Virtual Analog Character Compressor Design

by

Zhe Deng

Submitted in Partial Fulfillment of the

Requirements for the Degree

Master of Science

Supervised by Professor Ming-Lun Lee

Department of Electrical and Computer Engineering

Arts, Sciences and Engineering

Edmund A. Hajim School of Engineering and Applied Sciences

University of Rochester

Rochester, New York

2021

Table of Contents

Biographical Sketch

Zhe Deng was born in Wuhan, China, but spent most of his life in Dayton, Ohio. He attended Boston University and graduated with a Bachelor of Science degree in Computer Engineering (Cum Laude). Zhe began his Master's studies at the University of Rochester in 2020, pursuing Electrical Engineering with a concentration in musical acoustics and signal processing under the supervision of Professor Ming-Lun Lee. His research interests lie in nonlinear musical filter design.

Acknowledgments

I would like to acknowledge Professor Ming-Lun Lee for guiding me through my research and thesis. I also thank Professor Mark F. Bocko and Professor Scott H. Seidman for their involvement with the thesis committee. My gratitude extends to the staff of the University of Rochester for providing education under extraordinary circumstances. Furthermore, I thank the people of Rochester for teaching me life lessons outside of school. Finally, I would like to thank my family and friends for their financial and emotional support throughout my time at university.

Abstract

The character compressor is a ubiquitous audio effect archetype that reduces the dynamic range of signals. Many sensitive electronics such as tape recorders use compressors before their inputs to ensure that signals are above the noise floor and below the threshold for distortion. However, character compressors add noise and distortion known as ‘character.’ Their circuits consist of vintage, low-headroom components (e.g., vacuum tubes, transformers, etc.) that add musical imperfections to audio signals.

The demand for character compressors combined with their need for expensive and often unreliable components motivates a digital alternative. Furthermore, modeling the underlying mathematics of these circuits encourages novel designs. This is the philosophy of the ‘topology-preserving transform’ (TPT). In this thesis, filters are designed as continuous-time block diagrams and then discretized using minimal substitutions to preserve their intended response.

The implementation of the character compressor begins with an analysis of the compressor section. The description of the classical compressor covers a basic scheme for gain reduction on which more complicated models are based. In addition, common modifications enhance different blocks of the block diagram.

The character section is based on the Jiles-Atherton (JA) model of magnetic hysteresis. The proposed novel feedforward JA model modulates the cutoff frequency of a first-order TPT filter to simulate the effect of nonlinear passive components. Simplifications to the original JA model enable an analytical solution with guaranteed convergence.

The final model is optimized for musical digital signal processing (DSP). The previous abstractions yield independent parameters with audible results. Furthermore, the algorithm does not use iterative methods because the feedforward design bypasses nonlinear implicit equations. Experiments in the SOUL programming language verify these phenomena and introduce applications for further research.

Contributors and Funding Sources

This work was supported by a thesis committee consisting of Professor Ming-Lun Lee and Professor Mark F. Bocko at the Department of Electrical and Computer Engineering and Professor Scott H. Seidman at the Department of Biomedical Engineering. All work conducted for the thesis was completed by the student independently without outside funding support.

List of Figures

List of Symbols

$\int_{t_0}^{t} x(\tau)\, d\tau \Leftrightarrow \int x$ Implied time integral

$\frac{d}{dt}x \Leftrightarrow \dot{x}$ Newton's time derivative notation

$\int x \Leftrightarrow \frac{1}{s} \cdot x$ Laplace pair

$\dot{x} \Leftrightarrow s \cdot x$ Laplace pair

$\dot{x} \Rightarrow x = x(t)$ Implied time series

$s \cdot x \Rightarrow x = X(s)e^{st}$ Implied complex exponential

$H(s) \Leftrightarrow \frac{y}{x}$ Definition of transfer function

$x[0], x[1], \ldots, x[n-1], x[n] \Leftrightarrow x[0{:}\,n]$ Set of samples

Chapter 1

Introduction

Raw recordings typically have erratic dynamic ranges.[1] For example, a guitar pluck may fade out too quickly and become lost in a song with other instruments and vocals. To increase guitar sound clarity, audio engineers use compressors to decrease the volume difference between the initial pluck transients and the sustained string vibrations.

Dynamic processors use gain modulation to apply a dynamic volume. The time-varying gain, V_c(t), depends on the current and past values of a sidechain input, $\varsigma(t)$ (i.e., The system has memory.).[2]

$$y(t) = V_c\big(\varsigma(t)\big) \cdot x(t) \tag{1.1}$$

The main processing occurs in the sidechain as it maps $\varsigma(t)$ to V_c(t) (Figure 1.1).

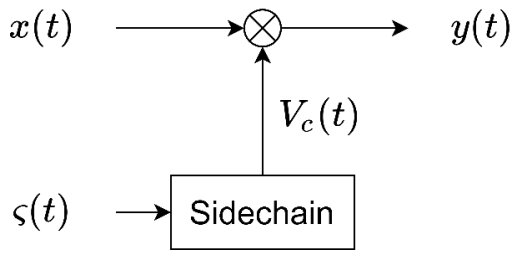

Figure 1.1: Generic dynamics processor block diagram

[1] Dynamic range – the ratio between the maximum and minimum sound levels.

[2] A common circuit realization uses voltage-controlled amplifiers (VCAs) to apply $V_c(t)$. VCAs have two inputs: an input signal, $x(t)$, and a 'control voltage,' $V_c(t)$, which controls the gain.

According to this definition, the system is nonlinear and possibly time-varying, depending on the sidechain input.

A classical compressor is a specific type of dynamics processor with a three-stage sidechain (Figure 1.2). The detector first rectifies $\varsigma(t)$ by mapping negative values to positive (or mostly positive) values. Unless otherwise stated, $\varsigma(t)$ is the input, $x(t)$, but compressors can also operate in feedback mode ($\varsigma(t)$ is the output, $y(t)$) or sidechain mode ($\varsigma(t)$ is an external signal, $\eta(t)$) [1]. After the detector, an envelope filter smooths the rectified signal. The envelope filter is usually a special type of low-pass filter called a 'ballistics filter,' but any filter with a similar smoothing property is sufficient. This condition implies that the envelope filter must have memory. Lastly, the compressor transfer function (CTF) maps the smoothed, rectified sidechain input to $V_c(t)$.[3] The value of $V_c(t)$ lies between 0 to 1 and applies compression via gain reduction [2].

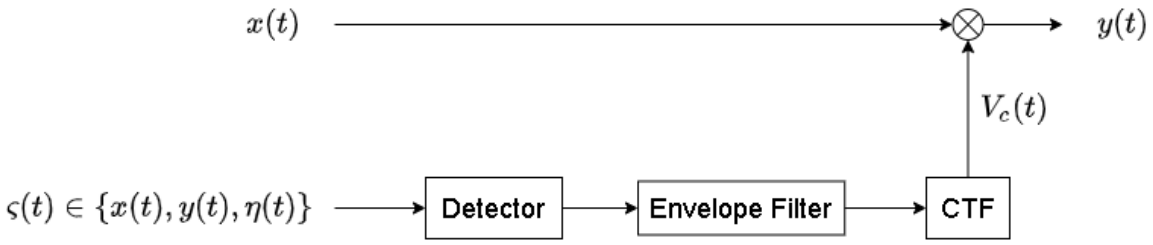

Figure 1.2: Generic compressor block diagram

While the previous definition defines most digital compressors, analog compressors contain a multitude of nonlinear components: vacuum tubes, transistors, diodes, etc. [3].[4] This

[3] The compressor transfer function is simply a static mapping (i.e., a function); not to be confused with the Laplace transform.

[4] Except for the term 'virtual analog,' 'analog' refers to a circuit and not continuous-time equations.

thesis focuses on nonlinear passive components (i.e., resistors, capacitors, and inductors), which are directly modeled by parameter modulation. Filter circuits with nonlinear passive components have different frequency responses depending on the system memory; this effect can be described by filter parameters that change per sample (modulate) to reproduce a desired behavior. These components appear in the envelope filter and increase or decrease the smoothing of $V_c(t)$ depending on the amplitude of $\varsigma(t)$ (Figure 1.3).

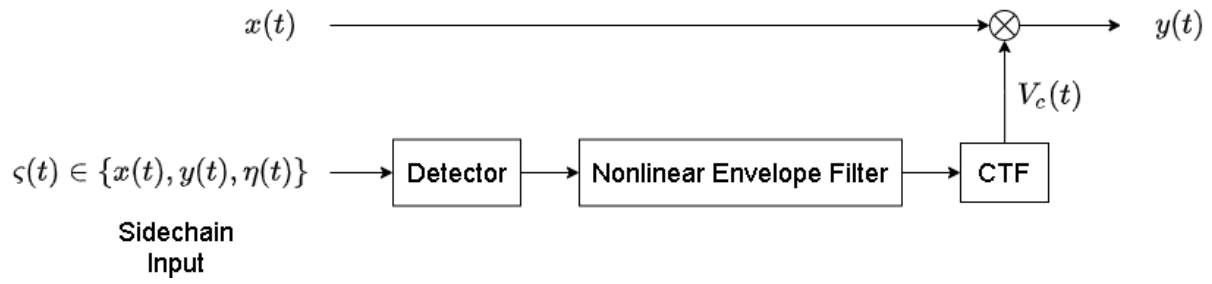

Figure 1.3: Block diagram of a generic compressor with a nonlinear envelope filter

Reusing the math of the nonlinear envelope filter as an input filter transforms Figure 1.3 into the character compressor in Figure 1.4. The character compressor adds frequency content via frequency nonlinearities regardless of the sidechain's parameters. Audio engineers call this phenomenon 'character' because it is intrinsically tied to the effect.

The character block is before $V_c(t)$ (Figure 1.4), but no specific position is required (in fact, character may exist in multiple locations). Different connections to the character block's input and output add another layer of expression for effect designers.

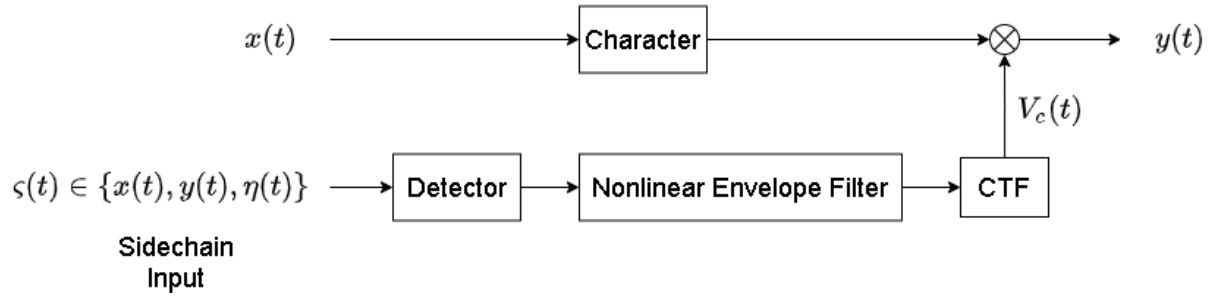

Figure 1.4: Block diagram of a generic character compressor with a nonlinear envelope filter

1.1 Review of Existing Models

Because literature about compressors is scarce, experiments on existing software provide a better understanding of key system specifications. This section uses three popular compressor plugins as case studies.

First, a classical compressor demonstrates the basic parameters. Second, a slightly more complex 'colored' compressor expands on the classical design with a modified sidechain. Third, an analog modeling plugin exemplifies character and envelope filter nonlinearities.

The experimental setup uses the Waveform digital audio workstation (DAW) to pass audio through each audio plugin [4]. For user, plugins are black boxes; the only way to alter the sound is to change the parameters via the user interface (UI). Thus, the best way to evaluate each plugin is to measure each plugin's responses to different input and parameter combinations.

Three different input signals are used to test different aspects of each plugin: a decibel (dB) scale ramp, a pulse, and a 1kHz sine wave.[5] The ramp tests the response of the CTF's gain

[5] All audio signals are 96kHz, 24-bit lossless wav files.

reduction. It has a duration of two seconds and increases linearly (on the dB scale) from -60 dB to 0 dB (Figure 1.5).

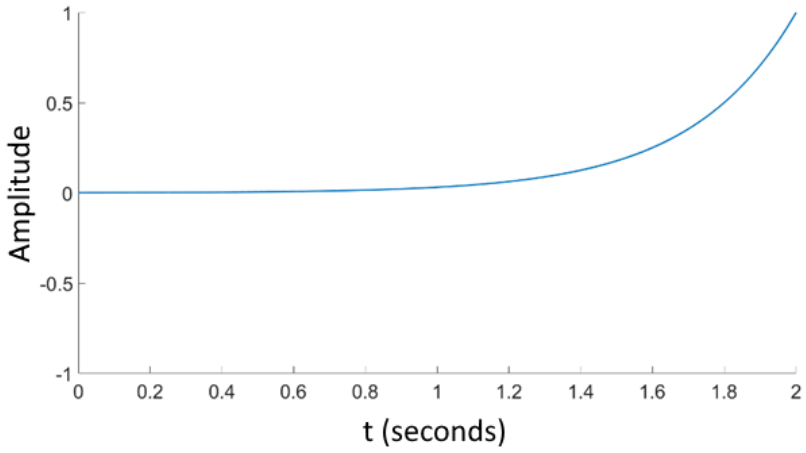

Figure 1.5: Ramp

The pulse has a value of 1 for 0.5 seconds and 0.5 for 0.5 seconds (Figure 1.6). The response to this signal demonstrates the transient response of the envelope filter as it smooths $V_c(t)$.

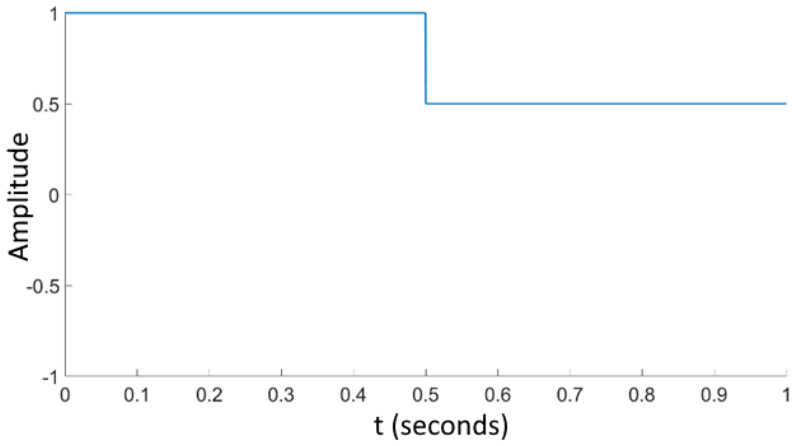

Figure 1.6: Pulse

Lastly, the sine wave is a simple test for frequency nonlinearity (Figure 1.7). The signal has a frequency of 1 kHz and an amplitude of 0.89 or -1 dB; it passes through plugins with the sidechain bypassed. If the spectrum of the output contains peaks other than the fundamental frequency, the effect is a character compressor.

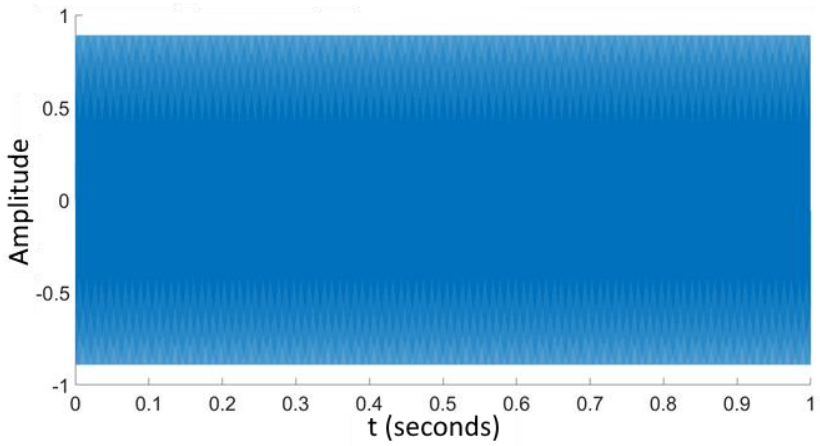

Figure 1.7: Sine

1.1.1 A Classical Compressor

Waveform's default compressor (Figure 1.8) is a classical compressor because it only provides these four main parameters: attack, release, threshold, and ratio. All compressors must demonstrate the effect of these four parameters, regardless if they are visible on the UI.

The ramp measures the 'threshold' and 'ratio' settings in the CTF. Below the threshold, $V_c(t) = 1$; above the threshold, $V_c(t) < 1$. The speed at which $V_c(t)$ approaches 0 is set by the ratio. The higher the ratio, the faster $V_c(t)$ decreases for each additional decibel over the threshold.

The attack and release settings are set to the lowest values to mitigate their effects. Using these settings, the compressor approximates a memoryless function or ‘waveshaper’; the output shows the effect of the ratio and threshold.

Figure 1.8: Waveform compressor parameters for the ramp input

The input-output plot is consistent with the threshold and ratio parameters (Figure 1.9). Some smoothing still exists because the CTF is curved past the threshold, but for each 5 dB past the threshold, the output amplitude increases by approximately 1 dB (i.e., the reciprocal of the ratio). Below the threshold, the amplitude remains the same.

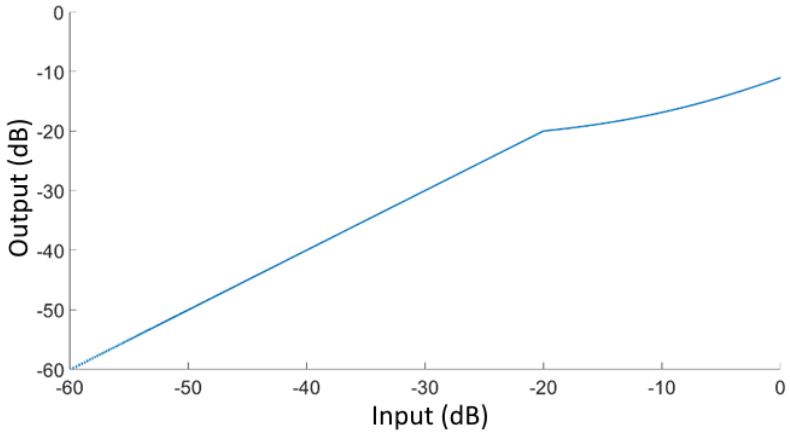

Figure 1.9: Waveform compressor output plotted against the ramp input (dB scale)

The pulse stimulates the ‘attack’ and ‘release’ responses of the envelope filter (Figure 1.10). Both the attack and release are 100 milliseconds and noticeable in the output. In addition, the threshold is -2.5 dB or about 0.75 (the average value of the pulse signal from $t = 0$ to $t = 1$).

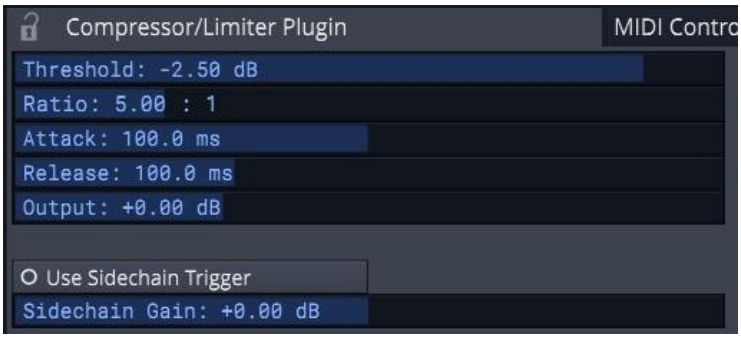

Figure 1.10: Waveform compressor parameters for the pulse input

Attack is the amount of smoothing for rising signals (after rectification) and release is the amount of smoothing for falling signals. The result (Figure 1.11) shows that a slow attack has a moment of zero gain reduction when the system is initially excited. Furthermore, a slow release produces an overshoot in gain reduction when the signal first falls below the threshold.

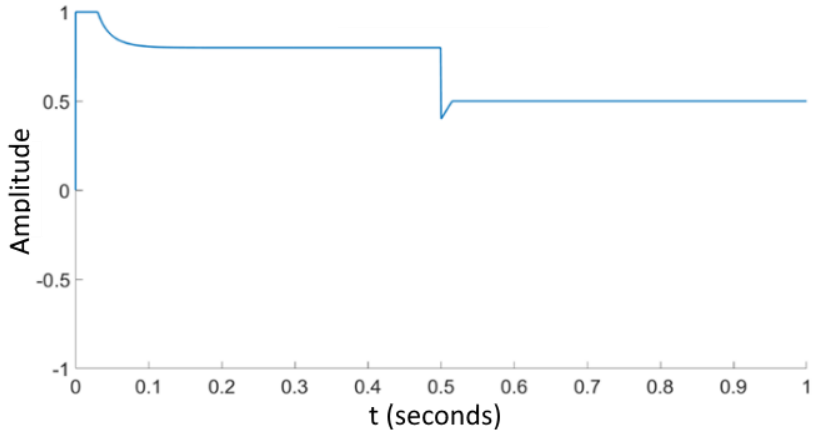

Figure 1.11: Waveform compressor’s response to the pulse input

Lastly, the compression (sidechain) is bypassed in the sine test by setting the threshold to zero and $\varsigma(t)$ to a nonexistent external source (i.e., $\varsigma(t) = 0$) (Figure 1.12).[6] Because this plugin is not an analog modeling compressor, the output should have the same frequency content as the input.

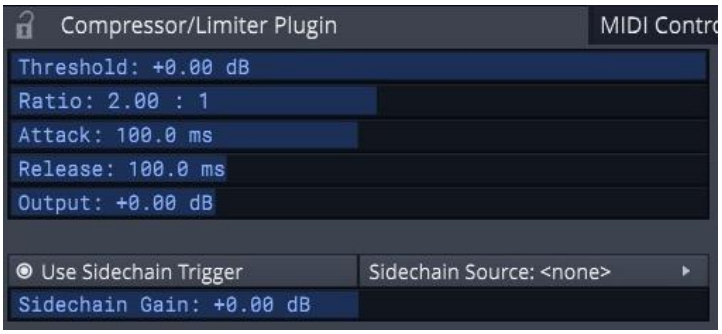

Figure 1.12: Waveform compressor parameters for the sine input

Because the output (Figure 1.13) only shows the fundamental frequency, the Waveform compressor is not a character compressor.

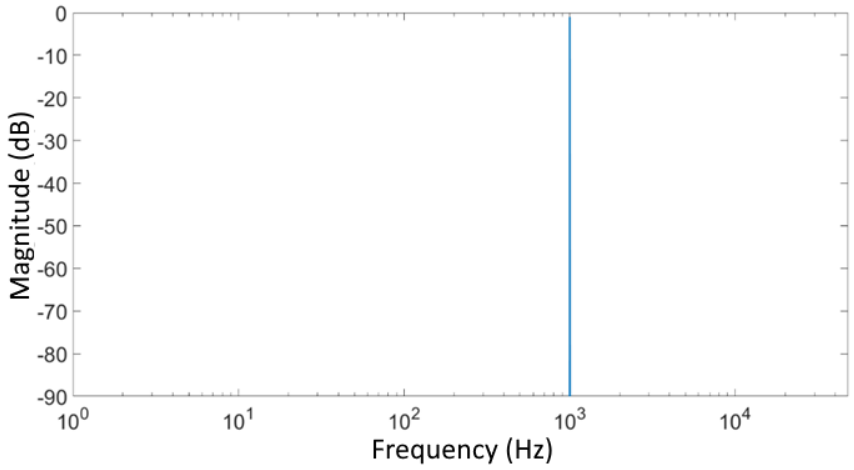

Figure 1.13: Waveform compressor output spectrum for the sine input

[6] Linearity is not guaranteed with compression. Fast attacks and releases also generate harmonics.

1.1.2 A Modified Transfer Function

The ATK Colored Compressor (Figure 1.14) defines ‘color’ as the height of the overshoot in the hard-knee (threshold) of the CTF.[7] In addition, the ‘quality’ knob sets the width of this overshoot, like the quality factor in a typical second-order filter [5].

Figure 1.14: ATK Colored Compressor parameters for the ramp input

Because the attack, release, and character are not significantly different compared to the Waveform compressor, only the ramp test is shown (Figure 1.15). With maximum color and quality, the knee has a round bell-shaped bump.

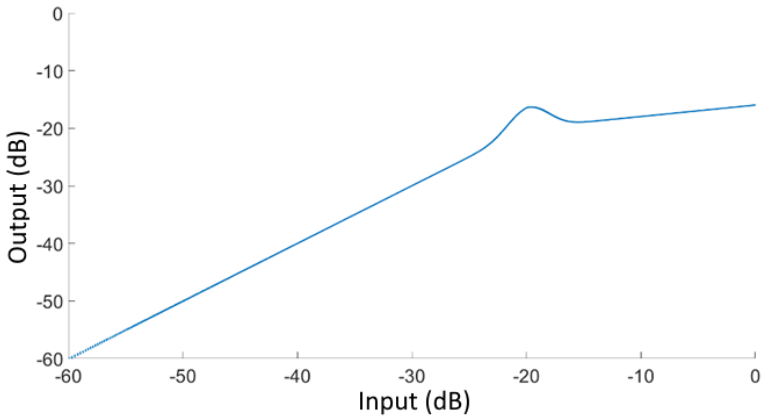

Figure 1.15: ATK Colored Compressor output plotted against the ramp input (dB scale)

[7] In this case, the ‘hard-knee’ is the threshold; ‘hard-knee’ also refers to any sharp or non-differentiable corner in a function.

The Colored Compressor also has two other non-traditional parameters. ‘Softness’ converts the hard-knee into a rounded, differentiable ‘soft-knee’; ‘RMS’ (root mean square) sets the window size of a moving RMS of $x(t)$.[8] The output of the moving RMS connects to $\varsigma(t)$ to create a perception of slowed attack/release times [6, Sec. 2.3.1]. While their mechanisms are new, these two parameters have a similar effect to knee color and varying attack/release times; therefore, they will not be covered in detail.

1.1.3 Dynamic Nonlinearities

The Teletronix Model LA-2A Leveling Amplifier is a well-known (vacuum) tube optical compressor. Input signals increase the brightness of an internal light panel. The panel then shines onto an optical receiver which reduces $V_c(t)$ [7]. ‘LALA’ is a plugin simulation of this analog effect (Figure 1.16) [8].

One of the main features of the LA-2A is the input-dependent attack and release times [9]. To test this property using the pulse test, the peak reduction is set at 50% and all other parameters are left at their default values (Figure 1.16).

Figure 1.16: LALA parameters for the pulse input

[8] Given a set of samples and a window size N, the moving RMS is defined as: $MovingRMS(x[n]) = \sqrt{\frac{\sum_{i=n-N}^{n} x[i]^2}{N}}$

The attack and release apply quickly (Figure 1.17), but a closer look reveals that the step responses are not a simple combination of sinusoidal and exponential terms. Four distinct stages (Figure 1.18) slow down (become less steep) as the input holds its value. Stages one to three and stage four combine to form the LA-2A's signature 'two-stage' smoothing. This behavior does not match the step responses of any first-order or second-order linear filters. Therefore, an accurate model of the envelope filter must contain nonlinearities and/or higher-order filters.

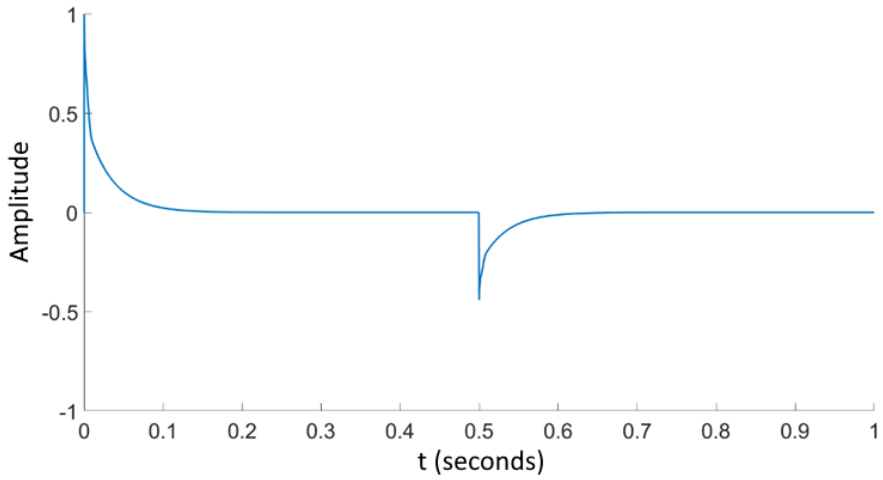

Figure 1.17: LALA's response to the pulse input

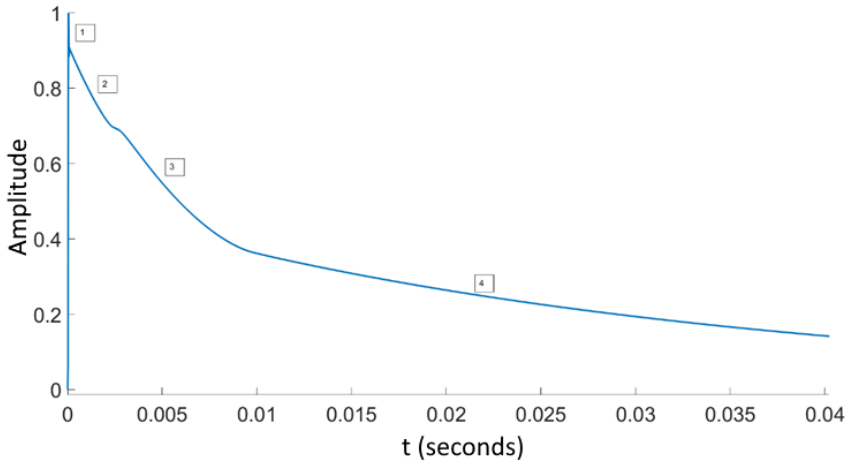

Figure 1.18: Detailed view of the beginning of LALA's pulse response

The input/output transformers, as well as the tubes themselves, are prone to distortion, even with the LA-2A's sidechain bypassed [7]. To bypass the sidechain, LALA's sidechain input is set to 'EXT' (external), and the peak reduction is set to zero (Figure 1.19).

Figure 1.19: LALA parameters for the sine input

The output spectrum of the sine test shows that lower-order even harmonics (2nd and 4th) dominate the harmonic distortion (Figure 1.20). In addition, the character adds low frequency and direct current (DC) components.[9]

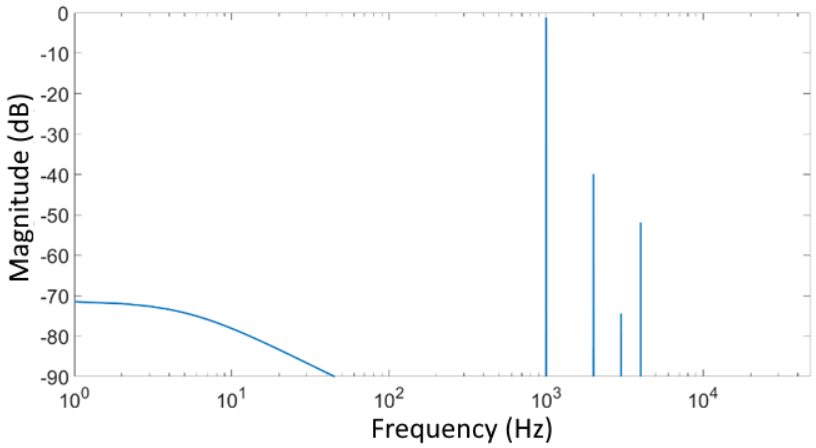

Figure 1.20: LALA output spectrum for the sine input

[9] DC – 0 Hz.

The input-output plot justifies the low frequency content (Figure 1.21). The system's nonlinearity is dynamic, as shown by the non-functional mapping. For each input value, there may exist more than one possible output value depending on the system memory. Furthermore, this mapping does not pass through the origin and maps DC values of zero to non-zero values. This phenomenon is called 'hysteresis' and each instance of a loop in the input-output plot is called a 'hysteresis loop.'

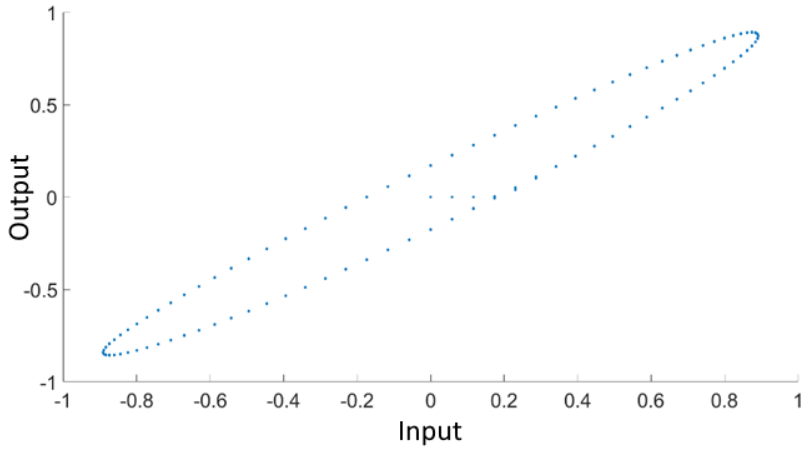

Figure 1.21: LALA output plotted against sine input

Lastly, the four sidechain knobs are unique to LALA. These parameters determine the frequency response of a linear filter between $\varsigma(t)$ and the detector. This filter is commonly used to block excessive low frequency and DC signals.

1.2 Thesis Organization

The remainder of this thesis implements the four main blocks of the character compressor. While the detector and CTF are typically static nonlinearities, the character and envelope filter exhibit dynamic nonlinearities.

Chapter 2 establishes a framework for modeling these dynamic nonlinearities using parameter-modulating virtual analog (VA) filters. Linear filters are defined by a single impulse response; in contrast, parameter-modulating filters have a different impulse response for each instantaneous cutoff value (i.e., not linear). This chapter leads to an important conclusion: classical linear, time-invariant discrete-time filters are not stable under parameter modulation. In *The Art of VA Filter* Design, Vadim Zavalishin proposes a 'topology-preserving transform' (TPT) that achieves stability under cutoff modulation by preserving the topology of continuous-time filters [10].

Chapter 3 begins by defining the equations of the classical compressor. Then, some common sidechain modifications introduce more complicated gain reduction schemes. Specifically, the modifications include input filters on the detector, cutoff-modulating envelope filters, and CTF variants.

Chapter 4 expands the cutoff modulation in Chapter 3 to create a character block. The Jiles-Atherton (JA) model adds hysteresis to the previously anhysteretic cutoff modulation. This study proposes a novel JA model with simplified parameters an analytical solution; the resulting filter is stable and requires less instructions to compute.

Finally, Chapter 5 realizes the effect in the SOUL domain-specific programming language. Tests demonstrate how this model overcomes common challenges related to parameter determination and filter stability. Furthermore, the same set of tests from Chapter 1 prove that the resulting filters can function as a character compressor.

1.3 Contributions

This thesis intends to inspire new ideas related to VA compressor design. Each of the character compressor's blocks follow very loose requirements. For example, any arbitrary mapping can be a CTF. Modifications are another endpoint for further dynamical behavior and musical expression. The proposed JA model is especially flexible; it may apply to the CTF to add hysteresis or combine with higher-order filters for a more complex character block. The SOUL programming language accentuates this flexibility and allows each 'processor' (block) to be used in an entirely different 'graph' (effect). The structure of the character compressor encourages modular design.

Chapter 2

Discretizing Differential Equations

Classical DSP emphasizes filters that preserve stability, causality, time-invariance, and linearity.[10] In contrast, musical filters are often time-varying and nonlinear; a generalized approach is necessary for stability under these conditions.

The filters in Chapters 3 and 4 are nonlinear due to cutoff modulation and may also be time-varying. Zavalishin provides in-depth analysis on topology-preserving transform (TPT) filters that discretize continuous-time nonlinearities with stability [10]. This section reviews relevant topics from [10] and proposes some case-specific enhancements.

2.1 One-Pole Filter

Consider the set of analog first-order passive filters with complex impedances (Figure 2.1):

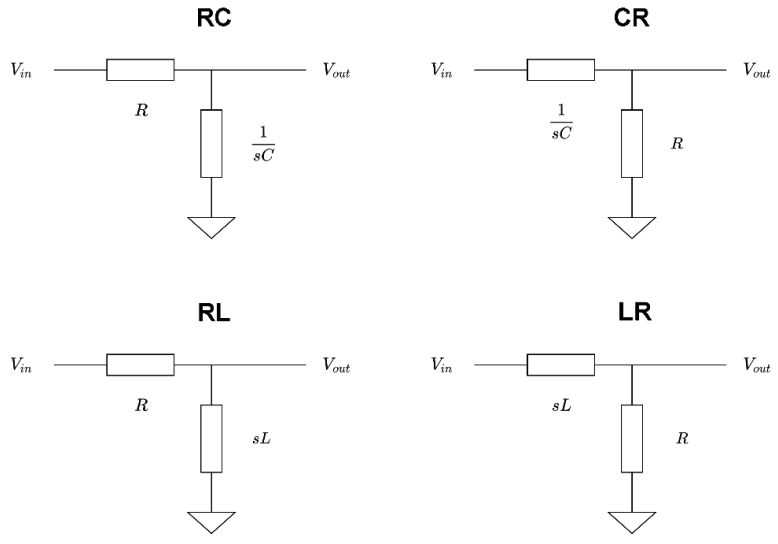

Figure 2.1: Analog first-order passive filters with complex impedances

[10] All filters in this thesis are causal (i.e., they only depend on past and present samples).

Voltage division yields the transfer functions:

$$H_{RC}(s) = \frac{\frac{1}{sC}}{R + \frac{1}{sC}} = \frac{\frac{1}{C}}{sR + \frac{1}{C}} = \frac{\frac{1}{RC}}{s + \frac{1}{RC}} \tag{2.1}$$

$$H_{CR}(s) = \frac{R}{R + \frac{1}{sC}} = \frac{sRC}{sRC + 1} = \frac{s}{s + \frac{1}{RC}} \tag{2.2}$$

$$H_{RL}(s) = \frac{sL}{sL + R} = \frac{s}{s + \frac{R}{L}} \tag{2.3}$$

$$H_{LR}(s) = \frac{R}{sL + R} = \frac{\frac{R}{L}}{s + \frac{R}{L}} \tag{2.4}$$

The continuous-time angular cutoff frequency, Ω_c, is $\frac{1}{RC}$ for the capacitor circuits and $\frac{R}{L}$ for the inductor circuits.[11] Equations 2.1 and 2.4 are in the form of the first-order low-pass filter (LPF1) transfer function, $H_{LP}(s)$:

$$H_{LP}(s) = \frac{\Omega_c}{s + \Omega_c} \tag{2.5}$$

Similarly, Equations 2.2 and 2.3 are first-order high-pass filter (HPF1) transfer functions, $H_{HP}(s)$:

$$H_{HP}(s) = \frac{s}{s + \Omega_c} \tag{2.6}$$

[11] The angular cutoff frequency is measured in radians per second ($2\pi\ rad/s\ =\ 1\ Hz$).

Both filters have a pole, $p_1 = -\Omega_c$, on the left half of the s-plane, hence the name 'one-pole' filter. The HPF1 has a zero, $z_1 = 0$, at the origin, whereas the LPF1 has an implicit zero, $z_1 = j\infty$.[12]

Equations 2.5 and 2.6 only differ in their numerator; they are related by the following equation in Laplace domain:

$$y_{HP} = \frac{s}{\Omega_c} y_{LP} \tag{2.7}$$

The relation as a differential equation is:

$$y_{HP} = \frac{\dot{y_{LP}}}{\Omega_c} \tag{2.8}$$

Since all modes are related via Equation 2.8, the solution of one mode trivially solves the other mode. The best strategy for solving 'multimode' filters in this thesis will be to solve one mode and derive the others from their relations to the initially solved mode.

Assume that the low-pass mode is the mode to solve. The mode should match the specifications of an LPF1. Indeed, with cutoff ($\Omega_c = 1$), the low-pass mode has the following magnitude response properties: $-6dB/oct$ roll-off, $|H_{LP}(j0)| = 0\ dB$, $|H_{LP}(j\infty)| = -\infty\ dB$, and $|H_{LP}(j1)| = -3\ dB$ (Figure 2.2). Low-pass-to-high-pass substitution yields a high-pass magnitude response (flipped magnitude response about the cutoff).[13]

[12] The imaginary axis is interpreted as a generalized circle. In other words, $+j\infty = -j\infty$, where $\pm j\infty$ is the theoretical point of continuous-time aliasing.

[13] Low-pass to high-pass substitution is defined as: $s \leftarrow \frac{1}{s}$. It is equivalent to replacing integration with differentiation or vice versa. This substitution sets $|H(j)|$ equal to $|H(-j)|$. Since transfer functions of real systems are Hermitian ($H(j\Omega) = H^*(-j\Omega)$), $|H(j)| = |H(-j)|$ and the cutoff position of filters with a unit cutoff does not change [10, Ch 2.5].

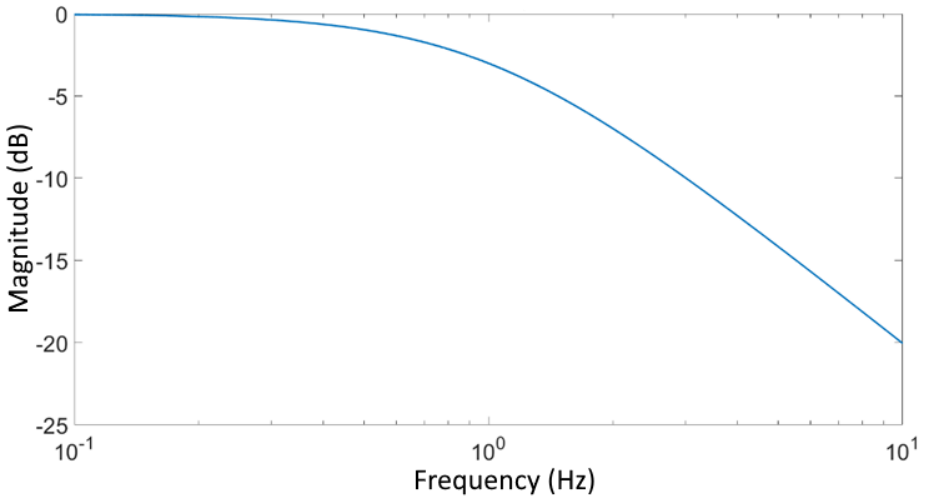

Figure 2.2: One-pole filter low-pass mode magnitude response

The low-pass step response is important because of its smoothing property (Figure 2.3). The value reaches approximately 0.63 at $t = \frac{1}{\Omega_c} = \tau$ [11]:[14]

$$step_{LP}(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}H_{LP}(s)\right\} = \mathcal{L}^{-1}\left\{\frac{\Omega_c}{s(s+\Omega_c)}\right\}$$

$$= \mathcal{L}^{-1}\left\{\frac{1}{s} - \frac{1}{s+\Omega_c}\right\} = 1 - e^{-\Omega_c t} = 1 - e^{\frac{-t}{\tau}} \tag{2.9}$$

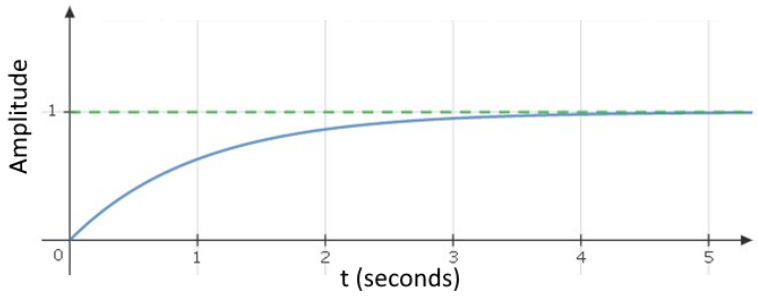

Figure 2.3: One-pole filter low-pass mode step response ($\tau = 1$)

[14] $step(t) = u(t) * h(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}H(s)\right\}$, $u(t) = \begin{cases} 0, & t < 0 \\ 1, & t \geq 1 \end{cases}$

After confirming the correct filter specifications, the next step is to derive a differential equation (i.e., the inverse Laplace transform of the transfer function):

$$\frac{y_{LP}}{x} = \frac{\Omega_c}{s + \Omega_c}$$

$$s \cdot y_{LP} = \Omega_c(x - y_{LP})$$

$$\dot{y_{LP}} = \Omega_c(x - y_{LP})$$

$$y_{LP} = \mathrm{y}_{LP}(t_0) + \Omega_c \int (x - y_{LP}) \tag{2.10}$$

Two key assumptions ensure that the filter is linear and accurate. First, $\mathrm{y}(t_0) = 0$ because there is no nonlinearity from DC bias [10, Ch 2.3]. Second, cutoff modulation implies that Ω_c is a function of time and should remain within the integral [10, Ch 2.1]:

$$y_{LP}(t) = \int_{t_0}^{t} \Omega_c(\tau)\big(x(\tau) - y_{LP}(\tau)\big)d\tau \tag{2.11}$$

Not only is the placement of $\Omega_c(t)$ accurate, but it also ensures stability. Given that $\Omega_c(t)$ is positive for all values of t, cutoff modulation warps the time and does not affect the boundedness of the output. There exists a bijection between the warped time and the un-warped time; the function mapping t to $\tilde{t}$ is monotonic and increasing, and the peak amplitude doesn't change [10, Eq. 2.27].

$$d\tilde{t} = \Omega_c(\mathrm{t})dt$$

$$\tilde{t} = \int_0^t \Omega_c(\tau)d\tau \tag{2.12}$$

Finally, applying Equation 2.8 to Equation 2.11 solves for $y_{HP}(t)$:

$$y_{HP}(t) = \frac{\dot{y_{LP}}}{\Omega_c} = \mathrm{x} - y_{LP} \tag{2.13}$$

The one-pole multimode filter block diagram visualizes $y_{HP}(t)$ as the node before the integrator and cutoff gain (Figure 2.4). Furthermore, accurate cutoff modulation corresponds to a cutoff gain before the integrator.

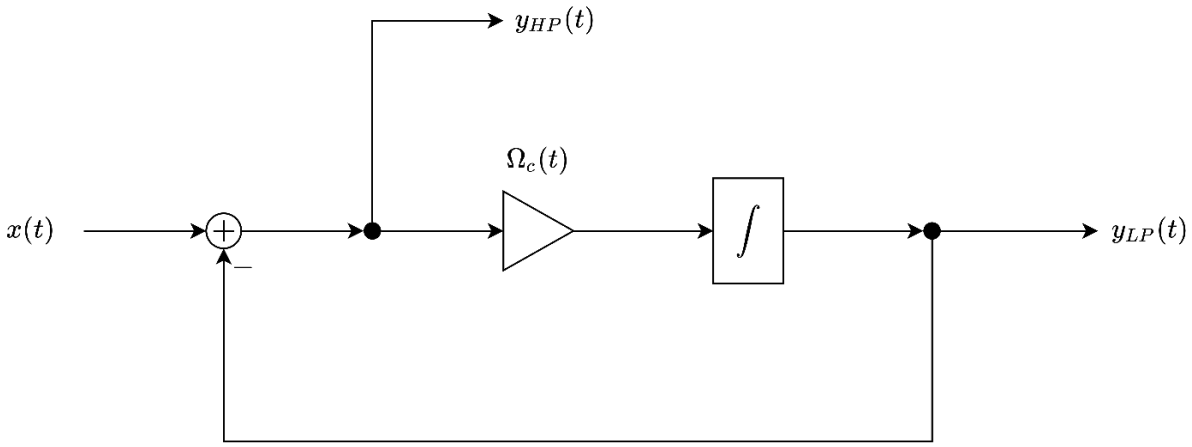

Figure 2.4: One-pole filter multimode block diagram

2.2 Topology-Preserving Transform

The bilinear transform (BLT) replaces continuous-time integrators with discrete-time integrators in the transfer function (frequency domain).[15] This approach ignores the topology of the differential equation and changes the placement of the cutoff gain [10, Fig. 3.34]. Alternatively, integrator replacement in the block diagram (time domain) preserves the location of the cutoff gain in Figure 2.4.

Numerical integration discretizes continuous integration. Integrators used in audio must involve minimal calculations and operate on evenly spaced samples (the sampling period of T)

[15] The bilinear transform is defined by the substitution: $\frac{1}{s} \leftarrow \frac{T}{2} \cdot \frac{z+1}{z-1}$

[12].[16] This project uses the trapezoidal integrator because of its minimal warping and simplicity.[17]

The non-recursive trapezoidal integrator equation is:

$$\mathrm{y}[n] = \sum_{i=n_0}^{n} \frac{T}{2}(x[i] + x[i-1]) \tag{2.14}$$

Given a set of regularly sampled points, $x[0:n]$, the trapezoidal integrator returns the area under the linearly interpolated trapezoids between each existing pair $\{x[i], x[i-1]\}$ (Figure 2.5).

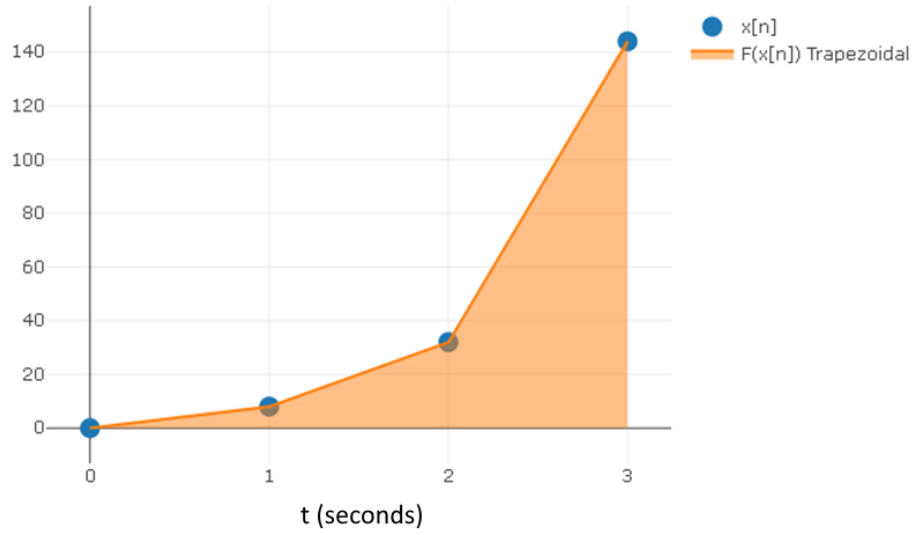

Figure 2.5: Trapezoidal integration on a set of arbitrary samples, $x[0:3]$ ($T = 1$)

Performance-sensitive applications (e.g., DAW plugins) use the infinite impulse response (IIR) in four different direct forms: Direct-Form-I, Direct-Form-II, and their transposes [13]. Transposed-Direct-Form-II (TDF-II) is the ideal digital topology for VA filters because it is the only canonical (with respect to delay) direct form that preserves cutoff gain location (Figure 2.7).

[16] Here, 'order' means number of delay taps (e.g., the trapezoidal integrator is a first-order method).
[17] Even more implementations exist, but they are outside of the scope of this paper.

In other words, the TDF-II integrator contains only one delay element, and the cutoff gain comes first (Figure 2.6). Consequently, fewer load/store instructions are required, and cutoff modulation is stable [10, Ch. 3.6].

The TDF-II solution derives from Equation 2.14 using a single state variable, $u[n-1]$:[18]

$$\mathrm{y}[n] = \sum_{i=n_0}^{n} \frac{T}{2} \cdot x[i] + \sum_{i=n_0}^{n} \frac{T}{2} \cdot \mathrm{x[i-1]} = \sum_{i=n_0}^{n} \frac{T}{2} x[i] + \sum_{i=n_0}^{n-1} \frac{T}{2} \cdot \mathrm{x[i]}$$

$$\mathrm{y}[n] = \frac{T}{2} \cdot x[n] + \sum_{i=n_0}^{n-1} \mathrm{T} \cdot x[i] \; = \frac{T}{2} \cdot x[n] + u[n-1]$$

In addition, $u[n]$ is saved each sample to solve $y[n+1]$:

$$u[n] = u[n-1] + T \cdot x[n] = y[n] \; + \frac{T}{2} \cdot x[n]$$

The algorithm (Algorithm 1) contains one multiplication and two additions ($T/2$ is constant):

Algorithm 1: TDF-II trapezoidal integrator

Input: x, s
Output: y

$v = (T/2) \cdot x$
$y = v + s$
$s = y + v$

[18] All samples before $t = 0$ or $n = 0$ are assumed to be zero.

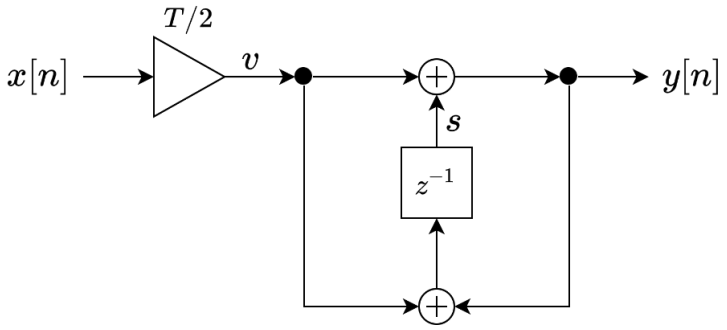

Figure 2.6: Transposed-Direct-Form-II trapezoidal integrator

Replacing the continuous-time integrator in Figure 2.4 yields the following block diagram (Figure 2.7):

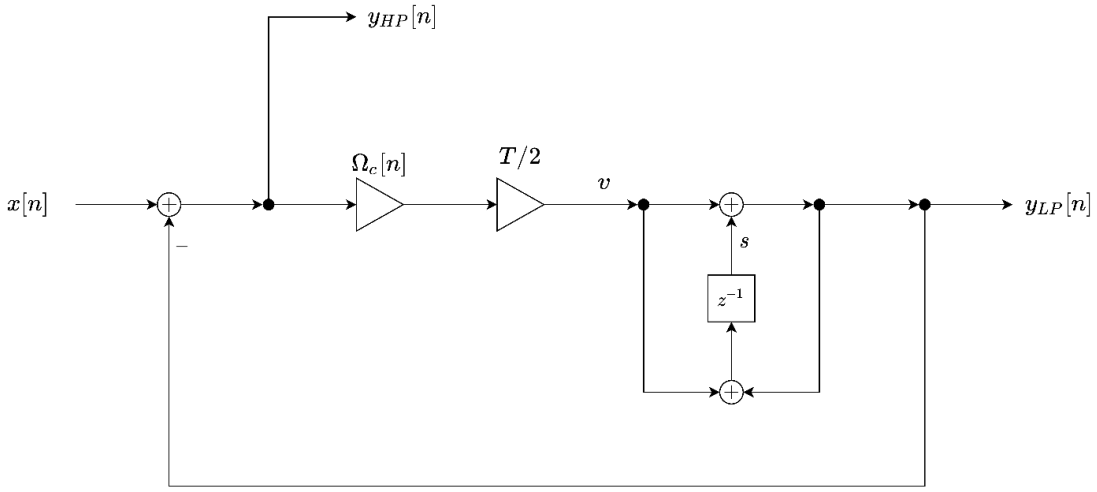

Figure 2.7: TPT one-pole filter block diagram

The Z-transform proves that Figure 2.6 is equivalent to the BLT in the time domain.

$$y = v + s$$

$$y = \left(y + \frac{Tx}{2}\right)z^{-1} + \frac{Tx}{2}$$

$$\mathrm{y}(z-1) = \frac{Tx}{2}(z+1)$$

$$\mathrm{H}(z) = \frac{T}{2} \cdot \frac{z+1}{z-1} \tag{2.15}$$

Replacing continuous-time integrators with the TFD-II trapezoidal integrator in the block diagram is analogous to the substitution $s \leftarrow \frac{2}{T} \cdot \frac{z-1}{z+1}$ in the transfer function [10, Ch. 3.6]:

$$H(s) = H\left(\frac{2}{T} \cdot \frac{z-1}{z+1}\right) \tag{2.16}$$

The imaginary axis in the s-plane maps to the unit circle in the z-plane with some warping:

$$H\left(e^{j\Omega T}\right) = H\left(\frac{2}{T} \cdot \frac{e^{j\Omega T}-1}{e^{j\Omega T}+1}\right) = H\left(\frac{2}{T} \cdot \frac{e^{j\Omega T/2}-e^{-j\Omega T/2}}{e^{j\Omega T/2}+e^{-j\Omega T/2}}\right) = H\left(j\frac{2}{T}tan\left(\frac{\Omega T}{2}\right)\right)$$

The frequency response for any frequency Ω into the trapezoidal integrator is equal to the continuous-time integrator's frequency response at $\frac{2}{T}tan\left(\frac{\Omega T}{2}\right)$. To ensure that both integrators match frequency response at Ω, a pre-warping function μ(Ω) must map exist such that $\Omega = \frac{2}{T}arctan\left(\frac{\mu(\Omega)\,T}{2}\right)$:

$$\mu(\Omega) = \frac{2}{T}tan\left(\frac{\Omega T}{2}\right) \tag{2.17}$$

To combine the gains in Figure 2.7, Ω_c is pre-warped and scaled by $\frac{T}{2}$ (Figure 2.8). The analog filter's frequency response and the discrete-time filter's frequency response match at the cutoff frequency.

$$\mathrm{g} = \frac{T}{2}\mu(\Omega_c) = \tan\left(\frac{\Omega_c T}{2}\right) \tag{2.18}$$

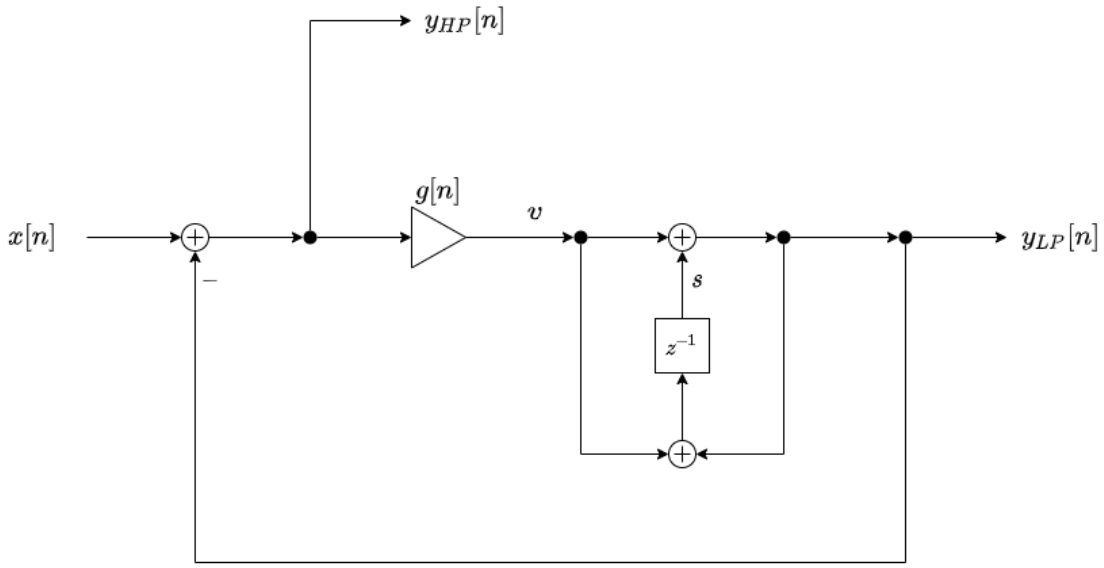

Figure 2.8: TPT one-pole filter block diagram written with total pre-warped cutoff gain, g

Solving a zero-delay feedback equation for any edge in the block diagram preserves the zero-delay feedback loop. Typically, the easiest edge is one that is right before or after the cutoff gain. In this case, the edge is v. Trapezoidal integrator replacement combined with preserved zero-delay feedback loops is necessary for TPT.

While [10, Ch. 3.9] takes a two-step approach by first solving for y_{LP}, solving for v directly is simpler and requires less steps. The following zero-delay feedback equation is based on Figure 2.8:

$$\mathrm{v} = \mathrm{g}(x - v - s)$$

Since multiplication by g is a linear operation, an explicit solution exists for v:

$$v(1 + g) = g(x - s)$$

$$v = g\frac{(x - s)}{1 + g} = G \cdot (x - s) \tag{2.19}$$

Visually, the feedback loop starts at s rather than y_{LP} (Figure 2.9):

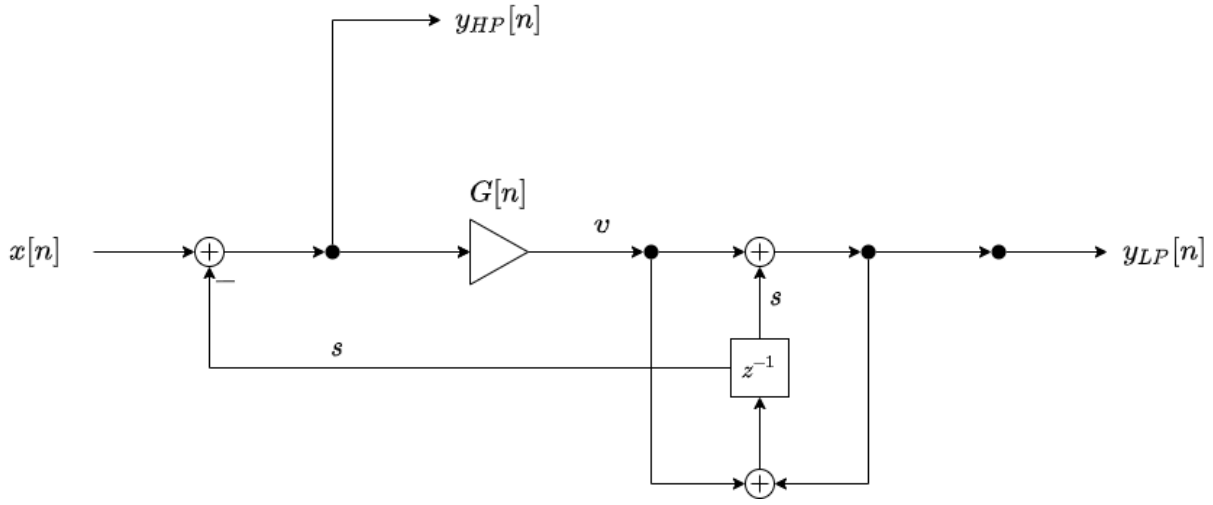

Figure 2.9: TPT one-pole filter block diagram with the zero-delay feedback resolved

G is instantaneously unstable if the denominator is zero; therefore, $g > -1$ [10, Ch 3.13]. Furthermore, values of g near negative one result in digital clipping and/or floating-point error due to high gain values.[19] For this reason, in this thesis, all filter parameters found in denominators are positive or zero. In addition, $\frac{\Omega_c T}{2}$ is limited to the interval $\left[0, 0.9 \cdot \frac{\pi}{2}\right)$ to prevent instability in g (Equation 2.18).

The remaining portion of the trapezoidal integrator solves for $y_{LP}[n]$ and updates s; thus, the final algorithm is:

[19] The floating-point error results from truncation (i.e., adding numbers of significantly different magnitude).

Algorithm 2: One-pole filter modes

Input: x, s
Parameters: g
Output: y_{HP}, y_{LP}

$$let\ G = \frac{g}{1+g}$$

$$let\ v = (x - s) \cdot G$$

$$y_{LP} = v + s$$
$$s = y_{LP} + v$$

$$y_{HP} = x - y_{LP}$$

2.3 Nonlinear Virtual Analog Filters

Most nonlinear implicit equations do not have closed-form solutions; their solutions require numerical methods. Nonetheless, filters with nonlinear feedback loops have solutions in this form.

This section considers two types of feedback loop nonlinearities: parameter-modulating and non-parameter-modulating. While [10] focuses on the second type of nonlinearity, the filters in this study will focus on nonlinearities due to internally modulated parameters (parameter-modulating).

Consider a non-parameter-modulating example. A one-pole filter has a static nonlinearity, $tanh(x)$, right before its cutoff gain (Figure 2.10):

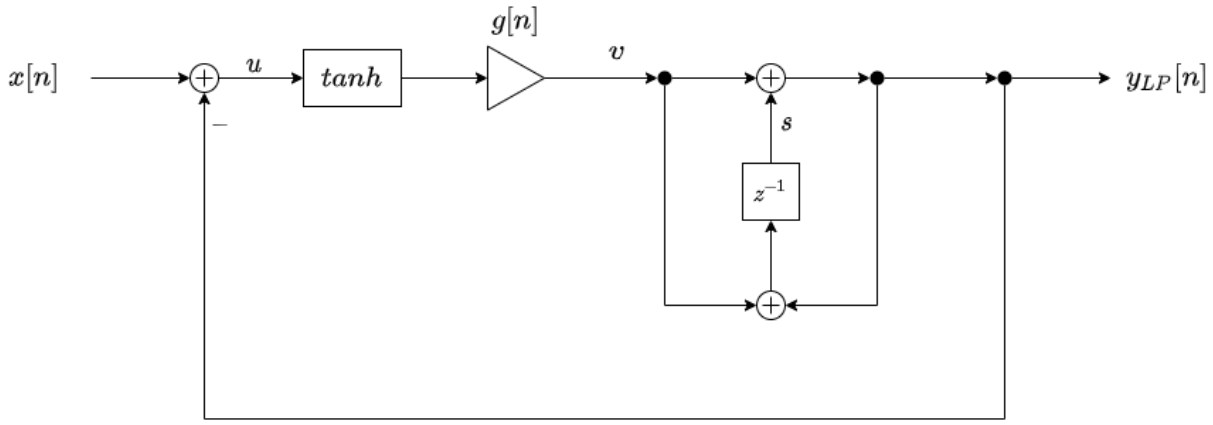

Figure 2.10: One-pole filter with a non-parameter-modulating feedback nonlinearity

Solving the delay-free loop starts with a similar form to Equation 2.18, but in this instance, the edge u is the unknown:

$$u = x - g \cdot tanh(u) + s \tag{2.20}$$

Setting the equation equal to zero, presents it as a root-finding problem:

$$0 = f(u) = x - g \cdot tanh(u) + s - u \tag{2.21}$$

Root-finding is an application of numerical analysis; a myriad of algorithms exists with varying accuracies and convergence speeds. A popular method for music DSP is the Newton-Ralphson method; this method is preferred because it only requires the first derivative of a function, and the error reduces quadratically [14]:

$$u_{n+1} = u_n - \frac{f(u_n)}{f'(u_n)} \tag{2.22}$$

Nonetheless, convergence is not guaranteed. Asymptotes and local extrema result in wrong or infinite answers [14]. Furthermore, functions with multiple roots require multiple sets

of iterations and a way to determine the correct root. Both factors make Newton-Ralphson applicable to a small set of well-conditioned single-root functions.

The implicit equation can be bypassed. Changing the nonlinearity to feedforward in Figure 2.8 yields a one-pole filter with a waveshaper before the input (Figure 2.11). The nonlinearity is no longer dynamic because it exists as a static function separate from the filter. In this case, an explicit design is detrimental for musical purposes because it results in a static nonlinearity that is much less reactive to the input signal.

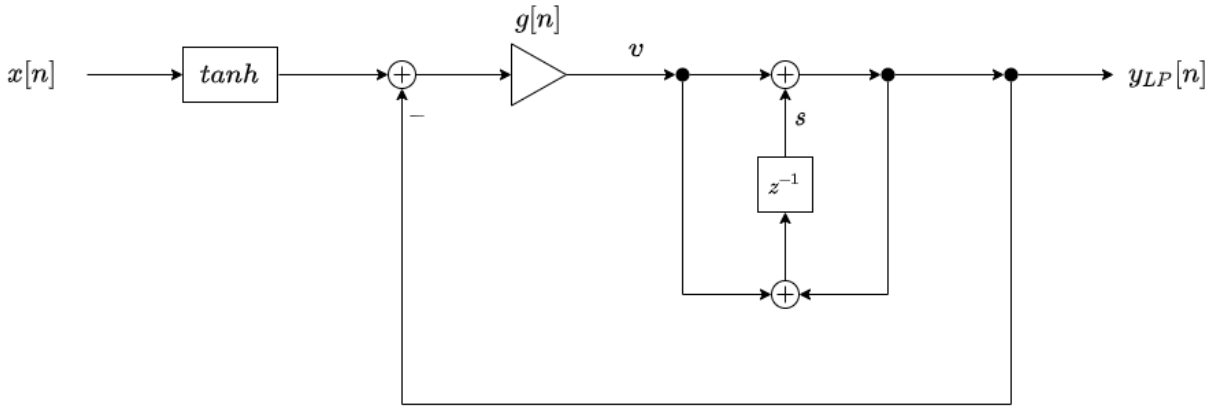

Figure 2.11: One-pole filter with a non-parameter-modulating feedforward nonlinearity

In contrast, a parameter-modulating nonlinearity must modify at least one part of its filter. For example, a feedback cutoff-modulating nonlinearity maps the output to a cutoff gain value (Figure 2.12). The dependence of the cutoff on the output implies that g is a nonlinear function of $y(t)$, and the zero-delay feedback equation is implicit.

$$y = g(y)(x - y) + s \tag{2.23}$$

Furthermore, since g is pre-warped, the nonlinear cutoff function passes through yet another nonlinear function: $tan(x)$. It is very hard and often impossible to avoid asymptotes, local extrema, and multiple roots with this type of nonlinearity.

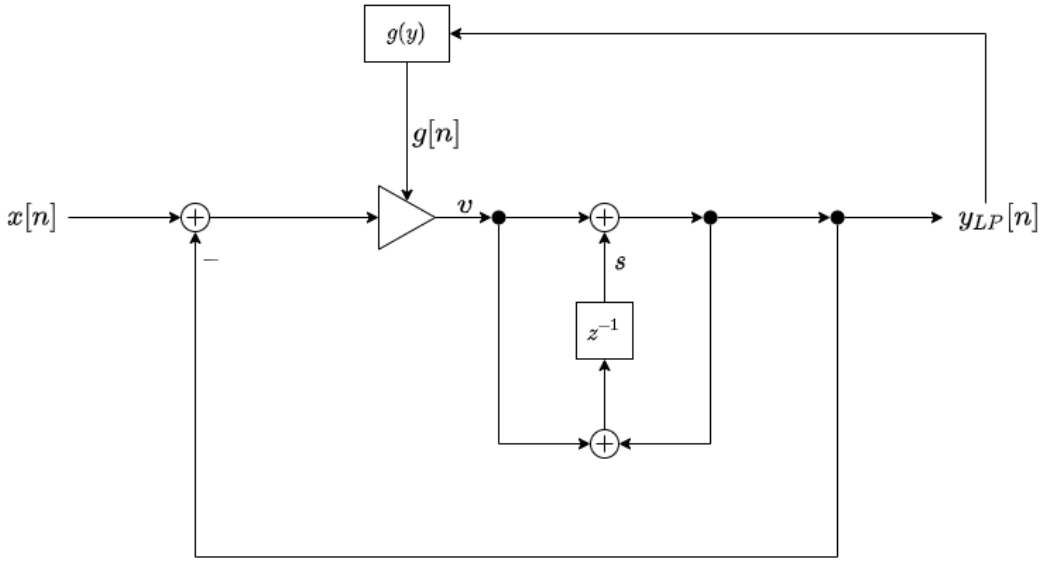

Figure 2.12: One-pole filter block diagram with a cutoff-modulating feedback nonlinearity

Transforming the cutoff-modulating feedback nonlinearity to a feedforward version preserves the dynamic nonlinearity (Figure 2.13). Since the cutoff gain must be nonlinear, the nonlinearity is tied to the filter; a static nonlinearity will not occur if g is a function of $x(t)$. In addition, the feedforward version's algorithm (Algorithm 3) is identical to the non-iterative linear one-pole filter algorithm (Algorithm 2), except $G[n]$ is updated every sample:

Algorithm 3: One-pole filter with a cutoff-modulating feedforward nonlinearity

Input: x, s
Parameters: $Depends\ on\ G(x)$
Output: y_{HP}, y_{LP}

$let\ G = G(x)$

$let\ v = (x - s) \cdot G$

$y_{LP} = v + s$
$s = y_{LP} + v$

$y_{HP} = x - y_{LP}$

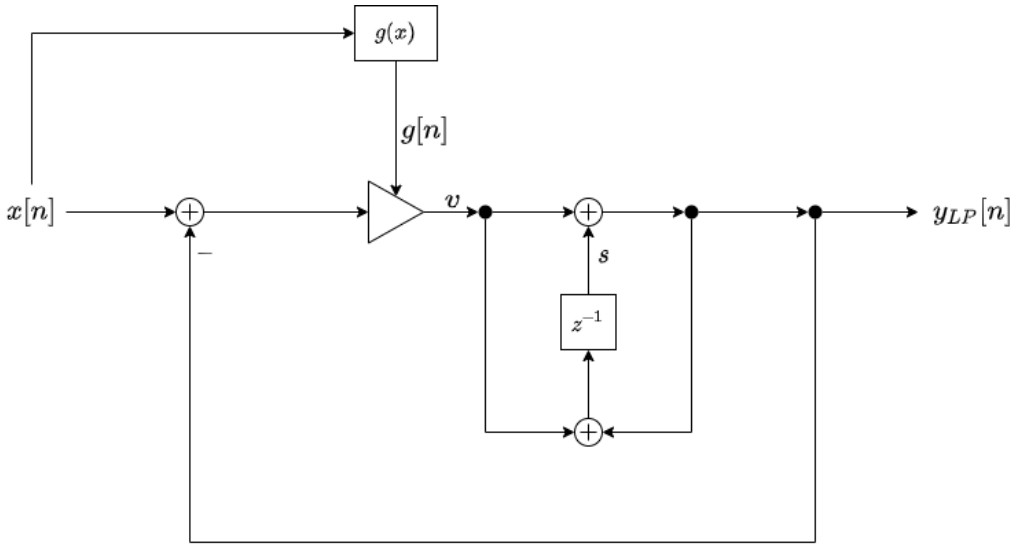

Figure 2.13: One-pole filter block diagram with a feedforward cutoff-modulating nonlinearity

Lastly, slow-varying nonlinear effects have analytical solutions that approximate feedback loops. Since this class of effects applies their sound gradually, the error from inserting a unit delay into the feedback loop is negligible. The overall trend of the effect is preserved, and the sound is effectively unchanged (Figure 2.14).

For example, if the nonlinear LPF1 in Figure 2.12 is used for its step response, it gradually applies a smoothing effect. Equation 2.20 is effectively equivalent to:

$$u[n] = x[n] - g[n] \cdot tanh(u[n-1]) + s[n] \tag{2.24}$$

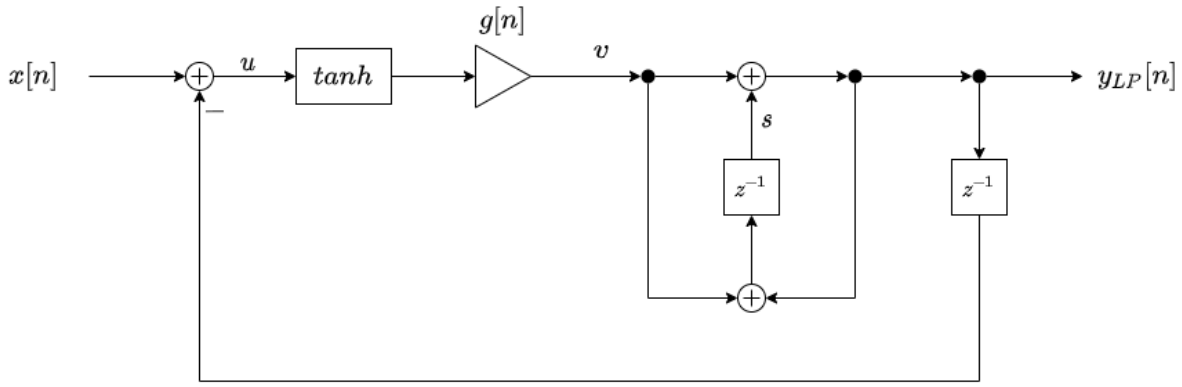

Figure 2.14: TPTz version of Figure 2.12

This method also solves the feedback loop in the classical compressor because compression gradually applies gain reduction. The discretized block diagram of the compressor in Figure 1.2 uses $y[n-1]$as $\varsigma[n]$ in feedback mode (Figure 2.15).

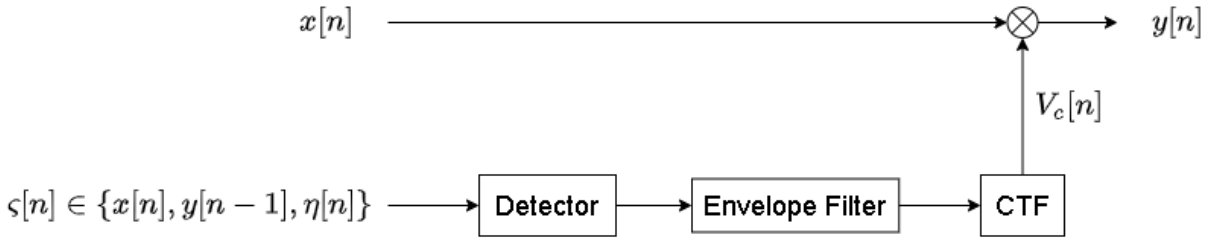

Figure 2.15: TPTz version of the compressor in Figure 1.2

This method will be called ‘TPTz’ because it uses trapezoidal integrators but does not preserve zero-delay feedback loops. Even though the accuracy of this method is much lower than Newton-Ralphson, the difference is not audible if the effect takes more than one millisecond to apply an audible effect (i.e., the effect is gradual).

Chapter 3

Sidechain Design

3.1 Classical Design

It is possible to implement all blocks of the character compressor using the math covered in the previous chapter. This chapter starts the implementation process with a classical compressor (Figure 1.2).

The detector converts $\varsigma(t)$ to positive or mostly positive signals. In other words, this block is a rectifier. The simplest detector is a perfect rectifier: $|x|$.

The envelope filter is a LPF1 with a cutoff that depends on the input and output. While the effect is a cutoff-modulating-nonlinearity, the cutoff can only be one of two values depending on if the input is greater than or less than the output. Therefore, this cutoff is more accurately referred to as 'piecewise linear.'

The modulating low-pass is called a ballistics filter (Figure 3.1). A branching function $b(x, y)$ determines sets $\Omega_c(t)$ to an attack or release value, Ω_a or Ω_r, given $x(t)$ and $y(t)$. Since the filter is used for its step response, the parameters are usually represented as τ_a and τ_r, where $\tau = \frac{1}{\Omega}$.

$$\Omega_c = b(x, y) = \begin{cases} \Omega_a, x > y \\ \Omega_r, otherwise \end{cases} \tag{3.1}$$

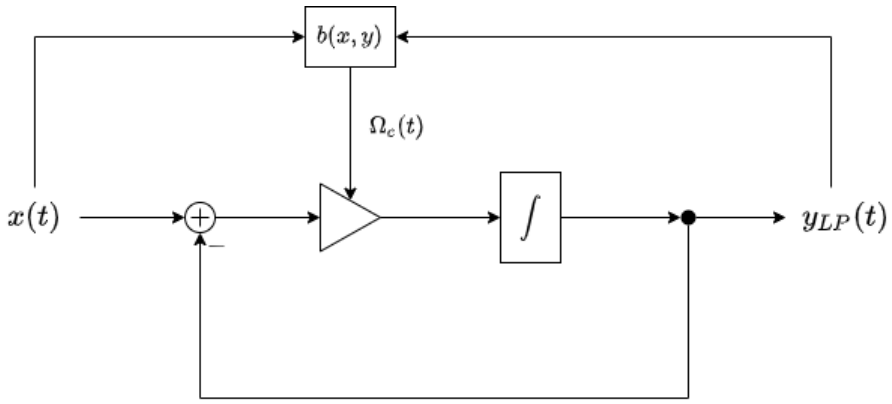

Figure 3.1: Ballistics filter block diagram

The branching depends on a comparison between two values; therefore, it cannot be transformed to be feedforward when discretized. Nonetheless, it is a slow-varying effect, and TPTz is sufficient (Figure 3.2)

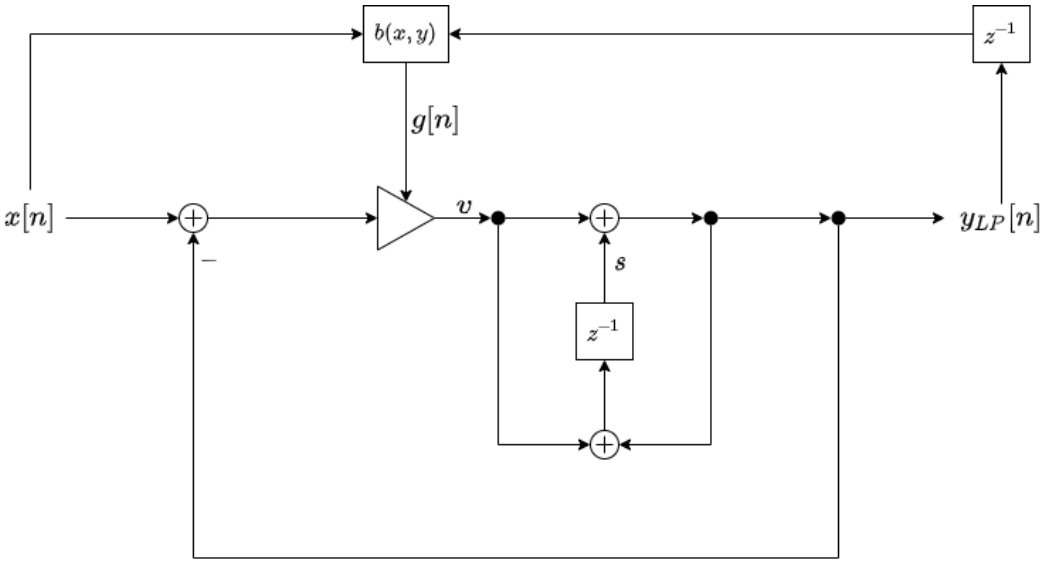

Figure 3.2: Ballistics filter TPTz block diagram

Lastly, the CTF calculates gain reduction based on a mapping on the dB scale. $V_c(t)$ requires a gain value; hence, the CTF maps to $V_c(t)$ via the following relation:[20]

$$V_c(t) = \frac{dBtoGain\left(CTF\left(gainTodB(x)\right)\right)}{x} = \frac{dBtoGain\left(CTF(x_{dB})\right)}{x} \tag{3.2}$$

The classical CTF is a piecewise linear hard-knee function ratio and threshold parameter (R and thr_{dB}):

$$CTF(x_{dB}) = \begin{cases} x_{dB}\ , x_{dB} \leq thr_{dB} \\ thr_{dB} + \dfrac{x_{dB} - thr_{dB}}{R}, otherwise \end{cases} \tag{3.3}$$

The plot of the ideal hard-knee CTF with $R = 5$ and $thr = -20\ dB$ resembles Figure 1.9, but without the curved gain reduction (when $x_{dB} > thr_{dB}$) (Figure 3.3):

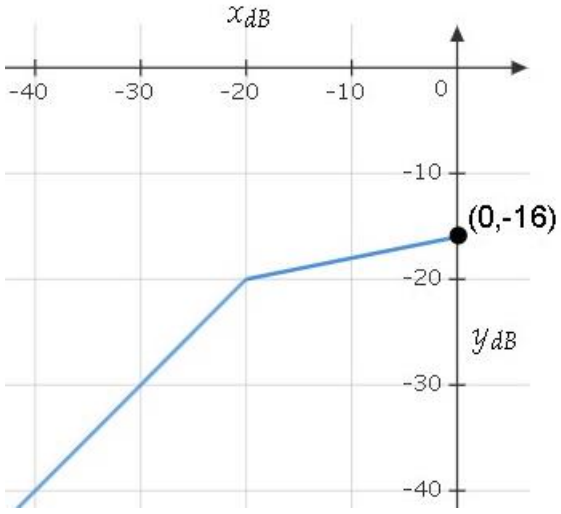

Figure 3.3: Hard-knee CTF ($R = 5, thr = -20dB$)

[20] $x_{dB} = gainTodB(x) = 20log_{10}(x);\ x = dBtoGain(x_{dB}) = 10^{(x_{dB}/20)}$

3.2 Advanced Features

3.2.1 Detector Variants

The detector can be any static function with a mostly positive output. For example, a simple alternative to the absolute value rectifier is the perfect half-wave rectifier:

$$f(x) = \begin{cases} x, x > 0 \\ 0, otherwise \end{cases} \tag{3.4}$$

Functions do not need to be piecewise linear. One example is a half-wave rectifier inspired by the Shockley diode equation [15]:

$$\mathrm{f}(x) = \frac{e^x - 1}{e - 1} \tag{3.5}$$

Furthermore, if a static function is not sufficient, the detector may generalize to any loudness metric. For example, one of the most common loudness metrics in audio DSP is Loudness Units relative to Full Scale (LUFS). The algorithm involves a second-order high-pass/high-shelf input filter and a modified moving RMS [16]. The input filter is especially important because it approximates the Fletch-Munson psycho-acoustic loudness model. Generally, human ears are less sensitive to low frequencies and more sensitive to high frequencies. The LUFS algorithm uses its K-weighting filter (Figure 3.4) to apply this approximate frequency weighting [17].

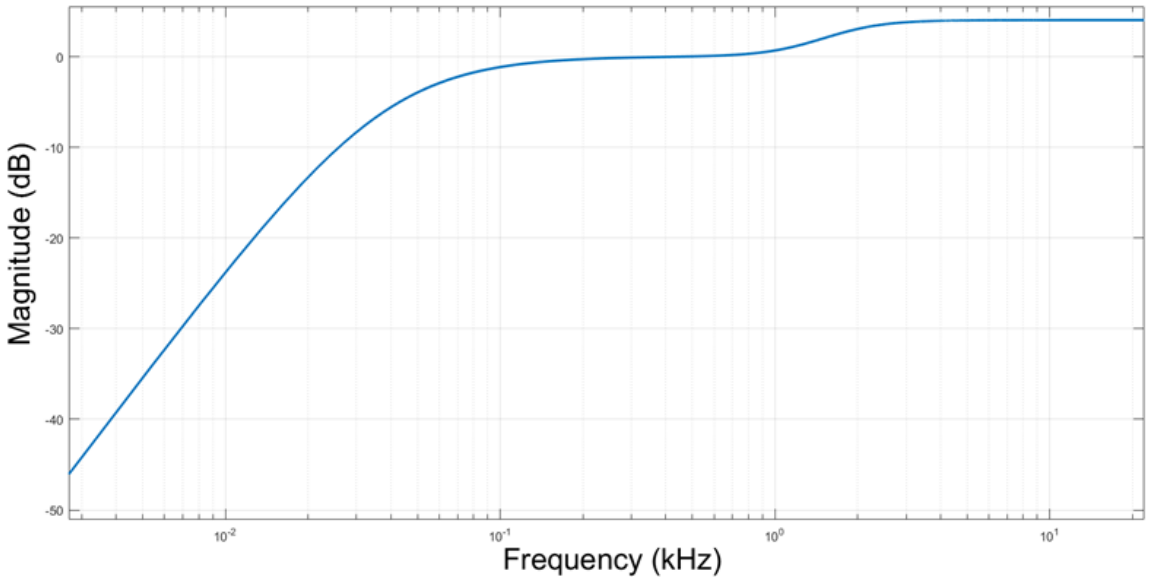

Figure 3.4: LUFS K-weighting filter magnitude response

3.2.2 Nonlinear Envelope Filter

The step response of the envelope filter also does not need to be piecewise linear. Rather, most analog envelope filters exhibit nonlinearities.

For example, if the attack and release values are applied via an LR LPF1, the values of Ω_a and Ω_r depend on $V_{out}(t)$ (Figure 3.5); an inductor with a magnetic core is significantly nonlinear such that $\Omega_c(\mathrm{t}) = \frac{R}{L(t)}$ [18]. In this case, Ω_a and Ω_r are not constant.

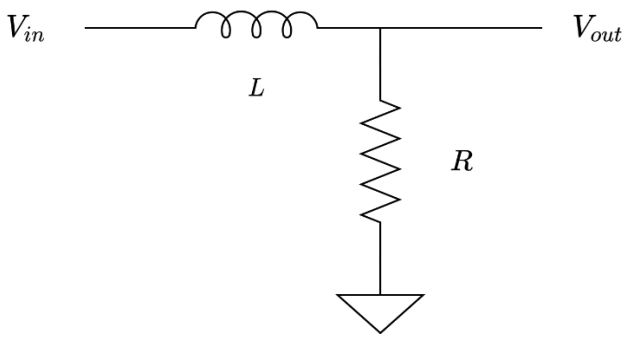

Figure 3.5: LR LPF1

A conductor that resists change in current exhibits self-inductance. In an inductor, a current, I, varies in a coil of wire around a magnetic core and changes the magnetic flux density, B, through the core. This change in B results in a current through the wire opposing the flow of I; therefore, the saturation of the magnetic core determines the nonlinear inductance.

A B-H relation describes B in terms of the magnetizing force, H; H depends on I flowing through the coil. The H-I relation depends on the geometry of the inductor. This fact also applies to the relation between $\frac{dB}{dH}$ and L.

$$H \propto I \tag{3.6}$$

$$\frac{dB}{dH} \propto L \tag{3.7}$$

This thesis is not concerned with physical accuracy; rather, the controllability of key parameters such as cutoff is more important for music DSP. Consequently, any constant multiples will generally be ignored because they do not contribute any new nonlinearity and complicate equations. Variables related by a constant multiple such as in Equations 3.6 and 3.7 will be treated as if they were equal.

Applying Ohm's law to Figure 3.5 converts I to y_{LP} [18]:

$$I = \frac{V_{out}}{R} = \frac{y_{LP}}{R} \tag{3.8}$$

Again, the value R is a constant multiple and ignored:

$$I \propto y_{LP} \Rightarrow H \propto y_{LP} \tag{3.9}$$

Using the previous assumptions, any B-H relation can map y_{LP} to L:

$$\frac{dB}{dH}(H) = L(y_{LP}) \tag{3.10}$$

Since L is an output-dependent parameter value, $L(y_{LP})$ is a parameter-modulating feedback nonlinearity (Figure 2.15); thus, the feedforward conversion (Figure 2.16) should be applied before defining the final mapping:

$$\frac{dB}{dH}(H) = L(x) \tag{3.11}$$

The Fröhlich-Kennelly (FK) relation is an anhysteretic B-H relation. In other words, the relation is the following static nonlinearity where c and b are parameters [18]:

$$B(H) = \frac{H}{c + b|H|} \tag{3.12}$$

$$\frac{dB}{dH}(H) = \frac{c}{(c + b|H|)^2} \tag{3.13}$$

Or in terms of filter parameters:

$$L(x) = \frac{c}{(c + b|x|)^2} \tag{3.14}$$

Removing the scaling by c allows the FK parameters to be represented by linear inductance, L_{lin}, and nonlinearity, N.[21] The final anhysteretic equation is:

$$L(x) = \frac{1}{(c + b|x|)^2} = \frac{1}{\left(\sqrt{\frac{1}{L_{lin}}} + N|x|\right)^2} \tag{3.15}$$

If $x(t)$ or N is zero, then $L(x) = L_{lin}$; otherwise, if N is non-zero, $L(x)$ decreases as $|x|$ increases (Figure 3.6).

[21] $N \geq 0$

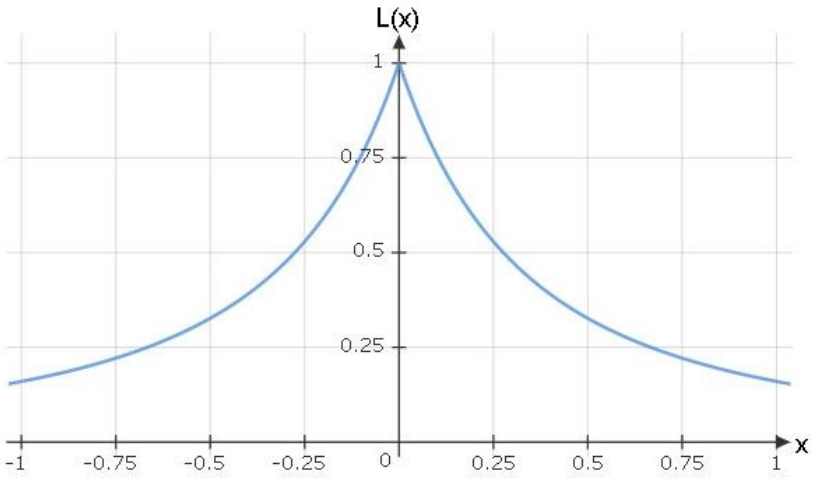

Figure 3.6: Plot of Equation 3.15 ($L_{lin} = 1$ and $N = 1.5$)

This behavior makes physical sense because the magnetic flux density within the inductor core saturates at high input amplitudes. Since the inductance is proportional to the flux density, it also drops, and the inductor becomes less effective (more nonlinear).

Resistors and capacitors are similarly nonlinear depending on their V-I and Q-V (charge-voltage) relations. Many models use low-order polynomial approximations like the denominator of Equation 3.15 [19, 20, 21]. Furthermore, resistance and capacitance can decrease at high voltages like in Figure 3.6 [19, 22]. For these reasons, the FK relation is a good approximation of the value of a generic nonlinear passive component.

3.2.3 Transfer Function Types

The four most common types of dynamic processor transfer functions are downward compression, upward compression, downward expansion, and upward expansion [23]. A hard-knee downward compression corresponds to the compressor transfer function (CTF) of the classical compressor.

Upwards expansion uses the same transfer function as the classical compressor, but $R \in (0, 1)$ (Figure 3.7). This effect increases or 'expands' the dynamic range of signals above the threshold. Unlike a downward transfer function, upward transfer functions can easily boost signals past the point of digital clipping ($0dB$) given small values of R.

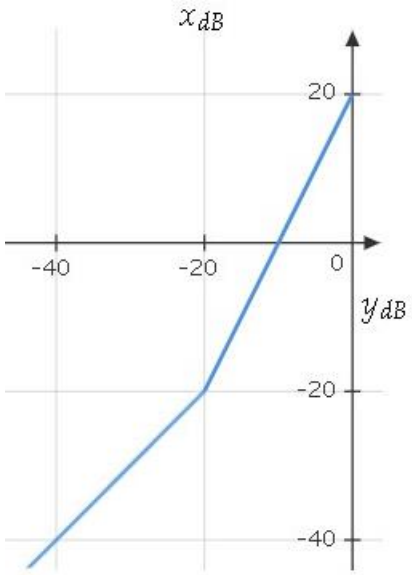

Figure 3.7: Transfer function of the hard-knee upward expander ($R = 0.5, thr = -20dB$)

The upward compressor and downward expander use the following CTF:

$$CTF(x_{dB}) = \begin{cases} x_{dB}, x_{dB} > thr_{dB} \\ thr_{dB} + \dfrac{x_{dB} - thr_{dB}}{R}, otherwise \end{cases} \quad (3.16)$$

$R \in (1, \infty)$ corresponds to the upward compressor (Figure 3.8). In this case, the CTF should change so that small values of x_{dB} do not map to high value of $CTF(x_{dB})$ and result in an extremely high gain. For example, the transfer function could be:

$$CTF(x_{dB}) = \begin{cases} x_{dB}, x_{dB} > thr_{dB} \\ thr_{dB} + \dfrac{x_{dB} - thr_{dB}}{R}, x_{dB} \leq thr_{dB} \; and \; x_{dB} > -90dB \\ 0, otherwise \end{cases} \quad (3.17)$$

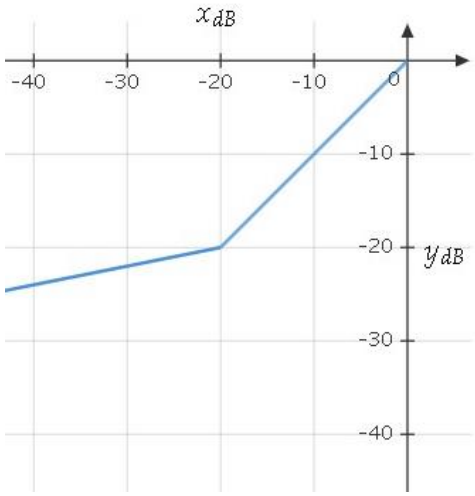

Figure 3.8: Transfer function of the hard-knee upward compressor ($R = 5, thr = -20dB$)

$R \in (0, 1)$ corresponds to the downward expander (Figure 3.9). The transfer function is also called a 'noise-gate' because it lowers the amplitude of the noise floor.

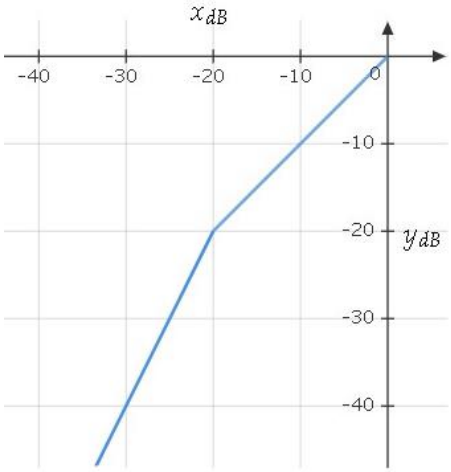

Figure 3.9: Transfer function of the hard-knee downward expander ($R = 0.5, thr = -20dB$)

'Auto swell' refers to a non-monotonic transfer function (Figure 3.10). For example, a hard-knee CTF with a negative ratio is classified as an auto swell.

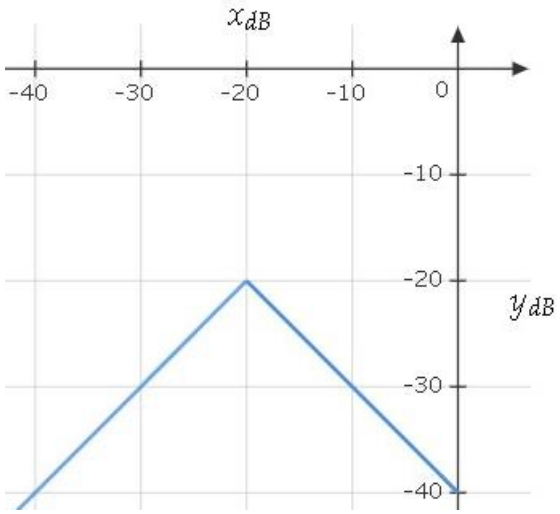

Figure 3.10: Transfer function of a hard-knee auto swell ($R = -1, thr = -20dB$)

The knee is another area for modification. An alternative to the hard-knee is the soft-knee. This variation is differentiable and produces a smoother sounding transition for input signals near the threshold [6].

$$CTF(x_{db}) = \begin{cases} x_{db}, 2(x_{db} - thr_{db}) < -W \\ x_{db} + \left(\frac{1}{R} - 1\right)\left(x_{db} - thr_{dB} + \frac{W}{2}\right)^2 /(2W), 2|x_{db} - thr_{db}| \leq W \\ thr_{db} + (x_{db} - thr_{db})/R, otherwise \end{cases} \quad (3.18)$$

The colored CTF from Chapter 1.1.2 not only smooths the knee but also can add emphasis/deemphasis via overshoot. The ATK Colored Compressor includes color by adding a Gaussian, $N(\mu, \sigma^2)$, centered at the threshold [24]. Color is equivalent to a height parameter, H, and quality is equivalent to width, W.

$$CTF(x_{dB}) = H \cdot N(thr_{dB}, W) + \begin{cases} x_{dB}\,, x_{dB} \le thr_{dB} \\ thr_{dB} + \dfrac{x_{dB} - thr_{dB}}{R}, otherwise \end{cases} \tag{3.19}$$

An algebraic bell-shaped function is a more computationally efficient alternative [25]:

$$\frac{d}{dx} arctan(x) = \frac{1}{1 + x^2}$$

Some minor modifications add parameters for height, width, and the center.

$$color(x) = \frac{H}{1 + \left(\dfrac{x - thr_{dB}}{W}\right)^2} \tag{3.20}$$

The algebraic colored CTF is (Figure 3.11):

$$CTF(x_{dB}) = color(x_{dB}) + \begin{cases} x_{dB}\,, x_{dB} \le thr_{dB} \\ thr_{dB} + \dfrac{x_{dB} - thr_{dB}}{R}, otherwise \end{cases} \tag{3.21}$$

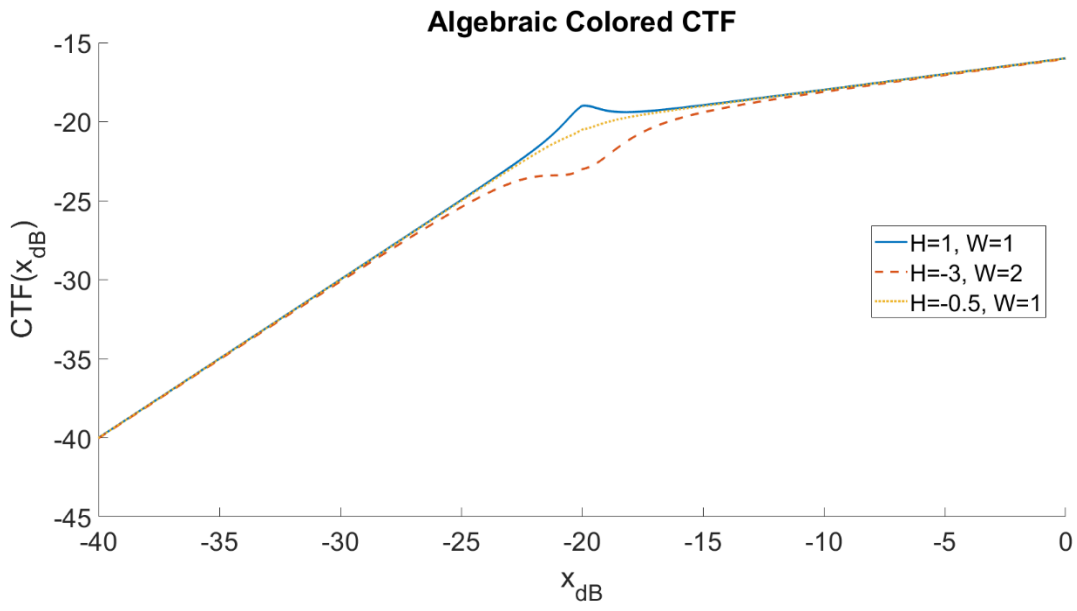

Figure 3.11: Algebraic colored CTF ($R = 5$ and $thr_{dB} = -20dB$) for different H and W

Lastly, the CTF can have memory. A common hysteresis scheme is to use different CTFs for attacks and releases.

$$CTF(x_{dB}[n]) = \begin{cases} CTF_a(x_{dB}[n]), x_{dB}[n] \geq x_{dB}[n-1] \\ CTF_r(x_{dB}[n]), x_{dB}[n] < x_{dB}[n-1] \end{cases} \tag{3.22}$$

For example, if an audio engineer wants attacks to compress at a higher threshold but releases to behave the same, the threshold should increase in $CTF_a(x_{dB}[n])$.

3.3 Final Design

The final design of this study's sidechain will prioritize the controllability of the gain reduction. The sidechain will consist of a classical detector, a nonlinear envelope filter, and a colored CTF.

Both the detector and CTF are static to ensure that level detection and gain reduction do not change based on the system's state. The colored CTF can approximate a soft-knee as well as a colored knee shape. In addition, it only contains two piecewise components (Equation 3.21) in contrast to the three components in the soft-knee CTF (Equation 3.18).

The nonlinear envelope filter assumes that $L(x)$ approximates the cutoff of a nonlinear RC circuit:

$$\Omega_c(t) = R \cdot C(t) = L(x)$$

Ω_a and Ω_r in Equation 3.1 are replaced with $L_a(x)$ and $L_r(x)$ where

$$\Omega_a = L_a(x) = \frac{1}{(c + b|x|)^2} = \frac{1}{\left(\sqrt{\tau_a} + N|x|\right)^2} \tag{3.23}$$

and

$$\Omega_r = L_r(x) = \frac{1}{(c + b|x|)^2} = \frac{1}{(\sqrt{\tau_r} + N|x|)^2} \tag{3.24}$$

Ω_a and Ω_r decrease with higher input levels; equivalently, the attack and release times increase with higher input levels.

Chapter 4

Character from Nonlinear Passive Component

Nonlinear passive components can also contribute character. Small nonlinearities in inductors, capacitors, and resistors combine to noticeably distort the input signal. The generated harmonics are an example of character.

The character block's only requirement is that it must be nonlinear. Consequently, it can handle more complex nonlinearities without sacrificing user control. This chapter expands the anhysteretic model in Chapter 3.2.2 to model a generic nonlinear passive component with memory.

The Jiles-Atherton (JA) model of hysteresis states that magnetization (M) is a linear interpolation of the irreversible and anhysteretic magnetizations with c as the interpolation parameter [26, Eqs. 1 and 2]. Each magnetization value depends on the effective field, H_e.

$$M(H_e) = (1 - c)M_{irr}(H_e) + cM_{an}(H_e) \tag{4.1}$$

M_{irr} represents the dynamic, irreversible contribution of the magnetic core whereas M_{an} is the static, anhysteretic nonlinear behavior. Exchanging interpolation with two gains, g_{irr} and g_{an}, enables both terms to mix with more flexibility.

$$M(H_e) = g_{irr}M_{irr}(H_e) + g_{an}M_{an}(H_e) \tag{4.2}$$

Furthermore, H_e is a sum of H and M, with α being the inter-domain coupling. Since α is on the order of 10^{-3} or less, H is a good approximation of H_e [27]:

$$H_e = H + \alpha M \approx H \tag{4.3}$$

The final modulating parameter requires M(H) to convert to $\frac{dB}{dH}(H)$. The formal relation between B and M adds H and scales by vacuum permeability, μ_0 [28]:

$$B(H) = \mu_0(M(H) + H) \tag{4.4}$$

Like in Chapter 3.3, constant multiples can be ignored:

$$B(H) = M(H) + H \tag{4.5}$$

In addition, the bias by H is omitted to retain asymptotic nonlinearity. Given a nonlinear M whose derivative is approximately 0 as H approaches positive and negative infinity, B(H) is asymptotically linear with the bias by H [10, Ch. 6.2]:

$$\frac{dB}{dH}(H_{\gg 0}) = \frac{dM}{dH}(H_{\gg 0}) + \frac{dH_{\gg 0}}{dH} \approx 0 + 1 \tag{4.6}$$

This effect is not desired since character should become more nonlinear (smaller derivate) for higher values of H. Although M is not physically equal to B, they will be treated as equivalent in the remaining derivations. The final from of the JA model without the anhysteretic and irreversible mappings defines a new model for modulating a component value:

$$\frac{dB}{dH}(H) = g_{irr}\frac{dB_{irr}}{dH}(H) + g_{an}\frac{dB_{an}}{dH}(H) \tag{4.7}$$

$$L(x) = g_{irr}L_{irr}(x) + g_{an}L_{an}(x) \tag{4.8}$$

$B_{an}(H)$ is based on the Langevin function:

$$\mathbb{L}(H) = coth(H) + \frac{1}{H} \tag{4.9}$$

The Langevin function combines with two parameters: saturation level, B_s, and input gain, $\frac{1}{a}$.

$$B_{an}(H) = B_s \cdot \mathbb{L}\left(\frac{H}{a}\right) \tag{4.10}$$

B_s is not relevant since it will multiply with g_{an}. The constant $\frac{1}{a}$ is included since it determines the amount of nonlinearity in the interval $x \in (-1,1)$; to represent this term as a gain, a moves to the numerator:

$$B_{an}(H) = \mathbb{L}(a \cdot H) \tag{4.11}$$

Sigmoidal functions are s-shaped functions whose output is bounded in the interval $(-1,1)$. $B_{an}(H)$ is a transcendental sigmoidal function with a numerical instability about $x = 0$ (dividing by 0 in Equation 4.9). As an alternative, this thesis uses the Fröhlich-Kennelly (FK) relation from Chapter 3.2.2 because it is an algebraic sigmoidal function that is numerically stable for all values of x. The equation for $B_{an}(x)$ is a modified version of Equation 3.12 (Equation 4.12); $L_{an}(x)$ is the same as Equation 3.15.

$$B_{an}(x) = \frac{a \cdot x}{1 + |a \cdot x|} \tag{4.12}$$

L_{irr}(x) is defined as a first-order differential equation:

$$L_{irr} = \frac{dB_{irr}}{dH} = \frac{B_{an}(x) - B_{irr}}{\delta k - \alpha(B_{an}(x) - B_{irr})} \tag{4.13}$$

Again, values multiplied by α are ignored. Furthermore, k affects the width of the resulting hysteresis loops, but, with $\alpha = 0$, it behaves as a constant multiple and is also ignored. $\delta = sgn\left(\frac{dx}{dt}\right) = sgn(dx)$; it can combine with the step size dx. The following equations derive B_{irr}:

$$\frac{dB_{irr}}{dH} = \frac{B_{an}(x) - B_{irr}}{\delta}$$

$$B_{irr} = \int_{x_0}^{?} \frac{B_{an}(\chi) - B_{irr}}{\delta} d\chi$$

$$B_{irr} = \int_{x_0}^{x_c} B_{an}(\chi) - B_{irr} |d\chi| \tag{4.14}$$

Where x_c is the integral of all magnitudes of $dx(t)$:

$$x_c = \int_{t_0}^{t} |dx(\tau)| \tag{4.15}$$

This differential equation is in the form of a modified one-pole filter with a feedforward non-cutoff-modulating nonlinearity. This approach is like the concept of timeless integration in [29], but the sampling period is not constant; rather, the effective sampling period, $T = dx$, changes per sample as shown by the total cutoff gain in Figure 4.1. Lastly, the input passes through a static nonlinearity before the filter, $B_{an}(x)$; the value of L_{irr} is the high-pass mode and the value of B_{irr} is the low-pass mode.

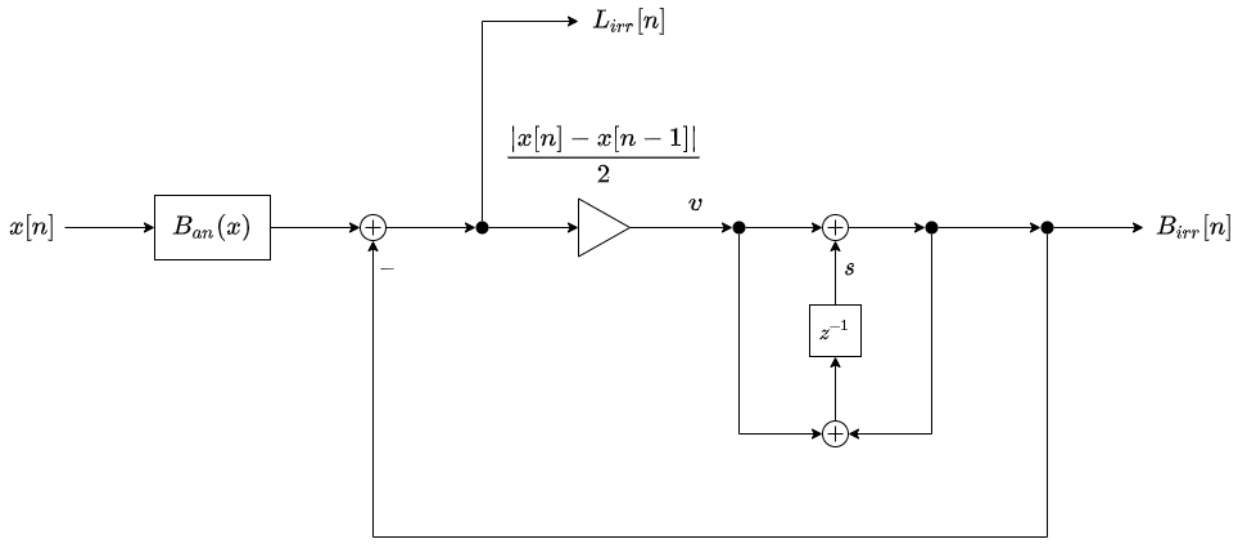

Figure 4.1: Block diagram of the irreversible inductance and magnetic flux density

The final definitions for magnetic flux density and inductance are

$$B(x) = g_{irr} \int_{x_0}^{x_c} B_{an}(\chi) - B_{irr}|d\chi| + g_{an} B_{an}(x) \tag{4.16}$$

And

$$L(x) = g_{irr}(B_{an}(x) - B_{irr}) + \frac{g_{an}}{\left(\sqrt{\frac{1}{L_{lin}}} + N|x|\right)^2} \tag{4.17}$$

Where is $B_{an}(x)$ is Equation 4.12.

Chapter 5

Implementation and Results

The SOUL domain-specific programming language provides a high-level framework for realizing the block diagrams. Each of the four blocks from the character compressor is a 'processor' or 'subgraph' (collection of processors and/or subgraphs); they are combined or isolated depending on the test. This flexibility allows easy measurements at each of the internal block diagram edges.[22]

Chapter 5.1 introduces new test signals to verify the stability and hysteresis of the modified Jiles-Atherton (JA) model. The sine input from Chapter 1.1 demonstrates character from harmonic distortion. Chapter 5.2 tests the sidechain described in Chapter 3.3 and produces responses like Figures 1.11 and 1.15.

5.1 Stability and Response of the Modified Jiles-Atherton Model

Like in Chapter 3.3, the hysteretic definition of $L(x)$ modulates the cutoff of a one-pole filter. However, depending on the filter, parameters may scale to normalize their effects on the output and ensure audible results.[23]

When using $L(x)$ as the cutoff, the following modifications are made. First, $\widetilde{g_{irr}}$ scales the irreversible component so that it is noticeable for high values of the anhysteretic component's linear cutoff, Ω_{lin}.

$$\widetilde{g_{irr}} = f_c \cdot g_{irr}$$

[22] Source code at: https://github.com/thezhe/VirtualAnalogCompressors

[23] For example, using $\frac{1}{L(x)}$ to approximate $\Omega_c(t) = \frac{R}{L(t)}$ requires a different scaling.

$$\Omega_{lin} = L_{lin}$$

Furthermore, to preserve Ω_{lin}, the anhysteretic gain is always one.

$$g_{an} = 1$$

Lastly, $\widetilde{N}$ and $\tilde{a}$ scale the nonlinearity parameters of the two anhysteretic functions so that N and a can combine into a single overall nonlinearity parameter.

$$\tilde{a} = 500 \cdot a$$

$$\widetilde{N} = \frac{N}{50}$$

$$nonlinearity = N = a$$

The final modified version has only three parameters: N, Ω_{lin}, and $\widetilde{g_{irr}}$.

$$L(x) = \widetilde{g_{irr}}(B_{an}(x) - B_{irr}) + \frac{1}{\left(\sqrt{\frac{1}{\Omega_{lin}}} + \frac{N}{50} \cdot |x|\right)^2} \tag{5.1}$$

$B_{an}(x)$ also contains the parameter N.

$$B_{an}(x) = \frac{\tilde{a} \cdot x}{1 + |\tilde{a} \cdot x|} = \frac{500 \cdot N \cdot x}{1 + |500 \cdot N \cdot x|} \tag{5.2}$$

$L(x)$ is also normalized to $\tilde{L}(x)$ such that the Nyquist frequency corresponds to $\frac{\pi}{2} \approx 1.57$, and cutoffs outside of the interval $[0, 1.41)$ are clamped. 1.41 corresponds to 90% of the Nyquist frequency; this limit prevent instability from pre-warping function (Equation 2.18). The resulting expression can be used as value of $g[n]$ in a TPT one-pole filter (Figure 2.8):

$$\tilde{L}(x) = max\left(min\left(L(x) \cdot \frac{T}{2}, 1.41\right)\right) \tag{5.3}$$

The definition of $B(x)$ remains the same as Equation 4.16.

5.1.1 Magnetic Flux Density

The $B(x)$ hysteresis loops should become wider when excited by higher amplitude signals. A 440 Hz sine wave that increases in amplitude (fades-in) triggers this hysteresis effect (Figure 5.1).

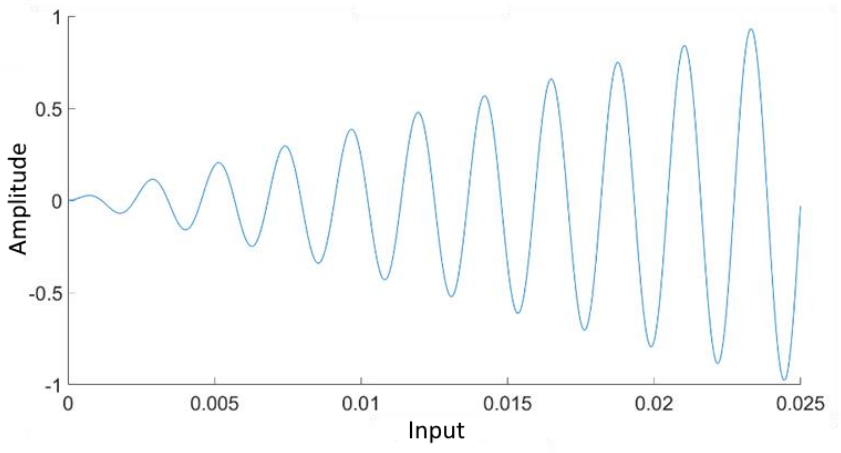

Figure 5.1: A 440 Hz sine wave that fades in linearly

The input-output plot shows increasing loop widths, and the overall shape is sigmoidal (Figure 5.2).

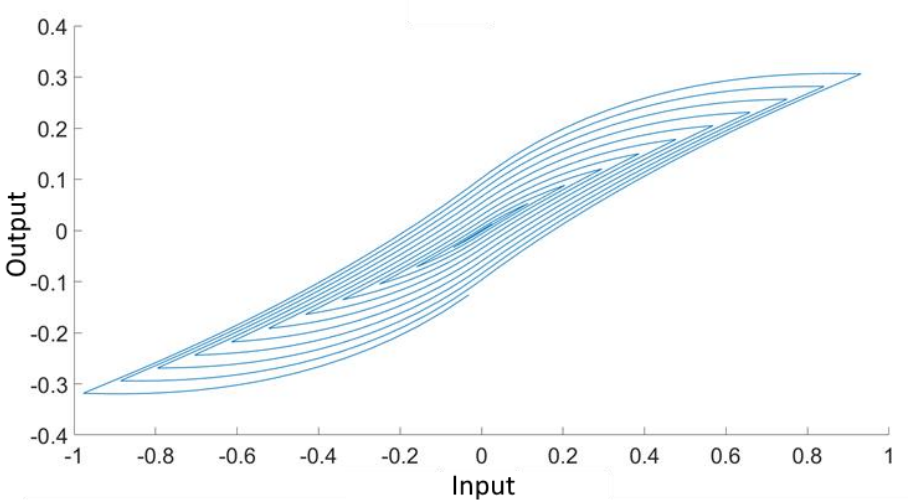

Figure 5.2: $B(x)$ plotted against the fade-in sine input ($a = 1, g_{irr} = 0.5, g_{an} = 0.5$)

Increasing a increases the nonlinearity of the overall shape (Figure 5.3):

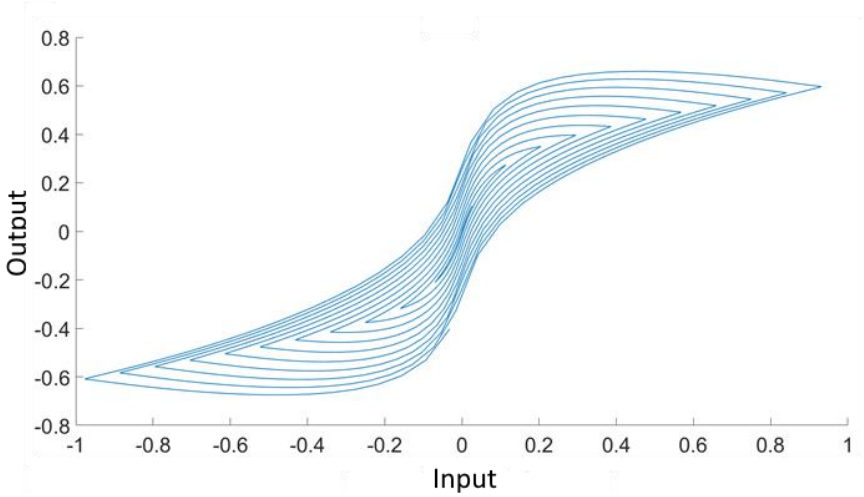

Figure 5.3: $B(x)$ plotted against the fade-in sine input ($\tilde{a} = 10, g_{irr} = 0.5, g_{an} = 0.5$)

Similarly, decreasing $\tilde{a}$ makes the overall shape more linear. In addition, the peak amplitude drops significantly (Figure 5.4).

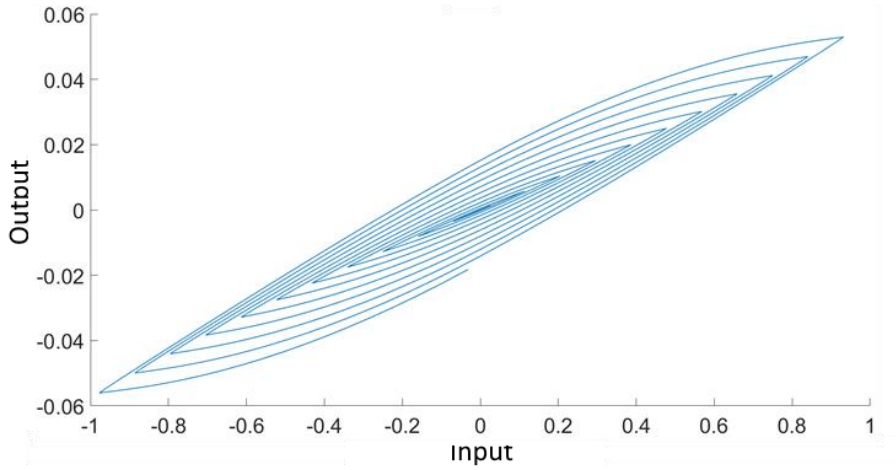

Figure 5.4: $B(x)$ plotted against the fade-in sine input ($\tilde{a} = 0.1, g_{irr} = 0.5, g_{an} = 0.5$)

If $g_{irr} = 0$, $B(x)$ is an anhysteretic sigmoidal curve (Figure 5.5).

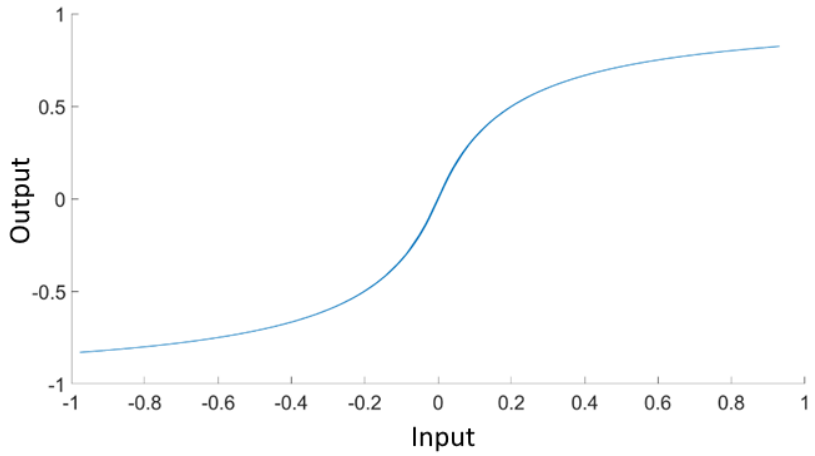

Figure 5.5: $B(x)$ plotted against the fade-in sine input ($\tilde{a} = 5, g_{irr} = 0, g_{an} = 1$)

Likewise, if $g_{an} = 0$, $B(x)$ is an irreversible flux density curve (Figure 5.6). This component adds a bias depending on the system's memory. The range of possible output values is especially large when the current input is zero.

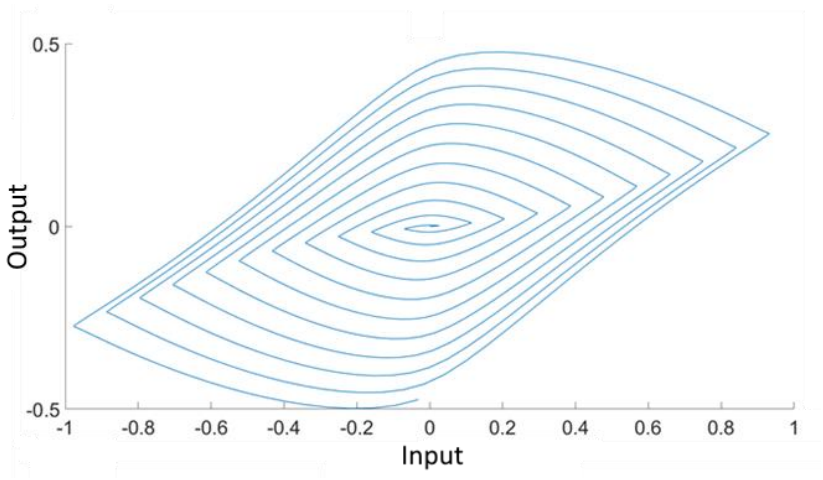

Figure 5.6: $B(x)$ plotted against the fade-in sine input ($\tilde{a} = 5, g_{irr} = 1, g_{an} = 0$)

Figures 5.5 and 5.6 show a separation between the static and dynamic components of the JA model.

A common way to achieve a constant sampling period is to find $\frac{dB}{dt}$ using the chain rule and then calculate $B(x)$ from the integral [26]:

$$\frac{dB}{dt} = \frac{dB}{dx}\frac{dx}{dt} \tag{5.4}$$

This step adds extra calculations and results in a longer, more complicated equation. [26] and [30] also omit the approximation from Equation 4.3; thus, they contain nonlinear feedback loops and require multiple iterations of Newton-Ralphson per sample. Other than the increased computational cost, biased sine inputs can cause instability. Higher order integrators and additional computation are required to achieve stability under these corner cases [30, Sec. 3.2].

A 2kHz sine wave biased by a 50kHz cosine wave (Figure 5.7) causes instability in the model from [30] when implemented using trapezoidal integration. In contrast, the JA model of this thesis should achieve stability using its trapezoidal integration and non-iterative feedforward

cutoff modulation. For this experiment, the sampling rate is set to 192kHz to prevent aliasing of the 50kHz bias signal. The biased input signal is defined as:

$$f[n] = \frac{1}{6}sin(2\pi 2000nT) + \frac{5}{6}cos(2\pi 50000nT) \tag{5.5}$$

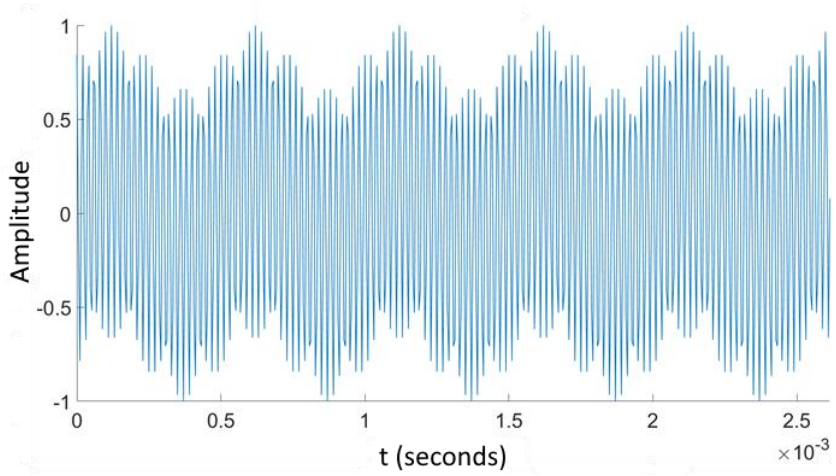

Figure 5.7: Equation 5.5 sampled at 192khz (entire signal is one second long)

$B(x)$ is bounded when excited by the biased sine wave (Figure 5.8). No instability exists since $B_{an}(x)$ is a bounded sigmoid, and B_{irr}(x) is the output of a stable filter with $B_{an}(x)$ as its input.

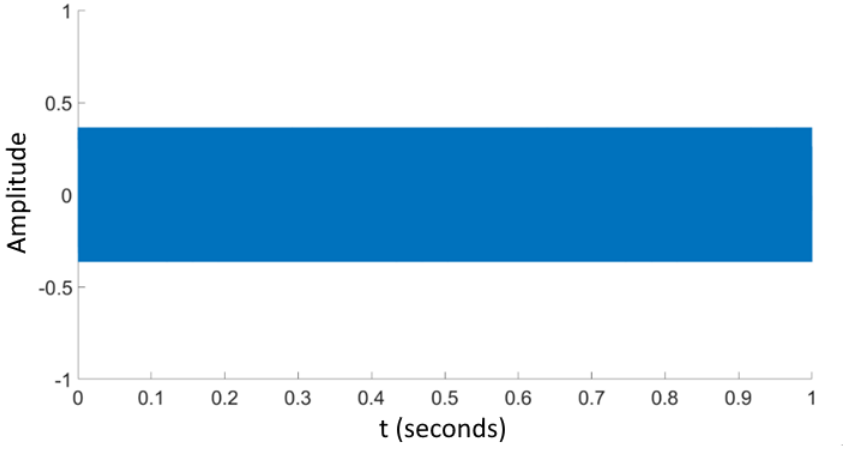

Figure 5.8: $B(x)$ response to the biased sine input ($a = 1, g_{irr} = 0.5, g_{an} = 0.5$)

5.1.2 Cutoff Modulation

Using $\tilde{L}(x)$ as the cutoff frequency of a simple one-pole filter modulates the cutoff and produces complex responses. The anhysteretic response of $\tilde{L}(x)$ is plotted against the sine fade-in input to measure the static response. All tests set the cutoff to 12kHz or approximately 0.39 on the normalized cutoff scale. The experimental result (Figure 5.9) is very similar to the results from the Fröhlich-Kennelly model (Figure 3.6).

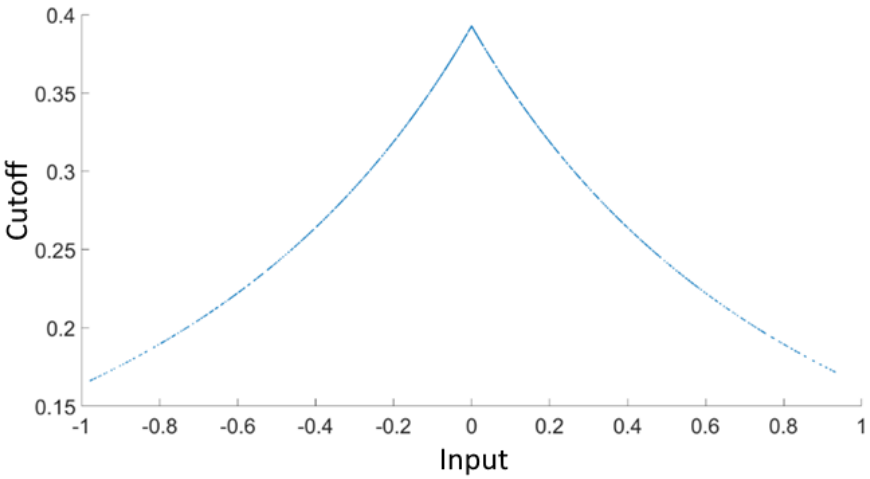

Figure 5.9: $\tilde{L}(x)$ plotted against the fade-in sine input ($N = 0.1, g_{irr} = 0$)

Adding the hysteretic component adds loops as expected (Figure 5.10). In addition, while the hysteretic plot of $B(x)$ shows odd symmetry, the following plot is noticeably asymmetric; $\frac{dB}{dH} = L$ treats positive and negative input values differently.

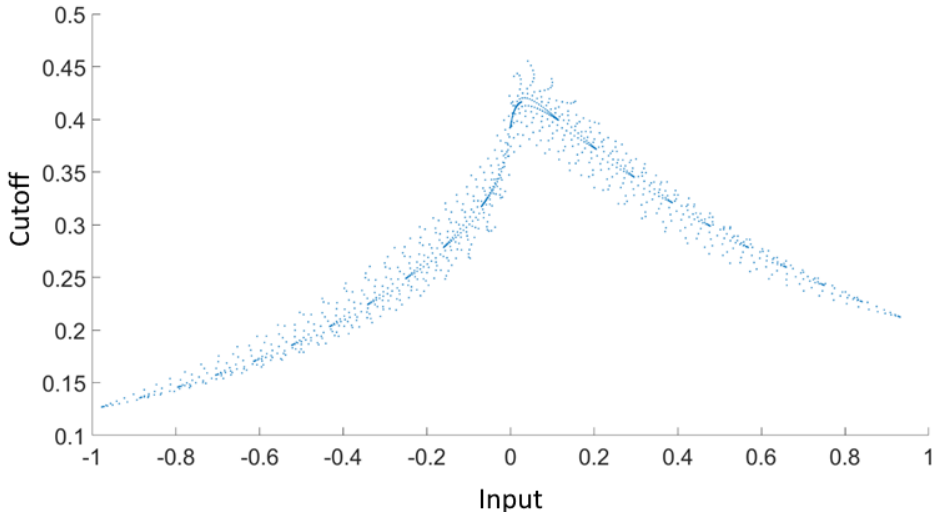

Figure 5.10: $\tilde{L}(x)$ plotted against the fade-in sine input ($N = 0.1, g_{irr} = 1$)

Increasing N creates an even more noticeable nonlinearity (Figure 5.11).

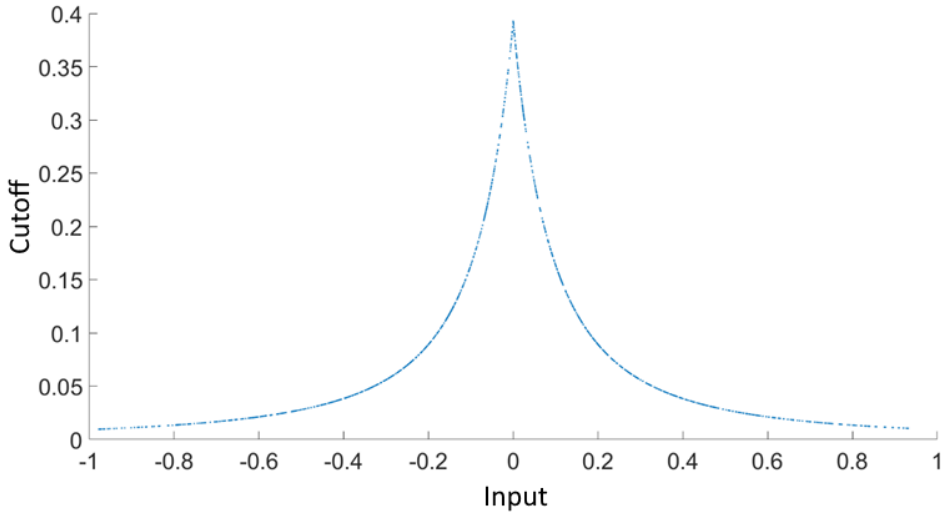

Figure 5.11: $\tilde{L}(x)$ plotted against the fade-in sine input ($N = 1, g_{irr} = 0$)

Since some values of the hysteretic component can be negative, the value of $\tilde{L}(x)$ may also be negative for high values of N; thus, clamping $L(x)$ is important because unstable, negative cutoffs can appear otherwise [29] (Figure 5.12).

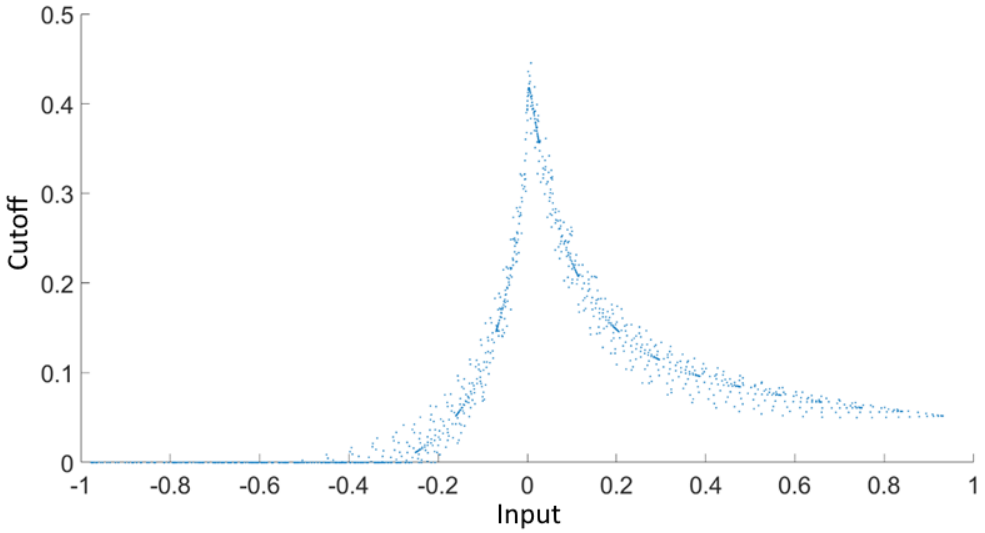

Figure 5.12: $\tilde{L}(x)$ plotted against the fade-in sine input ($N = 1, g_{irr} = 1$)

Passing the biased sine wave through a one-pole filter modulated by $\tilde{L}(x)$ is stable but produces a DC component (Figure 5.13). An extra HPF1 at the output should be in the final design to block unwanted DC. All plots will show the output of the low-pass mode.

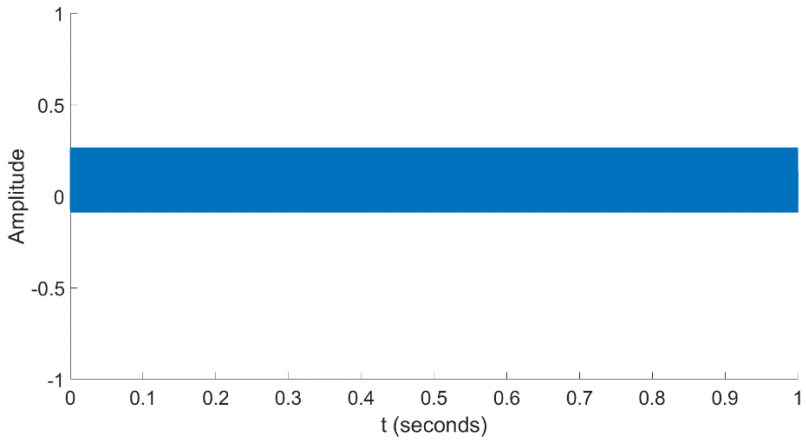

Figure 5.13: The modulating one-pole filter's response to the biased sine input ($N = 0.5, g_{irr} = 1$)

The modulated one-pole filter is nonlinear (i.e., it can be used as a character block), but the output's spectrum which specific harmonics are generated. The same test as Figures 1.13 and

1.20 measures the harmonic response, except the sampling rate for the following test is 384kHz.

If $g_{irr} = 0$, $\tilde{L}(x)$ is symmetric; odd harmonics appear and decrease monotonically (Figure 5.14).

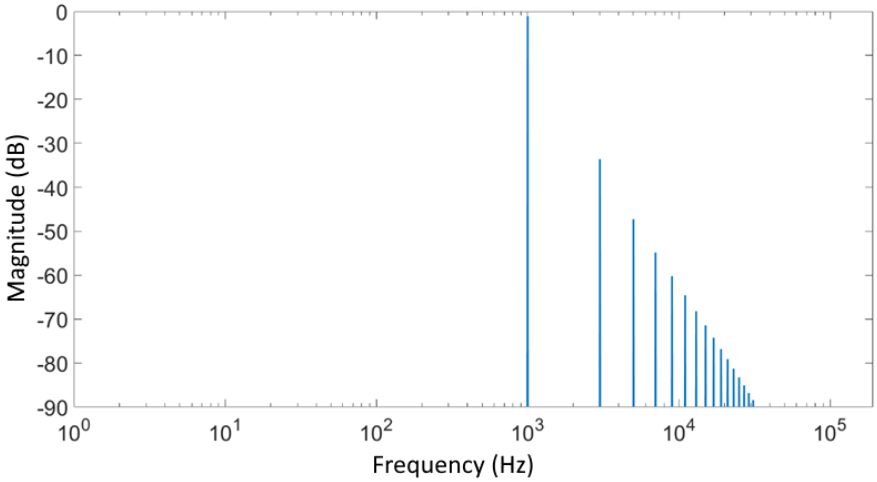

Figure 5.14: The modulating one-pole filter's output spectrum for the sine input ($N = 0.1, g_{irr} = 0$)

If $g_{irr} > 0$, even harmonics appear in the output because of the asymmetric cutoff modulation (Figure 5.15). Furthermore, the magnitudes of the even harmonics are more erratic than the odd harmonics.

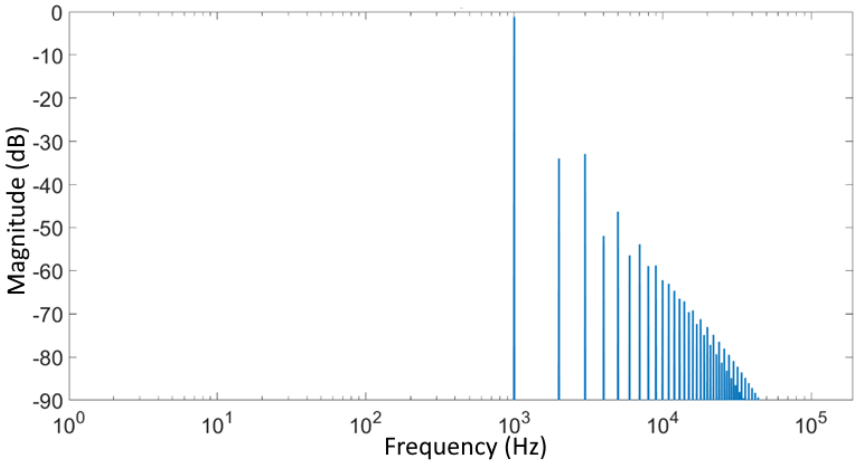

Figure 5.15: The modulating one-pole filter's output spectrum for the sine input ($N = 0.1, g_{irr} = 1$)

5.2 Sidechain Response

The ramp response of the sidechain (Figure 5.16) is very similar to the response of the ATK Colored Compressor (Figure 1.15).

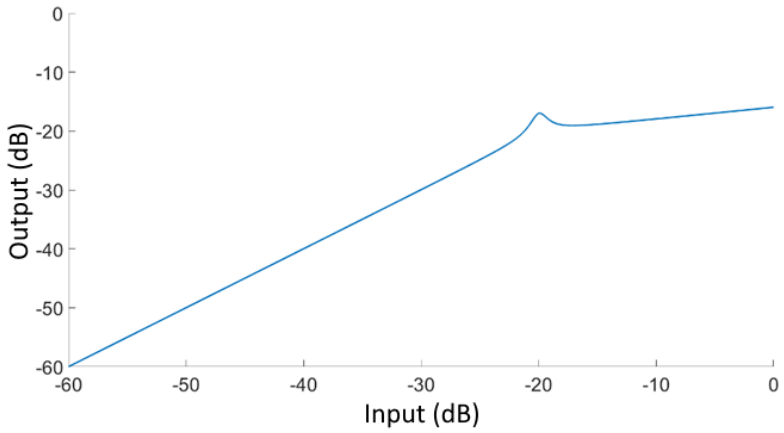

Figure 5.16: The compressor's output plotted against the ramp input (dB scale) ($thr_{dB} = -20$, $R = 5$, $H = 3$, $W = 1$, $\tau_a = \tau_r = 1$, $N = 0$)

The pulse response is like the Waveform compressor when the nonlinearity is set to zero (Figure 5.17).

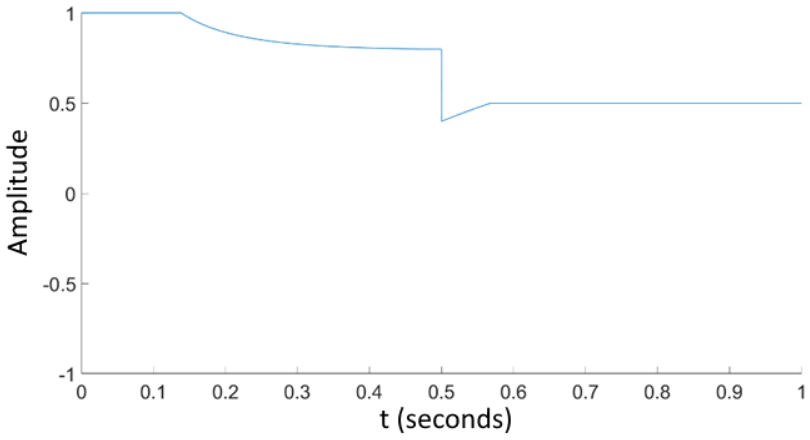

Figure 5.17: The compressor's response to the pulse input ($thr_{dB} = -2.5$, $R = 5$, $H = 0$, $W = 1$, $\tau_a = \tau_r = 100$, $N = 0$)

While the internal code of the Waveform compressor is unknown, the attack and releases are likely scaled by $\frac{1}{2\pi}$ as shown in [2]. After applying this change, the response (Figure 5.18) is almost identical to that of Figure 1.11.

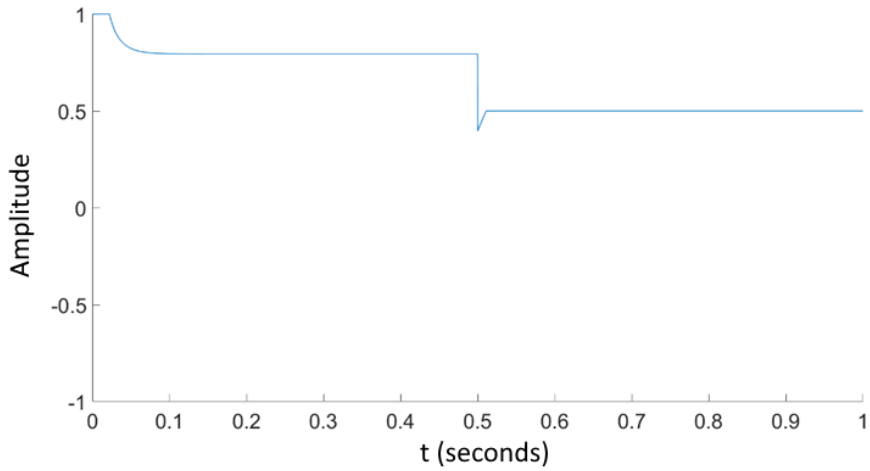

Figure 5.18: The compressor's response to the pulse input ($thr_{dB} = -2.5$, $R = 5$, $H = 0$, $W = 1$, $\tau_a = \tau_r = 100/(2\pi)$, $N = 0$)

Increasing the envelope filter's nonlinearity to 0.1 lengthens and warps the attack and release responses (Figure 5.19).

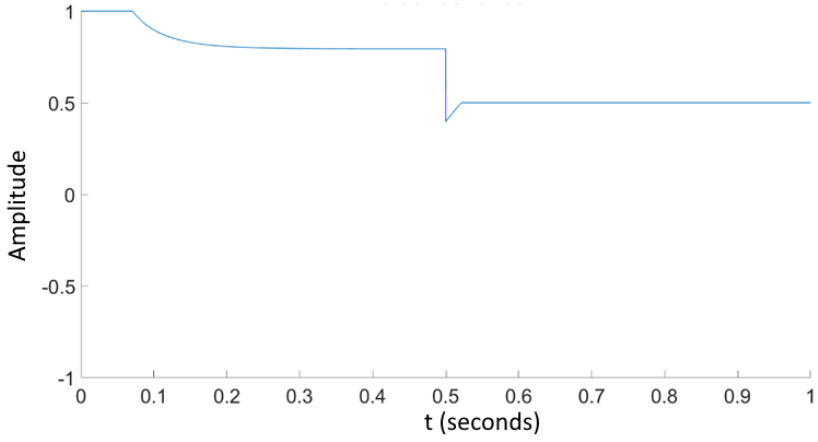

Figure 5.19: The compressor's response to the pulse input ($thr_{dB} = -2.5$, $R = 5$, $H = 0$, $W = 1$, $\tau_a = \tau_r = 100/(2\pi)$, $N = 0.1$)

Lastly, color adds nonmonotonicity to the response (Figure 5.20).

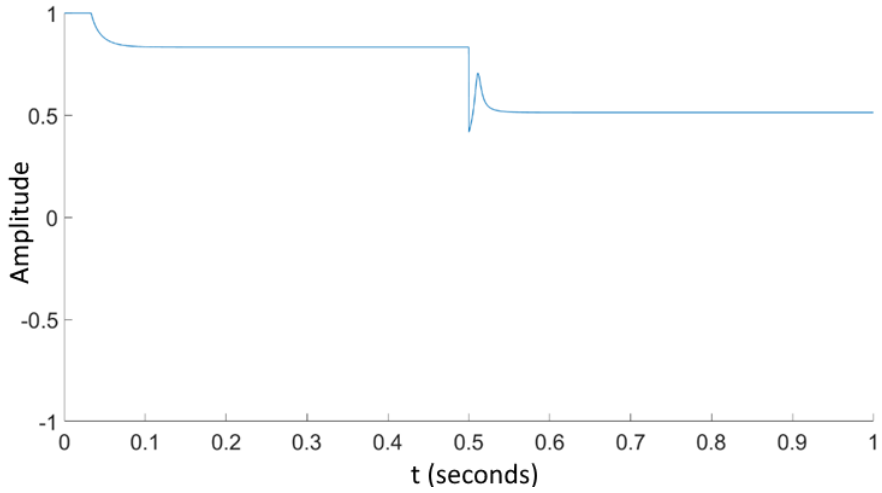

Figure 5.20: The compressor's response to the pulse input ($thr_{dB} = -2.5$, $R = 5$, $H = 3$, $W = 1$, $\tau_a = \tau_r = 100/(2\pi)$, $N = 0$)

Chapter 6

Conclusion

The classical compressor is an important audio engineering tool that can expand into more complex designs. By representing the main components as a block diagram, the designer can modify each block independently to fit a specific use case.

This thesis creates a character compressor by modifying the sidechain and adding a nonlinear input filter. Even though all blocks in the final effect are nonlinear, mathematical simplifications enable an efficient and accurate implementation. The topology-preserving transform interprets dynamic nonlinearities as modulating parameters in linear filter. Furthermore, approximations in the Jiles-Atherton (JA) and Fröhlich-Kennelly (FK) models produce stable, analytical solutions to the nonlinear envelope filter and character blocks.

Future research should include creating a professional audio plugin with higher quality standards. For example, the spectrum in Figure 5.15 shows a 1kHz sine wave producing harmonics near the Nyquist frequency, when sampled at 384kHz; the modulating one-pole filter is prone to aliasing. Antialiasing methods such as oversampling and low-pass filtering in a non-prototype build can better reproduce of the intended effect. Furthermore, this study pre-renders samples and does not measure computational efficiency. A professional, real-time plugin written in C++ can undergo much more performance profiling and provide more insight into the computational advantages of the approximate JA and FK models.

Finally, nonlinear passive components are an often-overlooked aspect of analog modeling. More literature about active components (e.g., vacuum tubes) exists because these

components are much more nonlinear. The techniques used in Chapters 3 and 4 can help develop more accurate models based on other passive components. For example, the memristor is specifically designed to distort and produce hysteresis different than that of an inductor [31]. Decoupling and isolating these effects from active component distortion can highlight sounds that are often overpowered in analog circuits.

Bibliography

[1] Z. Rezar. “Feed-back and Feed-forward Compressor Topology Explained.” Mix Analog. https://blog.mixanalog.com/compressor-topology-explained (accessed Nov. 12, 2020).

[2] I. Cohen, “Ivan Cohen - Fifty shades of distortion (ADC’17),” presented at Audio Developer Conference 2017, London, United Kingdom, Nov. 15, 2017. [Online]. Available: https://www.youtube.com/watch?v=oIChUOV_0w4. Accessed: Dec. 16, 2020.

[3] N. Messitte. “4 Types of Analog Compression—and Why They Matter in a Digital World.” Izotope. https://www.izotope.com/en/learn/4-types-of-analog-compression-and-why-they-matter-in-a-digital-world.html (accessed Nov. 12, 2020).

[4] “Waveform Free.” Tracktion. https://www.tracktion.com/products/waveform-free (accessed June 17, 2021).

[5] M. Brucher. “ATK Colored Compressor.” Plugins 4 Free. https://plugins4free.com/plugin/2340 (accessed June 21, 2021).

[6] D. Giannoulis, M. Massberg, and J. D. Reiss, "Digital Dynamic Range Compressor Design—A Tutorial and Analysis," *Journal of the Audio Engineering Society*, vol. 60, no. 6, pp. 399 - 408, June 2012. Available: https://www.eecs.qmul.ac.uk/~josh/documents/2012/GiannoulisMassbergReiss-dynamicrangecompression-JAES2012.pdf [Accessed Dec. 16, 2020].

[7] “Model LA-2A Leveling Amplifier.” Universal Audio. https://media.uaudio.com/assetlibrary/l/a/la-2a_manual.pdf (accessed June 22, 2021).

[8] “LALA by Analog Obsession.” Plugins 4 Free. https://plugins4free.com/plugin/3221 (accessed June 17, 2021).

[9] SAE Institute USA. “How To: Use the LA-2A Compressor // SAE Institute.” YouTube. https://www.youtube.com/watch?v=x0CmuYG6D4Y (accessed June 17, 2021).

[10] V. Zavalishin, *The Art of VA Filter Design*, 2.1.2. 2020. [Online]. Available: https://www.kvraudio.com/forum/viewtopic.php?t=350246. Accessed: Dec. 16, 2020.

[11] “Response of the First Order System.” Tutorials Point. https://www.tutorialspoint.com/control_systems/control_systems_response_first_order.htm (accessed July 8, 2021).

[12] W. Pirkle, “Virtual Analog (VA) Filter Implementation and Comparisons,” 2013. [Online]. Available: http://www.willpirkle.com/Downloads/AN-4VirtualAnalogFilters.pdf. Accessed: July 8, 2021.

[13] J. O. Smith III. “The Four Direct Forms.” CCRMA, Stanford University. https://ccrma.stanford.edu/~jos/filters/Four_Direct_Forms.html (accessed July 8, 2021).

[14] E. W. Weisstein. “Newton’s Method.” MathWorld. https://mathworld.wolfram.com/NewtonsMethod.html (accessed July 12, 2021).

[15] J. Chowdhury. “Complex Nonlinearities Episode 2: Harmonic Exciter.” Medium. https://jatinchowdhury18.medium.com/complex-nonlinearities-epsiode-2-harmonic-exciter-cd883d888a43 (accessed July 12, 2021).

[16] S. Mansbridge et al, “Implementation and Evaluation of Autonomous Multi-track Fader Control,” *Proc. 132nd Audio Engineering Society Convention*, Budapest, Hungary, Apr. 26-29, 2012. [Online]. Available: https://www.eecs.qmul.ac.uk/~josh/documents/2012/MansbridgeFinnReiss-AES1322012-AutoMultitrackFaders.pdf. Accessed: July 13, 2021.

[17] “weightingFilter.” MathWorks. https://www.mathworks.com/help/audio/ref/weightingfilter-system-object.html (accessed July 18, 2021).

[18] G. Moro and A. P. McPherson, “Approximating Non-Linear Inductors using Time-Variant Linear Filters,” *Proc. 18th Int. Conference on Digital Audio Effects (DAFx-15)*, Trondheim, Norway, Nov. 20-Dec. 3, 2015. [Online]. Available: https://www.dafx.de/paper-archive/2015/DAFx-15_submission_68.pdf. Accessed: July 12, 2021.

[19] P. L. Kirby, “The Non-linearity of Fixed Resistors,” *Electronic Engineering*, Vol. 37, No. 453, pp. 722-726, Nov. 1965. [Online]. Available: https://linearaudio.net/sites/linearaudio.net/files/CB-to-ES-V1-ref-3-IMG_0003.pdf. Accessed: July 18, 2021.

[20] “A Nonlinear Capacitor Model for Use in the PSpice Environment,” Cadence Design Systems, Inc., 2009. [Online]. Available: https://www.ema-eda.com/sites/ema/files/resources/files/nonlinear-capacitor-model.pdf. Accessed July 12, 2021.

[21] C. Johnson. “Capacitor2Proc.cpp.” GitHub. https://github.com/airwindows/airwindows/blob/master/plugins/WinVST/Capacitor2/Capacitor2Proc.cpp (accessed July 12, 2021).

[22] “Capacitance Change with Applied DC Voltage,” Vishay, Doc. 40144, 2012. [Online]. Available: https://www.vishay.com/docs/40144/capchange.pdf. Accessed: July 18, 2021.

[23] N. Messitte. “Expanding on Compression: 3 Overlooked Techniques for Improving Dynamic Range.” iZotope. https://www.izotope.com/en/learn/expanding-on-compression-3-overlooked-techniques-for-improving-dynamic-range.html (accessed July 12, 2021).

[24] M. Brucher. “Audio Toolkit: Additional Gain Functions.” Audio Toolkit. https://blog.audio-tk.com/2015/09/08/audio-toolkit-additional-gain-functions (accessed July 12, 2021).

[25] “Derivative of arctan(x),” Massachusetts Institute of Technology, Boston, Massachusetts, United States, 18.01SC, 2010. [Online]. Available: https://ocw.mit.edu/courses/mathematics/18-01sc-single-variable-calculus-fall-2010/1.-differentiation/part-b-implicit-differentiation-and-inverse-functions/session-15-implicit-differentiation-and-inverse-functions/MIT18_01SCF10_Ses15b.pdf. Accessed: July 18, 2021.

[26] M. Holters and U. Zölzer, “Circuit Simulation with Inductors and Transformers based on the Jiles-Atherton Model of Magnetization,” *Proc. 19th Int. Conference on Digital Audio Effects (DAFx-16)*, Brno, Czech Republic, September 5-9, 2016. [Online]. Available: https://www.hsu-hh.de/ant/wp-content/uploads/sites/699/2017/10/Holters_jamodel_DAFx16.pdf. Accessed: July 12, 2021.

[27] D. C. Jiles and D. L. Atherton, “Theory of ferromagnetic hysteresis,” *Journal of Magnetism and Magnetic Materials*, Vol. 61, pp. 48-60, Jan. 1986. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0304885386900661. Accessed July 18, 2021.

[28] W. Wang. Magnetism. [PowerPoint Slides]. Available: http://depts.washington.edu/mictech/optics/sensors/week2.pdf. Accessed: July 12, 2021.

[29] H. Al-Junaid et al, “Timeless Discretization of Magnetization Slope in the Modeling of Ferromagnetic Hysteresis,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 12, pp. 2757-2764, Dec. 2006, doi: 10.1109/TCAD.2006.882476. [Online]. Available: https://www.academia.edu/12708052/Timeless_Discretization_of_Magnetization_Slope_in_the_Modeling_of_Ferromagnetic_Hysteresis. Accessed: July 12, 2021.

[30] J. Chowdhury, “Real-Time Physical Modelling for Analog Tape Machines,” *Proc. 22nd Int. Conference on Digital Audio Effects (DAFx-19)*, Birmingham, United Kingdom, September 2-6, 2019. [Online]. Available: https://dafx2019.bcu.ac.uk/papers/ DAFx2019_paper_3.pdf. Accessed: July 12, 2021.

[31] B. Hayes. “The Memristor.” American Scientist. https://www.americanscientist.org/article/the-memristor (accessed July 18, 2021).