Audio Engineering Society

Convention Paper 10626

Presented at the 153rd Convention
2022 October



# Reverse Engineering Memoryless Distortion Effects with Differentiable Waveshapers

Joseph T Colonel[1], Marco Comunità[1], and Joshua Reiss[1]

[1]*Centre for Digital Music, Queen Mary University of London, UK*

Correspondence should be addressed to Joseph T Colonel (j.t.colonel@qmul.ac.uk)

**ABSTRACT**

We present a lightweight method of reverse engineering distortion effects using Wiener-Hammerstein models implemented in a differentiable framework. The Wiener-Hammerstein models are formulated using graphic equalizer pre-emphasis and de-emphasis filters and a parameterized waveshaping function. Several parameterized waveshaping functions are proposed and evaluated. The performance of each method is measured both objectively and subjectively on a dataset of guitar distortion emulation software plugins and guitar audio samples.

## 1 Introduction

There is a wealth of research in the fields of linear or nonlinear system identification. The audio research community makes no exception and have been investigating these problems for decades, with a specific attention to virtual analog modeling, i.e. the process of recreating analog devices through digital signal processing [1]. Great attention has been devoted to understand and simulate distortion circuits and vacuum-tube devices (e.g. amplifiers, filters, compressors, oscillators) [2].

More recently, work has been done to reverse engineer audio effects given a dry and wet pair of audio samples [3]. This theoretical framework casts audio effects in such a manner that they can be implemented in a differentiable framework such as Tensorflow. In [3] the authors emulate a linear mixing chain that incorporates gain, panning, equalisation (EQ), and convolutional reverb.

This work expands on reverse engineering audio effects with differentiable digital signal processing by proposing methods for reverse engineering distortion effects using Wiener-Hammerstein (W-H) models.[1] The performance of each method is measured both objectively and subjectively on a dataset of guitar distortion emulation plugins and guitar audio samples.

## 2 Background

Depending on the degree of prior knowledge applied to model a target device, the existing approaches for system identification can be divided into three categories: white-, grey- and black-box.

### 2.1 White-box Modelling

White-box modeling is based on the complete knowledge of the system, uses ordinary/partial differential equations to describe its behaviour and adopts numerical methods to solve them in the continuous or discrete domain. Since they reproduce all the important characteristics of a target device, such models can achieve

[1] Examples can be found here

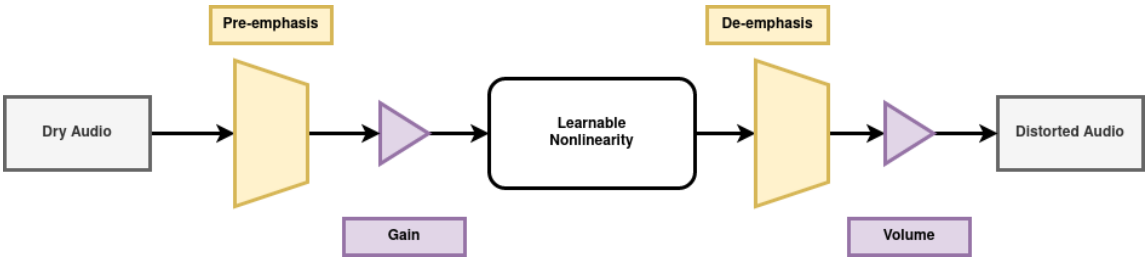



**Fig. 1:** Block diagram of the proposed system.

very good results and are preferable when the sound of a specific analog device is to be replicated with high accuracy. Although, such models can be very time consuming to develop, require exact knowledge of the equations describing nonlinear elements and can result in substantial computational load. Simple systems can be modeled manually by solving differential equations [4, 5, 6]; but, for more complex cases, there exist general-purpose frameworks like: state-space models [7, 8, 9], wave digital filters [10, 11, 12, 13, 14, 15], port-hamiltonian systems [16].

### 2.2 Black-box Modelling

Black-box modeling requires no prior knowledge about the system and relies exclusively on input-output measurements. The main advantage of such approaches is that they simplify the modeling procedure to gathering sufficient data, but they might require time-consuming optimisations and they seldom offer any interpretability. Examples of black-box methods are: Volterra series [17, 18], dynamic convolution [19], neural networks [20, 21].

### 2.3 Grey-box Modelling

Grey-box approaches combine a partial theoretical structure, referred to as block-oriented model, with data - typically input/output measurements - to complete the model. Grey-box models have the advantage of greatly reducing the prior knowledge necessary to model a device while maintaining a degree of interpretability, thanks to the block oriented approach. Although, the specific structure - together with the measurement and optimization procedures - are critical to achieve a good approximation, especially for nonlinear systems where the output is a function of the input signal amplitude. These models are typically represented as an interconnection of linear filters and static nonlinearities, such as: Hammerstein models (static nonlinearity followed by linear filter), Wiener models (linear filter followed by static nonlinearity) or Wiener-Hammerstein models (static nonlinearity inbetween two linear filters) [22, 23, 24, 25]. But they also include more complex arrangements like cascaded and parallel blocks (see [26] for other examples of block-oriented models). In the case of distortion circuits and amplifiers, Wiener-Hammerstein models have been extended to include: non-static nonlinearities (i.e. hysteresis and memory) [27, 28], pre- and power-amp modeling [29, 30, 31].

### 2.4 Neural Networks in White-, Grey- and Black-box Modelling

With the recent rise of machine learning the research community started applying neural networks and differentiable DSP to white-, grey- and black-box modelling. In [32], the authors adopt a deep neural network in the context of a state-space model, which they call state trajectory network. The network uses both the input signal and an internal state to predict the output. The authors apply the method to a first-order and a second-order diode clipper. In [33] the authors introduce the concept of differentiable white-box virtual analog modeling, with the idea of using backpropagation to optimize the components' values in an analog circuit. The authors apply this method to find the resistors' and capacitors' values that best approximate the frequency response of an RC filter and a tone-stack. Another recent work [34] uses recurrent neural networks with fast convolutional layers to model partial-differential equations governed systems. Specifically, they use the proposed approach to investigate: lossy dispersive string, 2D wave equation and tension modulated string.

Many examples of deep learning based black-box modeling are cited in a recent review paper [21]; while applications of DDSP for grey-box modelling are given in [35, 36]. In the first case the authors implement differentiable biquad filters and cascade them with tanh nonlinearities to model a guitar distortion pedal. In the second case, differentiable IIR filters are implemented and used together with a multilayer perceptron to derive a Wiener-Hammerstein model of another distortion pedal.

### 2.5 Wiener-Hammerstein Models

The typical W-H model consists of a linear block, a nonlinear block, and a linear block cascaded in series. The W-H models in this work are time-invariant, formulated using graphic equalizer pre-emphasis and de-emphasis filters for the linear blocks and a parameterized waveshaping function as the nonlinearity. While previous literature focused on power series and Chebyshev polynomials to model the nonlinear blocks [37, 38], in this work we also investigate several waveshaping functions. Figure 1 shows a diagram of the proposed W-H model used in this work.

## 3 Dataset

We assembled a novel dataset of processed electric guitar samples following the same procedure described in [39], using unprocessed recordings from the IDMT-SMT-Audio-Effects dataset [40].

The source dataset[2] includes monophonic (624 single notes) and polyphonic (420 intervals and chords) recordings (wav - 44.1kHz, 16bit, mono) from 2 different electric guitars, each with two pick-up settings and up to 3 plucking styles. The monophonic recordings cover the common pitch range of a 6-string electric guitar, and the polyphonic samples were obtained mixing single notes recordings to generate 2-note intervals and 3- or 4-note chords. All samples are 2 seconds long. The monophonic recordings required removal of background noise before the note onset, which we obtained using a python script together with *Librosa*'s [41] onset detection function.

To assemble our dataset we selected an overdrive, distortion and fuzz plug-ins (see Table 1) designed to emulate some of the most iconic and widely used analogue guitar effect pedals. By selecting 3 different types of distortion plug-ins from 3 different developers we aimed to cover a wide range of timbres and designs while keeping the amount of data limited. All the plugins have 2 or 3 controls and, regardless of the specific name adopted by the designer, the controls can be identified by their processing function: Level, Gain, Tone/Equalisation. A summary of the controls and settings is shown in Table 2. These values were chosen as they were found to be perceptually distinct from one another. The samples' were processed in MATLAB - making use of its VST plugin host features - and both unprocessed inputs and processed outputs were normalised to 0dBFS.

**Table 1:** Plugins used in this work

| *Designer* | *Plugin* | *Emulation of* | *Id* |
|---|---|---|---|
| Audified | Multidrive Pedal Pro | ProCo Rat | RAT |
| Mercuriall | Greed Smasher | Mesa/Boogie Grid Slammer | MGS |
| Analog Obsession | Zupaa | Vox Tone Bender | VTB |

**Table 2:** Plugin settings used to generate the dataset

| *Id* | *Level* | *Gain* | *Tone/Eq* |
|---|---|---|---|
| RAT | [1.0] | [0.2, 0.5, 1.0] | [0.2, 0.8] |
| MGS | [1.0] | [0.2, 0.5, 1.0] | [0.2, 0.8] |
| VTB | [1.0] | [0.1, 0.2, 0.5, 0.8, 1.0] | — |

## 4 Methods

### 4.1 Learnable Graphic EQ

Similar to the formulation in [3], separate 20-band finite impulse response (FIR) graphic EQs are learned for the pre-emphasis and de-emphasis linear blocks [42]. These EQs are calculated using the frequency sampling method as in [43]. First, a frequency transfer curve is specified. The inverse short-time Fourier transform (ISTFT) of this magnitude response is taken using a zero-phase response to obtain the filter's impulse response (IR). Afterward the EQ is applied by

[2] https://www.idmt.fraunhofer.de/en/business_units/m2d/smt/audio_effects.html

multiplying the magnitude response of this new IR with the windowed STFT of the input audio signal.

The 20-band graphic EQ can be characterized using a 20 dimensional $\Theta_{EQ\ gains}$. The 20 values specify the gain of each octave band filters, which are centered at 40, 65, 80, 130, 200, 270, 400, 540, 800, 1000, 1500, 2000, 3000, 4000, 6000, 8000, 12000, and 16000 Hz respectively. Shelving filters are used for frequencies below 40Hz and above 16000 Hz that match the attenuation specified at the lowest and highest octave band respectively.

These 20 values are transformed via

$$\Theta_{EQ\ gains} \leftarrow \sigma(\Theta_{EQ\ gains}) \tag{1}$$

where $\sigma$ denotes the sigmoid function

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{2}$$

The values in the transformed $\Theta_{EQ\ gains}$ range from $(0,1)$ due to the bounds of the sigmoid function.

Finally a piecewise linear frequency transfer curve $\Theta_{EQ}$ is constructed using linear interpolation between the octave band attenuations specified by $\Theta_{EQ\ gains}$. Thus the EQ module's frequency transfer curve is bounded from (0,1) at all points. The estimated values are initialized with random uniform noise from [-1,1], which initializes the octave band gains from -6dB to -1dB.

Because this EQ formulation only allows for the attenuation of frequencies, a gain value is specified in tandem with the pre-emphasis filter, and a volume value is specified in tandem with the de-emphasis filter. It was found that this decoupling of gain and attenuation helps stabilize the optimization.

### 4.2 Waveshaping nonlinearity functions

#### 4.2.1 Tanh Nonlinearity

A hyperbolic tangent (Tanh) with DC offset is used as the baseline nonlinearity in this work. The tanh function

$$\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1} \tag{3}$$

is often used to model distortion effects due to its saturating behavior towards $\pm\infty$. To enable the modelling of nonsymmetric distortion, a DC offset $b_{DC}$ can be applied before the tanh nonlinearity. However, this offset must be removed after the nonlinearity to ensure the output signal maintains no DC offset

$$f(x, b_{DC}) = \tanh(x + b_{DC}) - \tanh(b_{DC}) \tag{4}$$

#### 4.2.2 SumTanh Nonlinearity

Proposed in this work is a family of functions called a "harmonic sum of tanh functions" with DC offset (SumTanh)

$$\begin{aligned} f(x) = a_1 \tanh(x) + a_2 \tanh(2x) + \dots \\ a_{n-1} \tanh((n-1)x) + a_n \tanh(nx) \end{aligned} \tag{5}$$

As a weighted sum of tanh functions, the SumTanh family exhibits saturation towards $\pm\infty$. As a sum of odd functions, the SumTanh family are odd functions, meaning they can model symmetric distortions. Note that $a_c \tanh(c \times 0) = 0$, meaning the waveshaper introduces no DC offset. To model asymmetric distortions, a DC offset can be introduced via

$$\begin{aligned} f(x, b_{DC}) = a_0 + a_1 \tanh(x + b_{DC}) + \dots \\ + a_n \tanh(n(x + b_{DC})) \end{aligned} \tag{6}$$

with $a_0$ set to $-\sum_{c=1}^{n} a_c \tanh(c \times b_{DC})$ to remove the DC component after the nonlinearity. For stability, $a_1$ is initialized close to 1 and $a_c$ are initialized close to 0 otherwise. During optimization, $f(x)$ is normalized such that $\max(|f(x)|) = 1$

#### 4.2.3 PowTanh Nonlinearity

Also proposed in this work is a family of functions called a "power sum of tanh functions" (PowTanh)

$$\begin{aligned} f(x) = a_1 \tanh(x) + a_2 \tanh(x^2) + \dots \\ a_{n-1} \tanh(x^{n-1}) + a_n \tanh(x^n) \end{aligned} \tag{7}$$

As a weighted sum of tanh functions, the PowTanh family exhibits saturation towards $\pm\infty$. However, as a sum of even and odd functions the PowTanh can model both symmetric and asymmetric distortions. Note that $a_c \tanh(0^c) = 0$, meaning the waveshaper introduces no DC offset. As such, no DC offset is included in this parameterization. For stability, $a_1$ is initialized close to 1 and $a_c$ are initialized close to 0 otherwise. During optimization, $f(x)$ is normalized such that $\max(|f(x)|) = 1$

#### 4.2.4 Fourier Series

Because they are well known for their modelling capability, a parameterized Fourier series waveshaper is investigated. An $N^{th}$ degree Fourier waveshaper takes the form

$$f(x) = a_1 \sin(x) + a_2 \sin(2x) + \cdots + a_{\frac{n}{2}} \sin(\frac{n}{2}x) + b_0 + b_1 \cos x + b_2 \cos(2x) + \cdots + b_{\frac{n}{2}} \cos(\frac{n}{2}x) \quad (8)$$

As a sum of even and odd functions, this waveshaper can model both symmetric and asymmetric distortions. However, this waveshaper can introduce a DC offset due to its cosine components, and thus $b_0$ is fixed to $-\sum_{c=1}^{n/2} b_c$.

Traditionally, the support of a Fourier series is $[-\pi, \pi]$. However, a fourier series can exhibit overshoot behavior towards the ends of that support. Therefore, the input audio signal to the waveshaper is normalized between $[-0.9\pi, 0.9\pi]$ to avoid overshoot artefacts. In practice this means the gain parameter in Figure 1 is ignored. As the waveshaper is expected to model distortion, the coefficients of the Fourier series are initialized with an $N^{th}$ order approximation to a square wave. During optimization, $f(x)$ is normalized such that $\max(|f(x)|) = 1$

#### 4.2.5 Legendre Polynomials

Because they are well known for their modelling capability, a parameterized Legendre polynomial waveshaper is also investigated. Legendre polynomials are a family of polynomials $P_n(x)$ orthogonal on $[-1, 1]$ with $P_n(1) = 1$. From Rodrigues' formula these polynomials can be expressed as

$$P_n(x) = \sum_{k=0}^{n} x^k \binom{n}{k} \binom{\frac{n+k-1}{2}}{n} \quad (9)$$

and the waveshaper takes the form

$$f(x) = a_0 + a_1 P_1(x) + a_2 P_2(x) + \cdots + a_n P_n(x) \quad (10)$$

As a sum of even and odd functions, this waveshaper can model both symmetric and asymmetric distortions. However, this waveshaper can introduce a DC offset due to its constant components, and thus $a_0$ is fixed to $-\sum_{c=1}^{n} a_c P_c(0)$.

Similar to the Fourier series, Legendre polynomials can exhibit overshoot behavior towards the edges of its support. Therefore, the input audio signal to the waveshaper is normalized between $[-0.9, 0.9]$ to avoid overshoot artefacts. In practice this means the gain parameter in Figure 1 is ignored. As the waveshaper is expected to model distortion, the coefficients of the Legendre polynomials are initialized with an $N^{th}$ order approximation to the square wave. During optimization, $f(x)$ is normalized such that $\max(|f(x)|) = 1$

**Table 3:** Mean multiscale spectrogram loss of the waveshapers evaluated across pedals

| *Waveshaper* | *RAT* | *MGS* | *VTB* | *Total* |
|---|---|---|---|---|
| Powtanh | 1.321 | **0.559** | **2.117** | **1.332** |
| Sumtanh | **1.321** | 0.573 | 2.400 | 1.431 |
| Fourier | 1.588 | 0.603 | 2.686 | 1.625 |
| Legendre | 1.703 | 0.640 | 2.893 | 1.746 |
| Tanh | 1.353 | 0.597 | 2.478 | 1.476 |

### 4.3 Optimization

Stochastic gradient descent (SGD) is used to update the W-H parameters to fit a dry/wet audio pair. These parameter include the attenuations of each band in both pre-emphasis and de-emphasis filters, the gain parameter, each coefficient in the waveshaper, and the volume parameter. The cost is calculated by passing a dry audio sample through the estimated W-H model and measuring the distance between the estimated and target audio signal. SGD is performed using the Adam method with an initial learning rate of $10^{-3}$. The cost function chosen is multiscale spectrogram loss with window sizes 46ms, 12ms, and 3ms [43]. The optimizations are allowed to run for a maximum of 40000 iterations. Early stopping is employed with a patience of 1000 iterations. The learning rate is dropped to $10^{-4}$ when the first early stopping is reached, or when the optimization reaches 20000 iterations.

## 5 Results

### 5.1 Objective Evaluation

The dataset outlined in Section 3 was used to measure the objective performance of each waveshaping

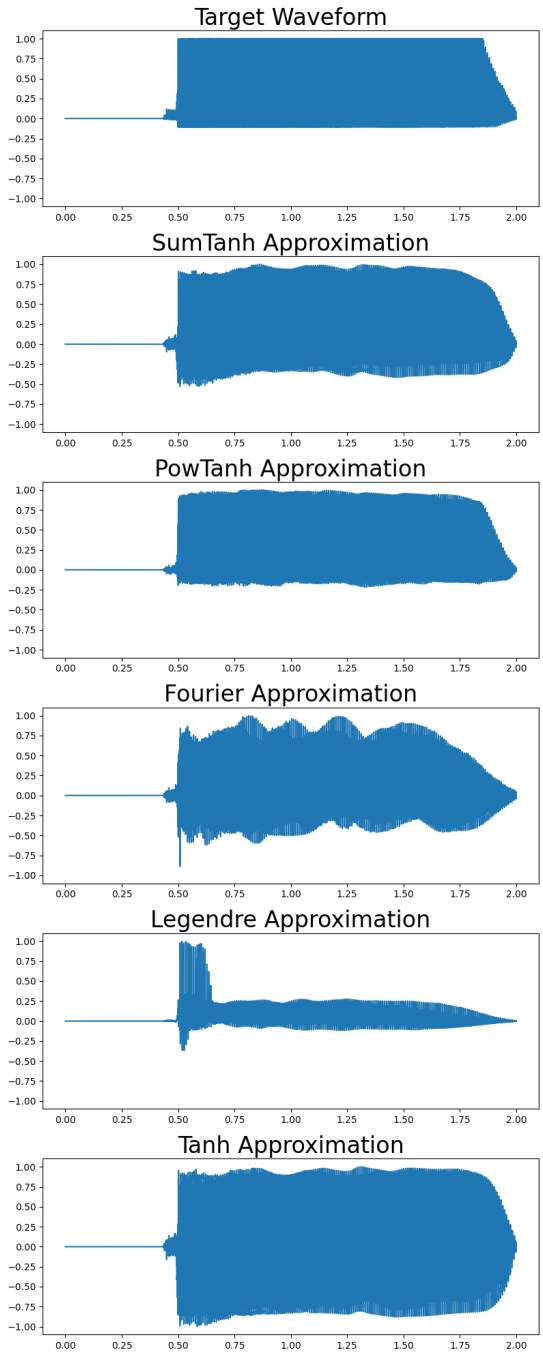



**Fig. 2:** Example of estimated waveforms for each waveshaping model on a VTB example

**Table 4:** Results of pairwise comparison of waveshaping method architecture on perceptual similarity rating across all stimuli, with Bonferroni Correction, o $>$ 0.9, *$<$0.001 ·=no comparison

| | Ref | Sum-Tanh | Pow-Tanh | Fourier | Legendre | Tanh |
|---|---|---|---|---|---|---|
| Ref | · | * | * | * | * | * |
| SumTanh | * | · | o | * | * | o |
| PowTanh | * | o | · | * | * | o |
| Fourier | * | * | * | · | * | * |
| Legendre | * | * | * | * | · | * |
| Tanh | * | o | o | * | * | · |

method. Each waveshaper was modeled using 10 degrees of freedom. Presented in Table 3 are the average multiscale spectrogram losses measured on each waveshaping method across each plugin and across all plugins. Across all plugins, the PowTanh method performs best. Similarly, the PowTanh method outperforms all other methods on the RAT and VTB plugins. On the MGS plugin, the SumTanh performs best. Both PowTanh and SumTanh methods outperform the Tanh baseline on all tasks. Notably, the Fourier and Legendre waveshapers did not outperform the Tanh baseline on any of the tasks.

### 5.2 Subjective Evaluation

A listening test was conducted using webMUSHRA on a subset of 12 samples from the dataset [44]. This subset consists of two unique monophonic and two unique polyphonic stimuli passed through each of the three distortion plugins. No plugin parameter settings were repeated across any of the stimuli. Participants were asked to rate how closely each of the learned W-H models' outputs matched a reference signal, with 0 representing a poor match and 1 representing a perfect match. The reference signal was included as a hidden anchor. A total of 17 participants took part in the study, with an average age of 31 years and standard deviation of 5.34. 8 participants identified as men, 7 as women, and 3 as nonbinary or gender nonconforming. 11 participants reported having at least 5 years of experience with music production or audio engineering, 2 reported having 3 years experience, and 4 participants reported no experience. No participants reported any diagnosed

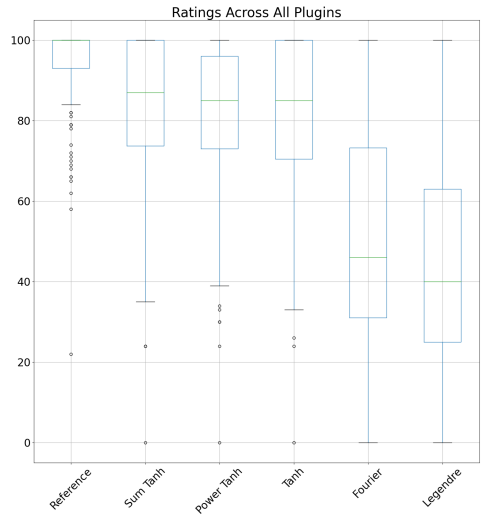



**Fig. 3:** Box and whisker plots of participant ratings across all plugins.

hearing impairments. Box and whisker plots of the participants' ratings are presented in Figure 3.

The analysis that follows is adapted from the perceptual study presented in [45]. The null hypothesis is that the perceptual evaluation scores are from the same distribution. A one-way ANOVA, with Bonferroni correction, shows for all stimuli that the effect each waveshaper had on user perception was statistically significant.

With the null hypothesis rejected, a post-hoc Tukey pairwise comparison, with Bonferroni correction to reduce the chance of type I errors, was used. Table 4 shows the results of these pairwise comparisons for all architectures used. The pairwise comparisons demonstrate that across all plugins, the perception of each waveshaping method differs significantly from the reference. However, when broken down by plugin type the perception of the SumTanh model does not differ significantly from the reference for the MGS and VTB effects. For the MGS effects the caluclated p-value between the reference stimulus and SumTanh stimulus is 0.372, and for the the VTB effects the calculated p-value is 0.064.

## 6 Discussion

Despite outperforming all models in the objective evaluation, the PowTanh waveshaper did not hold up to perceptual evaluation. Instead, the SumTanh model proved to be the most perceptually accurate model. This demonstrates that multiscale spectrogram loss does not necessarily correlate to perceptual closeness. Both the Fourier and Legendre waveshapers performed worse than the baseline Tanh model in the perceptual evaluations.

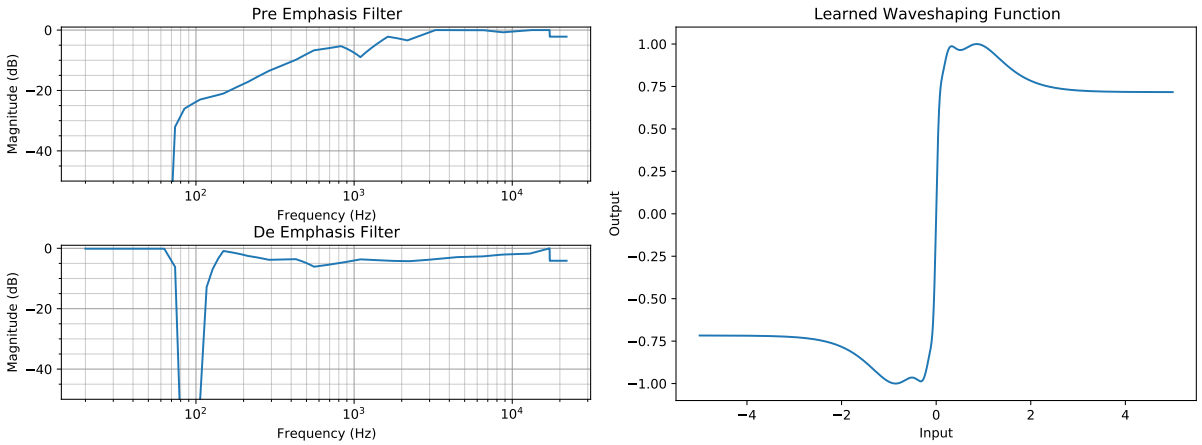
Figure 2 shows the estimated waveforms of learned W-H models for each waveshaper given a VTB target. Both the SumTanh and PowTanh models are able to match the asymmetric distortion well. The Fourier model exhibits an obvious tremolo-like artefact, and the Legendre model shows an overshoot behavior during the attack of the sample. These behaviors were typical across the objective study. The Tanh model was unable to learn the asymmetry in the waveform, a behavior that deserves future exploration.

Figure 4 shows the learned EQs and waveshaping function for the SumTanh model mentioned above. In this example the learned gain is 16.084, DC offset is $-0.026$, and volume is 1.170. The pre-emphasis filter appears to mimic a standard high-pass filter with a cut-off frequency of 1kHz, and the de-emphasis filter has a slight attenuation for most of the middle frequencies and a steep notch at 100Hz. The waveshaper learned to have a slight overshoot for values close to 0 and to saturate at a value less than 1. Future work may involve better understanding the trajectories of these learned parameters over the course of the optimization.

Further studies must be undertaken to fully understand where the SumTanh and PowTanh models underperformed in the perceptual evaluation. One potential issue may be that anti-aliasing is not explicitly addressed in either of these models. While the multiscale spectrogram loss would penalize aliasing harmonics, most waveshaping distortion models explicitly account for anti-aliasing. While internal oversampling may be feasible within the DDSP framework, it would certainly be possible to oversample the dry/wet audio pairs and apply a fixed lowpass filter after the de-emphasis filter when fitting parameters. The authors note that computation time to fit an example would scale with the oversampling factor.

## 7 Conclusion

A lightweight method for reverse engineering a memoryless audio distortion effect given a dry/wet pair




(A) (B)

**Fig. 4:** Example of learned parameters using the SumTanh waveshaping model on the VTB stimulus. (A) Pre-emphasis filter (top) and de-emphasis filter (bottom), (B) Learned waveshaping function

has been presented. This method utilizes Wiener-Hammerstein models implemented in a differentiable framework. These models consist of graphic EQs for their linear blocks and learnable waveshapers for the nonlinear block. Newly proposed families of learnable waveshaping functions have been outlined and are demonstrated to outperform a baseline Tanh nonlinearity as well as Fourier series and Legendre polynomial learnable waveshapers. Ultimately the proposed SumTanh family performs best in a perceptual evaluation, and the proposed PowTanh family performs best in an objective evaluation.

## 8 Acknowledgement

JC is supported by the Research and Development Division of Yamaha Corporation, Japan.

## References

[1] Yeh, D. T., Abel, J. S., Vladimirescu, A., and Smith, J. O., "Numerical methods for simulation of guitar distortion circuits," *Computer Music Journal*, 32(2), pp. 23–42, 2008.

[2] Pakarinen, J. and Yeh, D. T., "A review of digital techniques for modeling vacuum-tube guitar amplifiers," *Computer Music Journal*, 33(2), pp. 85–100, 2009.

[3] Colonel, J. T. and Reiss, J., "Reverse engineering of a recording mix with differentiable digital signal processing," *The Journal of the Acoustical Society of America*, 150(1), pp. 608–619, 2021.

[4] Yeh, D. T., Abel, J., and Smith, J. O., "Simulation of the diode limiter in guitar distortion circuits by numerical solution of ordinary differential equations," *Proceedings of the Digital Audio Effects (DAFx'07)*, pp. 197–204, 2007.

[5] D'Angelo, S. and Välimäki, V., "Generalized Moog ladder filter: Part II–explicit nonlinear model through a novel delay-free loop implementation method," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(12), pp. 1873–1883, 2014.

[6] Esqueda, F., Pöntynen, H., Parker, J. D., and Bilbao, S., "Virtual analog models of the Lockhart and Serge wavefolders," *Applied Sciences*, 7(12), p. 1328, 2017.

[7] Mačák, J., *Real-time digital simulation of guitar amplifiers as audio effects*, Ph.D. thesis, Ph. D. thesis, Brno University of Technology, Brno, 2012.

[8] Holters, M. and Zölzer, U., "A generalized method for the derivation of non-linear state-space models from circuit schematics," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 1073–1077, IEEE, 2015.

[9] Yeh, D. T., Abel, J. S., and Smith, J. O., "Automated physical modeling of nonlinear audio circuits for real-time audio effects—Part I: Theoretical development," *IEEE transactions on audio, speech, and language processing*, 18(4), pp. 728–737, 2009.

[10] Werner, K. J., Smith, J. O., and Abel, J. S., "Wave digital filter adaptors for arbitrary topologies and multiport linear elements," in *Proc. 18th Int. Conf. Digital Audio Effects*, pp. 379–386, 2015.

[11] Werner, K. J., Bernardini, A., Smith, J. O., and Sarti, A., "Modeling circuits with arbitrary topologies and active linear multiports using wave digital filters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(12), pp. 4233–4246, 2018.

[12] Dunkel, W. R., Rest, M., Werner, K. J., Olsen, M. J., and Smith III, J. O., "The Fender Bassman 5F6-A family of preamplifier circuits—a wave digital filter case study," in *Proceedings of the 19th International Conference on Digital Audio Effects (DAFx-16), Brno, Czech Republic*, pp. 5–9, 2016.

[13] Cauduro Dias de Paiva, R., Pakarinen, J., Välimäki, V., and Tikander, M., "Real-time audio transformer emulation for virtual tube amplifiers," *EURASIP Journal on Advances in Signal Processing*, 2011, pp. 1–15, 2011.

[14] De Sanctis, G. and Sarti, A., "Virtual analog modeling in the wave-digital domain," *IEEE transactions on audio, speech, and language processing*, 18(4), pp. 715–727, 2009.

[15] D'Angelo, S., Pakarinen, J., and Valimaki, V., "New family of wave-digital triode models," *IEEE transactions on audio, speech, and language processing*, 21(2), pp. 313–321, 2012.

[16] Falaize, A. and Hélie, T., "Passive guaranteed simulation of analog audio circuits: A port-Hamiltonian approach," *Applied Sciences*, 6(10), p. 273, 2016.

[17] Hélie, T., "On the use of Volterra series for real-time simulations of weakly nonlinear analog audio devices: Application to the Moog ladder filter," in *Proceedings of the 9th International Conference on Digital Audio Effects (DAFx'06)*, pp. 7–12, Citeseer, 2006.

[18] Orcioni, S., Terenzi, A., Cecchi, S., Piazza, F., and Carini, A., "Identification of Volterra models of tube audio devices using multiple-variance method," *Journal of the Audio Engineering Society*, 66(10), pp. 823–838, 2018.

[19] Primavera, A., Cecchi, S., Romoli, L., Gasparini, M., and Piazza, F., "Approximation of dynamic convolution exploiting principal component analysis: Objective and subjective quality evaluation," in *Audio Engineering Society Convention 133*, Audio Engineering Society, 2012.

[20] Ibnkahla, M., "Nonlinear system identification using neural networks trained with natural gradient descent," *EURASIP Journal on Advances in Signal Processing*, 2003(12), pp. 1–9, 2003.

[21] Vanhatalo, T., Legrand, P., Desainte-Catherine, M., Hanna, P., Brusco, A., Pille, G., and Bayle, Y., "A Review of Neural Network-Based Emulation of Guitar Amplifiers," *Applied Sciences*, 12(12), p. 5894, 2022.

[22] Novak, A., Simon, L., Lotton, P., and Kadlec, F., "Modeling of nonlinear audio systems using swept-sine signals: Application to audio effects," in *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, pp. 1–4, 2009.

[23] Novak, A., Simon, L., and Lotton, P., "Extension of Generalized Hammerstein model to non-polynomial inputs," in *2016 24th European Signal Processing Conference (EUSIPCO)*, pp. 21–25, IEEE, 2016.

[24] Cauduro Dias de Paiva, R., Pakarinen, J., and Välimäki, V., "Reduced-complexity modeling of high-order nonlinear audio systems using swept-sine and principal component analysis," in *Audio Engineering Society Conference: 45th International Conference: Applications of Time-Frequency Processing in Audio*, Audio Engineering Society, 2012.

[25] Rébillat, M., Hennequin, R., Corteel, E., and Katz, B. F., "Prediction of harmonic distortion generated by electro-dynamic loudspeakers using cascade of Hammerstein models," in *Audio Engineer-*

*ing Society Convention 128*, Audio Engineering Society, 2010.

[26] Schoukens, J. and Ljung, L., "Nonlinear system identification: A user-oriented road map," *IEEE Control Systems Magazine*, 39(6), pp. 28–99, 2019.

[27] Eichas, F., Möller, S., and Zölzer, U., "Block-oriented modeling of distortion audio effects using iterative minimization," *Proc. Digital Audio Effects (DAFx-15), Trondheim, Norway*, 2015.

[28] Eichas, F. and Zölzer, U., "Black-box modeling of distortion circuits with block-oriented models," in *Proceedings of the International Conference on Digital Audio Effects (DAFx), Brno, Czech Republic*, pp. 5–9, 2016.

[29] Kemper, C., "Musical instrument with acoustic transducer," 2014, uS Patent 8,796,530.

[30] Eichas, F., Möller, S., and Zölzer, U., "Block-oriented gray box modeling of guitar amplifiers," in *Proceedings of the International Conference on Digital Audio Effects (DAFx), Edinburgh, UK*, pp. 5–9, 2017.

[31] Eichas, F. and Zölzer, U., "Gray-box modeling of guitar amplifiers," *Journal of the Audio Engineering Society*, 66(12), pp. 1006–1015, 2018.

[32] Parker, J. D., Esqueda, F., and Bergner, A., "Modelling of nonlinear state-space systems using a deep neural network," in *Proceedings of the International Conference on Digital Audio Effects (DAFx), Birmingham, UK*, pp. 2–6, 2019.

[33] Esqueda, F., Kuznetsov, B., and Parker, J. D., "Differentiable White-Box Virtual Analog Modeling," in *2021 24th International Conference on Digital Audio Effects (DAFx)*, pp. 41–48, 2021.

[34] Parker, J. D., Schlecht, S. J., Rabenstein, R., and Schäfer, M., "Physical Modeling using Recurrent Neural Networks with Fast Convolutional Layers," 2022.

[35] Nercessian, S., Sarroff, A., and Werner, K. J., "Lightweight and interpretable neural modeling of an audio distortion effect using hyperconditioned differentiable biquads," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 890–894, IEEE, 2021.

[36] Kuznetsov, B., Parker, J. D., and Esqueda, F., "Differentiable IIR filters for machine learning applications," in *Proc. Int. Conf. Digital Audio Effects (eDAFx-20)*, pp. 297–303, 2020.

[37] Novak, A., Simon, L., Lotton, P., and Gilbert, J., "Chebyshev model and synchronized swept sine method in nonlinear audio effect modeling," in *Proc. 13th Int. Conference on Digital Audio Effects (DAFx-10)*, p. 15, 2010.

[38] Novak, A., Simon, L., and Lotton, P., "Analysis, synthesis, and classification of nonlinear systems using synchronized swept-sine method for audio effects," *EURASIP Journal on Advances in Signal Processing*, 2010, pp. 1–8, 2010.

[39] Comunità, M., Stowell, D., and Reiss, J. D., "Guitar Effects Recognition and Parameter Estimation With Convolutional Neural Networks," *J. Audio Eng. Soc*, 69(7/8), pp. 594–604, 2021.

[40] Stein, M., Abeßer, J., Dittmar, C., and Schuller, G., "Automatic detection of audio effects in guitar and bass recordings," in *Audio Engineering Society Convention 128*, Audio Engineering Society, 2010.

[41] McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., and Nieto, O., "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, volume 8, pp. 18–25, Citeseer, 2015.

[42] Välimäki, V. and Reiss, J. D., "All About Audio Equalization: Solutions and Frontiers," *Applied Sciences*, 6(5), 2016, ISSN 2076-3417, doi:10.3390/app6050129.

[43] Engel, J., Gu, C., Roberts, A., et al., "DDSP: Differentiable Digital Signal Processing," in *International Conference on Learning Representations*, 2019.

[44] Schoeffler, M., Bartoschek, S., Stöter, F.-R., Roess, M., Westphal, S., Edler, B., and Herre, J., "webMUSHRA—A comprehensive framework for web-based listening tests," *Journal of Open Research Software*, 6(1), 2018.

[45] Moffat, D. and Reiss, J. D., "Perceptual evaluation of synthesized sound effects," *ACM Transactions on Applied Perception (TAP)*, 15(2), pp. 1–19, 2018.