GENERALIZATIONS OF VELVET NOISE AND THEIR USE IN 1-BIT MUSIC

Kurt James Werner

The Sonic Arts Research Centre (SARC)
Queen's University Belfast (QUB)
Belfast, United Kingdom of Great Britain and Northern Ireland
kurt.james.werner@gmail.com

ABSTRACT

A family of spectrally-flat noise sequences called "Velvet Noise" have found use in reverb modeling, decorrelation, speech synthesis, and abstract sound synthesis. These noise sequences are ternary—they consist of only the values -1, 0, and $+1$. They are also sparse in time, with pulse density being their main design parameter, and at typical audio sampling rates need only several thousand non-zero samples per second to sound "smooth."

This paper proposes "Crushed Velvet Noise" (CVN) generalizations to the classic family of Velvet Noise sequences including "Original Velvet Noise" (OVN), "Additive Random Noise" (ARN), and "Totally Random Noise" (TRN). In these generalizations, the probability of getting a positive or negative impulse is a free parameter. Manipulating this probability gives Crushed OVN and ARN low-shelf spectra rather than the flat spectra of standard Velvet Noise, while the spectrum of Crushed TRN is still flat. This new family of noise sequences is still ternary and sparse in time. However, pulse density now controls the shelf cutoff frequency, and the distribution of polarities controls the shelf depth.

Crushed Velvet Noise sequences with pulses of only a single polarity are particularly useful in a niche style of music called "1-bit music": music with a binary waveform consisting of only 0s and 1s. We propose Crushed Velvet Noise as a valuable tool in 1-bit music composition, where its sparsity allows for good approximations to operations, such as addition, which are impossible for signals in general in the 1-bit domain.

1. INTRODUCTION

In 2007, Karjalainen and Järveläinen defined a new type of sparse noise sequences which they called "Velvet Noise" [1]. This was later more specifically termed "Original Velvet Noise" (OVN) by Välimäki *et al.* [2]. These noise sequences have some similarities to sparse noise investigated by Schreiber in 1960 [3] and have a few peculiar qualities. First, they are sparse in time—most of their samples are actually zero. Second, the non-zero samples only take the values -1 and $+1$. Specifically, OVN is produced by defining a pulse density, splitting time into equal-length windows, and distributing a single impulse into each window, with both its exact position within the window and its sign ($\pm$) randomized. OVN is clearly not i.i.d. (independent and identically distributed), since the value of each sample within a window is related to the values of all other samples in the window. Despite this, OVN remarkably has a flat magnitude spectrum and an autocorrelation which is nearly zero everywhere except at zero-lag, making it very similar to Gaussian white noise (an i.i.d. process which is neither sparse in time nor limited to particular values). With a sufficiently high pulse density, Velvet Noise can even sound just as "smooth" as Gaussian noise [2].

One fascinating property of Velvet Noise sequences is that they are very efficient to convolve by, since they are very sparse (mostly 0s) and the non-zero values (± 1) don't require an actual multiplication during convolution [4, 5]. These properties have led Velvet Noise to be used in reverb modeling [1, 5, 6, 7, 8, 9], the design of decorrelation filters [10, 11], speech synthesis [12, 13], and abstract sound synthesis [14, 15].

Related to OVN, several other sparse ternary noise sequences have been proposed, including Additive Random Noise (ARN), Totally Random Noise (TRN), Extended Velvet Noise (EVN), Random Integer Noise (RIN) [2]. ARN randomizes the spacing between pulses rather than distributing a single pulse per window. TRN has a random chance of generating a pulse of a random sign for every single sample. EVN takes OVN and restricts the sample location to only a portion of each window, enforcing a second level of sparsity. RIN is identical to ARN, although both the pulse offsets and sign are read from a precomputed table of integer random numbers rather than independent random numbers [2].

In this paper, I propose generalizations to the family of Velvet Noise sequences which are called "Crushed Velvet Noise" (CVN). Specifically, I propose new variants of OVN, ARN, and TRN now called COVN, CARN, and CTRN (the "C" denoting "Crushed" for each). In Crushed Velvet Noise Sequences, the signs of each pulse is not assigned based on a 50% probability, but rather this probability is exposed as a free parameter that can be manipulated along with the pulse density. This small change allows the creation of a variety of spectra with different properties. COVN and CARN both have low-shelf-like Power Spectral Densities (PSDs), with their cuttoff frequency controlled by the pulse density and their shelf attenuation controlled by the free parameter determining the probability of a positive or negative sign for each pulse.

Adjusting this probability to extreme settings gives sequences where either -1s or $+1$s do not appear[1]. Variants of CVN which only have 0s and 1s are of particular use to a niche approach to electronic music composition called "1-bit music," where the only allowable signal levels are 0 and 1. In the end of this paper, I explain how Velvet Noise and the proposed novel variants can be used in 1-bit composition, specifically highlighting their poten-

[1]The case where there is a 100% chance of a $+1$ and no chance of a -1 has already been investigated briefly in [13], where a unipolar variant of OVN is called Unipolar Velvet Noise (UVN). The sparse noise sequence explored by Schreiber [3] was also unipolar. With these in mind, we can say that this paper fills in the gaps between the proposed unipolar variant and the bipolar OVN sequence.

tial for spectral shaping, volume control, and layering of signals. These procedures are not possible *in general* in the 1-bit domain, since even the simplest operations like addition of signals do not exist in that domain, and so it is very hard to do anything LTI (linear and time-invariant). So, being able to control the spectrum and volume of a *particular* type of 1-bit signal, Crushed Velvet Noise, is extremely useful.

The prefix "crushed" refers both to bitcrushing (the classic lo-fi audio effect) and crushed velvet (the soft fabric). Bitcrushing involves reducing the number of possible signal levels of an audio signals, for instance down to $2^8 = 256$ levels for an 8-bit bitcrusher. Velvet Noise's ternary character is related to bitcrushing, and the proposed "crushed" variants, at their extreme settings, further reduce the set of possible sample values down to two. Crushed velvet is a particular kind of velvet fabric whose cut threads have been pressed in different directions in specific ways. Once the reader has understood the construction of Crushed Velvet Noise, a loose metaphorical connection is not hard to imagine.

In the rest of the paper I review the classic definitions of OVN, ARN, and TRN (§2), define and study the novel COVN, CARN, and CTRN sequences (§3), and explain an application to "1-bit music" (§4). §5 concludes and proposes avenues for future work.

2. CLASSIC VELVET NOISE DEFINITIONS

In this section, we will briefly review the classic Velvet Noise sequences: Original Velvet Noise (OVN), Additive Random Noise (ARN), and Totally Random Noise (TRN).

2.1. Original Velvet Noise (OVN)

Original Velvet Noise (OVN), proposed in [1] and termed OVN in [2], is defined by

$$s_{\text{ovn}}(n) = \begin{cases} 2\,\|r_2(m)\| - 1\,, & \text{if } n = k_{\text{ovn}}(m) \\ 0\,, & \text{otherwise} \end{cases}\,, \tag{1}$$

where $n = 0, 1, 2, \ldots$ is the discrete-time sample index, $\|\cdot\|$ is a function that rounds to the nearest integer, $r_2(m)$ is a sequence of random numbers uniformly distributed between 0 and $+1$, and k_{ovn} is a sequence of impulse locations defined by

$$k_{\text{ovn}}(m) = \|mT_{\text{d}} + r_1(m)(T_{\text{d}} - 1)\|\ , \tag{2}$$

where $m = 0, 1, 2, \ldots$ is a discrete pulse index, T_{d} is window width in samples, and $r_1(m)$ is another sequence of random numbers uniformly distributed between 0 and $+1$. T_{d} is related to the pulse density and sampling rate f_{s} by

$$N_{\text{d}} = f_{\text{s}}/T_{\text{d}}\,. \tag{3}$$

The sampling rate used throughout this paper is $f_{\text{s}} = 96\,\text{kHz}$.

The definition (1)–(2) of OVN is used widely [2, 9, 5, 10]. However, another variant exists [1, 7, 11] and actually precedes (1)–(2) [1]. Its definition of $k_{\text{ovn}}(m)$ is slightly different:

$$k_{\text{ovn,alt}}(m) = \|mT_{\text{d}} + r_1(m)T_{\text{d}}\| = \|(r_1(m) + m)T_{\text{d}}\|\ . \tag{4}$$

The second definition (4) has the potential to rarely have an impulse in the last sample of one window collide with an impulse in the first sample of the next window, if the pulse should occur on a window boundary.

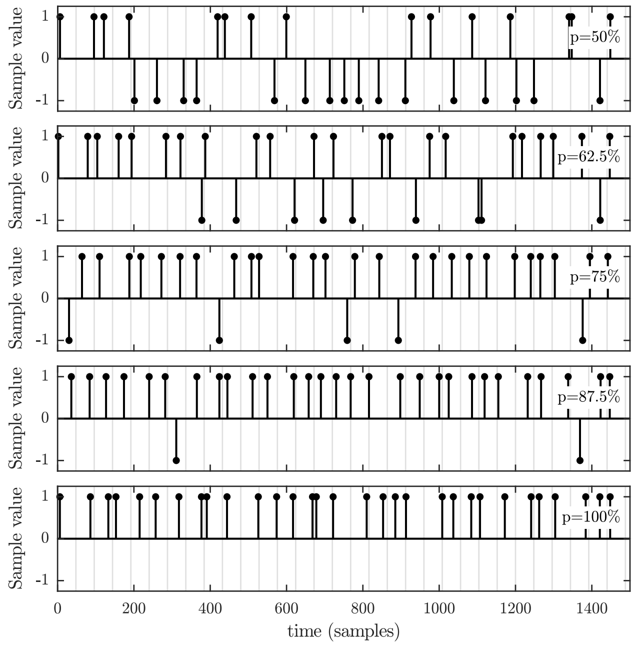


Figure 1: Crushed Original Velvet Noise (COVN), with various probability percentages $p \in \{50\%, 62.5\%, 75\%, 87.5\%, 100\%\}$.

Finally, we can mention that [11] reformulates (4) using a ceiling function $\lceil\cdot\rceil$ rather than $\|\cdot\|$.

An example of an OVN sequence ($N_{\text{d}} = 2000$) is shown in Fig. 1 (top).

2.2. Additive Random Noise (ARN)

Additive Random Noise (ARN) is defined by [2]

$$s_{\text{arn}}(n) = \begin{cases} 2\,\|r_2(m)\| - 1\,, & \text{if } n = \|k_{\text{arn}}(m)\| \\ 0\,, & \text{otherwise} \end{cases}\,, \tag{5}$$

where k_{arn} is a sequence of impulse locations defined by

$$\begin{aligned} k_{\text{arn}}(m) = k_{\text{arn}}(m-1) + 1 \ldots \\ + (1-\Delta)(T_{\text{d}} - 1) + 2\Delta(T_{\text{d}} - 1)r_1(m)\,. \end{aligned} \tag{6}$$

The parameter $\Delta \in [0, 1]$ controls a tradeoff between advancing time by a fixed amount and a random amount.

An example of an ARN sequence ($N_{\text{d}} = 2000$) is shown in Fig. 6 (top).

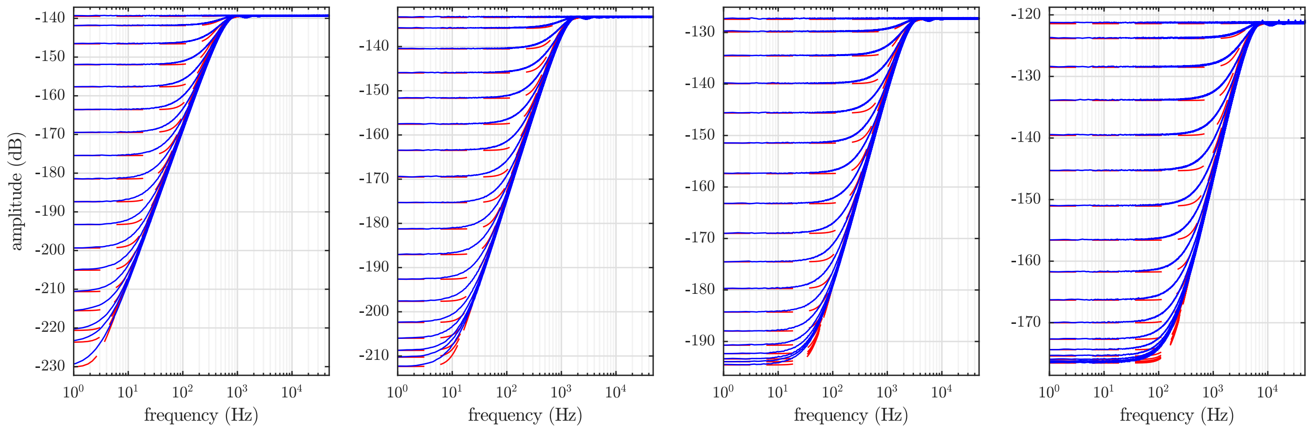
2.3. Totally Random Noise (TRN)

Totally Random Noise (TRN) is defined by [2]

$$s_{\text{trn}}(n) = \left\| \left(\frac{T_{\text{d}}}{T_{\text{d}} - 1}\right) \left(r_1(n) - \frac{1}{2}\right) \right\|\,. \tag{7}$$

TRN was originally investigated by Rubak and Johansen [16, 17].

An example of a TRN sequence ($N_{\text{d}} = 2000$) is shown in Fig. 9 (top).



(a) COVN, $N_\mathrm{d} = 1000$. (b) COVN, $N_\mathrm{d} = 2000$. (c) COVN, $N_\mathrm{d} = 4000$. (d) COVN, $N_\mathrm{d} = 8000$.

Figure 2: Power Spectral Density (PSD) of COVN at various pulse densities $N_\mathrm{d} \in \{1000, 2000, 4000, 8000\}$.

3. CRUSHED VELVET NOISE (CVN)

In this section, novel "Crushed" variants of OVN, ARN, and TRN are introduced.

3.1. Crushed Original Velvet Noise (COVN)

Crushed Original Velvet Noise (COVN) is defined by

$$s_\mathrm{covn}(n) = \begin{cases} 2 \cdot c(r_2(m), p) - 1\,, & \text{if } n = k_\mathrm{covn}(m) \\ 0 & \text{otherwise} \end{cases}\,, \tag{8}$$

which is identical to the traditional OVN sequence except the standard rounding function $\|\cdot\|$ has been replaced by the function $c(x, p)$, defined as

$$c(x, p) = \begin{cases} 1\,, & \text{if } x > p \\ 0\,, & \text{otherwise} \end{cases}\,. \tag{9}$$

The pulse timings $k_\mathrm{ovn}(m)$ are given by

$$k_\mathrm{covn}(m) = \|(r_1(m) + m)\, T_\mathrm{d}\|\,, \tag{10}$$

which is identical to the alternate definition from OVN (4). The reason for basing COVN off of the alternate definition is that the timing equation (2) introduces periodicities into COVN sequences (when $p \neq 0.5$), negatively affecting their noisy character.

Five examples of COVN with pulse density $N_\mathrm{d} = 2000$ and different polarity probabilities $p \in \{0.5, 0.625, 0.75, 0.875, 1.0\}$ are shown in Fig. 1. Notice that $p = 0.5$ is identical to traditional OVN, and that $p = 1.0$ is fully unipolar—it has only 0s and +1s, and no −1s. In this figure, the window boundaries are shown with vertical gray lines.

Power Spectral Density (PSD) estimates of COVN sequences with various pulse densities $N_\mathrm{d} \in \{1000, 2000, 4000, 8000\}$ are shown in Fig. 2. In each case, a family of many polarity probabilities p between 0.5 and 1.0 are shown, where the solid blue lines show the PSDs, with $p = 0.5$ on the top, and increasing to $p = 1.0$ below[2]. The polarity probabilties which are plotted follow

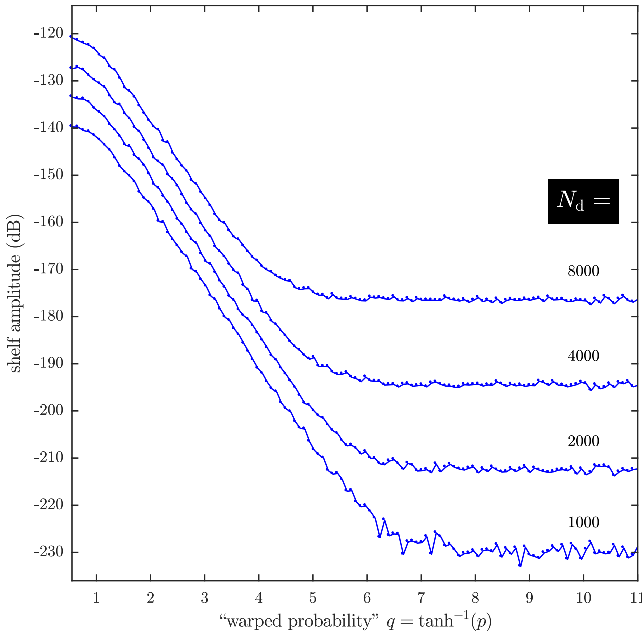


Figure 3: Low shelf amplitude of COVN at various pulse densities $N_\mathrm{d} \in \{1000, 2000, 4000, 8000\}$.

the following pattern:

$$p = \begin{cases} 1/2 & \tau = 0 \\ 1 - (1/2)^\tau & \tau = 1, 2, \ldots 16 \\ 1 & \tau = 17 \end{cases}\,. \tag{11}$$

PSD estimates are produced using Welch's method with a window size of $N_\mathrm{FFT} = 2^{18} = 262\,144$ samples, an overlap size of $N_\mathrm{FFT}/2$, and Hamming windows, and are plotted on a $20 \log_{10}(\cdot)$ scale rather than a $10 \log_{10}(\cdot)$ scale to facilitate comparison with shelf filter magnitude responses. Noise sequences used to generate PSD estimates in this paper are 24 hours long, i.e., $24 \times 60 \times$

[2] In this paper, we will always deal with probabilities between 0.5 and 1.0, biasing the distribution towards +1s. Of course, all of the conisderations of the paper would be identical (except for an opposite dc bias) if we instead considered probabilities between 0.5 and 0.0.

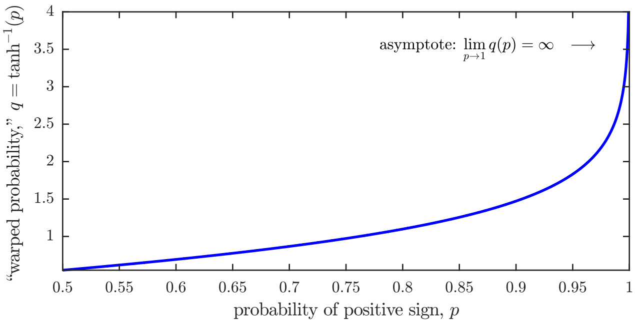


Figure 4: Relationship between probability of positive sign p and "warped probability" q.

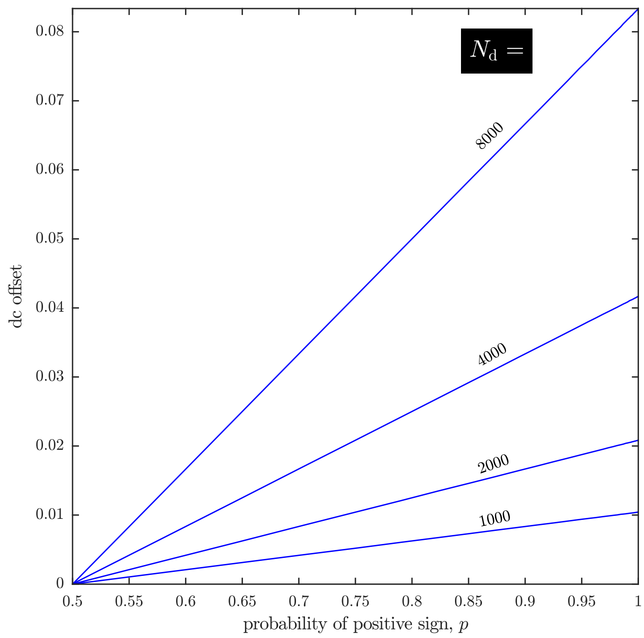


Figure 5: dc offset of COVN at various pulse densities $N_\mathrm{d} \in \{1000, 2000, 4000, 8000\}$.

$60 \times f_\mathrm{s} = 8.2944 \times 10^9$ samples. The purpose of this length, which may seem excessive, is simply to get better PSD estimates by having more frames to average using Welch's method, i.e., a better estimate of the shape of the PSD curves without using any artificial smoothing.

It is clear from Fig. 2 that, unlike OVN with its flat spectrum, COVN sequences have a *low-shelf* characteristic, where the low shelf has some attenuation but never a boost. These PSDs have a transition band slope of $+12$ dB/octave ($+40$ dB/decade), implying that they are similar in some way to white noise through a second-order shelf filter. There are various second-order shelf-filters defined in the literature [18, 19, 20]. The family of noise spectra produced by the COVN noise are similar to the second-order low-shelf filters proposed by Holters and Zölzer in [19]. They propose low-shelf filters with transfer function $H_\mathrm{LS}(s)$ de-

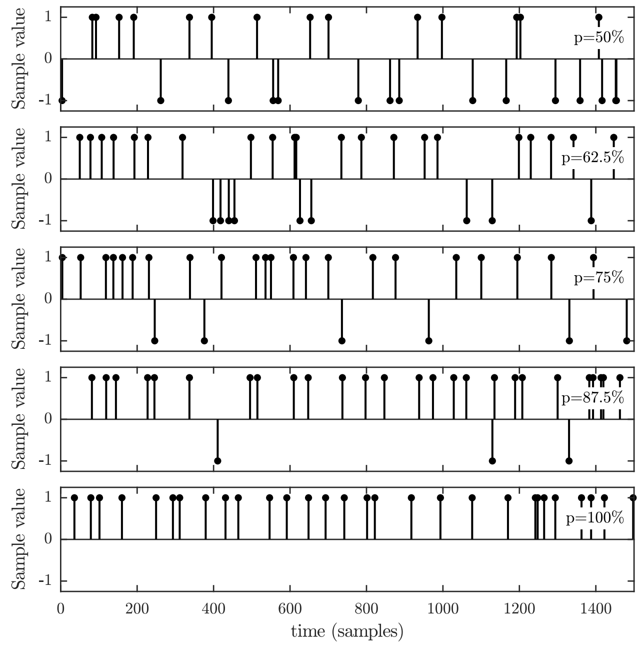


Figure 6: Crushed Additive Random Noise (CARN), with various probability percentages $p \in \{50\%, 62.5\%, 75\%, 87.5\%, 100\%\}$.

fined in pole-zero form on the s-plane by:

$$H_{\mathrm{LS},L}(s) = \prod_{l=1}^{L} \frac{s + \sqrt[L]{g}\,\omega_\mathrm{c} e^{j\left(\frac{1}{2} - \frac{2l-1}{2L}\right)}}{s + \omega_\mathrm{c} e^{j\left(\frac{1}{2} - \frac{2l-1}{2L}\right)}}\,, \tag{12}$$

where $L \geq 1$ is an integer defining the order of the shelf filter, g is the shelf level, and ω_c is the cutoff frequency in radians ($f_\mathrm{c} = \omega_\mathrm{c}/2\pi$ is the cutoff frequency in Hz). Here we are specifically interested in the 2nd-order case ($L = 2$):

$$H_{\mathrm{LS},2}(s) = \frac{\left(s + \sqrt{g}\omega_\mathrm{c} e^{j/4}\right)\left(s + \sqrt{g}\omega_\mathrm{c} e^{-j/4}\right)}{\left(s + \omega_\mathrm{c} e^{j/4}\right)\left(s + \omega_\mathrm{c} e^{-j/4}\right)}\,. \tag{13}$$

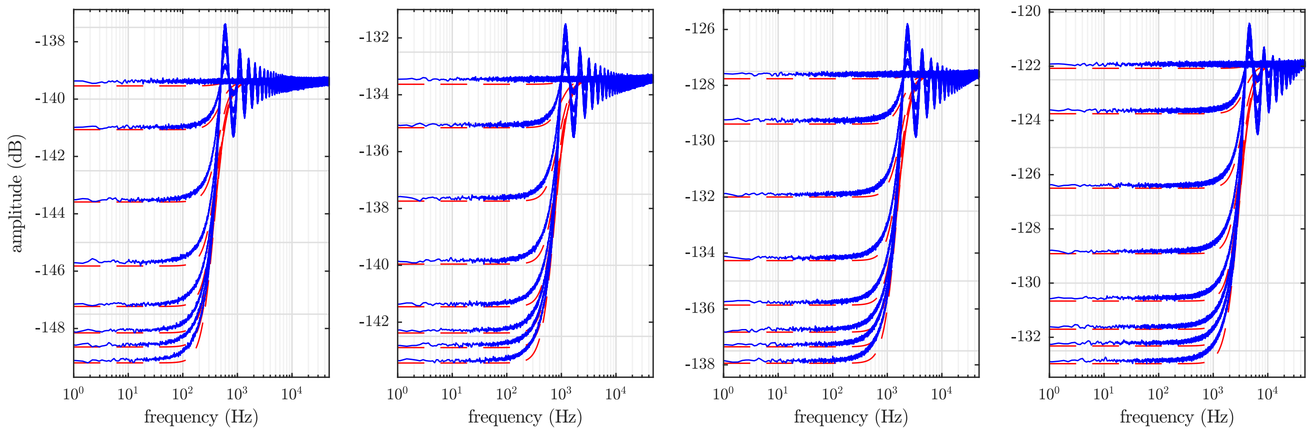
Relating to this family of shelf filter, the cutoff frequency of our PSDs is roughly $f_\mathrm{c} = N_\mathrm{d}/2$. Shelf filter responses that approximate the COVN sequences are shown on Fig. 2 by dashed red lines.

The shelf attenuation, obviously zero for $p = 0.5$, increases as p approaches 1.0. For the various pulse densities tested, the specific relationship between pulse density and shelf level, g in (13), is shown in Fig. 3. Plotting against p would overly compress the visual display of these traces near $p = 1$, so the horizontal axis is instead a "warped probability scale" q, defined simply as

$$q = \tanh^{-1}(p)\,. \tag{14}$$

The relationship between p and q is shown graphically in Fig. 4.

Finally, it is worth mentioning that disturbing the relative probabilities of -1s and $+1$s using p introduces a small dc offset of $(2p - 1)N_\mathrm{d}/f_\mathrm{s}$ to the signal. Notice that when $p = 0.5$, no dc offset is introduced, and that the maximum offset that can be introduced is $\pm N_\mathrm{d}/f_\mathrm{s}$. This is shown graphically in Fig. 5.



(a) CARN, $N_\text{d} = 1000$. (b) CARN, $N_\text{d} = 2000$. (c) CARN, $N_\text{d} = 4000$. (d) CARN, $N_\text{d} = 8000$.

Figure 7: Power Spectral Density (PSD) of CARN at various pulse densities $N_\text{d} \in \{1000, 2000, 4000, 8000\}$.

3.2. Crushed Additive Random Noise (CARN)

Crushed Additive Random Noise (CARN) is defined by

$$s_\text{carn}(n) = \begin{cases} 2 \cdot c(r_2(m), p) - 1\,, & \text{if } n = \|k_\text{carn}(m)\| \\ 0\,, & \text{otherwise} \end{cases}\,, \tag{15}$$

where k_carn is a sequence of impulse locations defined by

$$\begin{aligned} k_\text{carn}(m) = k_\text{carn}(m-1) + 1 \ldots \\ + (1-\Delta)(T_\text{d} - 1) + 2\Delta(T_\text{d} - 1) r_1(m)\,, \end{aligned} \tag{16}$$

which is identical to the traditional ARN (5)–(6), except that the rounding function $\|\cdot\|$ has again been replaced by the new function (9). In this paper, we will only consider the case $\Delta = 1$, i.e., the case where time advances by purely random steps.

Five examples of CARN with pulse density $N_\text{d} = 2000$ and different polarity probabilities $p \in \{0.5, 0.625, 0.75, 0.875, 1.0\}$ are shown in Fig. 6. Notice that $p = 0.5$ is identical to traditional ARN, and that $p = 1.0$ is again fully unipolar.

PSD estimates of CARN sequences with pulse densities $N_\text{d} \in \{1000, 2000, 4000, 8000\}$ are shown in Fig. 7. Again, a family of many polarity probabilities p between 0.5 and 1.0 are shown, where the solid blue lines show the PSDs, with $p = 0.5$ on the top, and increasing to $p = 1.0$ below. The polarity probabilities which are plotted follow the following pattern:

$$p = \begin{cases} 1/2 & \tau = 0 \\ 1 - (1/2)^\tau & \tau = 1, 2, \ldots 6 \\ 1 & \tau = 7 \end{cases}\,. \tag{17}$$

As with COVN, shelf filter responses that approximate the CARN sequences are shown on Fig. 7 by dashed red lines.

As with COVN, the PSDs of CARN sequences look a bit like the PSDs of low-shelf-filtered white noise. CARN sequences, however, have a more pronounced ripple around the cutoff frequency, and much less pronounced shelf attenuation than COVN. Again, the sheld attenuation, obviously zero for $p = 0.5$, increases as p approaches 1.0. For CARN, the relationship between p and shelf level is shown in Fig. 8. The dc offset considerations for COVN, which were based purely on pulse density N_d, apply identically to CARN.

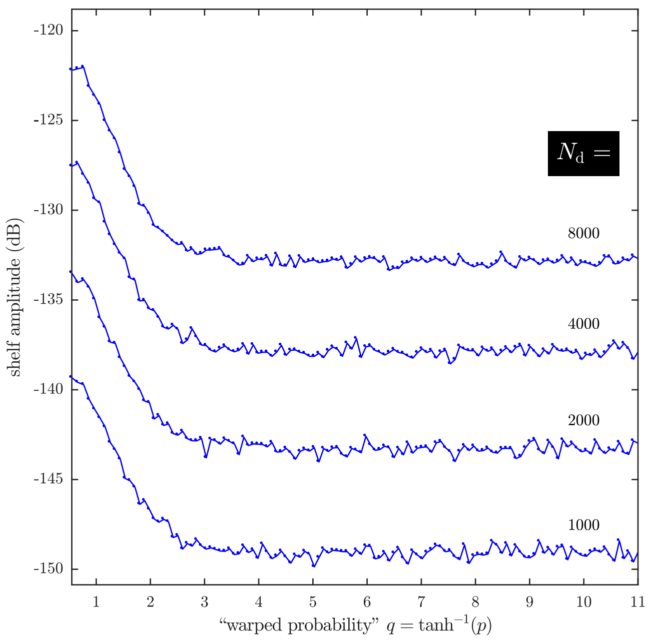


Figure 8: Low shelf amplitude of CARN at various pulse densities $N_\text{d} \in \{1000, 2000, 4000, 8000\}$.

3.3. Crushed Totally Random Noise (CTRN)

Crushed Totally Random Noise (CTRN) is defined by

$$s_\text{trn}(n) = c(r_2(n), p) \cdot \left| \left\| \frac{T_\text{d}}{T_\text{d} - 1} \left(r_1(n) - \frac{1}{2} \right) \right\| \right|\,, \tag{18}$$

where $|\cdot|$ represents the absolute value function, which is identical to traditional TRN (7), except that the polarity is defined by the new function (9) and another random noise sequence $r_2(n)$ rather than the rounding function $\|\cdot\|$.

Five examples of CTRN with pulse density $N_\text{d} = 2000$ and different polarity probabilities $p \in \{0.5, 0.625, 0.75, 0.875, 1.0\}$

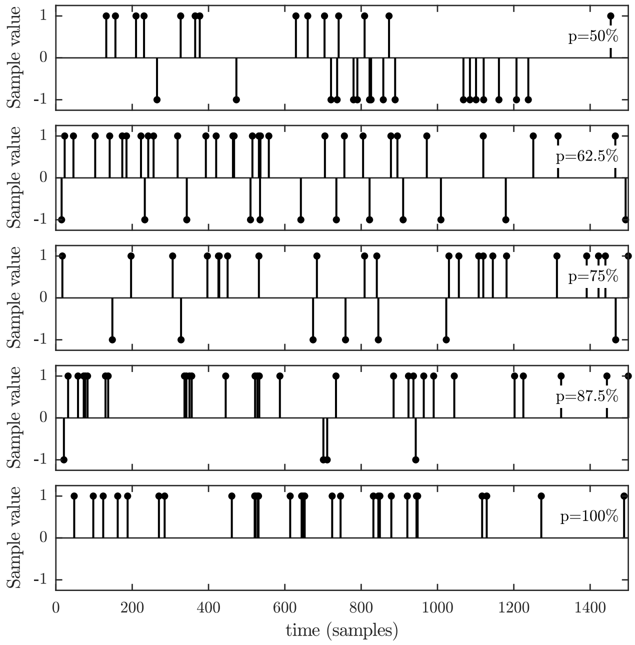


Figure 9: Crushed Totally Random Noise (CTRN), with various probability percentages $p \in \{50\%, 62.5\%, 75\%, 87.5\%, 100\%\}$.

are shown in Fig. 9. Notice that $p = 0.5$ is identical to traditional TRN, and that $p = 1.0$ is fully unipolar—it has only 0s and $+1$s, and no -1s.

PSD estimates of CTRN sequences with pulse densities $N_d \in \{1000, 2000, 4000, 8000\}$ are shown in Fig. 10. The polarity probabilities which are plotted are the same as for COVN. Unlike COVN and CARN, the PSDs for CTRN seem not to depend on p hardly at all—rather then are all entirely flat, just like traditional TRN.

The dc offset considerations for COVN and CARN apply identically to CTRN.

4. APPLICATIONS TO 1-BIT MUSIC

1-bit music is a style of electronic music production where only waveforms composed of 0s and 1s are allowed. To put it mathematically, an M-second-long 1-bit music composition with a sampling rate of f_s is defined by

$$x(n) \in \{0, 1\}\,, \quad n = 0, 1, 2, \ldots, M \cdot f_s\,. \qquad (19)$$

1-bit music (also called PC "beeper music") has its origins in retro computing platforms like the Apple //e and the Sinclair ZX Spectrum, whose sound systems consisted of a single *digital* CPU pin wired straight to a small speaker or output jack [21, 22, 23]. Conceptually, 1-bit music has some relationship to the idea of composition using "sieves," as developed by Iannis Xenakis [24, 25], digital audio effects and synthesizers based on manipulating binary data [26, 27], and especially to Σ-Δ modulator encoding [28, 29, 30, 31, 32, 33].

Today 1-bit music is still widely created, for instance by musicians such as Richard Hollins (Tufty) [34], utz with his "irrlicht project" [35], Mister Beep [36], and Blake Troise (Protodome). Composer Tristan Perich did a lot for popularizing 1-bit music with his albums "1-bit music" (2004) [37] and the very positively

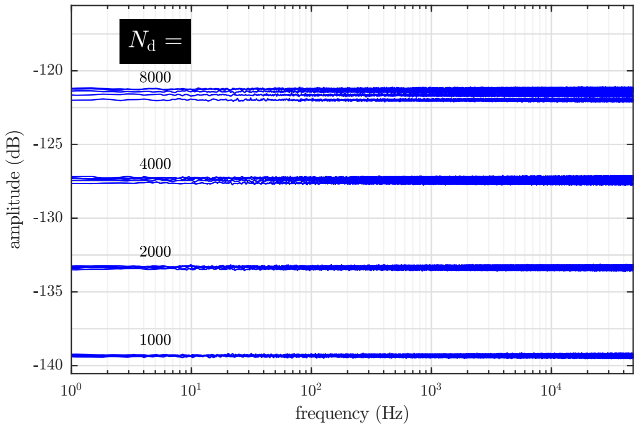


Figure 10: Power Spectral Densities (PSD) of CTRN at various pulse densities $N_d \in \{1000, 2000, 4000, 8000\}$.

reviewed "1-bit Symphony" [38], both of which create stereo 1-bit music from microcontrollers mounted inside of clear compact disc jewel cases. 1-bit music has even made a recent appearance in popular culture, in the ZX Spectrum software NOHZDYVE featured in the film "Bandersnatch" [39].

At first, it may seem that 1-bit music should have a very limited palette, perhaps consisting solely of square waves, pulse waves, impulse trains, and 1-bit noise. But in fact there are a wide variety of timbres that can be made in the 1-bit domain. Composers have managed to come up with a wide variety of techniques and some synthesisers use oscillators that essentially produce 1-bit signals. For example we can mention the Television Interface Adaptor (TIA) chip from Ataric VCS [21], which produces sounds by logical operations on binary signals, and the Korg MS-20 and ARP Odyssey which use "ring modulators" which are actually XOR chips operating on two square waves. The same technique of "ring-modulating" pulse waves is also used in "metallic noise" generators found in some analog drum machines [40, 41], for instance the Korg KR-55. Some old video game systems produce 1-bit noise, for example the Nintendo NES (Nintendo Entertainment System) and Commodore 64, which produce 1-bit noise using Linear Feedback Shift Register (LFSR) circuits [21].

Nonetheless, composing in the 1-bit domain is still incredibly difficult. Signal addition does not in general exist, which means linear operations like mixing and attenuating signals, filtering, etc. are difficult or impossible to achieve in general and must be accomplished through clever composition, construction of suitable signals to fake them, or other clever means. Here I'll explain how sparse noise sequences, including the unipolar ($p = 1.0$) version of the proposed family of Crushed Velvet Noise sequences, can be particularly useful in the 1-bit music context, and can actually solve a few of these issues.

4.1. (Noise) Problems in 1-bit Music

One of the biggest challenges in 1-bit music is combining multiple signals. Traditionally, this was handled for tonal sounds by using logical operations (e.g. XOR) to combine multiple streams of pulse waves with very narrow pulses (down to a single impulse, i.e. to an impulse train) [42]. Since the pulses would only rarely line up, XOR acts essentially like addition, and you can get the sound of signals added together without technically being able to perform addition on binary signals in general.

This technique could not be used so well with 1-bit noise, which is not sparse. So, for noise, this problem would be usually handled on the compositional level, leveraging masking phenomenon in human hearing. For instance, it is possible to overwrite very short 1-bit noise bursts onto another 1-bit musical signal without producing a perceived interruption, giving the illusion of two simultaneous musical lines[3]. If the interruption is short enough, the illusion can be very effective. However, this technique is obviously limited to short noise bursts and is no help at all for layering longer noisy sounds.

Another issue with noise in 1-bit music is that its spectrum is not easy to control. Setting each sample to 0 or 1 randomly produces a flat spectrum. This has its uses, but is timbrally limiting. Since LTI signal processing essentially does not exist in the 1-bit domain, it is difficult to filter a white signal like this further, as you would in traditional sound synthesis. However, it is possible to produce a somewhat passable imitation of lowpass-filtered noise by downsampling a 1-bit noise signal. None of these signals will be sparse in time, making them hard to layer with other signals. Furthermore, it's not obvious how to produce an approximation of noise with a highpass quality.

Finally, the lack of any waveform levels beyond 0 and 1 means that there is not usually a good way to control the perceived amplitude of a 1-bit signal. It turns out that this is an issue with many different 1-bit signals, not only noise. For pulse waves, it can be possible to trick your ear into hearing pulse width modulation towards a pulse train as a level shift, but this also introduces a timbre shift and does not generalize well.

4.2. Solution Using Crushed Velvet Noise

These limitations of noise in 1-bit music—limitations on layering with noise, noise spectra, and noise amplitude—can all be addressed somewhat using various Crushed Velvet Noise sequences.

In general, the sparsity of CVN makes it relatively easy to layer with other 1-bit signals using logical operations such as XOR ($\oplus$). This works by the same mechanism as for impulse trains—since the impulses only happen very rarely, they aren't likely to line up with ones from another stream. The $1 \oplus 1 = 0$ case of XOR rarely occurs, so it largely sounds like addition. This is true for all three proposed varieties of CVN: COVN, CARN, and CTRN.

Unlike normal binary noise, two varieties of CVN—COVN and CARN—have controllable spectra. By changing the pulse density N_d, the cutoff frequency of the two types of shelf-filtered noise can be produced. Here the cutoff is linked inextricably to the density of the pulses, hence the signal energy, hence the perceived volume of the signal, which means that the volume of the noise will also increase with increasing pulse density.

That property could be used on its own as a simulacra of volume control. Or, if it was desired to adjust the CVN level independent of the noise spectrum, CTRN signals could be used. CTRN sequences remain largely white regardless of the pulse density. So, they can be used to control the perceived volume of the signal independent of its spectrum, so long as you are happy with the noise spectrum being white.

One limitation of these techniques is that Crushed Velvet Noise sequences cease to sound "smooth" when their densities are too low, presumably similar to the pulse density that is required for standard Velvet Noise. In, e.g., reverb modeling or decorrelation filter design, this would normally be seen as a limitation of Velvet Noise. However, in the context of 1-bit noise where everything sounds a bit raw and grainy, it could perhaps be considered a charming and characteristic quirk. One interesting effect is "sweeping" the pulse density of a COVN sequence downwards. Starting from a high pulse density, this will sound like a shelf filter sweep. However, as the pulse density gets low enough, the "smoothness" dissapears and it sounds like the signal "disintegrating."

An example of using COVN in the context of 1-bit composition can be heard in the author's cover [44] of "Unholy Captives" from the video game "Return of the Obra Din" by Lucas Pope [45].

5. CONCLUSION

In this paper, new generalizations of Velvet Noise sequences were proposed. These "Crushed" Velvet Noise sequences, which open up the probability of an impulse taking a positive or negative polarity as a free parameter, can be used to make colored noise spectra, as opposed to the white spectra of traditional Velvet Noise. Crushed Velvet Noise (specifically the unipolar case) is a suitable sequence for use in *1-bit music*, alongside classic binary noise sequences like Linear Feedback Shift Register (LFSR) noise. Especially Crushed Original Velvet Noise and Crushed Totally Random Noise should be considered very useful in 1-bit music composition, since their sparsity characteristics make them very easy to mix with other 1-bit signals using logical operations like XOR.

Crushed Velvet Noise leaves open a lot of scope for future work. We've seen in this paper that certain CVN sequences (COVN and CARN) have PSDs that resemble low-shelf-filtered white noise, while others (CTRN) have flat spectra just like their traditional versions. Perhaps future mathematical analysis can reveal how Velvet Noise manages to have a white spectrum without being i.i.d. Future work could also consider trying to create other standard colored noise spectra—e.g. lowpass, bandpass, notch, and high-shelf—from sparse ternary noise. In this paper, only the $\Delta = 1.0$ case of CARN was considered. Future work could consider the more general case of CARN to see if it has any interesting properties, as well as a "Crushed" variant of the related "Extended Velvet Noise" (EVN) [2]. Although the PSD estimates shown in this paper can be taken as very good estimates of the true, underlying PSD of the studied noise sequences, on account of their very long length (24 hours!), we don't need sequences nearly that long to hear the spectral character of different noise sequences. Your ear can guide you; you should find that the spectral character of each noise sequence reveals itself even with very short bursts of Velvet Noise. However, it would be interesting for future work to test our perception of short Velvet Noise bursts.

Finally, it seems clear that 1-bit synthesis, mixing, and audio effects have a close relationship to signal processing of Σ-Δ bit streams [28, 29, 30, 31, 32, 33]. Hopefully future work can gain new insights from that literature.

6. ACKNOWLEDGMENTS

Thank you to Elliot Kermit Canfield-Dafilou and Ólafur Bogason for helpful conversations and to the reviewers for their careful editing and suggestions. Thank you to Fabián Esqueda for bringing the Korg KR-55's metallic noise circuit to my attention.

[3] A phenomenon that is well-known in psychoacoustics, e.g. [43].

7. REFERENCES

[1] M. Karjalainen and H. Järveläinen, "Reverberation modeling using velvet noise," in *Proc. 30th Int. Conf. Audio Eng. Soc. (AES)*, Saariselkä, Finland, Mar. 15–17 2007.

[2] V. Välimäki, H.-M. Lehtonen, and M. Takanen, "A perceptual study on velvet noise and its variants at different pulse densities," *IEEE Trans. Audio, Speech, Language Process.*, vol. 21, no. 7, pp. 1481–1488, July 2013.

[3] L. Schreiber, "Was empfinden wir als gleichförmiges Rauschen?," *Frequenz*, vol. 14, no. 12, pp. 399–403, Dec. 1960.

[4] J. Borish, "An efficient algorithm for generating colored noise using a pseudorandom sequence," *J. Audio Eng. Soc. (JAES)*, vol. 33, no. 3, pp. 141–144, Mar. 1985.

[5] B. Holm-Rasmussen, H.-M. Lehtonen, and V. Välimäki, "A new reverberator based on variable sparsity convolution," in *Proc. 16th Int. Conf. Digital Audio Effects (DAFx-13)*, Maynooth, Ireland, Sept. 2–5 2013.

[6] K. S. Lee, J. S. Abel, V. Välimäki, T. Stilson, and D. P. Berners, "The switched convolution reverberator," *J. Audio Eng. Soc. (JAES)*, vol. 60, no. 4, pp. 227–236, Apr. 2012.

[7] V. Välimäki, J. D. Parker, L. Savioja, J. O. Smith, and J. S. Abel, "Fifty years of artificial reverberation," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 5, pp. 1421–1448, July 2012.

[8] V. Välimäki, J. D. Parker, J. O. Smith, and J. S. Abel, "More than fifty years of artificial reverberation," in *Proc. 60th Int. Conf. Audio Eng. Soc. (AES)*, Leuven, Belgium, Feb. 3–5 2016.

[9] V. Välimaki, B. Holm-Rasmussen, B. Alary, and H.-M. Lehtonen, "Late reverberation synthesis using filtered velvet noise," *Appl. Sci.*, vol. 7, no. 5, 2017, Article #483.

[10] B. Alary, A. Politis, and V. Välimäki, "Velvet-noise decorrelator," in *Proc. 20th Int. Conf. Digital Audio Effects (DAFx-17)*, Edinburgh, UK, Sept. 5–9 2017, pp. 405–411.

[11] S. J. Schlecht, B. Alary, V. Välimäki, and E. A. P. Habets, "Optimized velvet-noise decorrelator," in *Proc. 21st Int. Conf. Digital Audio Effects (DAFx-18)*, Aveiro, Portugal, Sept. 4–8 2018.

[12] H. Kawahara, K.-I. Sakakibara, M. Morise, H. Banno, T. Toda, and T. Irino, "Frequency domain variants of velvet noise and their application to speech processing and synthesis," in *Proc. Interspeech*, Hyderbad, India, Sept. 2–6 2018, pp. 2027–2031.

[13] H. Kawahara, "Application of velvet noise and its variants for synthetic speech and singing," Tech. Rep., IPSJ SIG Tech. Rep., 2018.

[14] V. Välimäki, J. Rämö, and F. Esqueda, "Creating endless sounds," in *Proc. 21st Int. Conf. Digital Audio Effects (DAFx-18)*, Aveiro, Portugal, Sept. 4–8 2018, pp. 32–39.

[15] S. D'Angelo and L. Gabrielli, "Efficient signal extrapolation by granulation and convolution with velvet noise," in *Proc. 21st Int. Conf. Digital Audio Effects (DAFx-18)*, Aveiro, Portugal, Sept. 4–8 2018, pp. 107–112.

[16] P. Rubak and L. G. Johansen, "Artificial reverberation based on a pseudo-random impulse response: part I," in *Proc. 104th Conv. Audio Eng. Soc. (AES)*, Amsterdam, the Netherlands, May 16–19 1998, Conv. paper #4725.

[17] P. Rubak and L. G. Johansen, "Artificial reverberation based on a pseudo-random impulse response ii," in *Proc. 106th Conv. Audio Eng. Soc. (AES)*, Munich, Germany, May 8–11 1999, Conv. paper #4900.

[18] f. harris and E. Brooking, "A versatile parametric filter using an imbedded all-pass sub-filter to independently adjust bandwidth, center frequency, and boost or cut," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, Oct. 26–28 1992, pp. 269–273.

[19] M. Holters and U. Zölzer, "Parametric higher-order shelving filters," in *Proc. 14th European Signal Process. Conf. (EUSIPCO)*, Florence, Italy, Sept. 2006.

[20] V. Välimäki and J. D. Reiss, "All about audio equalization: Solutions and frontiers," *Appl. Sci.*, vol. 6, no. 5, 2016, Article #129.

[21] K. B. McAlpine, *Bits and Pieces: A History of Chiptunes*, Oxford Univ. Press, New York, NY, 2019.

[22] K. B. McAlpine, "The sound of 1-bit: Technical constraint and musical creativity on the 48k Sinclair ZX Spectrum," *GAME: Italian J. Game Studies*, vol. 6, no. 1, 2017.

[23] S. N. Goodwin, *Beep to Boom: The Development of Advanced Runtime Sound Systems for Games and Extended Reality*, Focal Press, New York, NY, 2019.

[24] V. Adán, *Discrete time 1-bit music: Foundations and models*, D.M.A. thesis, Columbia Univ., New York, USA, 2010.

[25] C. Ariza, "The Xenakis sieve as object: A new model and a complete implementation," *Comput. Music. J. (CMJ)*, vol. 29, no. 2, pp. 40–60, Summer 2005.

[26] K. J. Werner and M. Sanganeria, "Bit bending: An introduction," in *Proc. 16th Int. Conf. Digital Audio Effects (DAFx-13)*, Maynooth, Ireland, Sept. 2–5 2013.

[27] J. Kleimola, "Audio synthesis by bitwise logical modulation," in *Proc. 11th Int. Conf. Digital Audio Effects (DAFx-08)*, Espoo, Finland, Sept. 1–4 2008, pp. 67–70.

[28] N. M. Casey and J. A. S. Angus, "One bit digital processing of audio signals," in *Proc. 95th Conv. Audio Eng. Soc. (AES)*, New York, USA, Oct. 7–10 1993, Conv. paper #3717.

[29] J. A. S. Angus, "The one-bit alternative for audio processing and mastering," in *Proc. 9th UK Conf. Audio Eng. Soc. (AES)*, York, UK, May 1994.

[30] J. A. S. Angus and N. M. Casey, "Filtering Σ–Δ audio signals directly," in *Proc. 102nd Conv. Audio Eng. Soc. (AES)*, Munich, Germany, Mar. 22–25 1997, Conv. paper #4445.

[31] J. A. S. Angus and S. Draper, "An improved method for directly filtering Σ–Δ audio signals," in *Proc. 104th Conv. Audio Eng. Soc. (AES)*, Amsterdam, the Netherlands, May 16–19 1998, Conv. paper #4737.

[32] J. A. S. Angus, "Direct digital processing "Super Audio CD" signals," in *Proc. 108th Conv. Audio Eng. Soc. (AES)*, Paris, France, Feb. 19–22 2000, Conv. paper #5102.

[33] J. Reiss and M. Sandler, "Digital audio effects applied directly on a DSD bitstream," in *Proc. 7th Int. Conf. Digital Audio Effects (DAFx-4)*, Naples, Italy, Oct. 5–8 2004.

[34] Tufty, "1-bit mechanistic," Compact Disc, 2014.

[35] irrlicht project, "dat fuzz," Compact Disc, 2004.

[36] Mister Beep, "A thousand furious bees," Album, 2011, COUCOU Micromusic Netlabel.

[37] T. Perich, "1-bit music," Circuit in Jewel Case, 2004, Cantaloupe Music.

[38] T. Perich, "1-bit symphony," Circuit in Jewel Case, 2010, Cantaloupe Music.

[39] "Black mirror: Bandersnatch," Film, 2018, Netflix.

[40] K. J. Werner, J. S. Abel, and J. O. Smith, "The TR-808 cymbal: a physically-informed, circuit-bendable, digital model," in *Proc. 40th Int. Comput. Music Conf. (ICMC) / 11th Sound Music Comput. Conf. (SMC)*, Athens, Greece, Sept. 14–20 2014, pp. 1453–1460.

[41] K. J. Werner, J. S. Abel, and J. O. Smith, "More cowbell: a physically-informed, circuit-bendable, digital model of the TR-808 cowbell," in *Proc. 137th Conv. Audio Eng. Soc. (AES)*, Los Angeles, CA, Oct. 9–12 2014, Conv. paper #9207.

[42] K. B. McAlpine, "All aboard the impulse train: An analysis of the two-channel title music routine in Manic Miner," *Comput. Games J.*, vol. 4, no. 3–4, pp. 155–168, Dec. 2015.

[43] G. A. Miller and J. C. R. Licklider, "The intelligibility of interrupted speech," *J. Acoust. Soc. Am. (JASA)*, vol. 22, no. 2, pp. 167–173, Mar. 1950.

[44] K. J. Werner, " 'Unholy Captives' from 'Return of the Obra Dinn' (Lucas Pope, 2018): a 1-bit cover/tribute," Soundcloud, 2019, `https://soundcloud.com/kurt-james-werner/unholy-captives-from-return-of-the-obra-dinn-1-bit-covertribute`.

[45] L. Pope, "Return of the Obra Dinn," Video Game, 2018.