Comparison of Spectrum of PLTV allpass output using FFT (solid line) and Complex FM equation (asterisks).

are being generated at higher frequencies than the Nyquist frequency, while they are apparent when analyzing using Eq. (32).

Returning to our example that was introduced in Section 1 with $\omega_0 = \omega_m = 2\pi 100$ rad/s, we first must take the phase modulation computed by Eq. (26) and represent it as a series of harmonics to fit with the expression on the right-hand side of Eq. (30). The magnitudes and phases of these harmonics will be I_1, \dots, I_K and $\varphi_1, \dots, \varphi_K$ respectively. Seven harmonics (i.e., $K = 7$) and their parameters were found using the super resolution CSPE algorithm [37]. The mean square error difference between the original and harmonic series approximation was 5×10^{-6} or -106 dB.

In Fig. 7, the spectrum of the PLTV allpass output computed using Eq. (32) is shown using asterisks alongside the spectrum of the time-domain output $y(n)$, calculated using an FFT. The length of this FFT was 44,100 points, resulting in a frequency resolution of 1 Hz, and a rectangular window was used. Although the results from Eq. (32) conform in a general sense with those of the FFT spectrum computation because both are lowpass in shape, according to Eq. (32) the magnitudes of the higher harmonics should be greater and it is thus inaccurate in detail.

The source of this discrepancy is due to the influence of the PLTV allpass filter on both the magnitude and phase of its input. By using Eq. (32), it is assuming that the output of the PLTV allpass filter is the result of a pure phase modulation. However, it was shown in Fig. 5 this is not the case. This can be confirmed by synthesizing a signal using the phase modulation shown by the dashed line in the upper panel of Fig. 6 only, and comparing this to the output computed using an inverse FFT on Eq. (32). The result is given in Fig. 8. This figure shows that a synthesized version of the output using the phase modulation only (upper panel), and a time-domain signal obtained (lower panel) are the same.

Interpreting Fig. 7, it can be seen that Eq. (35) does not exactly predict the spectral bandwidth of the output of the PLTV allpass filter but instead overestimates it. However, in

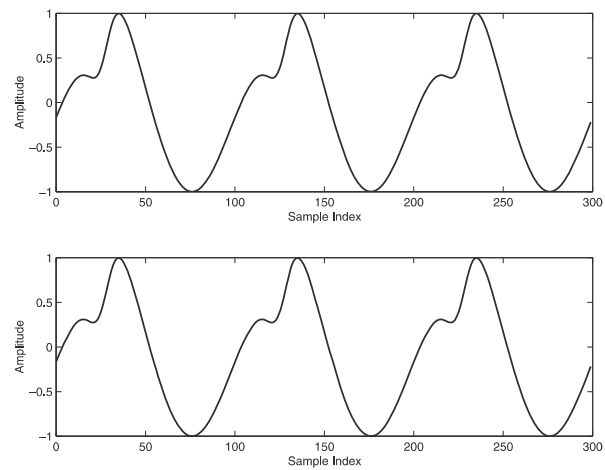


Fig. 8. Comparison of signal synthesized using time-varying phase only (upper panel) and the inverse Fourier transformed output of Eq. (32). (lower panel)

practice this is more preferable as it implies that more spectral space is needed than actually required, better protecting against the occurrence of audible aliasing distortion.

2.3 Analysis using the Bifrequency Function

As shown in the previous section, the time-varying nature of the PLTV system means that new spectral components appear at its output. These new components are related to the input signal's spectral components. The PLTV system shifts these to multiples of the fundamental frequency of the coefficient modulation [20]. The Bifrequency function is a two-dimensional function that describes this mapping between the spectral components of the input signal, which are present at frequencies ψ , and the spectrum of the output at frequencies ω . The general expression for the Bifrequency function is

$$B(\psi, \omega) = \sum_{n=0}^{\infty} \sum_{p=0}^{\infty} h(p, n) e^{j(\psi m - \omega n)} \quad (36)$$

It can be computed by applying a series of FFTs followed by a series of IFFTs to the matrix in Eq. (17). Fig. 9(a) is a plot of the Bifrequency function for the example. A Kaiser window ($\beta = 3$) was applied to the impulse response data before taking the FFTs. From Fig. 9(a), if the input signal has frequency components below 15 kHz there will be the original and new harmonically related components in the output. For higher frequency input, that is, above 15 kHz, no new components will appear in the output.

To interpret the mapping between the input and output as shown in Bifrequency function plot, a line can be drawn from a particular input frequency on the left hand axis. Where this line intersects the peaks along the diagonal indicates the magnitudes of new components that will result in the output. Then, dropping a line from the intersection points to the bottom axis will give the frequencies at which these new components will appear. Fig. 9(b) illustrates this by zooming in on the bottom left corner of Fig. 9(a). A white line is shown across the spectrum, and perpendicular to the

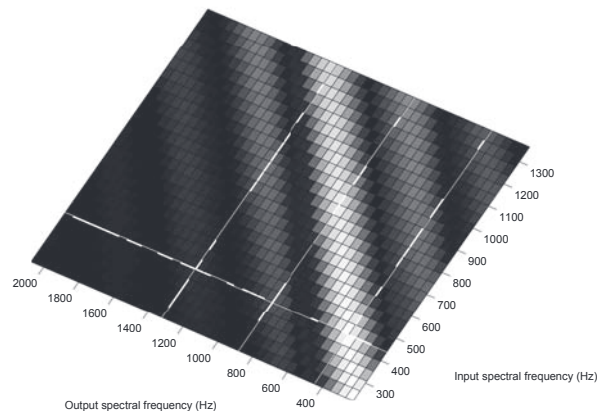
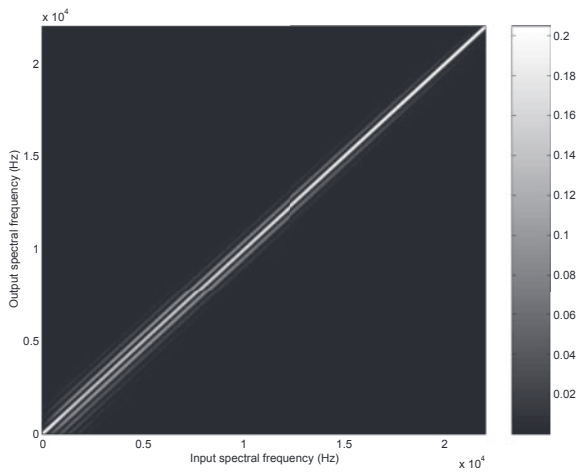


Fig. 9. (a) Bifrequency function for the example, (b) Zoom of corner of (a) to show correspondence between input component and output components.

input spectral frequency axis, which is at the frequency of the input, that is, 441 Hz. It intersects the diagonals of the Bifrequency function here at 441 and 882 Hz, corresponding to a fundamental and second harmonic in the output, highlighted using the three parallel white lines. Additionally, its color tells that the magnitude of the fundamental will be greater than that of the second harmonic.

If the Bifrequency function is available along with the spectrum of the input, then the spectrum of the output can be written [20]

$$Y(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\psi)B(\psi, \omega) d\psi \tag{37}$$

3. PLTV ALLPASS FILTER REALIZATIONS

The topology of the time-varying filter was found to have an impact on the filter output, as was demonstrated informally in [11] where it was shown that the filter output waveform depended on the particular implementation of the difference equation. This phenomenon will now be investigated thoroughly here by drawing on the implementation choices for a standard allpass filter, as given in [38], and then replacing the fixed coefficients with their time-varying

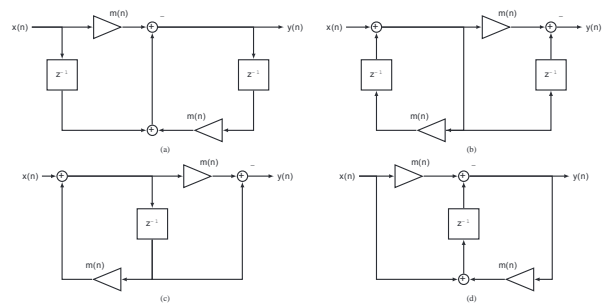


Fig. 10. Direct form implementations of PLTV allpass filter: (a) direct form I, (b) transposed direct form I, (c) direct form II, and (d) transposed direct form II.

versions. It is worth reiterating that the behavior of the time-varying version may not be strictly allpass. The next section will introduce a number of different topologies, illustrating them in block diagram form, showing their difference equations and presenting expressions for the impulse response of each.

3.1 Direct Form Structures

A direct form structure is when a digital filter is realized so that the multipliers of both the feedforward and the feedback parts of the realization are precisely the coefficients of the filter [38]. The direct form realizations of the time-varying first-order allpass filter with modulation signal $m(n)$ are depicted in Fig. 10.

The direct form structures differ from each other by the order in which the feedforward and the feedback parts of the filter are accessed and by the order in which the delay elements and the multipliers are within the filter parts. The state-space representation of the direct form I realization of Fig. 10(a) is given by

$$\begin{aligned} w_{ff}(n+1) &= x(n), w_{fb}(n+1) \\ &= -m(n)x(n) + w_{ff}(n) + m(n)w_{fb}(n), y(n) \\ &= -m(n)x(n) + w_{ff}(n) + m(n)w_{fb}(n) \\ &= -m(n)x(n) + x(n-1) + m(n)y(n-1) \end{aligned} \tag{38}$$

where $w_{ff}(n+1)$ and $w_{fb}(n+1)$ are the signals entering the delays of the feedforward and feedback parts of the realization, respectively. This filter has an impulse response given by

$$\begin{aligned} h(p, n) &= -m(n)\delta(n-p) + \sum_{k=1}^n \prod_{i=0}^{k-2} m(n-i) \\ &\times [1 - m(n-k+1)m(n-k)]\delta(n-k-p) \\ &- \prod_{k=0}^{n+1} m(n-k)\delta(p+1) + \prod_{k=0}^n m(n-k)\delta(p+2) \end{aligned} \tag{39}$$

For the transposed direct form I realization (Fig. 10(b)), the state-space representation is expressed as:

$$\begin{aligned} w_{ff}(n+1) &= x(n) + w_{fb}(n), w_{fb}(n+1) \\ &= m(n)x(n) + m(n)w_{fb}(n), y(n) \\ &= -m(n)x(n) - m(n)w_{fb}(n) + w_{ff}(n) \end{aligned} \quad (40)$$

The impulse response of this time-varying structure is

$$\begin{aligned} h(p, n) &= -m(n)\delta(n-p) + [1 - m(n)m(n-1)] \\ &\times \sum_{k=1}^n \prod_{i=2}^k m(n-i)\delta(n-k-p) \\ &+ [1 - m(n)m(n-1)] \prod_{k=2}^{n+1} m(n-k)\delta(p+1) \end{aligned} \quad (41)$$

The state-space representation of the direct form II realization of Fig. 10(c) is

$$\begin{aligned} w(n+1) &= x(n) + m(n)w(n), y(n) \\ &= -m(n)(x(n) + m(n)w(n)) + w(n) \\ &= -m(n)x(n) + (1 - m^2(n))w(n) \end{aligned} \quad (42)$$

where $w(n+1)$ is the signal entering the delay, and its impulse response is

$$\begin{aligned} h(p, n) &= -m(n)\delta(n-p) + [1 - m^2(n)] \\ &\times \sum_{k=1}^n \prod_{i=1}^{k-1} m(n-i)\delta(n-k-p) \\ &+ [1 - m^2(n)] \prod_{k=1}^{n-1} m(n-k)\delta(p+1) \end{aligned} \quad (43)$$

For the transposed direct form II realization (Fig. 10(d)), the state-space representation is expressed as:

$$\begin{aligned} w(n+1) &= (1 - m^2(n))x(n) + m(n)w(n), y(n) \\ &= -m(n)x(n) + w(n) \\ &= -m(n)x(n) + x(n-1) + m(n-1)y(n-1) \end{aligned} \quad (44)$$

and the impulse response of this filter is

$$\begin{aligned} h(p, n) &= -m(n)\delta(n-p) \\ &+ \sum_{k=1}^n \prod_{i=1}^{k-1} m(n-i) [1 - m^2(n-k)] \delta(n-k-p) \\ &+ [1 - m^2(-1)] \prod_{k=1}^n m(n-k)\delta(p+1) \end{aligned} \quad (45)$$

3.2 Parallel Form Structures

Alternatively, the PLTV allpass filter can be realized in a parallel form by writing the transfer function as a partial-fraction expansion. In a parallel form realization, the transfer function is expressed as a set of first- or second-order filter sections in parallel. However, it can be shown that the multipliers of a parallel form realization are more complex than for the direct form realizations of Fig. 10. This increased complexity for the multiplier structure discourages

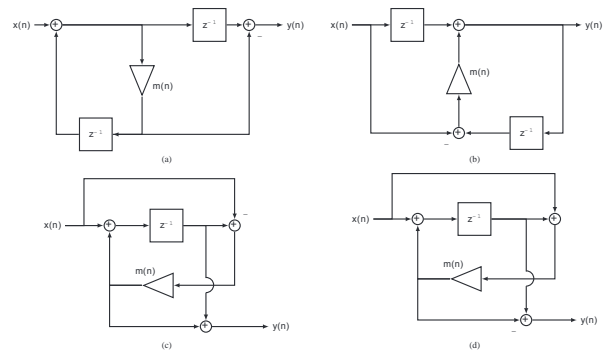


Fig. 11. Allpass type I implementations of PLTV allpass filter.

the use of parallel form realizations for time-varying allpass filtering. Therefore, they are not further investigated here.

3.3 Allpass Form Structures

Since the standard LTI allpass filter has identical coefficients in the numerator and denominator, another realization that takes an advantage of this property can be derived. For the first-order allpass filter, such a realization contains only a single multiplier. These single-multiplier realizations are called Allpass type I forms [38]. The Allpass type I realizations of the time-varying first-order allpass filter are given in Fig. 11.

For the allpass type IA realization (Fig. 11(a)), the state-space representation is expressed as:

$$\begin{aligned} w_{ff}(n+1) &= x(n) + w_{fb}(n), w_{fb}(n+1) \\ &= m(n)x(n) + m(n)w_{fb}(n), y(n) \\ &= w_{ff}(n) - m(n)x(n) - m(n)w_{fb}(n) \end{aligned} \quad (46)$$

where $w_{ff}(n+1)$ and $w_{fb}(n+1)$ are the signals entering the feedforward and feedback parts of the realization, respectively. This realization is the same as the transposed direct form I realization of Fig. 10(b) and thus its impulse response is given by Eq. (40).

The state-space representation of the transposed allpass type IA realization of Fig. 11(b) is given by

$$\begin{aligned} w_{ff}(n+1) &= x(n), w_{fb}(n+1) \\ &= -m(n)x(n) + w_{ff}(n) + m(n)w_{fb}(n), y(n) \\ &= -m(n)x(n) + w_{ff}(n) + m(n)w_{fb}(n) \\ &= -m(n)x(n) + x(n-1) + m(n)y(n-1) \end{aligned} \quad (47)$$

This realization is the same as the direct form I realization of Fig. 10(a), and its impulse response is given by Eq. (38).

For the allpass type IB realization (Fig. 11(c)), the state-space representation is

$$\begin{aligned} w(n+1) &= (1 - m(n))x(n) + m(n)w(n), y(n) \\ &= -m(n)x(n) + (1 + m(n))w(n) \end{aligned} \quad (48)$$

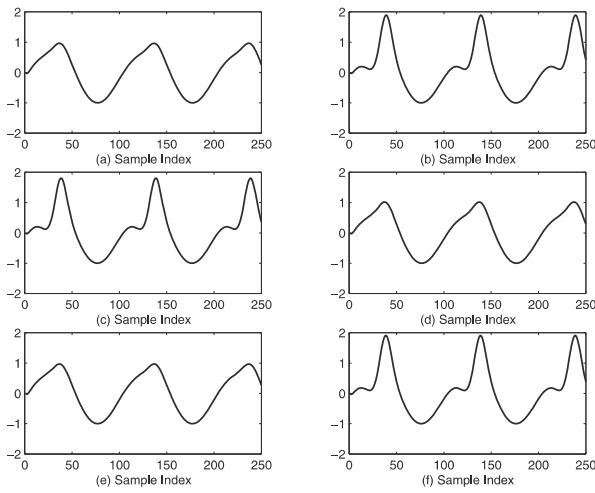


Fig. 12. Output of different allpass PLTV filter topologies: (a) direct form I and allpass type IA transposed, (b) direct form I transposed and allpass type IA, (c) direct form II, (d) direct form II transposed, (e) allpass type IB, and (f) allpass type IB transposed.

where $w(n+1)$ is the signal entering the delay. The impulse response of this structure is

$$\begin{aligned}
 h(p, n) = & -m(n)\delta(n - p) \\
 & + [1 + m(n)] \sum_{k=1}^n \prod_{i=1}^{k-1} m(n - i) \\
 & \times [1 - m(n - k)] \delta(n - k - p) \\
 & + [1 + m(n)] [1 - m(-1)] \\
 & \times \prod_{k=1}^n m(n - k) \delta(p + 1)
 \end{aligned} \quad (49)$$

The state-space representation of the transposed allpass type IB realization of Fig. 11(d) is given by

$$\begin{aligned}
 w(n + 1) = & (1 + m(n))x(n) + m(n)w(n), y(n) \\
 = & (1 - m(n))w(n) - m(n)x(n)
 \end{aligned} \quad (50)$$

and its impulse response is

$$\begin{aligned}
 h(p, n) = & -m(n)\delta(n - p) \\
 & + [1 - m(n)] \sum_{k=1}^n \prod_{i=1}^{k-1} m(n - i) \\
 & \times [1 + m(n - k)] \delta(n - k - p) \\
 & + [1 - m(n)] [1 + m(-1)] \\
 & \times \prod_{k=1}^n m(n - k) \delta(p + 1)
 \end{aligned} \quad (51)$$

Fig. 12 gives plots of a few periods of the output of the PLTV allpass system when implemented using, in essence, the six different topologies given. It can be seen that the Direct Form I (and Allpass type IA transposed), the Direct form II transposed and the Allpass type IB have similar shaped outputs that could be described as smooth. In contrast, the three remaining topologies produce a large transient in the output for each period. It is most pronounced for the Allpass type IB transposed. This suggests that when

selecting an implementation topology, it must be done carefully, particularly if there are other system elements that are sensitive to high crest factor [39] input that follow the time-varying filter in a specific application.

4 PLTV ALLPASS FILTER APPLICATIONS

In this section, we will examine two applications of the PLTV allpass filter. The first example shows how the PLTV allpass filter can be used to reproduce the sawtooth waveform associated with PD synthesis [8]. Second, the output of a waveform distorter will be analysed to find a suitable phase distortion function to describe the waveform deformation imparted to a cosine input. A version of this function will then be used to modulate the PLTV allpass filter to show that it can reproduce this distortion effect.

4.1 PLTV Allpass filter generation of PD Sawtooth

Phase Distortion [8] is a synthesis technique that allows the generation of complex spectra by modifying the phase of a sinusoidal signal. A nonlinear phase distorting function $PD(\cdot)$ is applied to the temporal phase of a cosine signal with the result that the transformed output signal $s(n)$ has a complex spectrum and therefore a more interesting timbre

$$s(n) = \cos(\omega_0 n + PD(\omega_0 n) - \pi) \quad (52)$$

where ω_0 denotes the fundamental frequency. A phase shift term of $-\pi$ is included to invert the waveform in keeping with the original patent [8].

This patent [8] demonstrated the technique by providing an example of how it could be used to create a sawtooth waveform, as is typically found in subtractive synthesizers. This phase distortion function actually took the form of a nonband-limited sawtooth ramp wave

$$PD_{\text{saw}}(\omega_0 n, d) = \frac{\pi}{4} (1 + \text{saw}(\omega_0 n, d)) \quad (53)$$

where the extra parameter d is the inflection point of the phase distortion function that determines the fraction of the complete period during which the sawtooth is rising [8].

Knowing the phase distortion function, it is necessary to convert this into a modulation function for the PLTV allpass filter. To do this we need to manipulate Eq. (26) to give [13]

$$m(n) = \frac{-(\phi(\omega, n) + \omega)}{2 \sin(\omega) - (\phi(\omega, n) + \omega) \cos(\omega)} \quad (54)$$

where in this particular case the phase is

$$\varphi_{\text{saw}}(\omega_0, n) = PD_{\text{saw}}(\omega_0 n, d) - \pi \quad (55)$$

The range of the function in Eq. (55) is between $-\pi/2$ and $-\pi$. When substituting this into Eq. (54), the maximum and minimum of the modulation function $m(n)$ were 0.9626 and 0.9248, respectively. It was found that the reproduction of the PD sawtooth by the allpass PLTV filter was much better if the phase in Eq. (55) was further shifted so that it resulted in a wider numerical range of values for the

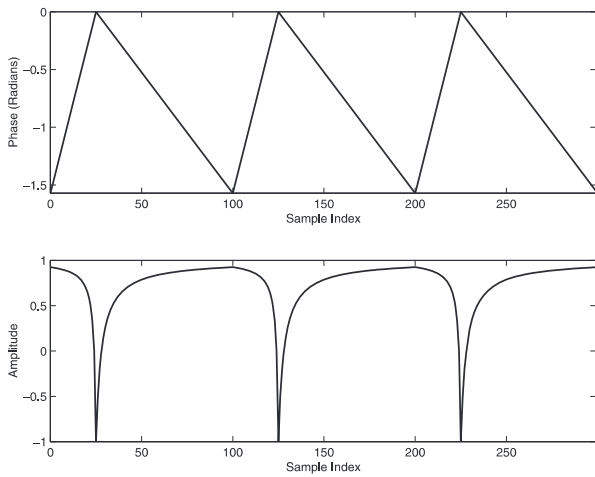


Fig. 13. The phase distortion function of (61) shifted by $\pi/2$ (upper panel) and the generated modulation function (lower panel).

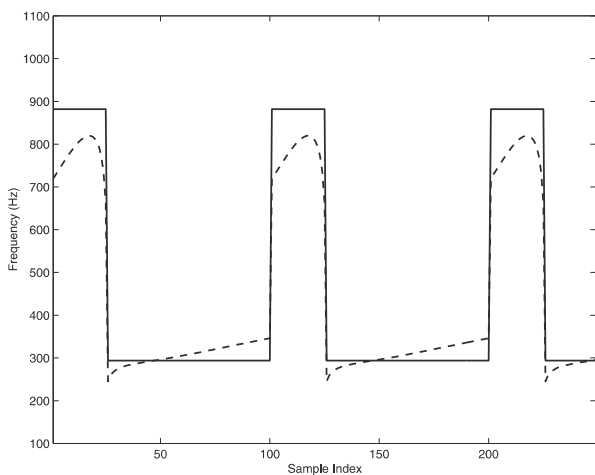


Fig. 14. Instantaneous frequency deviation of phase distortion function (solid line) compared with that given by using Eq. (27).

modulation function [13], thus a factor of $\pi/2$ was simply added to achieve this, that is,

$$\varphi_{\text{saw}}(\omega_0, n) = PD_{\text{saw}}(\omega_0 n, d) - \pi + \pi/2 \quad (56)$$

This effect of this added phase shift, done to improve the numerical range of the modulation function, on the output can be compensated for by shifting the phase of the input signal by the opposite value of $-\pi/2$.

Using a fundamental frequency of 441 Hz, a sampling frequency of 44100 Hz and $d = 0.25$, the upper panel of Fig. 13 shows the phase distortion function that was substituted into Eq. (56) and the lower panel shows the modulation function it produces. The nonlinear relationship between the two is apparent from the waveshape of the modulation function.

The respective frequency deviation, as given by Eq. (27), can be computed for this example, and compared with the frequency deviation of the phase distortion function found by differentiating Eq. (56). This is shown in Fig. 14. Both are centred at the fundamental frequency of 441 Hz. For

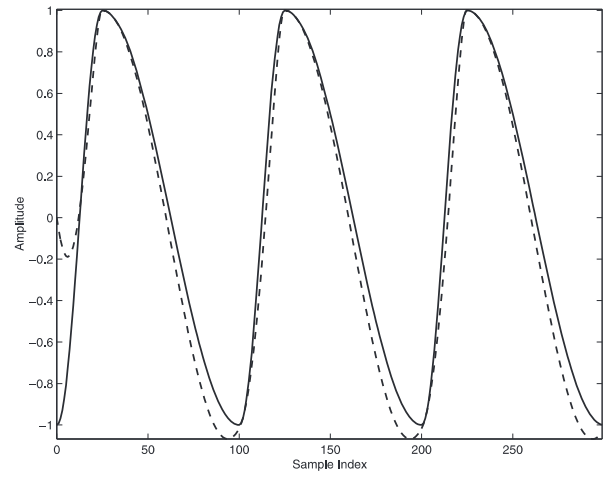


Fig. 15. Phase distortion sawtooth (solid line) compared with output of PLTV allpass filter (dashed line).

the phase distortion function (solid line), the instantaneous frequency in the first quarter period is 882 Hz, which is twice that of the input. The remainder is two thirds that of the input, that is, 294 Hz. In contrast, the dynamic behavior of Eq. (27) (dashed line), follows the solid line in a curving manner in keeping with the earlier assertion in Section 2 that it must be time-varying to introduce a frequency deviation to the signal. The maximum of this instantaneous frequency deviation does fall short of the maximum derived from the phase distortion function. This is attributable to the fact that Eq. (27) is an approximation.

Fig. 15 shows the sawtooth output of the phase distortion algorithm found by substituting $\varphi_{\text{saw}}(\omega_0, n)$ of Eq. (56) for $PD_{\text{saw}}(\omega_0 n, d) - \pi$ in Eq. (53) as a solid line. The dashed line shows the output of the PLTV allpass filter. A Direct form II transposed topology was used. The input to the PLTV filter was a cosine with a phase delay of $-\pi/2$ to compensate for the phase shift of the phase distortion function of Eq. (56). There is a good match between the two sawtooths with only a slight difference appearing at the minimum points of both where the phase distortion sawtooth is slightly faster. Notice that there is a small startup transient for the allpass filter at the very beginning of its output that quickly disappears.

Fig. 16 shows the spectra of the two waveforms given in Fig. 15. The upper panel is that of the phase distortion sawtooth. It exhibits a decaying spectrum like a textbook sawtooth but with very noticeable spectral zeros, whose presence was discussed in the original patent [8]. The lower panel shows the spectrum of the output of the PLTV allpass filter. No spectral zeros are present here, and thus the timbre of the sound in this case is richer [13].

4.2 PLTV Allpass filter emulation of a Waveform Distorter

Nonlinear distortion functions are commonly used to emulate analog electronics guitar distortion effects. Prior to advanced circuit models being used, for example [40, 41], simpler saturating nonlinear functions were applied

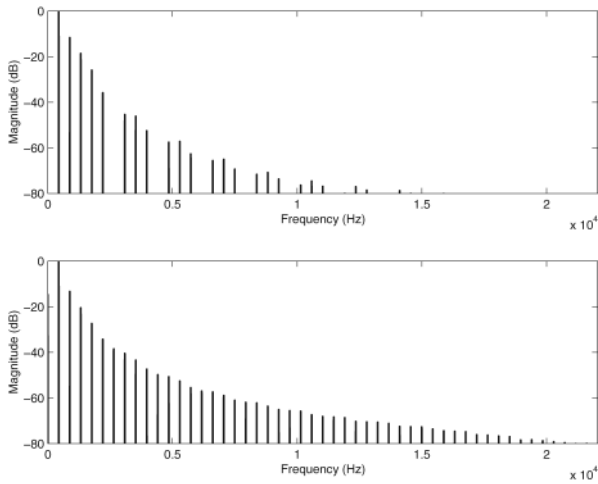


Fig. 16. Spectrum of phase distortion sawtooth (upper panel) and output of PLTV allpass filter (lower panel).

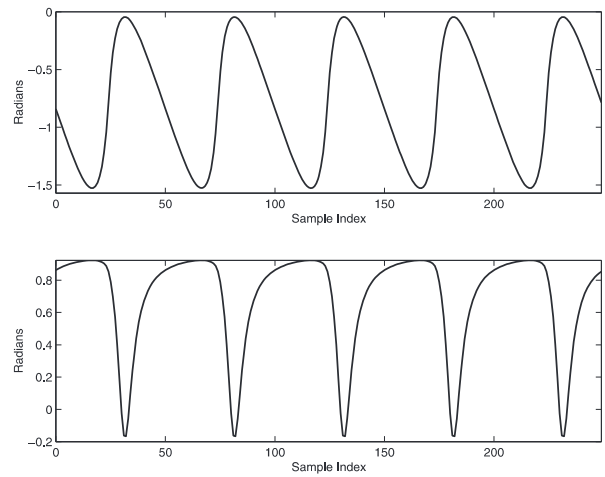


Fig. 18. Phase Modulation derived from distorter output (upper panel) and the generated modulation function (lower panel).

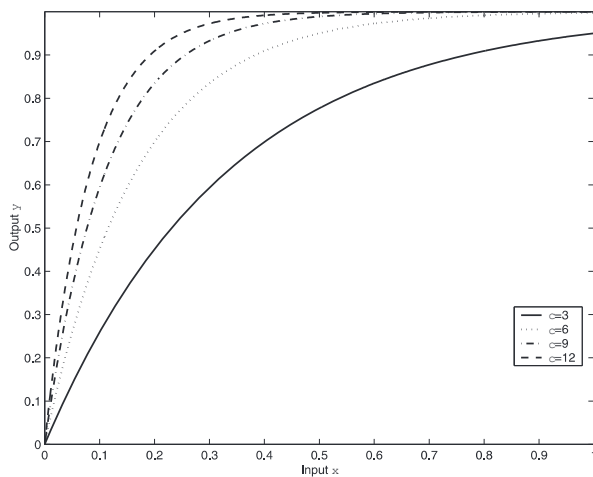


Fig. 17. Shape of the exponential saturation curve for four different values of parameter c .

[4]. One particular exponentially-based saturating function presented in [42] produces a distorted waveform $d(n)$ according to

$$d(x(n)) = A \operatorname{sgn}(x(n)) (1 - e^{-c|x(n)|}) \quad (57)$$

where $x(n)$ is the input, and $\operatorname{sgn}(\cdot)$ is the signum function

$$\operatorname{sgn}(x(n)) = \begin{cases} 1, & x(n) \geq 0 \\ -1, & x(n) < 0 \end{cases} \quad (58)$$

and A is the maximum value of $x(n)$ and Ac defines the slope of the curve at the origin.

The nonlinear input to output transfer function for Eq. (57) is shown in Fig. 17 for various positive values of c only as it is symmetric around the origin. The shape of the curves in Fig. 17 range from a very gentle slope to one that is more abrupt, suggesting it can produce a range of distortions, from a soft to hard clipping.

To find a phase distortion function that will emulate the effect of Eq. (58), it is necessary to compute the deviation

of the phase of the output waveform from that of the cosine input. It was found that a slightly more robust approach to the one proposed in [10] was to use the inverse cosine function to find the time-varying phase $\varphi_d(n)$, of the distorted waveform,

$$\varphi_d(n) = \cos^{-1}(d(n)) \quad (59)$$

The phase distortion function can then be found by

$$PD_d(n) = \varphi_d(n) - \cos^{-1}(x(n)) \quad (60)$$

There will be an ambiguity in the sign of the phase distortion function in Eq. (60). This can be resolved by tracking the way the waveform changes sign over time as it traverses from positive to negative or vice versa. Taking this into account produces an adjusted phase distortion function $PD'_d(n)$.

For this example the parameter was chosen to be $c = 6$. This adjusted phase distortion function was found for a cosine input of frequency 441 Hz and sampling frequency 44100 Hz. The adjusted phase distortion was found to be symmetric around zero so, as in the previous example, it was shifted to render it completely negative by subtracting the value of $\pi/4$ radians along with the normalized input frequency in radians. Fig. 18 plots the adjusted and now shifted phase distortion function which is a smooth sawtooth-like function. Eq. (54) was used then to find the coefficient modulation for the PLTV allpass filter which is given in the lower panel of Fig. 18. This has smoother transitions between the maximum and minimum of the function than in the previous example.

The output of the distorter is shown with the solid line in Fig. 19. It has the effect of converting the input cosine into a smooth squarewave. The input cosine signal to the PLTV allpass filter was first advanced by the same phase shift that was applied to phase distortion function. The coefficient modulation applied was that shown in the lower panel of Fig. 18. This produced the output given by the dashed line in Fig. 19. There is a very good match between the waveforms in Fig. 19 with just a slight difference in waveshape at the

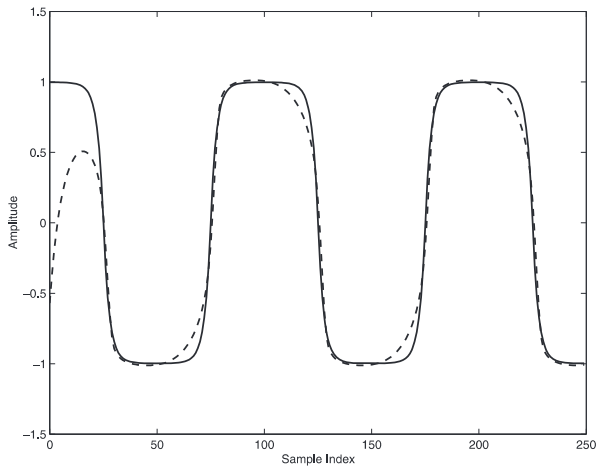


Fig. 19. Comparison of output of the distorter (solid line) and emulated version using PLTV allpass filter (dashed line).

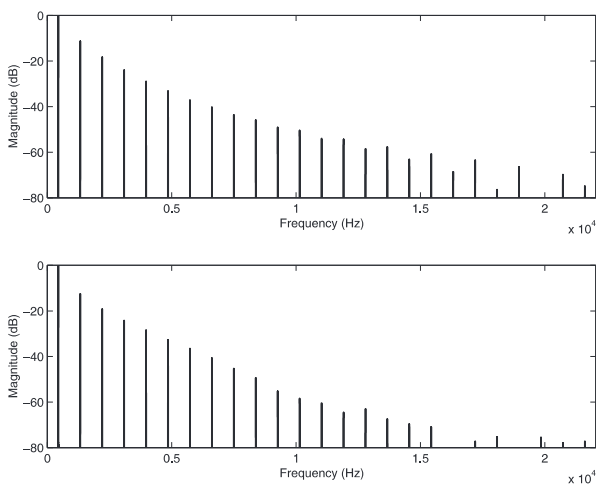


Fig. 20. Spectrum of distorter output (upper panel) and output of PLTV allpass filter (lower panel)

turning points of the wave. Again, there is a small transient present at the start of the filtering. Thus, the allpass PLTV filter is successful at synthesizing a clipping type distortion effect to its input.

Lastly, Fig. 20 shows the spectra of two waveforms in Fig. 19. The upper panel shows that of the distorter output while the lower panel is that of the PLTV allpass filter. In this case the bandwidth of the distorter output is greater than that of the PLTV allpass filter. The reason for this is because the PLTV allpass filter output is smoother at the turning points of the distorter wave, and thus has less high frequency content in comparison. However, there are no significant aliasing components in this example as a result.

5 CONCLUSIONS

This paper has described how the coefficient-modulated allpass filter can be analyzed in the time and frequency domains. It explained how the time-varying transfer function can be decomposed into nonrecursive and recursive elements, and illustrated the evolution of the 2D impulse response

matrix. It also introduced the stability criterion for these filters. A frequency domain analysis was also given which included the 2D time-varying magnitude and phase. It was shown that the magnitude response is not necessarily perfectly allpass across time and frequency. Approaches for computing the spectrum of a Complex FM signal were also examined and Eq. (32) in particular was determined to be useful for assessing the bandwidth of the filter output signal, from which the likelihood of aliasing distortion occurring could be inferred. This was followed by an explanation on how to interpret the Bifrequency function. A complete set of filter topologies were used to investigate their impact on the output waveshape and it was found that it is an important consideration and that only certain configurations produce outputs with a low crest factor. Lastly, two examples were given to illustrate its operation as a dynamic phase distortion device. Both examples showed that the output of the PLTV allpass filter matched well with that of the original distortion algorithm.

Overall, this article has explained how time-varying first-order allpass filters can be manipulated to create signal phase distortions which result in timbral modification. This is a new advantage regarding allpass filters when applied to the field of digital audio effects. These filters can process arbitrary input and subject them to prescribed modulation functions that could also be enhanced by the typical synthesis modifiers of envelopes and low frequency oscillators to form a complex signal effect chain. It is hoped that the set of tools introduced can be applied by sound designers wishing to take advantage of these versatile audio filters for any new sonic experiments and applications that they can imagine.¹

6 ACKNOWLEDGMENT

Joseph Timoney would like to acknowledge the College of Computer Science, Zhejiang University, Hangzhou, China as part of this work was prepared while on a study visit there.

7 REFERENCES

- [1] M. R. Schroeder, 'Natural-sounding artificial reverberation,' *J. Audio Eng. Soc.*, Vol. 10, no. 3, July 1962, pp. 219–223.
- [2] J. Dattorro, 'Effect Design, Part 1: Reverberator and Other Filters,' *J. Audio Eng. Soc.*, Vol. 45, no. 9, Sept. 1997, pp. 660–684.
- [3] V. Välimäki, J. Parker, and J. S. Abel, 'Parametric Spring Reverberation Effect,' *J. Audio Eng. Soc.*, vol. 58, no. 7/8, July 2010, pp. 547–562.
- [4] U. Zölzer (ed), *DAFX: digital audio effects*, John Wiley and sons, Chichester, UK, 2002.
- [5] J. O. Smith, 'An allpass approach to digital phasing and flanging', *Proc. of International Computer Music*

¹ A set of Matlab scripts to generate all the figures in the paper and sound examples can be downloaded <https://www.cs.nuim.ie/pltvpaperexamples>

- Conf. (ICMC)*, Paris, 1984, pp. 103–109. [available online: <http://ccrma.stanford.edu/STANM/STANM/stanm14/>.]
- [6] V. Välimäki, J. S. Abel, and J. O. Smith, ‘Spectral Delay Filters,’ *J. Audio Eng. Soc.*, vol. 57, no. 7/8, July 2009, pp. 521–531.
- [7] J. Pekonen, ‘Coefficient-modulated first-order all-pass filter as a distortion effect,’ *Proc. 11th Conference on Digital Audio Effects (DAFx’08)*, Espoo, Finland, Sept. 2008, pp. 83–87.
- [8] M. Ishibashi, Electronic Musical Instrument, U.S. Patent no. 4,658,691, 1987.
- [9] J. Chowning, ‘The synthesis of complex audio spectra by means of frequency modulation,’ *J. Audio Eng. Soc.*, Vol. 21, no. 7, Sept. 1973, pp. 526–534.
- [10] J. Timoney, V. Lazzarini, A. Gibney and J. Pekonen, ‘Digital Emulation of Distortion by Wave and Phase Shaping Methods’, *Proc. 13th Conference on Digital Audio Effects (DAFX ‘10)*, Graz, Austria, Sept. 2010.
- [11] V. Lazzarini, J. Timoney, J. Pekonen and V. Välimäki, ‘Adaptive Phase Distortion Synthesis,’ *Proc. 12th Conference on Digital Audio Effects (DAFx’09)*, Como, Italy, Sept. 2009.
- [12] V. Lazzarini, T. Lysaght and J. Timoney, ‘The generation of natural-synthetic spectra by means of Adaptive Frequency modulation’, *Computer Music Journal*, Vol. 32, no. 2, MIT Press, Cambridge, Mass., Summer 2008, pp. 9–22.
- [13] J. Timoney, V. Lazzarini, J. Pekonen and V. Välimäki, ‘Spectrally rich phase distortion sound synthesis using an allpass filter,’ *Proc. 2009 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP 2009)*, Taipei, Taiwan, Apr. 2009, pp. 293–296.
- [14] J. Timoney, V. Lazzarini, B. Carty and J. Pekonen, ‘Phase and amplitude distortion methods for digital synthesis of classic analogue waveforms’, *AES Convention 126*, Munich, Germany, May 7-10, 2009.
- [15] J. Kleimola, J. Pekonen, H. Penttinen, V. Välimäki and J. Abel, ‘Sound synthesis using an allpass filter chain with audio-rate coefficient modulation,’ *Proc. 12th Conference on Digital Audio Effects (DAFx’09)*, Como, Italy, Sept. 2009.
- [16] G. Surges and T. Smyth, ‘Spectral Distortion Using Second-Order Allpass Filters’, *Proc. of Sound and Music Computing (SMC2013)*, Stockholm, Sweden, July-August 2013.
- [17] M. Cherniakov, V.I. Sizov, and L. Donskoi, ‘Synthesis of a periodically time-varying digital filter,’ *IEE Proceedings - Vision, Image and Signal Processing*, Vol. 147, no. 5, Oct. 2000, pp. 393–399.
- [18] R. Rabenstein, ‘Minimization of Transient Signals in Recursive Time-Varying Digital Filters,’ *Circuits, Systems and Signal Processing*, vol. 7, no. 3, 1998, pp. 345–359.
- [19] V. Välimäki and T. I. Laakso, ‘Suppression of Transients in Variable Recursive Digital Filters with a Novel and Efficient Cancellation Method,’ *IEEE Trans. Signal Processing*, vol. 46, no. 12, Dec. 1998, pp. 3408–3414.
- [20] M. Cherniakov, *An introduction to parametric digital filters and oscillators*, J. Wiley and Sons, London, 2003.
- [21] J. Laroche, ‘On the stability of time-varying filters,’ *J. Audio Eng. Soc.*, Vol. 55, no. 6, June 2007, pp. 460–471.
- [22] M. Kawamata, X. Yang and T. Higuchi, ‘Fundamental Study on Periodically Time-Varying State-Space Digital Filters -Statistical Analysis, Scaling and Stability,’ *Intl. Conf. on Systems Engineering*, New York, USA, Sept. 1992, pp. 348–351.
- [23] G. Tohumoglu, ‘Analysis of periodically time-varying discrete-time systems in spectral domain’, *Applied Mathematics and Computation*, Vol. 162, no. 3, March 2005, pp. 1151–1165.
- [24] E. Louarroudi et al, Estimation of nonparametric Harmonic Transfer Functions for linear periodically time-varying systems using periodic excitations’, *IEEE International Instrumentation and Measurement Technology Conference - I2MTC*, Hangzhou, China, May 10-12, 2011, pp. 699–704.
- [25] T. Leou and J. Aggarwal, ‘Recursive Implementation of LTV Filters-Frozen-Time transfer Function Versus Generalized Transfer Function’, *Proc. IEEE*, Vol. 72, no. 7, July 1984, pp. 980–981.
- [26] V. Lazzarini, J. Kleimola, J. Timoney and V. Välimäki, ‘Five Variations on a Feedback Theme,’ *Proc. 12th Conference on Digital Audio Effects (DAFx’09)*, Como, Italy, Sept. 2008.
- [27] J. Kleimola, V. Lazzarini, V. Välimäki and J. Timoney, ‘Feedback Amplitude Modulation Synthesis,’ *EURASIP Journal on Advances in Signal Processing*, Vol. 2011, 2011.
- [28] P. Regalia, S. K. Mitra, and P. Vaidyanathan, ‘The digital all-pass filter: a versatile signal processing building block,’ *Proc. IEEE*, Vol. 76, no. 1, Jan. 1988, pp. 19–37.
- [29] T. Laakso, V. Välimäki, M. Karjalainen, and U. Laine, ‘Splitting the unit delay—Tools for fractional delay filter design,’ *IEEE Signal Processing Magazine*, Vol. 13, no. 1, Jan. 1996, pp. 30–60.
- [30] M. LeBrun, ‘A Derivation of the Spectrum of FM with a Complex Modulating Wave,’ *Computer Music Journal*, Vol. 1, no. 4, Winter 1977, pp. 51–52.
- [31] D. Benson, *Music: A Mathematical Offering*, Cambridge Univ. Press, Cambridge, UK, 2006.
- [32] B. Schottstaedt, ‘The Simulation of Natural Instrument Tones Using Frequency Modulation with a Complex Modulating Wave,’ *Computer Music Journal*, Vol. 1, no. 4, Winter 1977, pp. 46–50.
- [33] T. Mitchell and J. Sullivan, ‘Frequency modulation tone matching using a fuzzy clustering evolution strategy,’ *AES Convention 118*, Barcelona, Spain, May 2005.
- [34] G. Putland and B. Boashash, ‘Can a signal be both monocomponent and multicomponent?’, *Third Australasian Workshop on Signal Processing Applications (WoSPA 2000)*, Brisbane, December 14-15, 2000.
- [35] J. Timoney, V. Lazzarini, T. Lysaght, J. Kleimola and L. MacManus, ‘Considerations for a Computational Estimation of the Complex FM Spectrum’, *Irish Sound Science and Technology Conference (ISSTC) 2011*, Limerick, Ireland, August 2011.

[36] J. Timoney and V. Lazzarini, 'A homomorphic interpretation of the Complex FM expansion', *AES 45th Conference on Applications in Time-Frequency Processing*, Helsinki, Finland, March 2012.

[37] R. Garcia and K. Short, 'Signal analysis using the Complex Spectral Phase evolution (CSPE) method,' *AES Convention 120*, Paris, France, May 2006.

[38] S.K. Mitra, *Digital signal processing*, Mc-Graw hill, USA, 2005.

[39] P. Guillaume, J. Schoukens, R. Pintelton and I. Kollár, 'Crest-factor minimization using nonlinear Chebyshev approximation methods,' *IEEE Trans. on instrumen-*

tation and measurement, Vol. 40, no. 6, Dec 1991, pp. 982–989.

[40] D. Yeh, Digital implementation of musical distortion circuits by analysis and simulation, *Ph.D. thesis*, Stanford University, Stanford, CA, USA, June 2009.

[41] J. Pakarinen and D. Yeh, 'A Review on Digital Techniques for Modelling Vacuum-Tube Guitar Amplifiers,' *Computer Music Journal*, vol. 33, no. 2, Summer 2009, pp. 85–100.

[42] R. J. Kavanagh, 'An approximation to the harmonic response of saturating devices', *Proc. IEE - Part C*, Vol. 107, no. 11, March 1960, pp. 127–133.

THE AUTHORS



Joseph Timoney



Jussi Pekonen



Victor Lazzarini



Vesa Välimäki

Dr. Joe Timoney completed his Ph.D. in Electronic Engineering at TCD, Ireland in 1998. He joined the Dept. of Computer Science at Maynooth University in the following year. He teaches on undergraduate programs in Computer Science and in Music Technology. His research interests are based in the area of audio signal processing, with a focus on musical sound synthesis and the digital modelling of analogue subtractive synthesizers. He is also a DIY electronics enthusiast and has participated in a number of national Maker Fairs as part of the Maynooth University club. He is a member of the Audio Engineering Society.

Jussi Pekonen has worked in the field of audio signal processing for a decade. In 2004, when he joined the Laboratory of Acoustics and Audio Signal Processing at TKK Helsinki University of Technology, Espoo, Finland, as a research assistant. He completed his M.Sc. (Tech.) degree in 2007 and continued to pursue his D.Sc. (Tech.) degree at TKK (now called Aalto University). In the beginning 2012, he left the university and became a developer at LM Ericsson, Kirkkonummi, Finland, by day and an independent researcher by night. He (finally!) managed to complete his doctoral degree in April 2014 on the topic of filter-based oscillator algorithms for virtual analog synthesis. His research interests also include other sound synthesis methods and digital audio effects.

Victor Lazzarini is a Senior Lecturer at Maynooth University in Ireland. He is a graduate of the Universidade Estadual de Campinas (UNICAMP) in Brazil, where he was awarded a BMus in Composition. He completed his doctorate at the University of Nottingham, UK. His interests include musical signal processing and sound synthesis; computer music languages; electroacoustic and instrumental composition.

He coedited "The Audio Programming Book" (with R. Boulanger, Cambridge, Mass, MIT Press, 2010), and has published extensively in the above areas.

Vesa Välimäki is professor of audio signal processing at the Aalto University, Espoo, Finland. He received his M.Sc. in Technology, Licentiate of Science in Technology, and Doctor of Science in Technology degrees in electrical engineering from the Helsinki University of Technology (TKK) in 1992, 1994, and 1995, respectively. His doctoral dissertation dealt with fractional delay filters and physical modeling of musical wind instruments.

In 1996, he was a postdoctoral research fellow at the University of Westminster, London, UK. In 2001–2002, he was professor of signal processing at the Pori School of Technology and Economics, Tampere University of Technology, Pori, Finland. In 2006–2007, he was the head of the TKK Laboratory of Acoustics and Audio Signal Processing. During the academic year 2008–2009 he was a visiting scholar at the Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, Stanford, CA. His research interests include sound synthesis, audio effects processing, digital filter design, and acoustics of musical instruments.

Prof. Välimäki is a senior member of the IEEE Signal Processing Society, a life member of the Acoustical Society of Finland, and a member of the Finnish Musicological Society. He was the papers chair of the AES 22nd International Conference on Virtual, Synthetic, and Entertainment Audio in 2002. He was the chairman of the 11th International Conference on Digital Audio Effects (DAFx-08), which was held in Espoo, Finland, in 2008.