Vesa Välimäki and Antti Huovilainen

Antialiasing Oscillators in Subtractive Synthesis

Achieving classic signal waveforms in the digital domain without incurring the audible aliasing

Subtractive synthesis generates sound by filtering an input signal that has a rich spectral content. The input signal is provided by an oscillator that can produce a variety of periodic waveshapes, the most usual choices being the sawtooth, the rectangular-pulse, and the triangular waveform. The subtractive synthesis principle has been employed since the earliest music synthesizers, for example in the RCA synthesizer of the 1950s [1]. The method became very popular after Moog [2] improved the controllability of time-varying analog resonant filters to generate pleasing timbres [3].

In developing a digital implementation of subtractive synthesis, the main challenge is to find computationally efficient algorithms for generating periodic waveforms that do not suffer from excessive aliasing distortion. This was an obstacle for the use of subtractive synthesis in digital synthesizers in the late 1970s and early 1980s [4]–[7]. An example of an early digital music synthesizer featuring the subtractive technique was the system that Pete Samson designed for Stanford University [8], [9], which implemented bandlimited oscillators as a closed-form sum of a geometric series [10], [11]. Nevertheless, other sound generation methods, such as sampling, wavetable, additive, and FM synthesis, became common in the beginning of the digital era as well. Interestingly, the digital FM synthesis technique successfully employs the aliasing around 0 Hz in the generation of tones with an inharmonic spectrum [12].

The aliasing problem is illustrated in Figure 1 for a fairly high fundamental frequency, when the sampling rate is 44.1 kHz. The sawtooth waveform of Figure 1(a) has been generated using the trivial method, which samples the raising ramp signal: The counter increment per sampling interval is $2f/f_s$, where f is the desired fundamental frequency and f_s is the sampling rate, and whenever the counter value is about to exceed one, 2.0 is subtracted from it. The spectrum of this signal is shown in Figure 1(b). Figure 1(c) shows the waveform of the bandlimited sawtooth wave, which is

1053-5888/07/$25.00©2007IEEE

composed of only those harmonics that fit between 0 Hz and the Nyquist limit (in this case eight harmonics). The amplitude of each harmonic is inversely proportional to its harmonic index (1, 1/2, 1/3, etc.). The corresponding spectral peaks are seen in Figure 1(d). Note that the digital signal in Figure 1(c) does not appear to be periodic (i.e., the periods are nonidentical), because the period length is not an integer in this case. Figure 1(e) and (f) shows the difference of the trivial and bandlimited waveforms and their spectral difference, respectively. This is the aliasing error contained in the trivial sawtooth waveform.

The aliasing error can severely corrupt the sound quality. The typical audible disturbances caused by aliasing are inharmonicity, beating, and heterodyning. The aliased signal components generally fall between the desired harmonics, when the fundamental frequency does not divide the sampling frequency, thus rendering the overall signal inharmonic. Inharmonicity affects considerably the timbre. Particularly, the inharmonic components whose frequency is smaller than the fundamental frequency are clearly audible and thus very disturbing. When one of the aliased components appears in the vicinity of a harmonic component, say less than 10 Hz away, and their amplitude difference is not very large, the summing of these components generates the beating effect [13]. In the special case when the fundamental frequency divides the sampling frequency, the aliases add on to the desired harmonics causing amplitude distortion. Then the aliased components are not separable by listening, but they influence the tone's brightness. When the fundamental frequency is varied, which commonly occurs in vibrato or in glissando, the frequency of some aliased components are changing in the opposite direction with respect to the fundamental, a phenomenon known as heterodyning [13]. Then it is easy to separate the aliased components by listening.

However, it is known that when the level of aliased components is sufficiently low or they occur at very high frequencies, they become inaudible. The frequency-dependent sensitivity of hearing and the auditory masking phenomenon take care of this. Some oscillator algorithms utilize these phenomena trying to hide the aliasing from human ears without completely eliminating it.

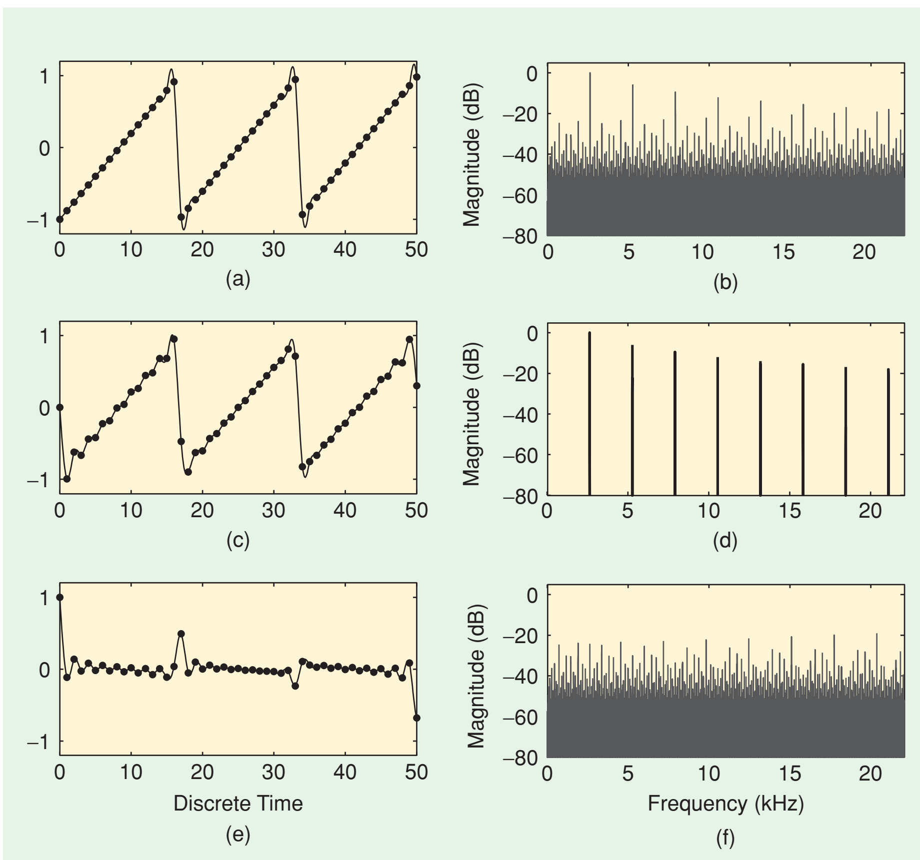
This article reviews the techniques for implementing digital waveform oscillators with reduced aliasing. These methods are useful, for example, in software-based subtractive synthesis, also called virtual analog synthesis, and in ringing tone synthesis for mobile phones and other portable devices. Currently known methods are classified, briefly reviewed, and compared. An auditory error measure, the noise-to-mask ratio (NMR), is used for evaluating the sound quality.

TAXONOMY OF ANTIALIASING OSCILLATOR ALGORITHMS

The algorithms to generate alias-free or nearly alias-free waveforms can be classified as follows [14], [15]:

1) bandlimited algorithms that produce a fixed number of harmonics
2) quasi-bandlimited algorithms that sample a low-pass filtered continuous-time function
3) algorithms based on spectral tilt modification.

The first class of methods includes additive synthesis [16] and related approaches that generate a fixed number of harmonics. This way aliasing can be completely avoided. For example, the sawtooth wave in Figure 1(c) was generated this way:



[FIG1] Waveform and magnitude spectrum of: (a), (b) the trivial, (c), (d) the ideal sawtooth signal; and (e), (f) their difference. The fundamental frequency of the sawtooth waves in this and other figures in this article is 2637.0 Hz (MIDI note #100) and the sampling rate is 44,100 Hz. The spectra presented in this article have been computed from a 1 s signal segment with a 65,536-point FFT using a Chebyshev window function with a 100-dB side-lobe attenuation.

$$s_{\rm id}(n) = -\sum_{k=1,2,3,\ldots}^{K} \frac{1}{k}\sin(2\pi fkn/f_s), \qquad (1)$$

where K is the number of harmonics, n is the sample number, f is the fundamental frequency, and f_s is the sampling frequency. We choose to have a minus sign in front of the equation to obtain a rising ramp.

The quasi-bandlimited methods of the second class control the amount of aliasing using a filtering operation. This class contains methods known as the bandlimited impulse train (BLIT) implemented with the sum of windowed sinc functions (BLIT-SWS) [17], [18] and the minimum bandlimited step sequence (MinBLEP) [19], which employ a bandlimited pulse train and a bandlimited step function, respectively.

IN DEVELOPING A DIGITAL IMPLEMENTATION OF SUBTRACTIVE SYNTHESIS, THE MAIN CHALLENGE IS TO FIND COMPUTATIONALLY EFFICIENT ALGORITHMS FOR GENERATING PERIODIC WAVEFORMS THAT DO NOT SUFFER FROM EXCESSIVE ALIASING DISTORTION.

In the third category of methods, aliasing is suppressed by changing the slope of the spectrum: These methods render the spectral tilt of the signal to be steep before sampling by applying an exponential decay function to the harmonic amplitudes and then restore the spectral tilt with a digital filter. Despite this operation, all frequency components that exceed the Nyquist limit are reflected to the audio band, but their amplitude is reduced with respect to the frequency components that were originally there. Thus, it is possible this way to reduce the amplitude of all aliased frequency components.

Lane et al. [20] proposed to full-wave rectify a sine wave and to modify the resulting signal with filters. The spectrum of the rectified sine wave contains all harmonics of the fundamental but its tilt is steeper than that of the sawtooth waveform, for example. This leads to reduction of the level of aliased components after the spectral envelope is corrected with a gentle highpass filter. The differentiated parabolic waveform-based techniques utilize a similar train of thought [21], [14], [15]. The simplest application of the spectral tilt modification is the oversampling of the waveform, see, e.g., [7, pp. 423–424].

In the following, we review specific algorithms belonging to the three classes above.

BANDLIMITED TECHNIQUES

Additive synthesis, as in (1), is in theory an ideal solution for bandlimited synthesis of periodic waveforms. It offers freedom to choose the number of harmonics as well as their frequency, amplitude, and phase. The computational cost of additive synthesis increases linearly with number of harmonics, and unfortunately it becomes enormous for low fundamental frequencies, if all harmonics up to the Nyquist limit are incorporated. As a replacement for the explicit summation of sine waves, the discrete summation formulas [10], [11], [22], [13], the wavetable [23], [24], the group additive [25], the multirate additive [26], or the inverse fast Fourier transform (FFT)-based [7 (pp. 458–467), 27] synthesis techniques may be used. Each of these has their own advantages and disadvantages.

Winham and Steiglitz [10] and Moorer [11] have proposed the use of a closed-form summation formula to generate a waveform containing a specified number of sinusoidal components. A low-order digital filter can shape the roll-off rate of the spectrum so that, for example, the sawtooth waveform can be closely approximated. Stilson and Smith categorize similar techniques based on filtering an impulse train under the term BLIT [17], [18]. While these methods do not require as many operations per sample as additive synthesis at low frequencies, they have a drawback: a division per sample is required, and this leads to numerical errors that must be controlled, when the denominator is close to zero.

Another way to avoid full-blown additive synthesis is to compute beforehand many versions of the bandlimited waveform with a different number of harmonic components, and then use them in wavetable synthesis [23], [24]. This is a successful technique that is efficient to compute and free of numerical problems. It offers the added possibility to store any waveshapes, such as the oscillator waveforms of analog synthesizers [28]. However, wavetable synthesis consumes much memory and requires interpolation in changing the fundamental frequency of a tone. A variation that saves memory consists of a small set of wavetables, such as one per octave, which are played at the various speeds using sample rate conversion techniques [23]. Group additive synthesis distributes the harmonics in different wavetables and uses a mix of two or more wavetables to generate the desired waveform. Horner and Wun have investigated the improvement of the signal-to-noise ratio in group additive synthesis by reducing the peak amplitude of each wavetable [29]. Maher [30] has recently discussed wavetable data compression techniques, which are of great concern in mobile applications.

Computational complexity can be traded off against memory usage in wavetable synthesis. Considering that the spectral envelope of typical waveforms is approximately of the 1/frequency type, bandlimiting the contents of wavetables to 1/4 or 1/8 of the Nyquist limit allows the use of linear interpolation without audible aliasing. This, however, presents a significant increase in memory consumption, especially in the case of low fundamental frequencies. When limitations of memory size are more stringent than those in computational power, a simple trick can be used to extend the range down to sub-audio frequencies without memory increase: The waveform can be stretched on the fly by powers of two by shifting out the least significant bits of the address. This produces mirror images of the spectrum and low-pass-filters it with a sinc-shaped frequency response. The waveform has to be critically bandlimited and this necessitates the use of a low-pass filter with nearly flat passband and high stop-band attenuation for interpolation.

The multirate additive synthesis technique [26] supplies the possibility to run sinusoidal oscillators at slow speed with the help of decimation and interpolation operations, which leads to savings in computational burden. A varying fundamental frequency brings about the inconvenience of swapping oscillators from one subband to another. It then becomes a challenge how to avoid audible disturbances during such changes.

QUASI-BANDLIMITED TECHNIQUES

Nearly alias-free waveforms can be synthesized when the continuous-time waveform is first filtered with an antialiasing filter to attenuate the frequencies above the Nyquist limit. In the special cases of simple geometric waveforms, the antialiased signal can be directly evaluated. We first consider bandlimited impulse trains and show how they can be used for synthesizing the sawtooth and pulse waveforms. The use of bandlimited step functions for improving the efficiency of the synthesis is discussed. Finally, a new method based on polynomial bandlimited step functions is introduced.

FILTERED BANDLIMITED IMPULSE TRAINS

A continuous-time impulse train $x(t)$ with period T and unit amplitude is a sum of shifted unit impulses

$$x(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT), \tag{2}$$

where $\delta(a) = 1$, when $a = 0$, and $\delta(a) = 0$ otherwise. To sample this signal without aliasing, it must first be low-pass filtered, i.e., convolved with the impulse response of a low-pass filter. As the sample value of the impulse train is 1 at $t = kT$ and 0 elsewhere, the convolution is equivalent to replacing each unit impulse with the low-pass filter's impulse response. This is called the BLIT [17].

Since the period T is not necessarily an integral multiple of the sampling interval, the sampled BLIT may consist of fractionally shifted filter impulse responses. The polyphase structure (see, e.g., [31]) for the finite impulse response (FIR) filter allows efficient calculation of the sampled BLIT with arbitrary period T, because the filtered signal is evaluated at the sample points only. This reduces to mixing the impulse response of the appropriate branch of the polyphase FIR filter to the output signal. Figure 2 illustrates how an FIR filter's impulse response [see Figure 2(a)] can be decomposed into four polyphase branches [see Figure 2(b)–(e)].

The waveform can be most conveniently generated by using a phase accumulator (i.e., a modulo counter) and by inserting an impulse every time the phase counter rolls over. Assuming a polyphase filter with M branches, the filter branch can be computed as $p = \text{round}(Mtf_s/f)$, where $0 \leq t < 1$ is the phase accumulator value after the roll-over. When the number of polyphase branches M is sufficiently large, the selection of the nearest available branch using the rounding operation does not cause audible disturbances. The BLIT-SWS is a special case of this approach, in which the impulse response of the FIR filter is chosen to be a windowed sinc function sampled at an appropriate phase [17]. We also call the polyphase implementation of this method BLIT-SWS, although the impulse response may be obtained with a filter design method.

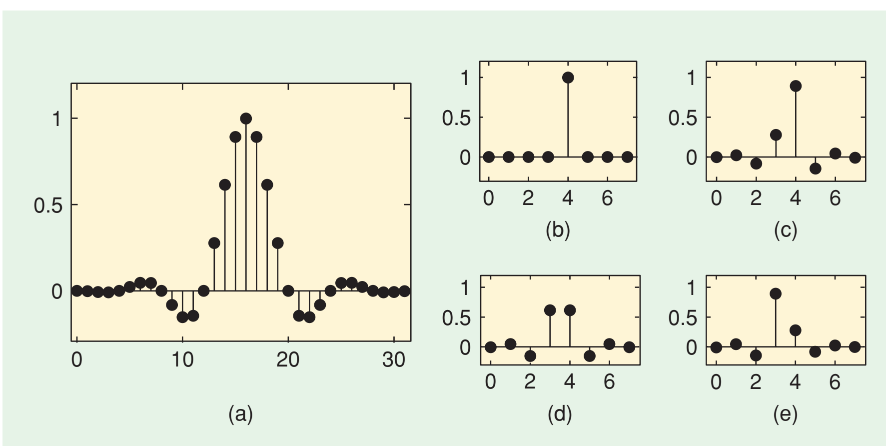
The sawtooth waveform can be generated by integrating the impulse train, i.e.,

$$s_{\text{saw}}(t) = -\frac{1}{2} + \int_0^t \delta(\tau) - C_1 \, d\tau \ \text{ with } C_1 = \int_0^T \frac{\delta(\tau)}{T} \, d\tau \tag{3}$$

where C_1 is the dc offset such that $s_{\text{saw}}(0-) = s_{\text{saw}}(T+)$. The same applies directly to the discrete-time case by substituting the bandlimited impulse for $\delta(t)$ and using numerical integration (running sum) [17]. Similarly, the pulse waveform can be generated by integrating two impulses, with the second one being time-delayed and sign reversed:

$$s_{\text{pulse}}(t) = w_{\text{pulse}} + \int_0^t \delta(\tau) - \delta(t - w_{\text{pulse}} T) \, d\tau, \tag{4}$$

where $0 < w_{\text{pulse}} < 1$ is the pulsewidth. Again a discrete-time waveform can be produced in same way as the sawtooth waveform by using the polyphase FIR filter structure.



[FIG2] (a) Impulse response of an FIR low-pass filter of length 32 and (b)–(e) its decomposition into four polyphase branches of length 8. The polyphase filters are generated by collecting every fourth sample from the original impulse response from a different starting point for each branch.

The numerical integration represents a problem in practical implementation. Even a tiny roundoff error can introduce a dc offset, which gets accumulated and may eventually result in a numerical overflow. Therefore, the ideal integrator may be replaced with a leaky integrator [17], [19], which can be implemented as a recursive low-pass filter.

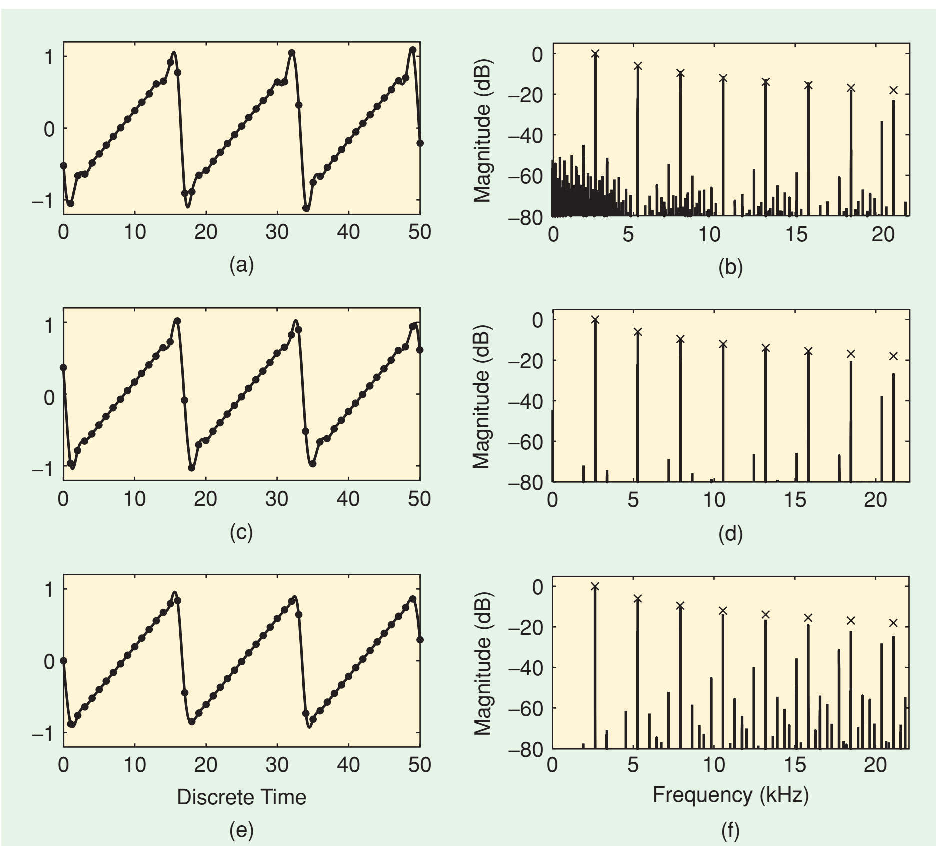
An example of sawtooth waveform synthesis using the BLIT-SWS method is given in Figure 3(a), when each impulse is replaced with a BLIT-SWS sequence of length 8. The BLIT-SWS signal has been designed using the least squares FIR filter design technique with the Kaiser window ($\beta = 3.5$). The polyphase structure has 32 branches, and thus the passband width in the filter specification has been $f_S/64$. Finally, the signal has been integrated with a one-pole filter whose pole radius is 0.9. The spectrum of this sawtooth approximation is presented in Figure 3(b). It shows that the aliasing is much reduced with respect to the trivial sawtooth wave [compare with Figure 1(b)]. Aliasing can be reduced more by extending the length of the BLIT-SWS sequence.

THE ALIASED SIGNAL COMPONENTS GENERALLY FALL BETWEEN THE DESIRED HARMONICS, WHEN THE FUNDAMENTAL FREQUENCY DOES NOT DIVIDE THE SAMPLING FREQUENCY, THUS RENDERING THE OVERALL SIGNAL INHARMONIC.

BANDLIMITED STEP SEQUENCES

An elegant method for avoiding the leaky integration in sawtooth wave generation is to replace the run-time integration with a sum of a rising ramp and the BLEP residual [19]. The BLEP signal can be produced by integrating a bandlimited impulse, which can be generated by using standard FIR filter design techniques. Figure 4 illustrates this principle. The bandlimited approximation for a unit step function in Figure 4(a) has been obtained by designing a long Mth-band FIR filter using the windowing technique. The Hann window was used for truncating the ideal impulse response of the Mth-band filter. Figure 4(b) shows the generated BLEP residual sequence, with the ideal unit step function subtracted from the integrated impulse response of the long FIR filter. When the BLEP pulse is implemented using the polyphase FIR filter structure, the integration can be performed in advance, when the filter coefficients are calculated. Then the BLEP synthesis reduces to inserting the appropriate branch of the integrated FIR filter response every time a discontinuity occurs in the waveform.

A bandlimited sawtooth wave can now be synthesized by mixing the output signal of the phase accumulator (i.e., the trivial sawtooth wave) with the BLEP residual inserted whenever the phase accumulator rolls over [19]. The BLEP residual needs to be centered around the roll-over point. In practice this requires either looking ahead half the BLEP residual length or, alternatively, delaying the output by the same amount. Further, to compensate for the discontinuity in the trivial sawtooth wave, the signal should have a unit step subtracted at the mid-point of the inserted BLEP residual, as seen in Figure 4(b). An example is given in Figure 3(c), when an 8-samples-long BLEP residual sequence is superimposed on each discontinuity of the trivial sawtooth signal. The BLEP sequence has been designed using the same design method and parameter values as the BLIT-SWS



[FIG3] Waveform and magnitude spectrum of sawtooth wave approximations produced by: (a), (b) the BLIT-SWS; (c), (d) the BLEP; and (e), (f) the PolyBLEP method. The x marks indicate the true level of harmonics of a sawtooth wave. The BLIT-SWS and BLEP methods use the same polyphase filters to illustrate the quality difference.

sequence above. The spectrum of this sawtooth approximation is presented in Figure 3(d). The aliasing has been reduced even more than using the BLIT-SWS method [see Figure 3(b)], which used the FIR filter of the same type and length.

The procedure for pulse wave synthesis using the BLEP is quite similar. However, now a BLEP residual is added whenever the phase accumulator either rolls over or the phase accumulator output exceeds the pulse width. Thus, two transitions per period are produced. The resulting train of BLEP residuals is again combined with the trivial version of the waveform. The trivial pulse waveform should be synchronized to the bandlimited steps. An obvious way is to use the output of a comparator with the same threshold as is used for inserting the BLEP residual sequence.

Obviously, the BLEP residual sequence can be used for bandlimiting any discontinuity in the waveform by simply mixing its properly shifted and scaled version with the signal at the appropriate time. The required lookahead causes a minor practical problem, because it introduces delay. It is unfortunately impossible to completely eliminate the lookahead delay, but it is possible to significantly reduce it. By transforming the low-pass FIR filter into a minimum-phase filter before integration [19], the required lookahead can be reduced to a few samples in typical practical cases. The MinBLEP must also have a unit step subtracted from it. However, the split point (and thus lookahead) is not quite trivial to calculate. It should be selected such that the sum of the MinBLEP table is as close to zero as possible. Typically, the split point is almost, but not exactly, the same as the group delay of the minimum-phase FIR filter at low frequencies.

There are no special requirements for the low-pass FIR filter. As the human ear is least sensitive at high frequencies, it makes sense to emphasize stopband attenuation at the expense of performance near the Nyquist limit. We have obtained good results using least squares FIR design. The BLEP method is less sensitive than BLIT-SWS to stopband attenuation, since the sampled function has been integrated before sampling. This corresponds to an inherent 6 dB/oct low-pass filtering as opposed to the BLIT-SWS method, which performs the integration after sampling. The number of polyphase branches is typically restricted by memory usage. Linear interpolation between adjacent branches helps to reduce the number of required branches, which can otherwise be quite large, if good aliasing rejection is required.

POLYNOMIAL BANDLIMITED STEP FUNCTION

Next we introduce a modification of the BLEP method, which avoids a lookup table by evaluating the approximately bandlimited transitions using closed-form formulas. We derive the new method by low-pass filtering a discontinuity in a continuous-time square waveform using a function that can be expressed in closed form. One of the simplest functions for this purpose is the triangular pulse, which is defined by two linear segments:

$$s_{\text{tri}}(t) = \begin{cases} t+1, & \text{when } -1 \le t \le 0 \\ 1-t, & \text{when } 0 < t \le 1 \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

This pulse is known to have a magnitude frequency response of the form sinc^2; that is, a gentle low-pass filter that can be used as an antialiasing filter. When the pulse is convolved with the rising edge of the square pulse, it gets integrated. The integrated triangular pulse can be expressed as

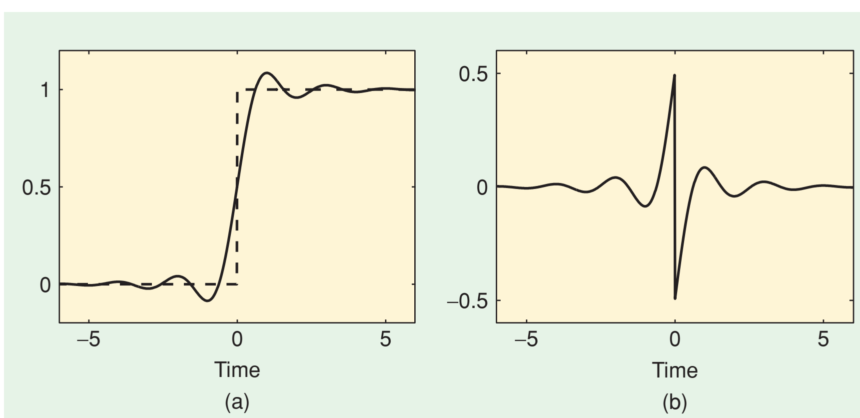
$$s_{\text{tri,int}}(t) = \begin{cases} 0, & \text{when } t < -1 \\ \frac{t^2}{2} + t + \frac{1}{2}, & \text{when } -1 \le t \le 0 \\ t - \frac{t^2}{2} + \frac{1}{2}, & \text{when } 0 < t \le 1 \\ 1, & \text{when } t > 1, \end{cases} \tag{6}$$

where the constant terms ½ have been selected based on continuity. We call this closed-form function the polynomial bandlimited step function, or PolyBLEP. We can now extract the BLEP residual by subtracting the unit step function. This yields the following two polynomial segments:

$$p_{\text{PolyBLEP}}(t) = \begin{cases} \frac{t^2}{2} + t + \frac{1}{2}, & \text{when } -1 \le t \le 0 \\ t - \frac{t^2}{2} - \frac{1}{2}, & \text{when } 0 < t \le 1. \end{cases} \tag{7}$$

This function can be used in a similar way as the BLEP residual sequences above for smoothing the discontinuities of any waveform in principle.

Figure 3(e) shows an example of the PolyBLEP-based sawtooth waveform approximation. Only two samples around every discontinuity have been affected by the PolyBLEP correction: one sample value before and one after the transition. The corresponding magnitude spectrum given in Figure 3(f)

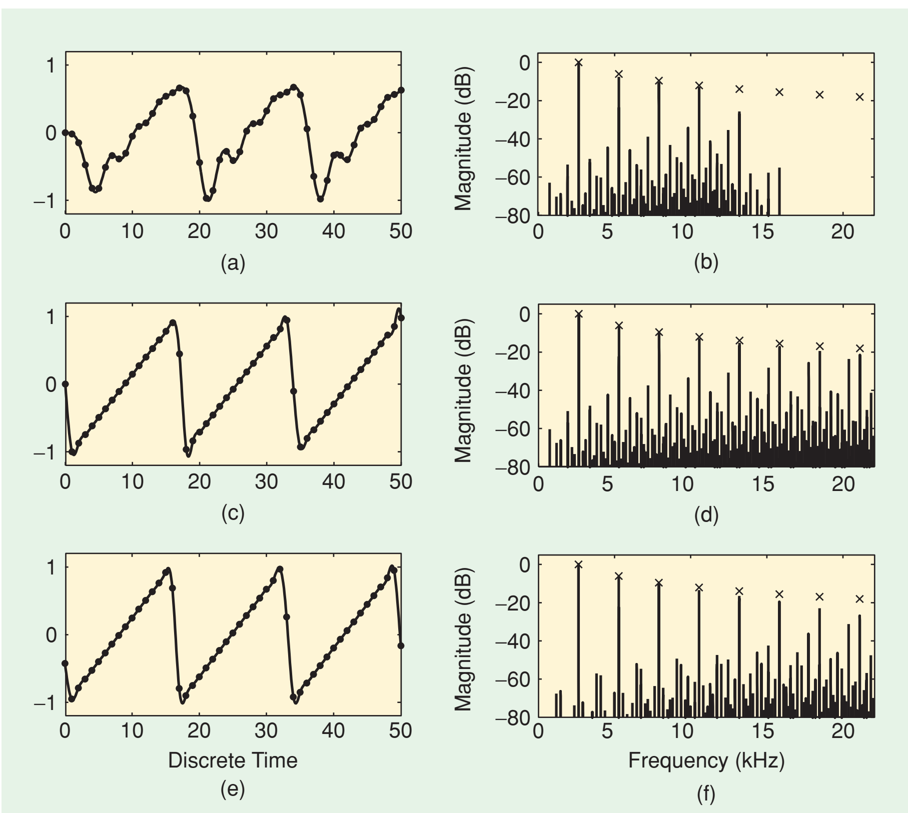


[FIG4] (a) An ideal unit step function (dashed line) and its approximately bandlimited version (solid line) obtained by integrating the impulse response of an FIR low-pass filter. (b) The approximated bandlimited step function (BLEP) residual, which is obtained by subtracting the ideal unit step function from the integrated impulse response [19].

demonstrates that the alias-suppression obtained is remarkable even with the simple integrated triangular pulse, which can be considered a second-order PolyBLEP technique. The implementation of this method requires a logic to identify the discontinuities, determination of the fractional delay or location of the discontinuity between the samples, evaluating the two second-order polynomials given in (7), and adding the results to the two sample values before and after the transition. The determination of the fractional delay requires a division by the fundamental frequency, just like other methods in this category. This method does not require a table for the filter coefficients. Furthermore, it does not require decomposing the filter into polyphase branches.

ADDITIVE SYNTHESIS IS IN THEORY AN IDEAL SOLUTION FOR BANDLIMITED SYNTHESIS OF PERIODIC WAVEFORMS.

Higher-order PolyBLEP functions can be derived similarly by integrating a polynomial antialiasing filter response in the symbolic form. The continuous-time impulse responses corresponding to Lagrange interpolation filters of all orders are suitable candidates to start with. The triangular pulse used above is the continuous-time impulse response of the linear-interpolation filter, which is the first-order Lagrange interpolation filter [32]. We leave the derivation of higher order cases and other polynomial BLEP techniques for future work.



[FIG5] Waveform and magnitude spectrum of sawtooth wave approximations produced by: (a), (b) Lane's method; (c), (d) the basic DPW method; and (e), (f) the two-times oversampled DPW (DPW2X) method with an FIR decimation filter of order 6. The x marks indicate the true level of harmonics of a sawtooth wave.

REDUCTION OF ALIASING BY MODIFYING THE SPECTRAL TILT

FILTERING OF RECTIFIED SINE WAVES

Lane et al. introduced algorithms that full-wave rectify a sine wave and modify the resulting signal with two digital filters to obtain approximations of classic periodic waveforms with reduced aliasing [20]. The sawtooth wave is used as the example waveshape. In Lane's sawtooth algorithm [20], a first-order infinite impulse response (IIR) filter corrects the spectral tilt to be close to that of the sawtooth signal and a Butterworth low-pass filter suppresses the aliased components (and everything else, including the harmonics) at high frequencies. The first-order filter has the transfer function

$$H(z) = \frac{\alpha(1 - z^{-1})}{1 - \gamma z^{-1}} \tag{8}$$

where $\gamma = \cos(2\pi \times 16 f/f_s)/[1 + \sin(2\pi \times 16 f/f_s)]$ and $\alpha = (1 + \gamma)/2$. This filter has a fixed transmission zero at dc and a tunable pole on the real axis. Figure 5(a) and (b) shows the sawtooth waveform approximation generated using Lane's method and its spectrum, respectively. The waveform is visually quite different from the sawtooth wave, but the levels of harmonics are approximately right up to about 10 kHz in the spectrum in Figure 5(b). The effect of the Butterworth low-pass filter (order 8, cutoff frequency 12 kHz) is quite obvious.

DIFFERENTIATED PARABOLIC WAVEFORM

Välimäki realized that a digital signal that closely resembles the sawtooth wave—but with attenuated aliasing distortion—can be produced by differentiating a piecewise parabolic waveform [21]. The simplest version of the differentiated parabolic waveform (DPW) algorithm generates the sawtooth waveform in four stages: First the trivial sawtooth waveform is generated using a bipolar modulo counter, then the waveform is raised to the second power, the signal is differentiated with a first difference filter with transfer function $1 - z^{-1}$, and, finally, the obtained waveform is scaled by factor

$$c = f_s/[4f(1 - f/f_s)]. \quad (9)$$

This scaling factor may well be replaced with $c = f_s/4f$, which gives a slightly too small gain for high fundamental frequencies but is accurate within 1 dB up to 4 kHz at the sampling rate of 44.1 kHz [15].

The waveform and spectrum of the sawtooth wave approximation obtained with the basic DPW technique are shown in Figure 5(c) and (d). The waveform differs from the trivial sawtooth wave at one sample only in each period [compare Figure 5(c) with Figure 1(a)], but the improvement in the frequency domain is dramatic [compare Figure 5(d) with Figure 1(b)]. Interestingly, a waveform identical to that of the basic DPW algorithm can be obtained by sampling a skewed triangle wave. Its duty cycle must be set so that the duration of the downward slope of the underlying continuous-time sawtooth function is equivalent to the sampling interval.

IN THE POLYBLEP ALGORITHM, AN INTEGRATED POLYNOMIAL INTERPOLATION FUNCTION IS USED FOR ACQUIRING SAMPLES TO CORRECT THE TRANSITION REGION OF THE WAVEFORM.

A multirate version of the DPW algorithm yields improved alias suppression and leaves room for optimization by using various choices of decimation filters [21]. In the two-times oversampled DPW algorithm (called the DPW2X [21]), the parabolic waveform is generated at a rate twice that of the output sample rate, the signal is filtered by a decimation filter, then the sampling rate is reduced by factor two by skipping every second sample, and finally the signal is differentiated. It has been previously shown that the sound quality of this algorithm is superior to the basic DPW method even with the simplest possible decimation filter, a two-point averager [21]. More recent experiments show that a three-point averager is still better in the sense that it suppresses aliasing well at high frequencies and helps to reduce beating artifacts. The performance can be further improved by using a high-order decimation filter or by oversampling the parabolic wave by a factor larger than two.

In Figure 5(e) and (f) we show an example waveform and spectrum obtained with the DPW2X algorithm, when the decimation filter is a halfband linear-phase FIR filter of order 6 (its coefficient values are –0.0228, 0, 0.275, 0.5, 0.275, 0, and –0.0228). When this filter is executed at the sample rate of 88.2 kHz, it has a notch at 40,650 Hz. After decimation, this notch effectively attenuates aliased components around the frequency 3,450 Hz (i.e., 44,100 Hz–40,650 Hz), which is in the most sensitive region of hearing. As a consequence, the aliasing is reduced more at low and middle frequencies in Figure 5(f) than in Figure 5(d). The waveform in Figure 5(e) has smoother transients than in Figure 5(c).

The pulse waveform can be generated using the DPW method either by adding two sawtooth waves with a phase shift corresponding to the duty cycle [15] or by filtering the DPW sawtooth wave with an FIR comb filter whose delay-line length is proportional to the duty cycle [33]. A DPW-based triangular waveform algorithm is detailed in [15].

COMPARISON

We compare the antialiasing algorithms in terms of sound quality and other characteristics, such computational complexity. As a measure of sound quality we use an algorithm called the NMR, which has been proposed for appraisal of perceptual audio coding methods [34], [35]. The NMR accounts for the frequency-dependent sensitivity of hearing and the auditory masking phenomenon in the frequency domain. We use a reduced version of the NRM that neglects the temporal masking. We briefly recapitulate the NMR algorithm in the following.

The NMR takes two signals as input: the reference and the corrupted signal. In this case the reference signal is the bandlimited waveform and the corrupted signal is its approximation that suffers from aliasing. The error signal is computed as the difference of the reference and the corrupted signal. To avoid the influence of amplitude and phase errors of partials in the NMR figure, we estimate the amplitude and phase of harmonics from the corrupted signal and compose the reference signal from them. The algorithm then computes the 1,024-point magnitude spectrum of the reference and error signals with the FFT using the Hann window. The spectra are divided into segments that are approximately the size of the critical bands of hearing. The average energy of each band is scaled by dividing by its width in bins and is converted to the decibel scale. To simulate the frequency masking phenomenon, a fraction of the signal energy is copied from each critical band to all neighboring bands using an interband spreading function. The hearing threshold is applied as an additive term, the final energy per band is computed, and energies of all bands are added up. Finally, a single NMR figure is obtained as a ratio of the error to the mask threshold. We use the 44.1-kHz sampling rate in this evaluation. This affects the choice of critical bandwidths and spreading functions [35].

Figure 6(a) shows the NMR of the trivial sawtooth and the sawtooth approximations obtained with Lane's method, the basic DPW algorithm, and the two-times oversampled DPW algorithm employing the above sixth-order decimation filter. The NMR is computed for 88 fundamental frequency values that span the frequency range of the piano from 27.5 Hz to 4,186 Hz. Small NMR values are considered good while large values are considered poor. In audio coding, it has been said that when the NMR is below –10 dB, it is safe to assume that the signal is free from audible artifacts [35]. Figure 6(b) illustrates the amplitude error of the sawtooth signals.

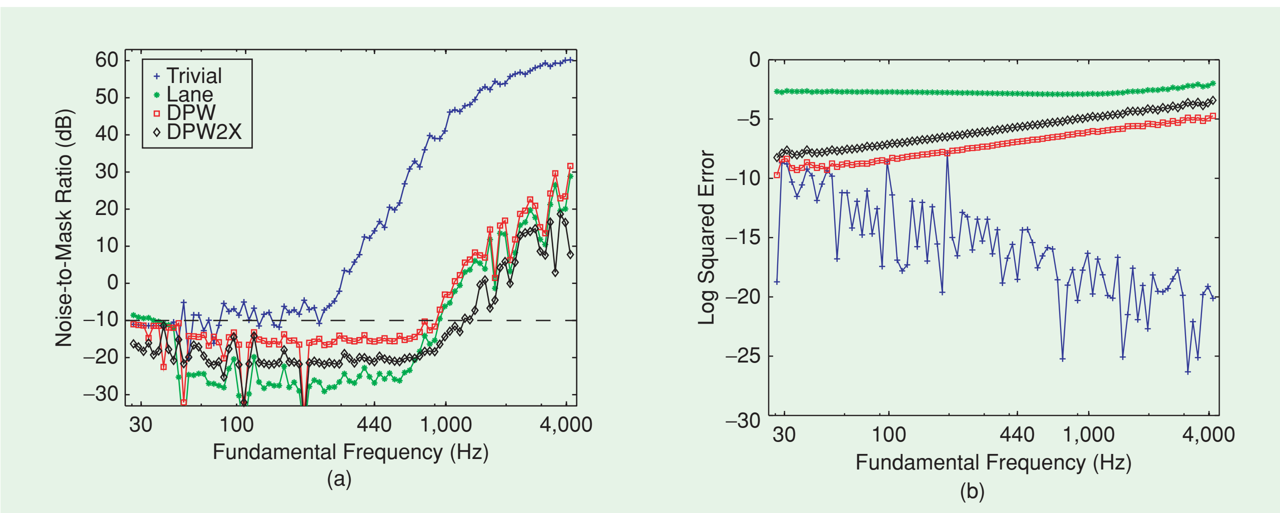
From Figure 6 one can read that the trivial sawtooth waveform has a good sound quality only when the fundamental frequency is very low. The Lane and the DPW method are practically free of aliasing distortion below about 800 Hz and the DPW2x method is perfect below about 1,200 Hz. However, these methods suffer from amplitude distortion, as seen in Figure 6(b).

The properties of the nine oscillator algorithms discussed in this article have been gathered in Table 1. The computational cost, memory consumption, and sound quality are informally appraised. Several methods offer a high sound quality, but it is realistic to expect perfect quality at all fundamental frequencies from the additive synthesis, the BLEP, and the BLIT-SWS methods. The additive synthesis method has a very high computational complexity, because it computes each harmonic component separately. In mobile audio processors and other cheap implementations, a very limited amount of RAM memory, such as a few kilowords, may be available. These platforms may allow only a dozen instructions per sample for a synthesis algorithm, because several voices must be computed simultaneously. The DPW, the DPW2X, and the PolyBLEP methods are fit for such restricted applications, because the algorithms are simple and the memory consumption is very low.

A MULTIRATE VERSION OF THE DIFFERENTIATED PARABOLIC WAVEFORM (DPW) ALOGORITHM YIELDS IMPROVED ALIAS SUPPRESSION AND LEAVES ROOM FOR OPTIMIZATION BY USING VARIOUS CHOICES OF DECIMATION FILTERS.

CONCLUSIONS

In this article, oscillator algorithms for digital subtractive synthesis were reviewed. The algorithms were divided into three categories: bandlimited, quasi-bandlimited, and alias-reducing methods. In the first category, the most interesting methods are in practice those that utilize wavetable techniques. The second category consists of methods that low-pass-filter the underlying continuous-time signal prior to sampling. The optimization of the previously introduced BLIT and BLEP methods were considered as a filter design problem. A new technique called the PolyBLEP method was introduced as a variation of the BLEP method that



[FIG6] (a) Noise-to-mask ratio (NMR) and (b) log squared error in harmonic amplitudes for sawtooth wave approximations obtained with the trivial, the Lane, the basic DPW, and the DPW2x methods. The hypothetical threshold of audibility for aliasing noise is indicated with a horizontal dashed line at –10 dB.

[TABLE 1] COMPARISON OF OSCILLATOR ALGORITHMS IN ORDER OF SOUND QUALITY FROM POOREST TO HIGHEST.

ALGORITHM	COMPUTATIONAL COMPLEXITY	MEMORY CONSUMPTION	SOUND QUALITY	REMARKS
TRIVIAL	VERY LOW	NO	POOR	**PRACTICALLY USELESS, EXCEPT AT LOW** *f*
LANE	AVERAGE	VERY SMALL	GOOD	**REQUIRES A SINE OSCILLATOR**
DPW	VERY LOW	VERY SMALL	GOOD	**SIMPLE AND USEFUL**
DPW2X	LOW	VERY SMALL	VERY GOOD	**SLIGHTLY BETTER THAN DPW**
POLYBLEP	LOW	VERY SMALL	VERY GOOD	**A DIVISION PER DISCONTINUITY**
WAVETABLE	AVERAGE…HIGH	VERY LARGE	VERY GOOD	**REQUIRES INTERPOLATION**
BLIT-SWS	AVERAGE…HIGH	SMALL	VERY GOOD	**A DIVISION PER DISCONTINUITY**
BLEP	AVERAGE	SMALL	EXCELLENT	**A DIVISION PER DISCONTINUITY**
ADDITIVE	**VERY HIGH**	**VERY SMALL**	**EXCELLENT**	**COMPUTATIONALLY INTENSIVE**

does not require a table lookup but is based on a closed-form formula. In the PolyBLEP algorithm, an integrated polynomial interpolation function is used for acquiring samples to correct the transition region of the waveform.

In the third category, the DPW oscillator algorithm generates an approximate sawtooth waveform that has reduced aliasing. This recently proposed method is probably the simplest useful technique for this purpose, because only the trivial sawtooth is simpler, but it is practically useless due to its heavy aliasing. An alternative decimation filter was proposed for the DPW2X algorithm to suppress aliasing well in the frequency region where human hearing is most sensitive.

ACKNOWLEDGMENTS

This work has been financed partly by the Academy of Finland (project no. 104934, "Control, Analysis, and parametric Synthesis of Audio Signals"). The authors are grateful to Dr. Teppo Karema and Mr. Tomi Valkonen of VLSI Solution Oy (Tampere, Finland) for helpful discussions.

AUTHORS

Vesa Välimäki (vesa.valimaki@tkk.fi) is a professor at the Laboratory of Acoustics and Audio Signal Processing, TKK (Helsinki University of Technology), Espoo, Finland. He received the M.S., the Licentiate of Science, and the Doctor of Science degrees in technology, all in electrical engineering, from TKK in 1992, 1994, and 1995, respectively. His research interests include music synthesis, acoustics of musical instruments, and digital filter design. He is a Senior Member of the IEEE.

Antti Huovilainen (antti.huovilainen@iki.fi) was born in Espoo, Finland, in 1978. He is currently finishing his M.S. in electrical engineering at TKK (Helsinki University of Technology), Espoo, Finland. His research interests lie in the modeling of musical circuits for digital implementation, particularly guitar amplifiers and analog synthesizers. He has written several papers on virtual analog synthesis techniques.

REFERENCES

[1] H.F. Olson and H. Belar, "Electronic music synthesis," *J. Acoust. Soc. Amer.*, vol. 27, no. 3, pp. 595–612, May 1955.

[2] R.A. Moog, "Voltage-controlled electronic music modules," *J. Audio Eng. Soc.*, vol. 13, no. 3, pp. 200–206, 1965.

[3] J. Chadabe, *Electric Sound: The Past and Promise of Electronic Music*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[4] J.A. Moorer, "Signal processing aspects of computer music: A survey," *Proc. IEEE*, vol. 65, no. 8, pp. 1108–1137, Aug. 1977.

[5] H.G. Alles, "Music synthesis using real time digital techniques," *Proc. IEEE*, vol. 68, no. 4, pp. 436–449, Apr. 1980.

[6] J.W. Gordon, "System architectures for computer music," *ACM Computing Surveys*, vol. 17, no. 2, pp. 191–233, June 1985.

[7] H. Chamberlin, *Musical Applications of Microprocessors*, 2nd ed. Rochelle Park, NJ: Hayden Books, 1985.

[8] P.R. Samson, "A general-purpose digital synthesizer," *J. Audio Eng. Soc.*, vol. 28, no. 3, pp. 106–113, Mar. 1980.

[9] J.O. Smith, "Viewpoints on the history of digital synthesis," in *Proc. Int. Computer Music Conf.*, Montreal, Canada, Extended version available online at http://www-ccrma.stanford.edu/~jos/kna/. pp. 1–10, 1991.

[10] G. Winham and K. Steiglitz, "Input generators for digital sound synthesis," *J. Acoust. Soc. Amer.*, vol. 47, part 2, pp. 665–666, Feb. 1970.

[11] J.A. Moorer, "The synthesis of complex audio spectra by means of discrete summation formulae," *J. Audio Eng. Soc.*, vol. 24, no. 9, pp. 717–727, Nov. 1976.

[12] J. Chowning, "The synthesis of complex audio spectra by means of frequency modulation," *J. Audio Eng. Soc.*, vol. 21, no. 7, pp. 526–534, 1973.

[13] F.R. Moore, *Elements of Computer Music*. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[14] A. Huovilainen and V. Välimäki, "New approaches to digital subtractive synthesis," in *Proc. Int. Computer Music Conf.*, Barcelona, Spain, pp. 399–402, Sept. 2005.

[15] V. Välimäki and A. Huovilainen, "Oscillator and filter algorithms for virtual analog synthesis," *Computer Music J.*, vol. 30, no. 2, pp. 19–31, 2006.

[16] A. Chaudhary, "Bandlimited simulation of analog synthesizer modules by additive synthesis," in *Proc. Audio Engineering Society 105th Conv.*, San Francisco, CA, paper no. 4779, 1998.

[17] T. Stilson and J. Smith, "Alias-free digital synthesis of classic analog waveforms," in *Proc. Int. Computer Music Conf.*, Hong Kong, China, pp. 332–335, 1996.

[18] T. Stilson, "Efficiently-variable non-oversampling algorithms in virtual-analog music synthesis—A root-locus perspective." Ph.D. dissertation, Dept. Elect. Eng., Stanford Univ., Stanford, CA, Available: http://ccrma.stanford.edu/~stilti/papers/. 2006.

[19] E. Brandt, "Hard sync without aliasing," in *Proc. Int. Computer Music Conf.*, Havana, Cuba, pp. 365–368, 2001.

[20] J. Lane, D. Hoory, E. Martinez, and P. Wang, "Modeling analog synthesis with DSPs," *Computer Music J.*, vol. 21, no. 4, pp. 23–41, 1997.

[21] V. Välimäki, "Discrete-time synthesis of the sawtooth waveform with reduced aliasing," *IEEE Signal Processing Lett.*, vol. 12, no. 3, pp. 214–217, Mar. 2005.

[22] J. Lazzaro and J. Wawrzynek, "Subtractive synthesis without filters," in *Audio Anecdotes II—Tools, Tips, and Techniques for Digital Audio*, K. Greenebaum and R. Barzel, Eds. Wellesley, MA: A.K. Peters, pp. 55–64, 2004.

[23] D.C. Massie, "Wavetable sampling synthesis," in *Applications of Digital Signal Processing to Audio and Acoustics*, M. Kahrs and K. Brandenburg, Eds. Norwell, MA: Kluwer, pp. 311–341, 1998.

[24] P. Burk, "Band limited oscillators using wave table synthesis," in *Audio Anecdotes II—Tools, Tips, and Techniques for Digital Audio*, K. Greenebaum and R. Barzel, Eds. Wellesley, MA: A.K. Peters, pp. 37–53, 2004.

[25] P. Kleczkowski, "Group additive synthesis," *Computer Music J.*, vol. 13, no. 1, pp. 12–20, 1989.

[26] D.K. Phillips, "Multirate additive synthesis," *Computer Music J.*, vol. 23, no. 1, pp. 28–40, 1999.

[27] X. Rodet and P. Depalle, "Spectral envelopes and inverse FFT synthesis," in *Proc. Audio Engineering Society 83rd Conv.*, San Francisco, CA, preprint no. 3393, 1992.

[28] J. Kleimola, "Design and implementation of a software sound synthesizer," M.S. thesis, Helsinki Univ. of Tech., Espoo, Finland, Available: http://www.acoustics.hut.fi/publications/. 2005.

[29] A. Horner and S. Wun, "Low peak amplitudes for group additive synthesis," *J. Audio Eng. Soc.*, vol. 53, no. 6, pp. 475–484, June 2005.

[30] R.C. Maher, "Wavetable synthesis strategies for mobile devices," *J. Audio Eng. Soc.*, vol. 53, no. 3, pp. 205–212, Mar. 2005.

[31] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[32] T.I. Laakso, V. Välimäki, M. Karjalainen, and U.K. Laine, "Splitting the unit delay—Tools for fractional delay filter design," *IEEE Signal Processing Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.

[33] D. Lowenfels, "Virtual analog synthesis with a time-varying comb filter," in *Proc. Audio Engineering Society 115th Conv.*, New York, paper no. 5960, Oct. 2003.

[34] K. Brandenburg, "Evaluation of quality for audio encoding at low bit rates," in *Proc. Audio Engineering Society 82nd Conv.*, London, UK, preprint no. 2433, 1987.

[35] K. Brandenburg and T. Sporer, "NMR and masking flag: Evaluation of quality using perceptual criteria," in *Proc. Audio Engineering Society 11th Int. Conf. on Test and Measurement*, Portland, OR, pp. 169–179, May 1992.