Digital Wavefolder and Bandlimiting

Samuel Thibodeau[*]
McGill Undergraduate in Physics

[†]
McGill Musical Science and Technology and
McGill Department of Physics

(Dated: April 24, 2019)

I. INTRODUCTION

The paper I worked with for this project is titled "Virtual Analogue Buchla 259 Wavefolder" [1]. As a brief overview, the paper starts with the analysis of a simplified 259 Wavefolder circuit. From analyzing the circuit using ideal op-amp laws, as well as some knowledge of the specific op-amps in the original circuit, the authors develop a static, memoryless function to describe the non-linear behaviour of the output (described in Section III). This mapping is confirmed with LTSpice simulations. Next, the wavefolder is implemented digitally using the mapping found earlier, and the signal is bandlimited using a Bandlimited Ramp (BLAMP) method. Because the paper deals only with static sine waves, their method of applying the bandlimiting is simple and fast, but not flexible. Lastly, the authors compare the signal to noise ratio (SNR), calculated as the power ratio between the desired harmonics and the aliasing frequencies, between non-antialiased $44.1kHz$ implementation, $44.1kHz$ BLAMP implementation, $64 \cdot 441.kHz$ oversampling, and $8 \cdot 44.1kHz$ oversampling with BLAMP. This report will describe the results of testing and implementing this wavefolder, as well as possible improvements, both from a technical and a musical standpoint.

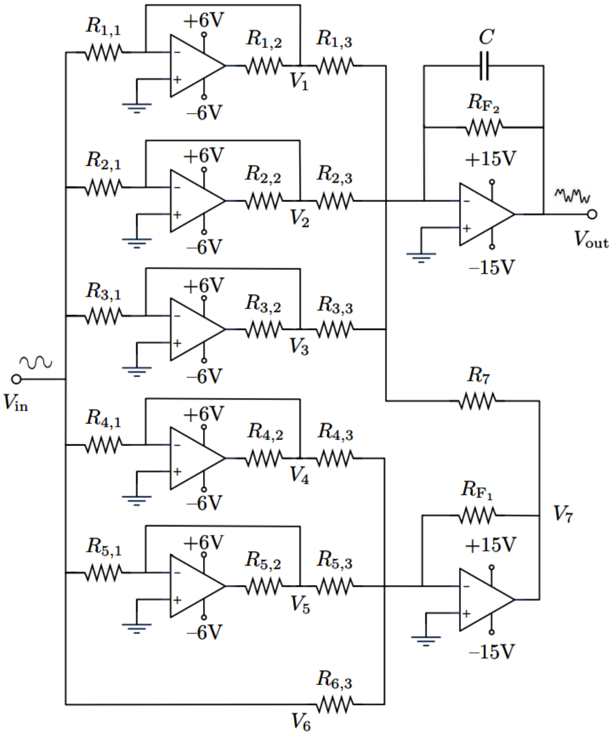


FIG. 1: The full wavefolding circuit. [1]

II. CIRCUIT ANALYSIS AND SPICE SIMULATIONS

We begin with the analysis of the circuit, shown in Figure 1. This circuit consists of multiple parts; the central part consists of five folding cells in parallel (shown in Figure 2). These take as input the signal V_{in}, a sine wave in the original wavefolder. Due to the virtual ground at the op-amp's non-inverting input, it attempts to maintain $0V$ at the negative input as well. Due to the specific op-amps in the circuit (a CMOS op-amp which can output the rail voltages), the actual voltage at V_k (the points labelled $V_1, V_2, \dots$) will be $0V$ below a threshold voltage, and a version of the input signal offset by that threshold voltage above this threshold. Each folding cell has a different threshold, which is based on the ratio $\frac{R_1}{R_2}$ and the rail voltage V_R (which is $\pm 6V$ in the original circuit). The output of these cells then gets added to the original by the two inverting summing amplifiers. Because the first summing amplifier feeds into the second, everything summed by that first op-amp is not inverted, while the cells whose outputs are added directly by the second summing amplifier have their voltages inverted. The addition is a weighted sum, where each cell's amplitude is multiplied by $\frac{R_F}{R_3}$, where R_3 denotes the resistor in front of the signal before the summing op-amp. Lastly, the capacitor around the inverting input and output of the second summing amplifier acts as a one-pole low pass filter with cutoff frequency $\frac{1}{2*\pi*R_{F2}*C} \simeq 1.33kHz$. This depends on the op-amp operating within its linear region, which is expected with input signal with amplitude below the rail voltages of the summing op-amp. Section VI will look in the physical meaning of some of these values, as well as the parameters which may be modulated for

[*] samuel.thibodeau@mail.mcgill.ca; McGill ID 260605279
[†]

pleasing musical effects.

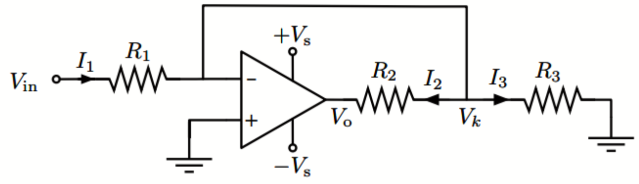


FIG. 2: Single folding cell. [1]

The circuit is simulated in LTSpice to confirm the mapping equation (which will be discussed in section VI), and to test the circuit's behaviour to other input waves. As per the author's recommendations, the LTC6088 op-amp model from the LTSpice library is used to replicate the behaviour of the real circuit. The authors found that the mapping function and the circuit simulations agreed to within $10^{-5}V$ [2]. This indicates that the mapping equation is a good recreation of the circuit in LTSpice, which itself should be a good approximation of the original Buchla circuit. The full LTSpice circuit is shown in the appendix A. Of note is that, because the rail voltages are $V_R = \pm 6V$ for the folding cells, the input signal voltage must be somewhat close to this, and the circuit is tested with a $5V$ signal. This activates 4 of the 5 folding cells. The expected output is shown in Figure 3.

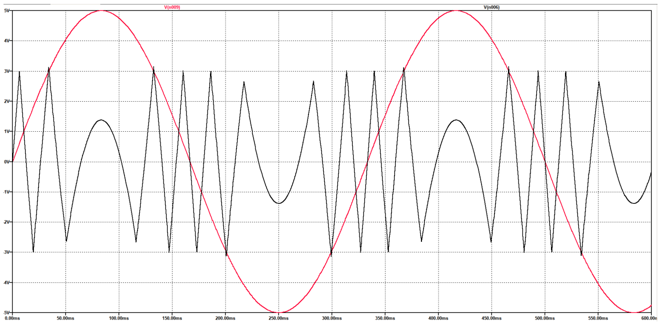


FIG. 3: Output (in black) of the LTSpice simulation for a $5V$ sine wave input (in red).

III. IMPLEMENTATION

The digital implementation of this circuit using the mapping function is done in MATLAB. The main file is "wavefolder.m". The file contains different methods of implementing the same circuit with different levels of fidelity: the naive direct implementation of the mapping function, the 2-point polyBLAMP using static sine input, the 2-point polyBLAMP with clipping detection, and the 4-point polyBLAMP with clipping detection. These are anti-aliasing techniques which will be discussing in Section IV.

The mapping function derived in the original paper consists of an equation of the output signal, and individual equations for the voltage at the output of each cell. The latter of these all share the same general form, with the specific resistor values determining the exact behaviour of each cells. If $k = 1, 2, 3, 4, 5$ denotes the k^{th} folding cell, V_{in} the input signal, V_r the rail voltage of the op-amps, then

$$V_k = \begin{cases} \dfrac{R_{k,2}}{R_{k,1} + R_{k,2} + \frac{R_{k,1}R_{k,2}}{R_{k,3}}} V_{in} - \dfrac{R_{k,1}}{R_{k,1} + R_{k,2} + \frac{R_{k,1}R_{k,2}}{R_{k,3}}} sgn(V_{in}) V_R, & |V_{in}| > \dfrac{R_{k,1}}{R_{k,2}} V_r \\ 0, & |V_{in}| < \dfrac{R_{k,1}}{R_{k,2}} V_r \end{cases} \tag{1}$$

$$V_{out} = V_{in} \frac{R_{F1}}{R_{6,3}} \frac{R_{F2}}{R_7} + V_5 \frac{R_{F1}}{R_{5,3}} \frac{R_{F2}}{R_7} + V_4 \frac{R_{F1}}{R_{4,3}} \frac{R_{F2}}{R_7} - V_3 \frac{R_{F2}}{R_{3,3}} - V_2 \frac{R_{F2}}{R_{2,3}} - V_1 \frac{R_{F2}}{R_{1,3}} \tag{2}$$

These equations fully describe the behaviour of the system to any input signal within the linear operation of the op-amps, which limited by the rail voltages of the summing amplifiers. The folding-cell op-amps behave linearly until their output exceeds the rail voltage. Another source of clipping is the saturation of the summing amplifiers, see Figure 4. This is important since, as will be discussed in Section VI, one of the most interesting modulation destinations is the ratio of of amplitudes between the input signal and the rail voltages of the folding cells, which requires interplay between the input voltage, the rail voltages of the folding cells, and the rail voltages of the summing amplifiers.

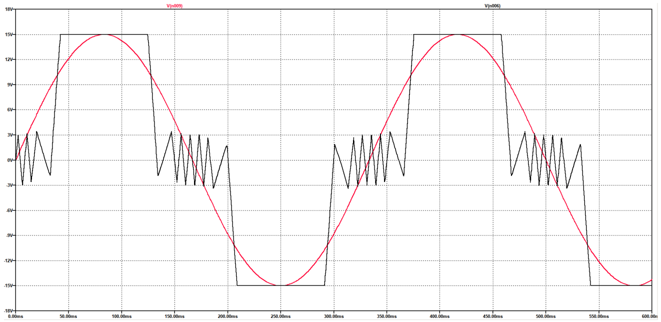


FIG. 4: Output (in black) of the LTSpice simulation for a 15V sine wave input (in red).

IV. BLAMP

The sharp edges introduced by the wavefolding is a case of non-linear waveshaping. We see that, even with a bandlimited input like a sine wave, the output will have many edges, leading to a non-bandlimited output (since discontinuities in the signal require infinitely many harmonics). The original paper employed Bandlimited Ramp (BLAMP) to bandlimit the signal, and this technique will be explore here.

The idea of BLAMP is an extension of the bandlimited impulse (BLIT) and the bandlimited step (BLEP). Where BLEP is the first derivate of BLIT, BLAMP, is the double integral. Of interest is the BLAMP residual, the difference between the trivial ramp ($R(t) = 0,\ t < 0;\ R(t) = t,\ t > 0$). These are all shown in Figure 5 and 6.

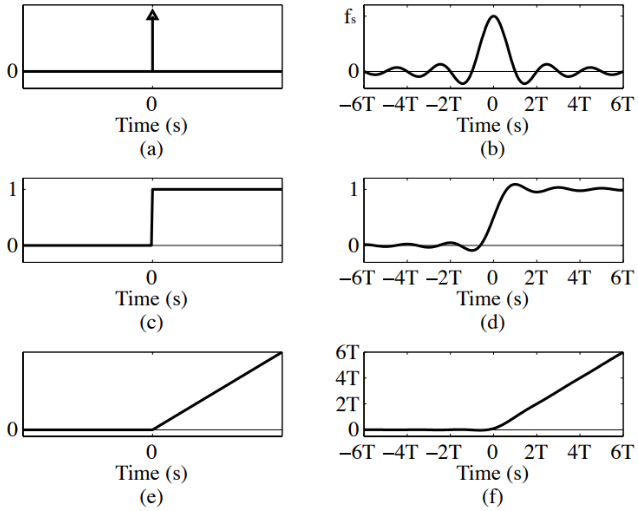


FIG. 5: (a) Trivial impulse. (b) Bandlimited impulse (BLIT). (c) Trivial step function. (d) Bandlimited step (BLEP), integral of b). (e) Trivial ramp. (f) Bandlimited ramp (BLAMP), double integral of b). [3]

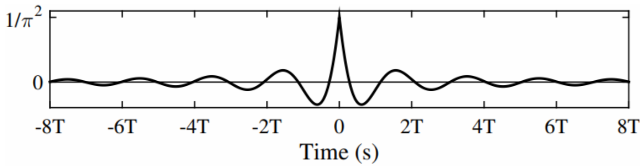


FIG. 6: The BLAMP residual, the difference between f) and e) in Figure 5. [3]

The general idea of bandlimiting with BLAMP is to superimpose the BLAMP residual with discontinuities in the signal, especially 90° corners, which are common in clipping circuits. This leads to a smoothing of the corner, approximating the Gibbs phenomenon [2] (the result of truncating an infinite Fourier series, similar to a square wave constructed from a few of the first harmonics). An example of this is shown in Figure 7.

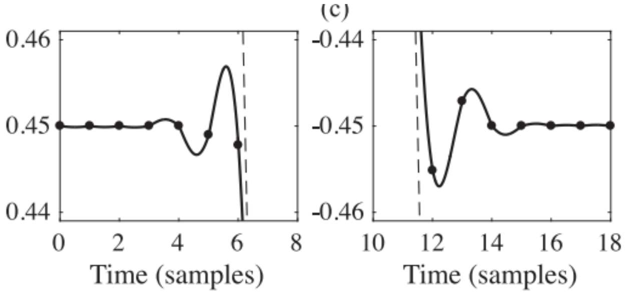


FIG. 7: The effect of adding the BLAMP residual to the corner of a square wave discontinuity. [2]

The points where this needs to be applied in the wavefolder circuit are the values of the input which cross the threshold for any of the folding cells. The original authors implement BLAMP bandlimiting by forcing clipping in V_k at the point the threshold frequencies using an inverse clipper (see Figure 8) before the folding. V_k will follow the input as long as the input is above $\frac{R_{k,1}}{R_{k,2}}V_r$. When equal to or below this threshold, V_k will be equal to the threshold, jumping to threshold at opposite polarity when the input signal crosses 0.

The BLAMP residual is then added or subtracted (based on polarity) to the corner values: the values where the original input reaches the cell threshold. The residual is also first scaled by the value of the slope of the input signal, and must be centered on the clipping point. This often requires a fractional delay d; a measure of how many fractions of a sample ahead the actual clipping point is from the nearest sample behind the clipping point. d should always be on the order $(0, 1]$. As a comparison with the brickwall filter, this method of applying BLAMP bandlimiting only requires $8 - 16$ operations per period of the signal (between $2 - 4$ per intersections, with 4 intersections per period). In contrast, an FIR Hamming windowed low pass filter (similar to the one seen in assignment 5) with 0.003dB of stop-band attenu-

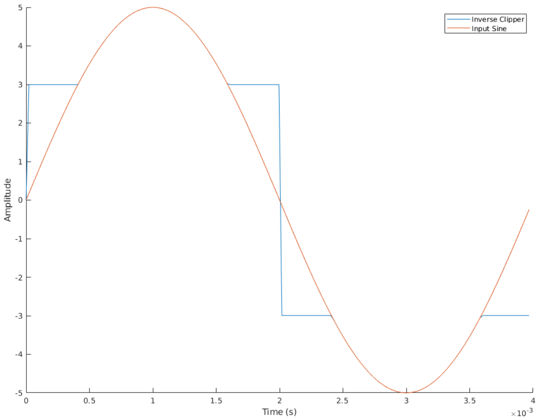


FIG. 8: The inverse clipping applied to V_2 before the folding (blue). The original signal is shown in orange. The threshold voltage of cell 2 is 2.994.

ation would require 1584 coefficients, meaning that each sample of the signal adds 1584 operations. Therefore, by judiciously applying bandlimiting where it is necessary, we can improve the efficiency of algorithms while maintaining sound quality.

The equation for the BLAMP function is given by

$$R_{BL}(t) = t\left(\frac{1}{2} + \frac{1}{\pi}\int_0^{\pi F_s t} \frac{\sin(t')}{t'}dt'\right) + \frac{\cos(\pi F_s t)}{\pi^2 F_s} \quad (3)$$

While this equation, and its residual given by its difference from $R(t)$, can greatly reduce the aliasing in a signal, the calculation of of the BLAMP is extremely costly in terms of computation. As such, the BLAMP and its residual are usually approximated. The most common approximations are polynomial B-splines.

A. 2-point polyBLAMP

The simplest of the spline approximations to the BLAMP residual is the 2-point polyBLAMP. In this regime, the BLIT is approximated by the triangle of width $2T_s$ (technically a first order B spline), and the BLEP and BLAMP are its first and second integrals. These are shown in Figure 9, and the spline values on either side of 0 are given in Figure 10 where d is the fractional delay.

Once the clipping points are found, the the 2-point polyBLAMP residual (multiplied by the slope and with appropriate polarity) are added to the two samples on either side of the clipping point. The original authors determine the clipping point from the equation

$$t = \frac{\sin^{-1}(\frac{V_r R_{k,1}}{A R_{k,2}})}{2 * \pi * f_0} \quad (4)$$

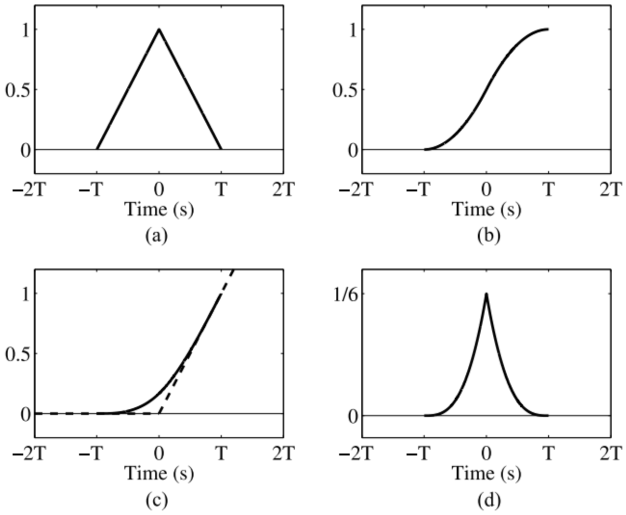
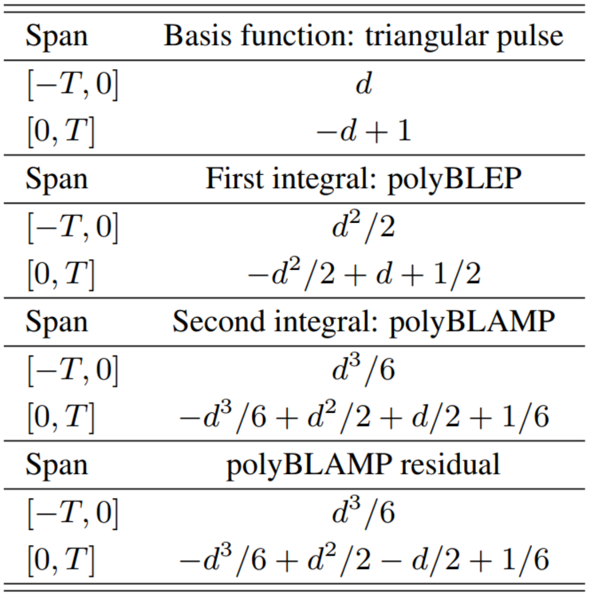


FIG. 9: (a) Triangle approximation of the BLIT. (b) Integral of (a) 2-point BLEP. (c) Integral of (b), the 2-point BLAMP with the trivial ramp. (d) 2-point BLAMP residual approximation. [2]



Span	Basis function: triangular pulse
$[-T, 0]$	d
$[0, T]$	$-d + 1$
Span	First integral: polyBLEP
$[-T, 0]$	$d^2/2$
$[0, T]$	$-d^2/2 + d + 1/2$
Span	Second integral: polyBLAMP
$[-T, 0]$	$d^3/6$
$[0, T]$	$-d^3/6 + d^2/2 + d/2 + 1/6$
Span	polyBLAMP residual
$[-T, 0]$	$d^3/6$
$[0, T]$	$-d^3/6 + d^2/2 - d/2 + 1/6$

FIG. 10: Spline parameters to (a), (b), (c), and (d) from 9. [4]

Where f_0 is the frequency of the input sinusoid. This is efficient and removes the need from detecting clipping points. It is, however, not flexible as it prevents frequency changes and inputs other than sinusoid (I call it the 2-point static sine BLAMP). An alternative 2-point polyBLAMP algorithm is presented by Esqueda [2] in which the BLAMP bandlimiting is applied as V_k is calculated and clipped from the input, which eliminates the necessity of knowing when the input signal will reach the threshold of the folding cell. The slope is calculated from the difference between the two samples on either side of the clipping, and this is used to estimate the fractional delay using linear interpolation.

B. 4-point polyBLAMP

A more complex approximation of the BLAMP residual is a third order B-spline, resulting in the 4-point polyBLAMP. This method increases the computational cost, but can extend the bandlimiting to four points around the clipping point. The derivation of the 4-point polyBLAMP is similar to that shown in IV A. Figure 11 shows the functions themselves, and Figure 12 shows the spline values within the range of each of the four samples ($[-2T, -T]$, $[-T, 0]$, $[0, T]$, $[T, 2T]$).

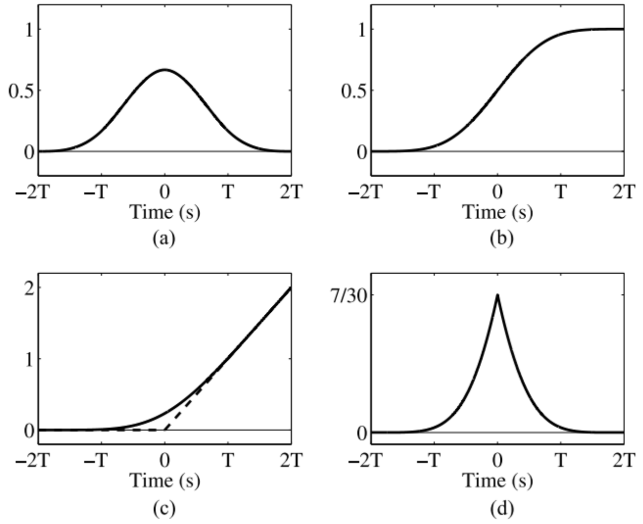
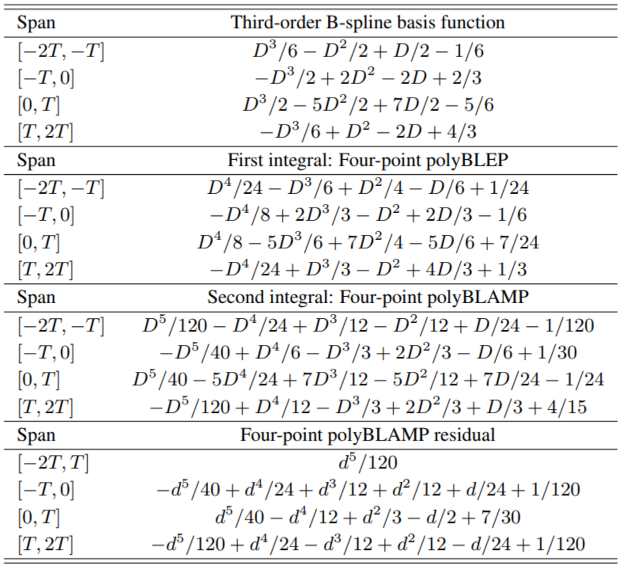


FIG. 11: (a) 3^{rd} order B-Spline approximation of the BLIT. (b) Integral of (a) 4-point BLEP. (c) Integral of (b), the 4-point BLAMP with the trivial ramp. (d) 4-point BLAMP residual approximation. [3]

This implementation is much more complex, and is adapted in the code from Esqueda's code [3]. This code not only calculates the slope of the original function as a second order polynomial, it also estimates the exact clipping point using a converging iterative method, the Newton-Raphson algorithm. Once the clipping point estimated and the fractional delay calculated, the 4-point polyBLAMP residual is addedd to the four points surrounding the clipping point.

V. RESULTS

Below are plots of various results for the comparison of the different antialiasing methods.



Span	Third-order B-spline basis function
$[-2T, -T]$	$D^3/6 - D^2/2 + D/2 - 1/6$
$[-T, 0]$	$-D^3/2 + 2D^2 - 2D + 2/3$
$[0, T]$	$D^3/2 - 5D^2/2 + 7D/2 - 5/6$
$[T, 2T]$	$-D^3/6 + D^2 - 2D + 4/3$
Span	**First integral: Four-point polyBLEP**
$[-2T, -T]$	$D^4/24 - D^3/6 + D^2/4 - D/6 + 1/24$
$[-T, 0]$	$-D^4/8 + 2D^3/3 - D^2 + 2D/3 - 1/6$
$[0, T]$	$D^4/8 - 5D^3/6 + 7D^2/4 - 5D/6 + 7/24$
$[T, 2T]$	$-D^4/24 + D^3/3 - D^2 + 4D/3 + 1/3$
Span	**Second integral: Four-point polyBLAMP**
$[-2T, -T]$	$D^5/120 - D^4/24 + D^3/12 - D^2/12 + D/24 - 1/120$
$[-T, 0]$	$-D^5/40 + D^4/6 - D^3/3 + 2D^2/3 - D/6 + 1/30$
$[0, T]$	$D^5/40 - 5D^4/24 + 7D^3/12 - 5D^2/12 + 7D/24 - 1/24$
$[T, 2T]$	$-D^5/120 + D^4/12 - D^3/3 + 2D^2/3 + D/3 + 4/15$
Span	**Four-point polyBLAMP residual**
$[-2T, T]$	$d^5/120$
$[-T, 0]$	$-d^5/40 + d^4/24 + d^3/12 + d^2/12 + d/24 + 1/120$
$[0, T]$	$d^5/40 - d^4/12 + d^2/3 - d/2 + 7/30$
$[T, 2T]$	$-d^5/120 + d^4/24 - d^3/12 + d^2/12 - d/24 + 1/120$

FIG. 12: Spline parameters to (a), (b), (c), and (d) from 11. [3]

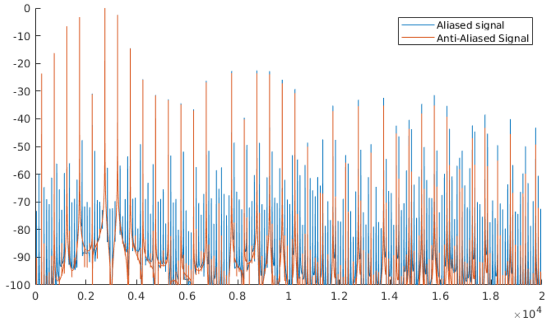


FIG. 13: Spectrum of the direct wavefolding (blue) output and the 2-point static sine BLAMP (orange). The input wave is a 250Hz sine wave.

Figure 13 shows a comparison between the direct folding output and the 2-point static sine BLAMP bandlimiting. This plot clearly shows many aliased components in the range of −50dB and below, which are removed from the signal through BLAMP. The antialiasing does affect the amplitude of higher frequency harmonics which had not been aliased.

The above three figures (Figures 14, 15, and 16) show spectrograms of the wavefolding with 2-point clipping detection BLAMP, 4-point clipping detection BLAMP, and no antialiasing respectively. Figure 14 shows improvements with respect to Figure 16 in that the background colour is less yellow, indicating a lower aliasing amplitude. Figure 15 also shows this behaviour within the first third of the time, although this gives way to very high amplitude noise artifacts. This can also be seen at the end of Figure 14. This is similar to the final results from one of Esqueda's first papers [4], shown in Figure

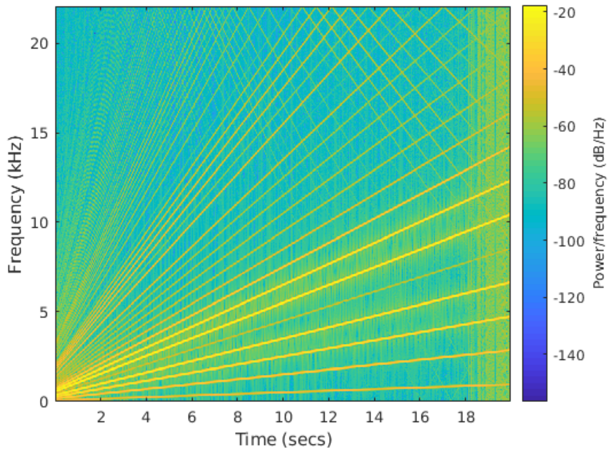


FIG. 14: Output of the 2-point polyBLAMP with clipping estimating to a sine wave ramping in frequency from 50Hz to 500Hz

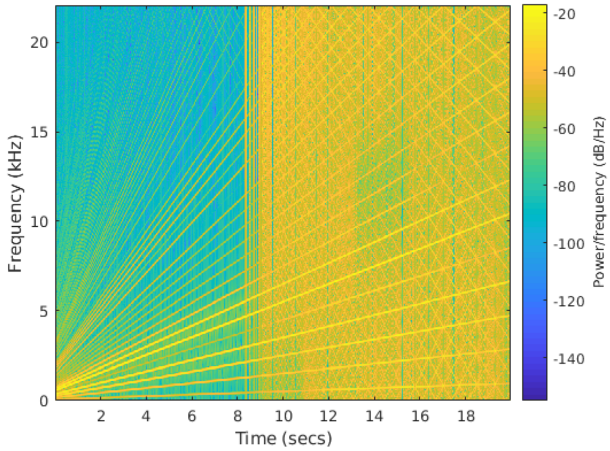


FIG. 15: Output of the 4-point polyBLAMP with clipping estimating to a sine wave ramping in frequency from 50Hz to 500Hz

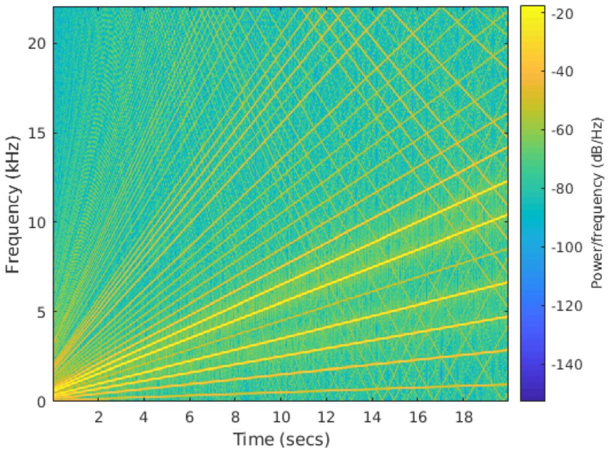


FIG. 16: Output of the direct wavefolding (fully aliased) with input sine wave ramping in frequency from 50Hz to 500Hz.

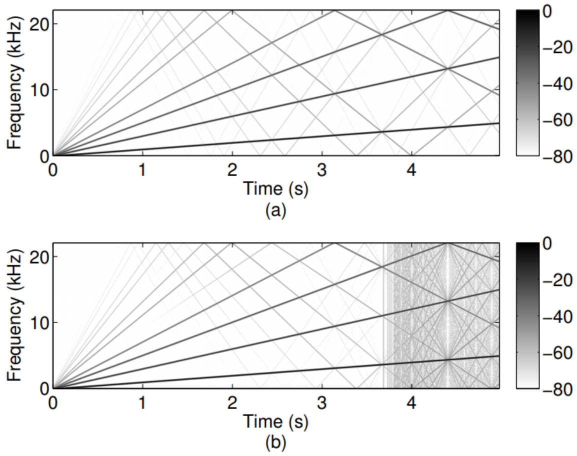


FIG. 17: Results from Esqueda's research into BLAMP bandlimiting for soft clipping. [4]

17. Esqueda states that this could be eliminated with better centering of BLAMP residuals with the clipping points and better estimates of the slope values.

VI. PARAMETER MEANING AND MODIFICATIONS

While the bandlimiting of the system is the primary area of improvement for this project, this section will focus on the musical capabilities of the circuit, specifically the parameters which can be modulated for interesting musical outcomes.

The primary source of timbral movement in the sound is the ratio of amplitudes between the input signal and the rail voltage. As the input signal's amplitude changes, different number of folding cells will activate, changing the amount of folding in the sound as well as its depth. This is shown in MATLAB script "Modulations.m". The frequency of the input signal can also be changed, although this does not directly affect the behaviour of the wavefolding. In the mapping function used to discretize the behaviour of the wavefolder, there is no limit on the ratio $\frac{V_{in}}{V_r}$. Realistically however, the folding cells will clip when the sum of the input voltage and the threshold voltage $\frac{R_{k,1}}{R_{l,2}}V_r$ is greater than V_r.

As discussed throughout the paper, the ratio $\frac{R_{k,1}}{R_{l,2}}$ determines the threshold voltage of a folding cell. Additionally, the ratios $\frac{R_{k,3}}{R_{F,2}}$ will determine the weight of a folding cell in the sum. This is because the values of R_{F1} and R_7 are the same in the schematic, reducing the weight of all folding cells in the sum to the ratio of the summing resistor R_{F2} and their respective third resistor. Any given cell can also be made to invert or not by the choice of which summing amplifier it is attached to. Being fed into the bottom summing amplifier (R_{F1}) will lead to two inversions, so no net change in polarity. Being fed into the top summing amplifier (R_{F2}) will lead to only one inversion, effectively subtracting the output of those folding cells from the input signal. Lastly, the number of cells can be changed; fewer cells will have a mellower quality as the output will more closely resemble the original sine wave, while greater number of cells will results in harsher sounds. This could be implemented in a similar way to phasers with many stage settings (2-stage, 4-stage, 8-stage, etc).

VII. CONCLUSION

The wavefolder was analyzed, tested in LTSpice, discretized to a high accuracy, and implemented digitally. Efficient antialiasing was then applied to the system through various methods which were compared. While the 2-point static sine BLAMP gave the best results, it's inflexible requirements make it a poor choice for implementation, leading to 2-point and 4-point clipping detecting BLAMP methods. These gave promising results, although they introduce artifacts in the signal. It is likely that the implementations could be improved, through more accurate calculations of clipping positions, slope, and higher order splines. Lastly, recommendations are given as to the modifications and modulations which can be applied to the circuit/code to introduce more musically pleasing results. The script used to generate 2-point static, 2-point clipping detection, and 4-point clipping detection is attached, title “wavefolder.m”, while a script implementing 2-point clipping detection with various commented commands for wavefolder modulations is attached, titled “modulations.m”.

Appendix A: LTspice Circuit

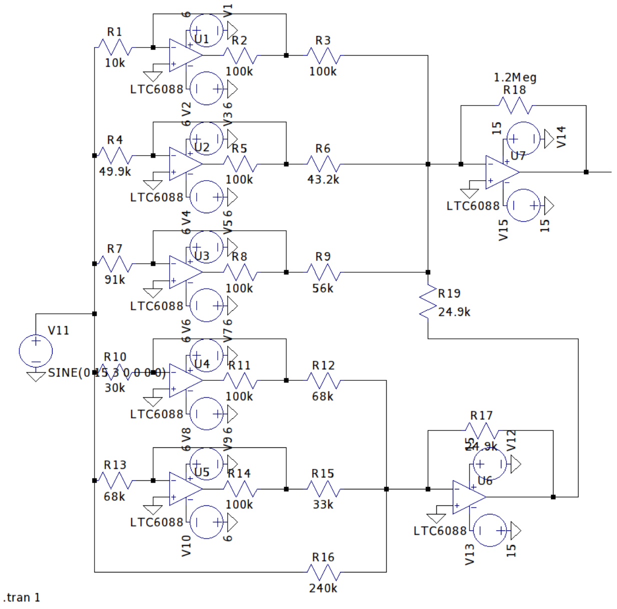


FIG. 18: Schematic implemented in LTSpice from 1 with component values taken from the original paper. [1]

[1] F. Esqueda, H. Pöntynen, V. Välimäki, and J. D. Parker (2018).

[2] F. Esqueda, S. Bilbao, and V. Välimäki, IEEE Transactions on Signal Processing **64**, 5255 (2016).

[3] F. Esqueda and S. Bilbao (2016).

[4] F. Esqueda, V. Välimäki, and S. Bilbao, 2015 23rd European Signal Processing Conference (EUSIPCO) , 2014 (2015).