

ALIAS-FREE VIRTUAL ANALOG OSCILLATORS USING A FEEDBACK DELAY LOOP

Juhan Nam

CCRMA, Stanford University
Stanford, USA
juhan@ccrma.stanford.edu

Vesa Välimäki

Department of Signal Processing and Acoustics
Helsinki Univ. of Tech., TKK, Espoo, Finland
vesa.valimaki@tkk.fi

Jonathan S. Abel

CCRMA, Stanford University
Stanford, USA
abel@ccrma.stanford.edu

Julius O. Smith

CCRMA, Stanford University
Stanford, USA
jos@ccrma.stanford.edu

ABSTRACT

The rich spectra of classic waveforms (sawtooth, square and triangular) are obtained by discontinuities in the waveforms or their derivatives. At the same time, the discontinuities lead to aliasing when the waveforms are digitally generated. To remove or reduce the aliasing, researchers have proposed various methods, mostly based on limiting bandwidth or smoothing the waveforms. This paper introduces a new approach to generate the virtual analog oscillators with no aliasing. The approach relies on generating an impulse train using a feedback delay loop, often used for the physical modeling of musical instruments. Classic waveforms are then derived from the impulse train with a leaky integrator. Although the output generated by this method is not exactly periodic, it perceptually sounds harmonic. While additional processing is required for time-varying pitch shifting, resulting in some high-frequency attenuation when the pitch changes, the proposed method is computationally more efficient than other algorithms and the high-frequency attenuation can be also adjusted.

1. INTRODUCTION

The principle of subtractive synthesis is to generate sound by shaping a sound source with rich harmonics. The sound source is commonly chosen among simple periodic oscillators, such as sawtooth, square and triangular waveforms. They contain discontinuities in the waveforms or their derivatives, thereby resulting in a theoretically infinite bandwidth. In generating the waveforms in the digital domain, such wideband characteristics has become a challenging issue because harmonic contents above the Nyquist limit are aliased, which causes unpleasant noise particularly at high fundamental frequencies.

A number of algorithms to remove or reduce the aliasing have been proposed. According to [1], they are classified into three groups; bandlimited, quasi-bandlimited, and alias-suppressing algorithms. The first group features algorithms with no aliasing by generating only harmonic partials. A straightforward method to generate bandlimited waveforms is the additive synthesis technique, which represents waveforms as a sum of sinusoids. Another method is the so-called “discrete summation formula”,

which allows the simultaneous generation of N harmonics of a fundamental from a closed-form expression [2][3]. The second group does not eliminate aliasing completely but rather suppresses it sufficiently to make it less disturbing. These quasi-bandlimited algorithms include the bandlimited impulse train (BLIT) method and the bandlimited step (BLEP) method [4][5]. Both methods basically reduce the aliasing by smoothing the discontinuities using oversampled lowpass filters. The third group reduces aliasing by modifying the spectral tilt of the input signal. A simple method is to oversample trivial waveforms and apply a lowpass filter before downsampling the signal [6]. The differentiated parabolic waveform (DPW) method also belongs to this group [7]. Although the aliasing-suppressing methods are computationally efficient, they suffer from the audible aliasing noise at high fundamental frequencies.

This paper proposes an alias-free oscillator generation algorithm, classified under the first group. Unlike the previous methods that generate a periodic waveform, however, this method produces a pseudo-periodic signal, whose waveform is not periodic but perceptually sounds harmonic. Sections 2 and 3 begin by introducing the basic structure to generate an impulse train and follow with its harmonic properties. In Section 4, classic waveforms (sawtooth, square and triangular) are derived from the impulse train using the BLIT method. Section 5 explores the relationship between our proposed method and a sinusoid generation algorithm. Lastly, Section 6 suggests numerical techniques for computational efficiency.

2. OSCILLATORS USING A FEEDBACK DELAY LOOP

The feedback delay loop has been extensively used in the field of sound synthesis, especially for physical modeling of musical instruments [8]. Its principle lies in rendering the periodicity, the fundamental nature of musical sounds, by repeating the previous state with a length corresponding to the number of samples in the delay line. The feedback delay loop is typically combined with linear filters or non-linear transfer functions to simulate complex physical behaviors of musical instruments. On the other hand, the basic structure can be also applied to generating simple sound oscillators.

2.1. Impulse trains

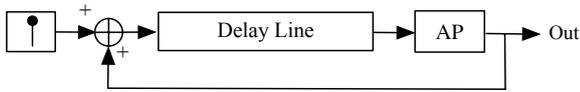


Figure 1: Diagram for generating an impulse train using a feedback delay loop. The block AP contains an allpass filter.

Figure 1 shows a diagram for generating an impulse train. The idea is to inject an impulse into the feedback delay loop so that the impulse is repeated periodically. The feedback delay loop is composed of a delay line and a fractional delay filter. The two delay units correspond to the integer and fractional parts of a period in samples, respectively. The fractional delay filter is usually chosen from either Lagrange polynomial interpolators or Thiran allpass filters for maximal flatness at dc [9]. In order to generate a signal with a constant level of energy over time, an allpass filter (AP) can be used, as shown in Figure 1. The details of Thiran allpass filters are examined in [9]. Here the first- and second-order Thiran allpass filters are reviewed.

The first-order Thiran allpass filter is computed by the following recursive equation

$$y(n) = a_1 x(n) + x(n-1) - a_1 y(n-1) \quad (1)$$

where the coefficient a_1 is given by setting the phase delay at dc to D

$$a_1 = \frac{1-D}{1+D} \quad (2)$$

Delay D is chosen between 0.418 and 1.418, which is the optimal range for the first-order allpass filter to have a maximally flat delay over the entire frequencies [9]. Since the phase delay is nearly constant at low frequencies, Eq. (2) is well approximated for practically used fundamental frequencies. For high fundamental frequencies, which have slight detuning with Eq. (2), a more accurate formula can be used as follows:

$$a_1 = \frac{\sin((1-D)\omega_0/2)}{\sin((1+D)\omega_0/2)} \quad (3)$$

where ω_0 is the fundamental frequency in radians per sample, and D is the phase delay at ω_0 by the allpass filter [10]. Note that Eq. (3) reduces to Eq. (2) when ω_0 is small.

The second-order Thiran allpass filter is computed by the following recursive equation.

$$y(n) = a_2 x(n) + a_1 x(n-1) + x(n-2) - a_1 y(n-1) - a_2 y(n-2) \quad (4)$$

where coefficients a_1 and a_2 can be given as

$$a_1 = -\frac{D-2}{D+1} \quad a_2 = \frac{(D-1)(D-2)}{(D+1)(D+2)} \quad (5)$$

Delay D is chosen between 1.5 and 2.5 for maximal flatness at dc [9]. Since the second-order Thiran allpass filter has a flatter

group delay than the first-order filter at low frequencies, Eq. (5) is approximated over a wider range of fundamental frequencies at the expense of more computation.

The waveform of the impulse train using the first-order Thiran allpass filter is presented in Figure 2. Its initial shape is recognizable as an impulse train although it slightly spreads over time. After one second, however, it becomes completely dispersed due to the frequency-dependent phase delay of the allpass filter. This dispersion results in inharmonic overtones in the frequency domain as shown in Figure 3. Depending on the fractional delay or the fundamental frequency, the locations of inharmonic partials vary around harmonic overtones.

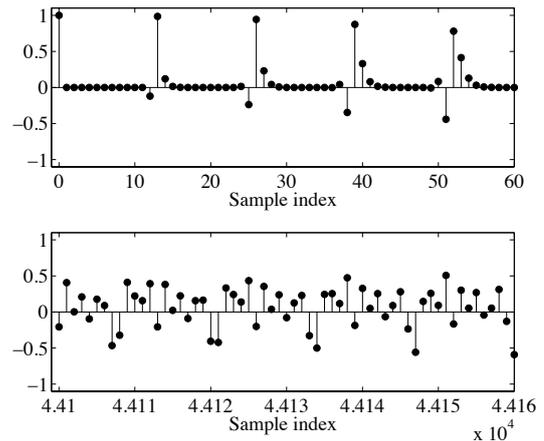


Figure 2: Impulse train generated by the feedback delay loop: (top) from zero second and (bottom) after one second. The fundamental frequency is 3322 Hz (MIDI note #104) and the sampling rate is 44.1 kHz.

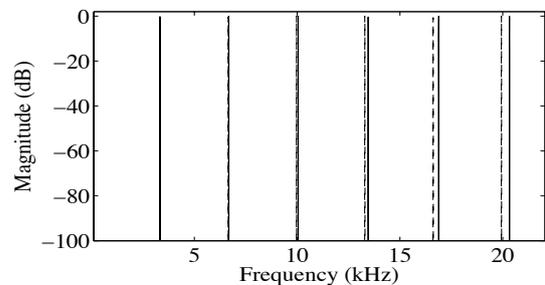


Figure 3: The solid lines are the magnitude response of the impulse train in Figure 2. The dotted lines are harmonic partials generated from the sum of sine functions. The spectrum was computed with a 32,768-point FFT using a Chebyshev window with a 120-dB side-lobe attenuation.

The deviation from the harmonic partials can be represented by inharmonicity coefficient B . For non-zero B , the inharmonic partials are calculated as

$$f_k = k f_0 \sqrt{1 + B k^2} \quad (6)$$

where k is the partial number, f_0 is the nominal fundamental frequency for the ideal non-dispersive case, and f_k is the partial frequency. Several methods are known to estimate the inharmonicity coefficient. We used a fast estimate algorithm proposed by Rauhala et al. for the impulse trains using the first- and second-order Thiran allpass filters [11]. Figure 4 shows the results for 88-key piano notes. The inharmonic coefficients were estimated only when the inharmonic partials were greater than harmonic overtones because Eq. (6) is invalid otherwise. Perceptual testing to examine the audibility of inharmonicity for five notes of synthetic sounds was performed in [12]. The inharmonicity coefficients for the corresponding notes in Figure 4 were all below the threshold of audibility. However, in an informal listening test, which compared the oscillator output with the reference sound (by additive synthesis), a slight time-varying phase shift effect was detected at high fundamental frequencies.

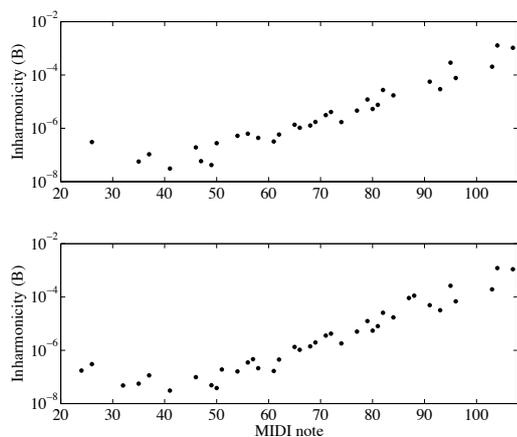


Figure 4: Inharmonicity coefficients (B) estimated for the impulse trains using the feedback delay loop: (top) the first-order and (bottom) second-order Thiran allpass filters. They were estimated only when inharmonic partials are greater than integer multiples of the fundamental.

Although this method does not generate an exactly periodic signal, it is seen to perceptually produce a harmonic spectrum. Above all, the impulse train has no aliasing and uses a small number of operations, two significant advantages over other algorithms.

2.2. Bipolar Impulse Trains

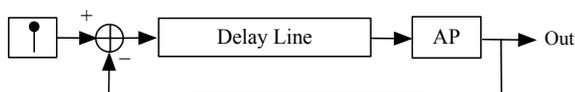


Figure 5: Diagram for generating a bipolar impulse train using a feedback delay loop. The only difference between this structure and Figure 1 is the minus sign.

The feedback delay loop in Figure 1 can be modified to generate a bipolar impulse train by including a negative sign in the loop as seen in Figure 5. The bipolar impulse train can be used to derive oscillators with only odd harmonics, such as square and triangu-

lar waveforms. Note that the length of the delay line and the fractional delay should be reduced by half for a given fundamental frequency.

The waveform of the bipolar impulse train using the feedback delay loop is shown in Figure 6. As the “unipolar” impulse train in Figure 2, the impulses are dispersed over time so that the shape of the waveform looks random after one second. The magnitude response of the bipolar impulse is presented in Figure 7. It contains only odd partials, but they are slightly deviated from harmonic overtones due to the frequency-dependent phase delay by the allpass filter.

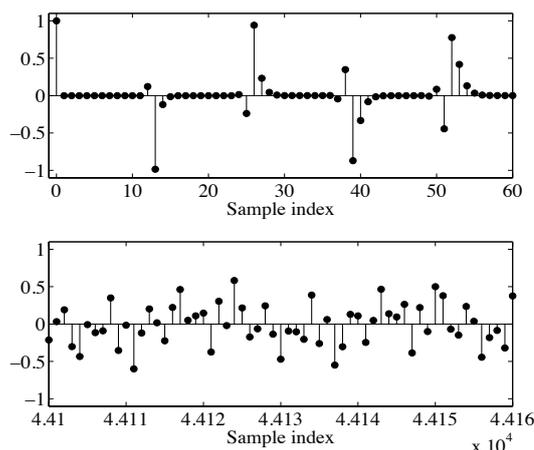


Figure 6: Bipolar impulse train generated by the feedback delay loop: (top) from zero second and (bottom) after one second. The fundamental frequency is 1661 Hz and the sampling rate is 44.1 kHz.

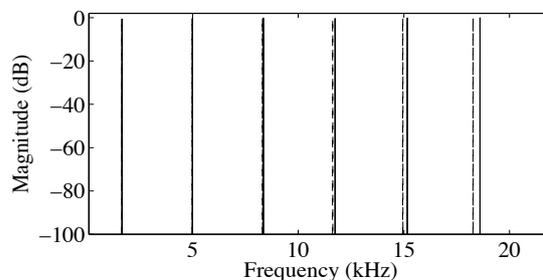


Figure 7: The solid lines are the magnitude response of the bipolar impulse train in Figure 2. The dotted lines are harmonic partials generated from the sum of sine functions.

3. TIME-VARYING PITCH SHIFTING

Time-varying pitch shifting is typically used for musical expressions such as vibrato and glissando. Since pitch is determined by the length of delay in the feedback delay loop scheme, time-varying pitch shifting needs to change the length of the delay line or the phase delay of the allpass filter on the fly. Pitch shifting is

usually implemented by moving the reading pointer to the delay line faster or slower than the writing pointer or by varying the allpass filter coefficient. Using a single reading pointer such as that shown in Figure 1 can introduce discontinuities into the loop when the length of the delay line or the allpass filter coefficient suddenly changes, resulting in undesirable artifacts such as audible clicks [13][14].

Solutions for the discontinuity have been proposed by several authors [13][14][15]. To avoid the abrupt change of the reading pointer, Jaffe and Smith suggested using two reading pointers corresponding to two different pitches and cross-fading them to make a smooth transition [13]. Van Duyne et al. improved on this method by minimizing the transient effect of the allpass filter [14]. They suggested setting the range of the allpass filter delay D between 0.618 and 1.618 to make the transient decay fast, and modifying the cross-fader to ramp up after 5 warm-up samples, corresponding to the time that the transient dies out (the range of the allpass filter delay D is a little greater than 0.418 to 1.418 to have maximally flat phase delay, but they both had similar results). Figure 8 illustrates time-varying pitch shifting based on the ideas. Note that the curve of the cross-fader in the dotted box remains at zero during the first 5 samples.

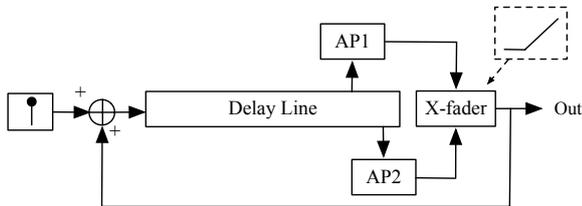


Figure 8: Diagram for time-varying pitch shifting in the impulse train using a feedback delay loop.

Another significant issue is that the feedback delay loop must conserve energy to be used as an oscillator. The cross-fader tends to cause some lowpass filtering, reducing the energy level at high frequencies. It is associated with the linearly weighted sum in the cross-fader, which attenuates the output level when the input signals are not in phase. Results show that the phase difference between signals from the two read pointers varies more in high frequencies. Although a different curve shape, such as a half of cosine cycle, can be used for the cross-fader, it is difficult to find an optimized curve that conserves the energy over the entire frequencies.

However, it turns out that the high-frequency attenuation can be alleviated by increasing the transition rate between two read pointers. Figure 9 shows the spectrograms of the impulse train played with vibrato during 10 seconds when the transition is updated every 16 (suggested by Van Duyne et al. in [14]) and 256 samples, respectively. At the 16-sample transition rate, high frequency energy slowly decays, whereas, at the 256-sample transition rate, it rapidly fades out particularly for harmonic partials above 5 kHz. In terms of the total energy per period, the level after 10 seconds decreased by -7.4 dB at the 16-sample transition rate, while it dropped to -26dB at the 256-sample transition rate. Note that the fast transition rate between the read pointers also minimizes the transient effect of the allpass filter by updating the coefficient more frequently although it increases computation.

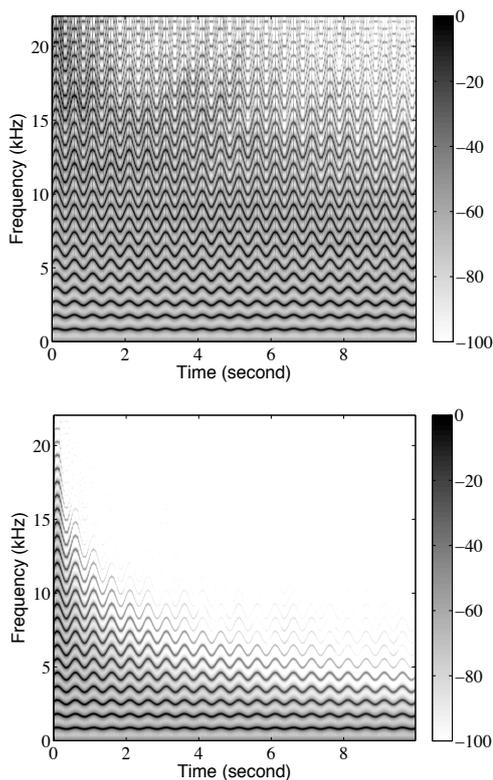


Figure 9: Spectrograms of impulse trains played with vibrato. The transition rate between two read pointers in the feedback delay loop is set to 16 (top) and 256 (bottom) samples.

4. GENERATING CLASSIC WAVEFORMS

4.1. Sawtooth Waveform

The sawtooth waveform has harmonics at all integer multiples of the fundamental frequency with a -6 dB per octave roll-off. Therefore, the sawtooth waveform can be derived by integrating the impulse train [4]. In practice, this integration is performed with a leaky integrator to avoid an accumulation of numerical errors [4] and is expressed as follows:

$$y(n) = x(n) + (1 - \epsilon)y(n - 1) \quad (7)$$

The leak rate ϵ is typically set to a small value (e.g. 0.005). Before integrating the impulse train, the dc component must be removed to prevent the waveform from drifting. Since the sum of the allpass filter output is equal to that of the input, the dc offset of the impulse train, which is given as the average of the output over a period, can be approximated as f_0/f_s , where f_0 is the fundamental frequency and f_s the sampling rate. Therefore, the resulting dc offset can be cancelled by subtracting the pre-computed dc offset. Another method to remove the dc offset is to use a dc blocker, which is especially necessary for time-varying pitch shifting where the dc level varies over time.

Figure 10 shows the sawtooth waveform and its spectrum generated by the feedback delay loop. Note that the waveform differs from the typical sawtooth shape due to the frequency-dependent phase delay of the allpass filter. Although partials of the spectrum slightly deviate from the harmonic locations, the deviation is nearly imperceptible.

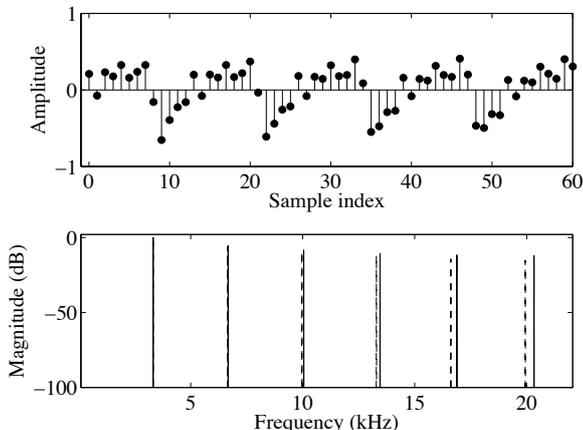


Figure 10: Sawtooth waveform generated by the feedback delay loop: (top) waveform after one second (bottom) its spectrum. The dotted lines in the bottom are harmonic partials generated from the sum of sine functions. The fundamental frequency is 3322 Hz (MIDI note #104) and the sampling rate is 44.1 kHz.

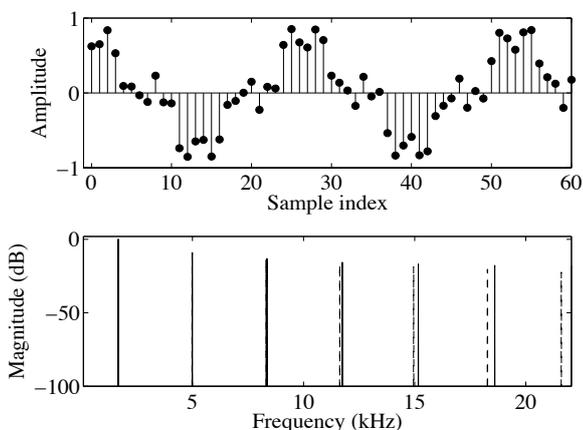


Figure 11: Square waveform generated by the feedback delay loop: (top) waveform after one second and (bottom) its spectrum. The fundamental frequency is 1661 Hz.

4.2. Square and Triangular Waveforms

The square waveform has harmonics at odd multiples of the fundamental frequency with a -6 dB per octave roll-off. Thus, the square waveform can be derived by integrating the bipolar impulse train [4]. Since the dc level of the bipolar impulse train is zero, dc removal is not necessary. Figure 11 shows the square waveform and its spectrum generated by integrating the bipolar impulse train. As the sawtooth waveform above, the square

waveform is different from the typical square shape, and partials of the spectrum are not exactly on harmonic locations. However, the output sounds nearly the same as the one generated by the additive synthesis.

The triangular waveform contains odd harmonics like the square waveform, but with a -12 dB per octave roll-off because it has discontinuities in its first derivative. Therefore, the triangular waveform can be generated by integrating the square waveform [4]. The additional integration, however, scales the amplitude according to the fundamental frequency. This can be corrected by multiplying by a scale factor $4f_0/f_s$, which is obtained by setting the integration over a quarter period of a square waveform equal to the peak value of a triangular waveform.

5. RELATION TO THE SECOND-ORDER DIGITAL WAVEGUIDE OSCILLATOR

Smith and Cook introduced a sinusoid oscillator algorithm using the second-order waveguide filter [16]. This filter is composed of two unit delay elements combined with a scattering junction where one terminates with an inverting reflection and the other with a non-inverting reflection as shown in Figure 12. The coefficient k is given as $\cos(\omega_0)$ for the fundamental frequency ω_0 .

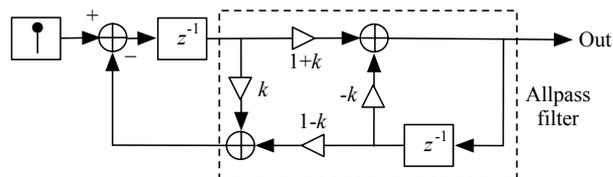


Figure 12: Second-order digital waveguide oscillator

The scattering junction and a delay unit in the dotted box can be seen as a first-order allpass filter when its output is fed back through the delay element on the left side. As a result, the diagram in Figure 12 is equivalent to that in Figure 5 when the length of the delay line is one. In fact, the bipolar impulse train generates a single sinusoid at high fundamental frequencies (greater than $f_s/6$).

The coefficient of the allpass filter in Figure 5 is also related to the tuning coefficient in the digital waveguide oscillator. When the length of the delay line is one, the total delay $1+D$ and the fundamental frequency ω_0 have the following relation:

$$1 + D = \frac{f_s}{2f_0} = \frac{\pi}{\omega_0} \tag{8}$$

By plugging Eq. (8) into Eq. (3), we obtain

$$\begin{aligned} a_1 &= \frac{\sin((1-D)\omega_0/2)}{\sin((1+D)\omega_0/2)} = \frac{\sin((\omega_0 - (1+D)\omega_0)/2)}{\sin((1+D)\omega_0/2)} \\ &= \sin(\omega_0 - \frac{\pi}{2}) = -\cos(\omega_0) \end{aligned} \tag{9}$$

Since Eq. (3) is originally given for the unipolar impulse train, the minus sign in Eq. (9) is cancelled out by the negative sign in the feedback delay loop. Thus, the result is the same as the coefficient k in Figure 12. In consequence, the second-order digital

waveguide oscillator can be interpreted as a special case of the bipolar impulse train.

6. NUMERICAL TRICKS TO AVOID DIVISIONS

The coefficient computation in the Thiran allpass filters includes a division, generally an expensive operation. For the sake of computational efficiency, the division can be replaced by multiplications. In the first-order allpass filter, Eq. (2) can be approximated as shown below,

$$a_1 = \frac{1-D}{1+D} = \frac{v}{1-v} = v(1+v)(1+v^2)(1+v^4)\dots \quad (10)$$

where $v = \frac{1-D}{2}$

Since v is much less than one ($-0.319 \leq v \leq 0.191$) and factors in Eq. (10) have a form of $1+v^{\text{power of } 2}$, Eq. (10) converges fast. For example, the first three factors amount to about 99% and the first four factors to 99.9% of the coefficient. In the second-order case, Eq. (5) also can be replaced with multiplications with following approximations:

$$a_1 = -2 \frac{D-2}{D+1} = -2 \frac{0.25-v_1}{1-v_1} \quad (11)$$

$$= -2(0.25-v_1)(1+v_1)(1+v_1^2)(1+v_1^4)\dots$$

$$a_2 = -\frac{a_1}{2} \frac{D-1}{D+2} = -\frac{a_1}{2} \frac{0.25-v_2}{1-v_2} \quad (12)$$

$$= -\frac{a_1}{2} (0.25-v_2)(1+v_2)(1+v_2^2)(1+v_2^4)\dots$$

where $v_1 = \frac{3-D}{4}$ $v_2 = \frac{2-D}{4}$

Since v_1 is between 0.125 and 0.375, and v_2 is between -0.125 and 0.125, Eq. (11) and (12) converge fast as well, so that the coefficients can be well approximated by the first three or four factors without divisions.

7. CONCLUSIONS

A new approach for generating virtual analog oscillators with no aliasing was proposed. The method produces impulse trains using a feedback delay loop composed of a delay line and an allpass filter, and derives classic waveforms with a leaky integrator. The waveforms generated by this method are not exactly periodic, but perceptually sound harmonic. A technique for time-varying pitch shifting was also reviewed. Although high-frequency energy is attenuated when pitch changes, the attenuation can be adjusted by fast transition between two read pointers. In addition, the relationship between the proposed method and the second-order waveguide oscillator was examined.

Future work includes minimizing the inharmonicity and improving time-varying pitch shifting to avoid the high-frequency attenuation. Another work is to generate oscillators other than classic waveforms, such as by initializing the delay line with various shapes of signals.

8. ACKNOWLEDGEMENT

This research was conducted in 2008-2009, when Prof. Vesa Välimäki was a visiting scholar at Stanford University. His visit was financed by the Academy of Finland (project no. 126310)

9. REFERENCES

- [1] V. Välimäki and A. Huovilainen, "Antialiasing oscillators in subtractive synthesis," *IEEE Signal Processing Mag.*, vol. 24, no. 2, pp. 116-125, Mar. 2007.
- [2] G. Winham and K. Steiglitz, "Input generators for digital sound synthesis," *J. Acoust. Soc. Amer.*, vol. 47, part 2, pp. 665-666, Feb. 1970.
- [3] J. A. Moorer, "The Synthesis of Complex Audio Spectra by Means of Discrete Summation Formulae", *Journal of the Audio Engineering Society*, vol. 24, pp. 717-727, Dec. 1975.
- [4] 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.
- [5] E. Brandt, "Hard sync without aliasing," in *Proc. Int. Computer Music Conf.*, Havana, Cuba, pp. 365-368, 2001.
- [6] H. Chamberlin, *Musical Applications of Microprocessors*, 2nd ed. New York: Hayden, 1985.
- [7] V. Välimäki, "Discrete-time synthesis of the sawtooth waveform with reduced aliasing," *IEEE Signal Processing Letters*, vol. 12, no. 3, pp. 214-217, Mar. 2005.
- [8] J. O. Smith, "Physical modeling using digital waveguides," *Computer Music Journal*, vol. 16, no. 4, pp. 75-87, 1992.
- [9] V. Välimäki, *Discrete-time modeling of acoustic tubes using fractional delay filters*, Ph.D. thesis in Helsinki University of Technology, Espoo, Finland, Dec. 1995. [Online]. Available: http://www.acoustics.hut.fi/~vpv/publications/vesa_ph_d.html [Accessed: Mar. 30. 2009]
- [10] D. A. Jaffe and J. O. Smith, "Extensions of the Karplus-Strong plucked string algorithm," *Computer Music Journal*, vol. 7, no. 2, pp. 56-69, 1983.
- [11] J. Rauhala, H.-M. Lehtonen, and V. Välimäki, "Fast automatic inharmonicity estimation algorithm," *J. Acous. Soc. Am. Express Letters*, vol. 121, no. 5, pp. EL184-EL189, 2007.
- [12] H. Järveläinen, V. Välimäki, M. Karjalainen, "Audibility of the timbral effects of inharmonicity in stringed instrument tones," *Acoustics Research Letters Online*, vol. 2, issue 3, pp. 79-84, Jul. 2001
- [13] D. A. Jaffe and J. O. Smith, "Performance expression in commuted waveguide synthesis of bowed strings," in *Proc. Int. Computer Music Conf.*, Banff, Canada, pp. 343-346, 1995.
- [14] S. A. Van Duyne, D. Jaffe, G. P. Scandalis, and T. Stilson, "A lossless, click-free, pitchbend-able, delay line loop interpolation scheme," in *Proc. Intl. Computer Music Conf.*, Thessaloniki, Greece, pp. 252-255, 1997.
- [15] V. Välimäki, T. I. Laakso, and J. Mackenzie, "Elimination of transients in time-varying allpass fractional delay filters with application to digital waveguide modeling," in *Proc. Int. Computer Music Conf.*, Banff, Canada, pp. 327-334, 1995.
- [16] J. Smith and P. Cook, "The second order digital waveguide oscillator," in *Proc. Int. Computer Music Conf.*, San Jose, USA, pp.150-153, Oct. 1992.