VIRTUAL ANALOG BUCHLA 259 WAVEFOLDER

Fabían Esqueda, Henri Pöntynen, Vesa Väläläki
Acoustics Lab, Dept. of Signal Processing and Acoustics
Aalto University
Espoo, Finland
firstname.lastname@aalto.fi

Julian D. Parker
Native Instruments GmbH
Berlin, Germany
julian.parker@native-instruments.com

ABSTRACT

An antialiased digital model of the wavefolding circuit inside the Buchla 259 Complex Waveform Generator is presented. Wavefolding is a type of nonlinear waveshaping used to generate complex harmonically-rich sounds from simple periodic waveforms. Unlike other analog wavefolder designs, Buchla’s design features five op-amp-based folding stages arranged in parallel alongside a direct signal path. The nonlinear behavior of the system is accurately modeled in the digital domain using memoryless mappings of the input–output voltage relationships inside the circuit. We pay special attention to suppressing the aliasing introduced by the nonlinear frequency-expanding behavior of the wavefolder. For this, we propose using the bandlimited ramp (BLAMP) method with eight times oversampling. Results obtained are validated against SPICE simulations and a highly oversampled digital model. The proposed virtual analog wavefolder retains the salient features of the original circuit and is applicable to digital sound synthesis.

1. INTRODUCTION

To talk about Don Buchla is to talk about the history of the analog synthesizer. Motivated by his early experiments with musique concrète, California native Donald “Don” Buchla was drawn to the San Francisco Tape Music Center in 1963, where he began collaborating with composers Morton Subotnick and Ramon Sender (1). Subotnick and Sender commissioned Buchla to design a voltage-controlled musical instrument that could manipulate the characteristics of sounds generated by function generators. This led to the development of Buchla’s first synthesizer, the Buchla 100 (12), completed in 1964.

From the beginning, Buchla’s approach to sound synthesis was fundamentally different to that of his contemporaries, particularly Robert Moog. In Moog synthesizers, sounds are sculpted by filtering harmonically-rich waveforms with resonant filters. This method is known in the literature as “subtractive” synthesis and is commonly dubbed “East Coast” synthesis as a reference to Moog’s New York origins. In contrast, Buchla’s synthesis paradigm (known as “West Coast” synthesis) concentrates on timbre manipulation at oscillator level via nonlinear waveshaping, frequency modulation or phase locking. A trademark module in Buchla synthesizers is the lowpass gate, a filter/amplifier circuit capable of producing acoustic-like plucked sounds by using photoresistive opto-isolators, or “vactrols”, in its control path (3). Buchla’s designs played a key role in the development of electronic music and can be heard across numerous recordings, such as in the works of the renowned composer Suzanne Ciani (4).

Recent years have seen a resurgence of interest in analog synthesizers, with music technology powerhouses such as Moog and Korg re-releasing modern versions of their now classic designs. Similarly, contemporary manufacturers of modular synthesizers like Make Noise. Sputnik Modular and Verbos Electronics, to name a few, have reinterpreted Buchla’s designs, rekindling the interest in analog West Coast synthesis. This rise in popularity serves as the motivation to study classic analog devices and to develop virtual analog (VA) models which can be used within digital audio environments. VA instruments are generally more affordable than their analog counterparts, and are exempt from issues such as electrical faults and component aging (5).

In this work we present a novel VA model of the timbre circuit inside the seminal Buchla 259, a complex waveform generator released in 1970 as part of the Buchla 200 synthesizer. The 259 is a dual oscillator module with frequency modulation and waveform synchronization capabilities that provide a wide timbral palette. However, its most distinctive feature is its wavefolding circuit capable of producing the rich harmonic sweeps characteristic to West Coast synthesis. Wavefolding is a type of nonlinear waveshaping in which parts of the input signal that exceed a certain value are inverted or “folded back”. This process introduces high levels of harmonic distortion and thus alters the timbre of the signal.

The use of nonlinear distortion to generate complex sounds has been widely studied within the context of digital synthesis. Well-known methods include the use of nonlinear waveshaping functions, such as Chebyshev polynomials, to expand the spectrum of simple sinusoids (9, 9), and frequency modulation (FM) synthesis (10). Other methods include modified FM synthesis (11), bitwise logical modulation and vector phaseshaping synthesis (12, 13). Previous research on VA modeling of nonlinear analog audio systems has covered a wide spectrum of topics, including Moog’s ladder filter (14–18), other nonlinear filters (19–21), distortion circuits (22–26) and effects units (27–29).

One of the major challenges in VA modeling is to minimize the effects of aliasing distortion. Aliased components are known to be perceptually disturbing and unpleasant, but become negligible if attenuated sufficiently (30, 31). The brute force method to reduce aliasing is oversampling, but, if the nonlinearity introduces high levels of distortion, the sample rate may have to be very high to obtain good audio quality. Aliasing suppression techniques have been thoroughly studied in the field of digital audio synthesis (32–35) and, more recently, in nonlinear audio processing (36, 39). In this work we propose the use of the previously introduced bandlimited ramp (BLAMP) method (36, 37) which can be used to bandlimit the corners, or edges, introduced by the wavefolding operation. The BLAMP method significantly reduces the oversampling requirements of the system.

This paper is organized as follows. Section 2 details the analysis of the circuit. Section 3 deals with its implementation in the digital domain with emphasis on aliasing suppression. Finally, results and concluding remarks are presented in Sections 4 and 5, respectively.
2. CIRCUIT ANALYSIS

Figure 1 shows a simplified schematic of the Buchla 259 timbre circuit. This figure has been adapted from Andre Theelen’s DIY circuit. This figure has been adapted from Andre Theelen’s DIY circuit. The main difference between Fig. 1 and Buchla’s original design is the omission of the “Symmetry” and “Order” controls, which are not considered in this study. The following treatment of the circuit adheres, for the most part, to the analysis presented by Prof. Aaron Lanterman as part of his lecture series “Electronics for Music Synthesis” [40].

The wavefolder inside the Buchla 259 consists of five non-identical op-amp-based folding cells arranged in parallel along side a direct signal path, as shown in Fig. 1. The two op-amps on the right-hand side of the schematic are set up as summing amplifiers and are used to combine the outputs of all six branches. Overall, this parallel topology differs from that of the more common transistor/diode-based foldfiers, where multiple folding stages are usually cascaded together, e.g. as in the middle section of the Serge Wave Multiplier [41]. The Intellijel μFold II and Topobrillo Triple Wavefolder [42] are examples of commercially-available designs built around a series topology.

To simplify the analysis of the circuit, we first derive the input–output voltage relationship of a single folding cell. Since the parallel paths share the same structure, this result can be applied to all folding branches. Component values for the circuit are given in Table 1. Indices have been used to indicate branch number, e.g. \( R_{3,2} \) denotes resistor \( R_2 \) in the fifth branch.

2.1. Single Folding Cell

Figure 2 shows the schematic for an op-amp circuit that is in the context of this work referred to as a folding cell. The variable \( V_{in} \) represents the voltage appearing at the input of all six branches. In the Buchla 259 the input of the timbre circuit is wired internally to the output of a sinusoidal oscillator. We denote the output voltage of each folding branch by \( V_k \), where \( k \) is the branch number as counted from top to bottom. The voltage \( V_o \) denotes the voltage at the output terminal of the op-amp. Since \( R_3 \) is connected to the virtual ground node formed at the inverting input terminal of the succeeding summing amplifier (see Fig. 1), we assume loading effects between the branches to be minimal, and thus treat each folding cell individually.

First, we assume ideal op-amp behavior and apply Kirchhoff’s voltage law (KVL). This results in the current–voltage relationships

\[
V_{in} = R_1 I_1 + V_i \quad \text{and} \quad V_o = V_i - R_2 I_2, \tag{1}
\]

where

\[
V_i = R_3 I_3. \tag{2}
\]

Rearranging these equations in terms of currents then gives us

\[
I_1 = \frac{V_{in} - V_i}{R_1}, \quad I_2 = \frac{V_k - V_o}{R_2} \quad \text{and} \quad I_3 = \frac{V_k}{R_3}. \tag{3}
\]

Next, we apply Kirchhoff’s current law (KCL) at node \( V_k \) to establish the current relation

\[
I_1 + I_2 = I_3. \tag{4}
\]

Plugging (3) into (4) results in the expression

\[
\frac{V_{in} - V_i}{R_1} = \frac{V_k - V_o}{R_2} + \frac{V_k}{R_3}. \tag{5}
\]

Table 1: Component values for the Buchla 259 circuit in Fig. 1

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Name</th>
<th>Value</th>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( R_{1,1} )</td>
<td>10 kΩ</td>
<td>( R_{1,2} )</td>
<td>100 kΩ</td>
<td>( R_{1,3} )</td>
<td>100 kΩ</td>
</tr>
<tr>
<td>( R_{2,1} )</td>
<td>49.9 kΩ</td>
<td>( R_{2,2} )</td>
<td>100 kΩ</td>
<td>( R_{2,3} )</td>
<td>43.2 kΩ</td>
</tr>
<tr>
<td>( R_{3,1} )</td>
<td>91 kΩ</td>
<td>( R_{3,2} )</td>
<td>100 kΩ</td>
<td>( R_{3,3} )</td>
<td>56 kΩ</td>
</tr>
<tr>
<td>( R_{4,1} )</td>
<td>30 kΩ</td>
<td>( R_{4,2} )</td>
<td>100 kΩ</td>
<td>( R_{4,3} )</td>
<td>68 kΩ</td>
</tr>
<tr>
<td>( R_{5,1} )</td>
<td>68 kΩ</td>
<td>( R_{5,2} )</td>
<td>100 kΩ</td>
<td>( R_{5,3} )</td>
<td>33 kΩ</td>
</tr>
<tr>
<td>( R_7 )</td>
<td>24.9 kΩ</td>
<td>( R_{6,1} )</td>
<td>24.9 kΩ</td>
<td>( R_{7} )</td>
<td>1.2 MΩ</td>
</tr>
<tr>
<td>( - )</td>
<td>( - )</td>
<td>( C )</td>
<td>100 pF</td>
<td>( - )</td>
<td>( - )</td>
</tr>
</tbody>
</table>
which we can solve for $V_k$ as:

$$V_k = -\frac{R_3}{R_1} \left( R_2 V_m + R_1 V_o \right). \quad (6)$$

Now, since the op-amp is in the inverting configuration, the value of $V_o$ is defined as

$$V_o = -\frac{R_2}{R_1} V_m. \quad (7)$$

This definition implies that the op-amp can provide a fixed gain of $-\frac{R_2}{R_1}$ for all values of $V_o$. If we were to substitute (7) into (6) we would find that $V_k = 0$, as required by ideal op-amp behavior (i.e., the op-amp maintains the input terminals at the same potential) \[41\]. In practice, however, the value of $V_o$ is limited by the supply voltages and the device is unable to maintain $V_k$ at ground potential when the input voltage is high. Note that the op-amps in the folding branches are connected to lower supply voltages than the rest of the circuit.

Buchla’s original design utilized CA3160 op-amps in its folding cells. This particular “rail-to-rail” op-amp features a CMOS output stage and is capable of swinging the output up to the supply voltages. As illustrated in its datasheet \[42\], the CA3160 exhibits a sharp saturating behavior similar to hard clipping. Therefore, we rewrite (7) as

$$V_o = \begin{cases} \frac{R_2}{R_1} V_m, & \text{if } |V_m| \leq \frac{R_1}{R_2} V_o \\ -\text{sgn}(V_m) V_o, & \text{otherwise} \end{cases} \quad (8)$$

where $V_o = 6$ V is the supply voltage of the op-amp and sgn() is the signum function.

By combining (6) and (8), we can derive a piecewise expression for the output of each folding branch in the original circuit:

$$V_k = \begin{cases} R_{k,3} \left( R_{k,2} V_m - \text{sgn}(V_m) R_{k,1} V_o \right), & \text{if } |V_m| > \frac{R_{k,1}}{R_{k,2}} V_o \\ R_{k,1} R_{k,3} + R_{k,2} R_{k,3} + R_{k,1} R_{k,2}, & \text{otherwise} \end{cases} \quad (9)$$

Figures 3(a)–(e) show the value of $V_{1-5}$ for values of $V_m$ between −10 V and 10 V measured at 1 mV steps using SPICE. Since no publicly available SPICE model for the CA3160 seems to exist, LTC6088 was used in the simulations instead. This device is similar to the CA3160 in that it also features a “rail-to-rail”-capable CMOS output stage \[43\]. These plots show that the output of each folding cell has a “deadband” in the input voltage region where the op-amp displays ideal behavior and maintains $V_k$ at ground potential. At larger input voltage values, the op-amp output saturates to the supply voltage and is unable to maintain the deadband.

### 2.2. Mixing Stages

Following the folding cells, the output voltages of the six parallel branches are combined with two inverting amplifiers. Voltage $V_7$, the output of the lower amplifier (cf. Fig. 1), is formed as the weighted sum of the voltages from the three lower branches

$$V_7 = -R_{1,3} \left( \frac{V_1}{R_{1,3}} + \frac{V_2}{R_{2,3}} + \frac{V_3}{R_{3,3}} \right). \quad (10)$$

This voltage is subsequently fed to the input of the upper amplifier along with voltages $V_{1-3}$. The upper amplifier is an active first-order integrator that lowpass filters the weighted combination of the input signals. Assuming that the op-amp is operating within its linear region, the summing and filtering operations commute. Therefore, we can simplify the analysis by representing this stage as an inverting amplifier cascaded with a first-order lowpass filter. By replacing capacitor $C$ with an open circuit we can then derive an expression for $V_{out}'$, the output of the circuit before filtering:

$$V_{out}' = -R_{2,3} \left( \frac{V_1}{R_{1,3}} + \frac{V_2}{R_{2,3}} + \frac{V_3}{R_{3,3}} + \frac{V_7}{R_7} \right). \quad (11)$$

Figure 4(a) shows a SPICE simulation of the input–output voltage relation of the entire circuit when the output filter is bypassed. It can be seen that the weighted sum of the individual branches (cf. Fig. 3) implements a piecewise linear waveshaping function. Figure 4(b) illustrates the outcome of driving the circuit with a sinusoidal signal. A fundamental frequency of 100 Hz and a peak voltage of 5 V were used in this simulation. The output of the circuit exhibits high levels of harmonic distortion which dramatically alters its timbral characteristics. In general, the output signal is perceived as harsher than the original input signal. Significant timbral variation can be achieved by simply modulating the amplitude of the input sinusoid. The filtering effect of the upper summing amplifier is discussed in Section 3.1.

### 3. DIGITAL IMPLEMENTATION

With the exception of the filtering stage at the output, the Buchla 259 timbre circuit can be categorized as a static system. This...
means that we can derive a digital model using discrete memoryless mappings of the voltage relationships derived in the previous section. First, we define our discrete-time sinusoidal input as

\[ V_{in}[n] = A \sin(2\pi f_0 n T), \quad (12) \]

where \( A \) is the peak amplitude, \( f_0 \) is the fundamental frequency and \( T \) is the sampling period, i.e. \( T = 1/f_s \).

From (12) we can then define explicit discrete-time expressions for the output of each folding branch. To facilitate their implementation, terms containing resistor values have been evaluated and replaced for their corresponding approximate scalar values:

\[ V_{1}[n] = \begin{cases} 0.8333 V_{in}[n] - 0.5000 s[n] & |V_{in}[n]| > 0.6000 \\ 0, & \text{otherwise} \end{cases}, \quad (13) \]

\[ V_{2}[n] = \begin{cases} 0.3768 V_{in}[n] - 1.1281 s[n] & |V_{in}[n]| > 2.9940 \\ 0, & \text{otherwise} \end{cases}, \quad (14) \]

\[ V_{3}[n] = \begin{cases} 0.2829 V_{in}[n] - 1.5446 s[n] & |V_{in}[n]| > 5.4600 \\ 0, & \text{otherwise} \end{cases}, \quad (15) \]

\[ V_{4}[n] = \begin{cases} 0.5743 V_{in}[n] - 1.0338 s[n] & |V_{in}[n]| > 1.8000 \\ 0, & \text{otherwise} \end{cases}, \quad (16) \]

\[ V_{5}[n] = \begin{cases} 0.2673 V_{in}[n] - 1.0907 s[n] & |V_{in}[n]| > 4.0800 \\ 0, & \text{otherwise} \end{cases}, \quad (17) \]

where \( s[n] = \text{sgn}(V_{in}[n]). \) From these branches we can then define a global summation stage:

\[ V'_{ou}[n] = -12.000V_{1}[n] - 27.777V_{2}[n] - 21.428V_{3}[n] 
+ 17.647V_{4}[n] + 36.363V_{5}[n] + 5.000V_{in}[n]. \quad (18) \]

Figures 2 and 3 show the input–output relation of these mappings against the previously presented SPICE simulations. These results show a good match between the original and modeled behavior, with an absolute error in the range of \( 10^{-5} \) V.

### 3.1. Filtering Stage

The filter at the output of the system is a one-pole lowpass filter. In the Laplace domain, the transfer function of this filter is given by

\[ H(s) = \frac{w_c}{s + w_c}, \quad (19) \]

where \( w_c = 2\pi f_c \) and \( f_c \) represent the cutoff frequency in radians and Hz, respectively \([44, 45]\). From Fig. 1, the cutoff of the filter is derived as

\[ f_c = \frac{1}{2\pi R_C C} \approx 1.33 \text{kHz}. \quad (20) \]

This relatively low cutoff frequency indicates the purpose of the filter is simply to act as a fixed tone control, attenuating the perceived brightness of the output by introducing a gentle 6-dB/octave roll-off. Equation (19) can be discretized using the bilinear transform, which results in the z-domain transfer function

\[ H(z) = \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1}}. \quad (21) \]

where

\[ b_0 = b_1 = \frac{w_c T}{2 + w_c T} \quad \text{and} \quad a_1 = \frac{w_c T - 2}{w_c T + 2}. \]

Due to the low cutoff parameter, the warping effects of the bilinear transform can be neglected. This transfer function can be implemented digitally, e.g. using Direct Form II Transposed \([44]\).

### 3.2. Antialiasing

Given the highly nonlinear nature of wavefolding, audio-rate implementations of the proposed model using \([13, 14, 15]\) will suffer from excessive aliasing distortion. This problem can be attributed to the corners or edges introduced by the folding cells of the system (cf. Fig. 4). These corners indicate that the first derivative of the signal is discontinuous and, as such, has infinite frequency content. In the discrete-time domain, frequency components that exceed the Nyquist limit will be reflected into the audio band as aliases.

To ameliorate this condition we propose the use of the BLAMP method, which has previously been used in the context of ideal nonlinear operations such as signal clipping and rectification \([36, 37]\). This method consists of replacing the corners with bandlimited versions of themselves. It is an extension of the bandlimited impulse (BLIT) synthesis method \([32]\).

The BLAMP function is a closed-form expression that models a bandlimited discontinuity in the first derivative of a signal. It is derived from the second integral of the bandlimited impulse \([33, 35]\), or sinc, function and is defined as

\[ R_{BLAMP}(t) := t \left[ \frac{1}{2} + \frac{1}{\pi} \text{Si}(\pi f_c t) \right] + \frac{\cos(\pi f_c t)}{\pi^2 f_c}, \quad (22) \]

where \( t \) is time and \( \text{Si}(x) \) is the sine integral

\[ \text{Si}(x) := \int_0^x \frac{\sin(t)}{t} dt. \quad (23) \]

Computing the difference between the BLAMP and the ideal ramp function

\[ R(t) := \begin{cases} t, & \text{when } t \geq 0 \\ 0, & \text{when } t < 0 \end{cases} \quad (24) \]
produces the BLAMP residual function shown in Fig. 5(a). In the discrete-time domain, this function is used to reduce aliasing by superimposing it on every corner within the waveform and sampling it at neighboring sample points. A crucial step in this process is centering the residual around the exact point in time where each discontinuity occurs, which is usually between samples.

Due to the high computational costs of evaluating (22), we will use its two-point polynomial approximation (polyBLAMP) instead [36]. Figure 5(b) illustrates the time-domain waveform of the two-point polyBLAMP residual function evaluated using the expressions given in Table 2. In this context $d \in [0, 1)$ is the fractional delay required to center the residual function between two samples.

In the case of the Buchla 259 timbre circuit, the BLAMP method is applied independently within each folding branch. To facilitate its implementation, we define an intermediate processing step in which the input-output relationships of the folding cells [13]–[17] are rewritten as inverse clippers. We then denote the output of the $k$th inverse clipper as $V'_k$, which can be written as

$$V'_k[n] = \begin{cases} V_{\mu}[n], & \text{if } |V_{\mu}[n]| > \frac{R_{k,1}}{R_{k,2}} V_k \\ \text{sgn}(V_{\mu}[n]) \frac{R_{k,1}}{R_{k,2}} V_k, & \text{otherwise.} \end{cases} \tag{25}$$

Figure 6 shows the input–output relation of this intermediate processing stage. The advantage of this seemingly unnecessary step is that now we can apply the BLAMP method following the same approach described in [36] and [37] for the case of the regular hard clipper. This process involves detecting the transition from non-clipping to clipping samples (i.e. detecting the corners), computing the exact fractional clipping point and adding the correction function to the samples immediately before and after each corner. Prior to addition, the polyBLAMP function must be scaled by the slope of the input signal at the clipping point. Since we know the input to the system is a sinusoidal waveform, we can compute the fractional clipping points and their respective slopes analytically, thus facilitating the implementation and improving the robustness of the method.

![Figure 5: Time domain representation of (a) the central lobe of the BLAMP residual function and (b) its two-point polynomial approximation.](image)

![Figure 6: Input–output relationship of the proposed intermediate processing step, the inverse clipper.(25)](image)

For an arbitrary inverse clipper stage (25) driven by an $f_0$-Hz sinewave starting at zero phase, the first clipping point (in seconds) is given by

$$t_1 = \frac{\sin^{-1}(V_{\mu}/AR_k)}{2\pi f_0}. \tag{26}$$

From this value, we can evaluate the three remaining clipping points within the first period of the signal:

$$t_2 = \frac{1}{2f_0} - t_1, \quad t_3 = \frac{1}{2f_0} + t_1 \quad \text{and} \quad t_4 = \frac{1}{f_0} - t_1. \tag{27}$$

Figure 7(a) shows the result of inverse-clipping the first period of a sinewave, all four clipping points are highlighted. Subsequent clipping points can then be computed by adding multiples of the fundamental period, i.e. $1/f_0$.

For a stationary sinewave, the magnitude of the slope is the same at all clipping points. Therefore, we can define a closed-form expression of the polyBLAMP scaling factor as

$$\mu = \left| 2\pi f_0 A \cos(2\pi f_0 t_{1-4} / f_s) \right|. \tag{28}$$

Figure 7(b) illustrates the process of centering the polyBLAMP residual function at each clipping point, scaling it and sampling it at neighboring samples. The polarity must be adjusted according to the polarity of the signal at the clipping point. Although in this study we only consider the case of sinusoidal inputs, the same approach can be adapted when other periodic signals are used as input to the wavefolder, e.g. sawtooth and triangular waveforms.

Now, if we then define $V'_k$ as the signal that results from applying the polyBLAMP method to $V_k$, we can write an expression for $V'_k$, the anti-aliased output of each folding cell:

$$V'_k[n] = \frac{R_{k,2} R_{k,3}}{R_{k,1} R_{k,3} + R_{k,2} R_{k,3} + R_{k,1} R_{k,2}} \left[ V_{\mu}[n] \right. - \text{sgn}\left( V_{\mu}[n] \right) \left\{ \frac{R_{k,1} R_{k,3}}{R_{k,2}} V_k[n] \right\} \tag{29}\right].$$

This step basically undoes the intermediate processing step (25) while preserving the anti-aliased behavior.

![Table 2: Two-point polyBLAMP function and its residual](image)

<table>
<thead>
<tr>
<th>Span</th>
<th>Two-point polyBLAMP</th>
<th>d ∈ [0, 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$[-T, 0]$</td>
<td>$-d^2/6 + d^2/2 + d/2 + 1/6$</td>
<td>$d \in [0, 1)$</td>
</tr>
<tr>
<td>$[0, T]$</td>
<td>$d^2/6$</td>
<td>$d \in [0, 1)$</td>
</tr>
<tr>
<td>$[0, T]$</td>
<td>$-d^2/6 + d^2/2 - d/2 + 1/6$</td>
<td>$d \in [0, 1)$</td>
</tr>
</tbody>
</table>
The delay is given for each corner.

The polyBLAMP residual at each clipping point. The fractional

including the output filter, is given in Fig. 9. Boxes labeled

by the inverse clipping stage (25).

Figure 7: (a) Time-domain representation of a sine wave processed
by the inverse clipping stage (25) and (b) the process of centering
the polyBLAMP residual at each clipping point. The fractional
delay is given for each corner.

A complete block diagram of the proposed wavefolder model,
including the output filter, is given in Fig. 9. Boxes labeled $w_{1-5}(t)$
consist of the inverse clipper (25) followed by polyBLAMP cor-
rection and the mapping function (29). Once again, a tilde has
been used to distinguish $V_{\text{out}}$, the output of the system with alias-
ing suppression, from $V_{\text{out}}$, its trivial counterpart.

4. RESULTS

Having compared the time-domain characteristics of the proposed
model against SPICE simulations (cf. Figs. 7 and 8), in this section
we move on to observe and evaluate its frequency-domain behav-
ior. The spectrogram in Fig. 8 shows the effect of sweeping the
input gain $A$ from 0 to 10 for a sine wave with fundamental fre-
quency $f_0 = 100$ Hz. Compared to typical saturating waveshapers
(e.g. the tanh function or the hard clipper), where the level of intro-
duced harmonics is directly proportional to input gain, wavefolding
generates complex harmonic patterns reminiscent of FM syn-
thesis. From a perceptual point of view, the folded waveform can be
described as being brighter and more abrasive than the original
input signal. It should be pointed out that due to the odd symmetry
of the wavefolding operation (cf. Fig. 4(a)), the system introduces
odd harmonics only.

Next, we analyze the effect of wavefolding on a static 890-Hz
input sine wave with amplitude $A = 5$. Figures 10(a)–(b) show
the waveform and magnitude spectrum, respectively, of the sys-

tem’s output when implemented at audio rate (i.e. $f_s = 44.1$ kHz)
and without polyBLAMP correction. The resulting signal is prac-
tically unusable, as it exhibits very high levels of audible alias-
ing distortion. In comparison, Figs. 10(c)–(d) show the outcome
of operating at the same rate but employing the two-point poly-
BLAMP method. As expected, the overall level of aliasing has
been considerably attenuated. Next, Figs. 10(e)–(f) show the out-
put of the system for a sample rate $f_s = 2.82$ MHz, i.e. 64 times
the previous rate. This example was generated by synthesizing the
input sine wave at the target rate and plotting only those frequency
components below 20 kHz. The output is virtually free from alias-
ing, with only a handful of components laying above the −100 dB
line. Lastly, Figs. 10(g)–(h) show the outcome of using eight times
oversampling and the proposed polyBLAMP method. Results ob-
tained are comparable to those in Fig. 10(f), which indicates that
the proposed method reduces the oversampling requirements of the
system.

The overall increase in signal quality provided by the two-
point polyBLAMP method was measured for a larger set of input
signals. Figure 11 shows the measured signal-to-noise ratio (SNR)
at the output of the wavefolder for input sinewaves with funda-
mental frequency between 100 Hz and 5 kHz. In this context, we
consider SNR to be the power ratio between the desired harmonics
and aliasing components. This plot shows that the two-point poly-
BLAMP method provides an SNR increase of approx. 12 dB over
a trivial audio-rate implementation. When combined with eight
times oversampling the proposed method yields an average SNR
increase of approx. 20 dB w.r.t. oversampling by factor 64.

In terms of computational costs, the two-point polyBLAMP
method is highly efficient in that only samples around disconti-
nuities are processed. Therefore, the complexity of the method
increases as a function of fundamental frequency, not oversam-
pling factor. For the case of 5-kHz sinusoidal input (i.e. the
worst-case scenario for the polyBLAMP method in terms of op-
eration count), Matlab simulations indicated that the proposed
method is approx. 6 times faster than oversampling by factor 64.
This estimate does not include the costs of any resampling filters
at the output of the system, which will also be more expensive
for the case of oversampling by 64. An implementation of the
proposed model and accompanying sound examples are available

5. CONCLUSIONS

In this work we have examined the underlying structure of the
Buchla 259 wavefolder, also known as the timbre circuit. The anal-
ysis of the circuit provides a glimpse into the unconventional de-
signs of Don Buchla and his approach to sound synthesis. A digi-
tal model of the wavefolder is derived using nonlinear memoryless mappings based on the input–output voltage relationships within the circuit. In an effort to minimize the high levels of aliasing distortion caused by the inherent frequency-expanding behavior of the system, the use of the BLAMP method has been proposed, more specifically in its two-point polynomial form. This method reduces the oversampling requirements of the system, allowing us to accurately process sinusoidal waveforms with fundamental frequencies up to 5 kHz at a sample rate of 352.8 kHz, which is eight times the standard audio rate. The proposed model is free from perceivable aliasing and can be implemented as part of a real-time digital music synthesis environment.

6. ACKNOWLEDGMENTS

The work of Fabián Esqueda is supported by the Aalto ELEC Doctoral School. Part of this work was conducted during Julian Parker’s visit to Aalto University in March 2017.
7. REFERENCES


