AES NEW YORK
AUDIO ENGINEERING CONVENTION

Reverse Engineering Memoryless Distortion Effects with Differentiable Waveshapers

Joseph T. Colonel, Marco Comunità, Joshua D. Reiss

Queen Mary University of London

centre for digital music

YAMAHA

Overview

Distortion modelling:

- Black-, Grey- and White-box modelling
- Wiener-Hammerstein models

Differentiable Digital Signal Processing

Differentiable Waveshapers

- Sumtanh family
- Powtanh family

Objective Evaluation & Listening Test

Distortion Effect Modelling

White-box Models:

- Based on complete knowledge of the system
- Ordinary/partial differential equations to describe behaviour
- Numerical methods to solve them in the continuous or discrete domain

Black-box Models:

- No prior knowledge about the system
- Only input-output measurements
- Includes end-to-end neural network models

Grey-box Models:

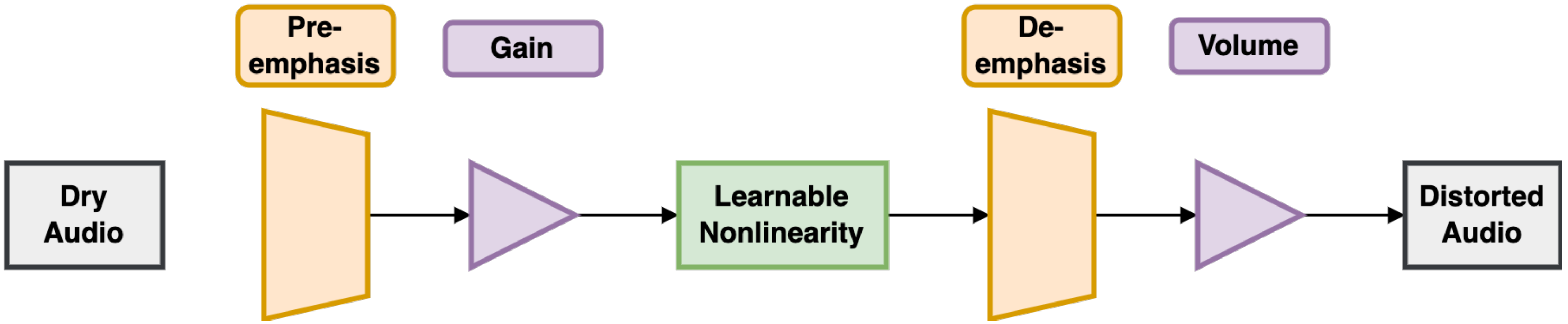
- Partial theoretical structure (Block-oriented model) + Data
- Reduce prior knowledge necessary to model device
- Maintain a degree of interpretability

Wiener-Hammerstein Grey-box Model

Differentiable Digital Signal Processing

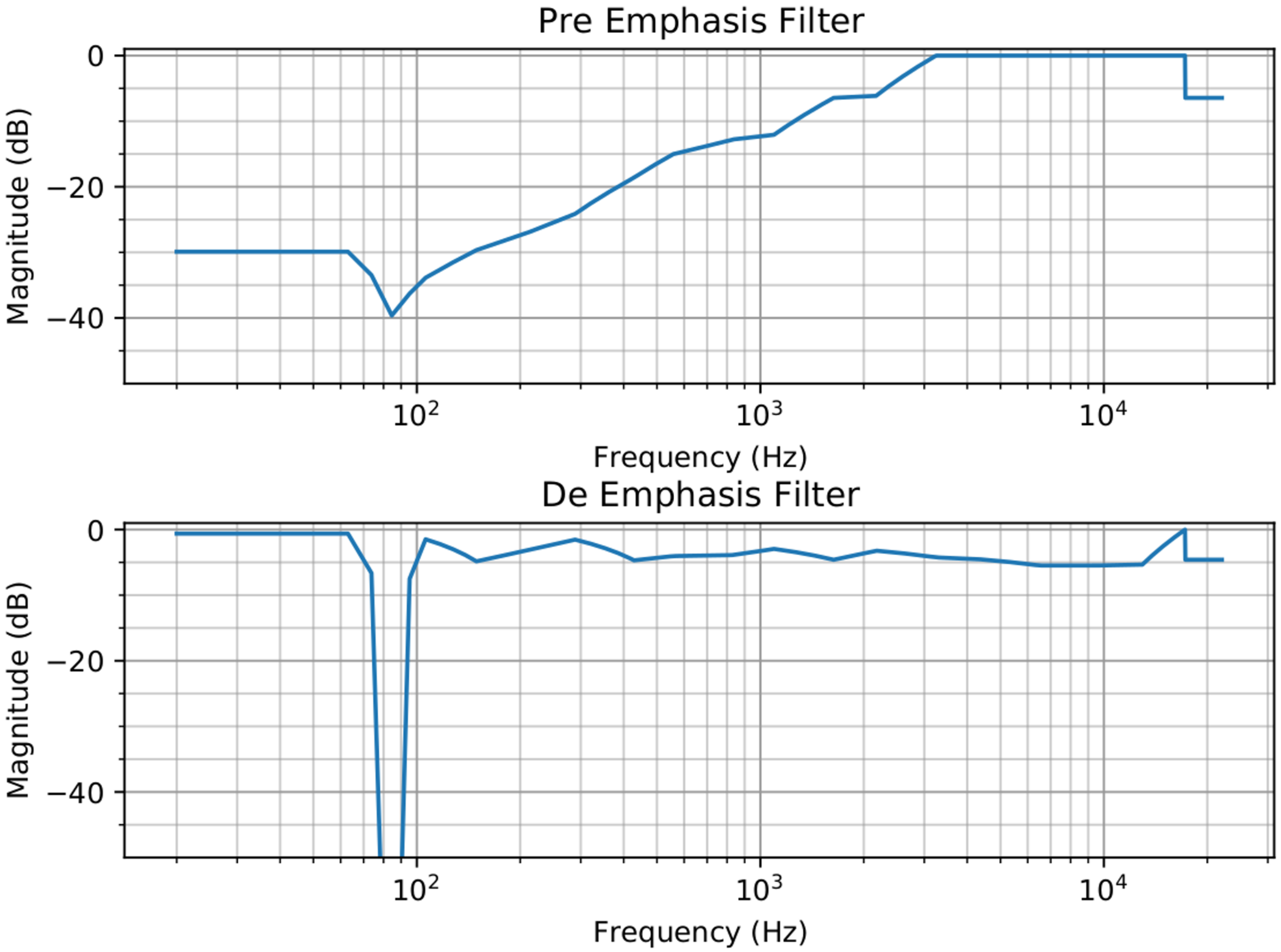
- Common DSP modules implemented in a differentiable framework (e.g., Tensorflow or Pytorch)
- Auto differentiation allows for these modules to be implemented in or controlled by neural networks due to their ability to backpropagate gradients

Proposed Wiener-Hammerstein Model



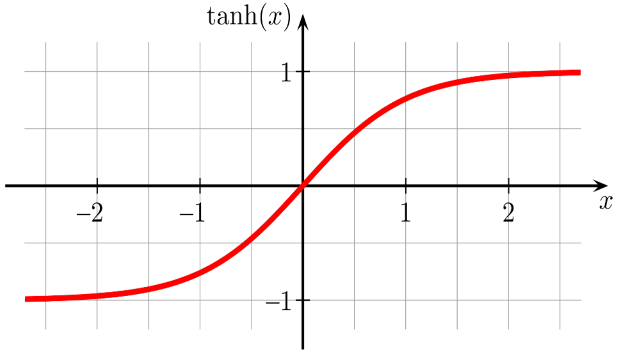
Pre-emphasis and De-emphasis topology

- “Graphic” FIR EQ
- Attenuations specified at semi-octave bands + low and high shelves
- Transfer curve interpolated between these values
- Windowing + spectral sampling method for inversion to time domain

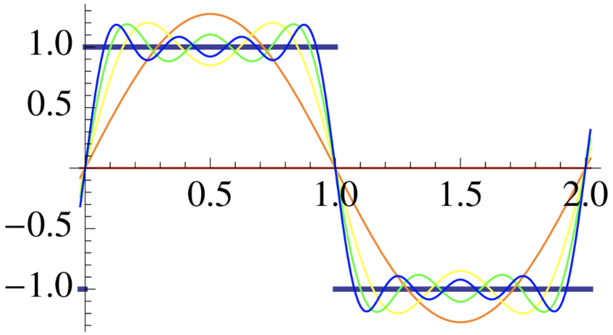


Differentiable Waveshapers

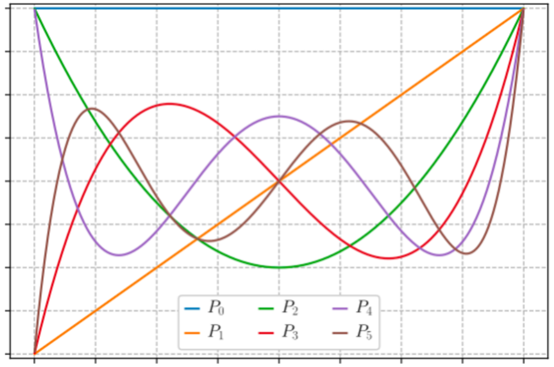
Hyperbolic tangent (tanh) nonlinearity



Fourier Series



Legendre Polynomial Series



Sumtanh nonlinearity

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

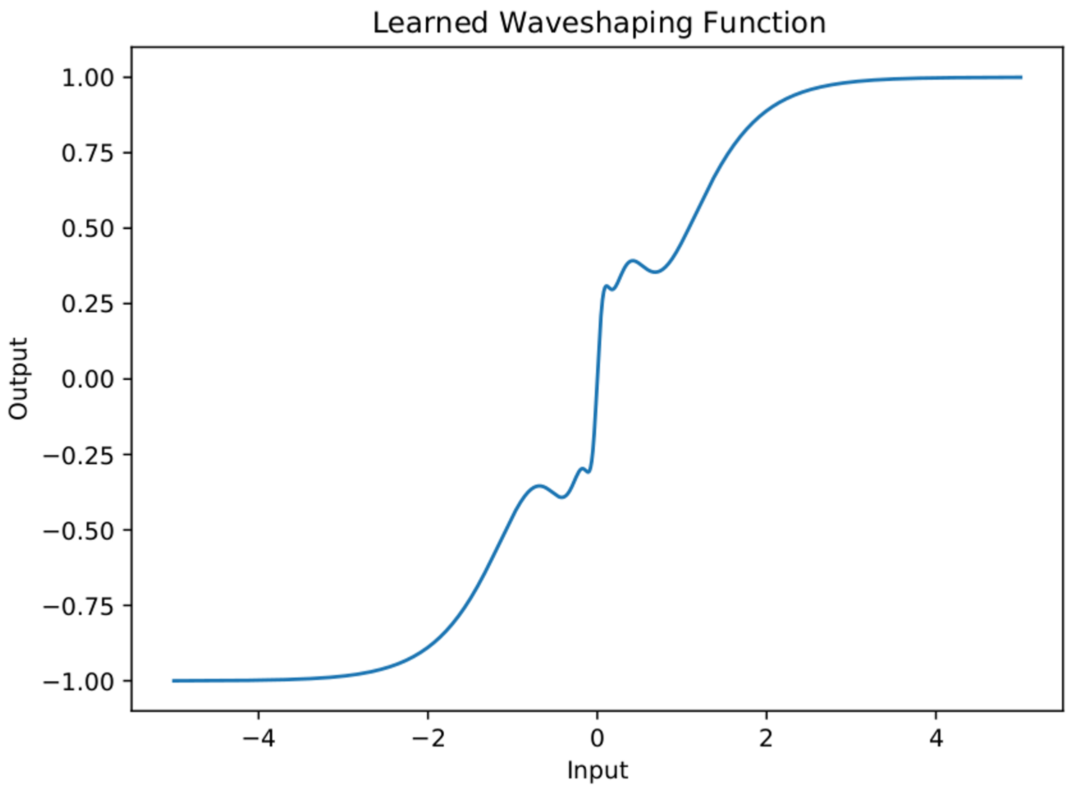
Powtanh nonlinearity

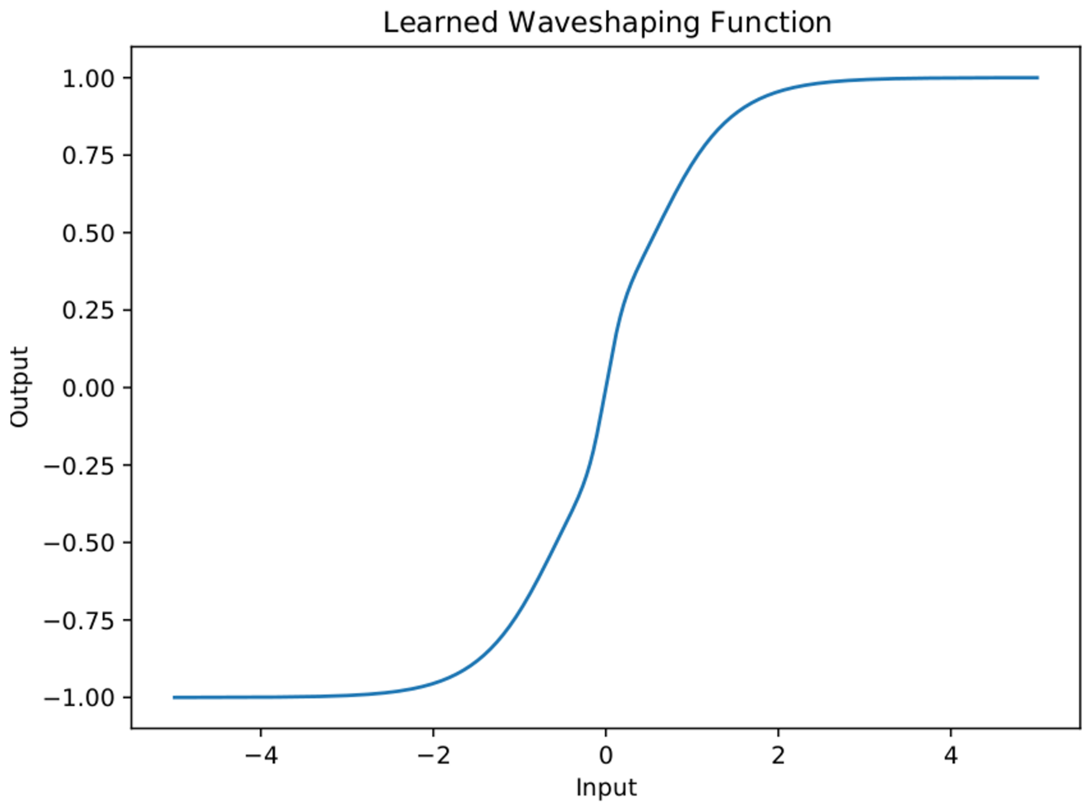
$$f(x) = a_1 \tanh(x) + a_2 \tanh(x^2) + \dots$$
$$a_{n-1} \tanh(x^{n-1}) + a_n \tanh(x^n)$$

Sumtanh family

$$f(x) = a_1 \tanh(x) + a_2 \tanh(2x) + \ldots + a_n \tanh(nx)$$

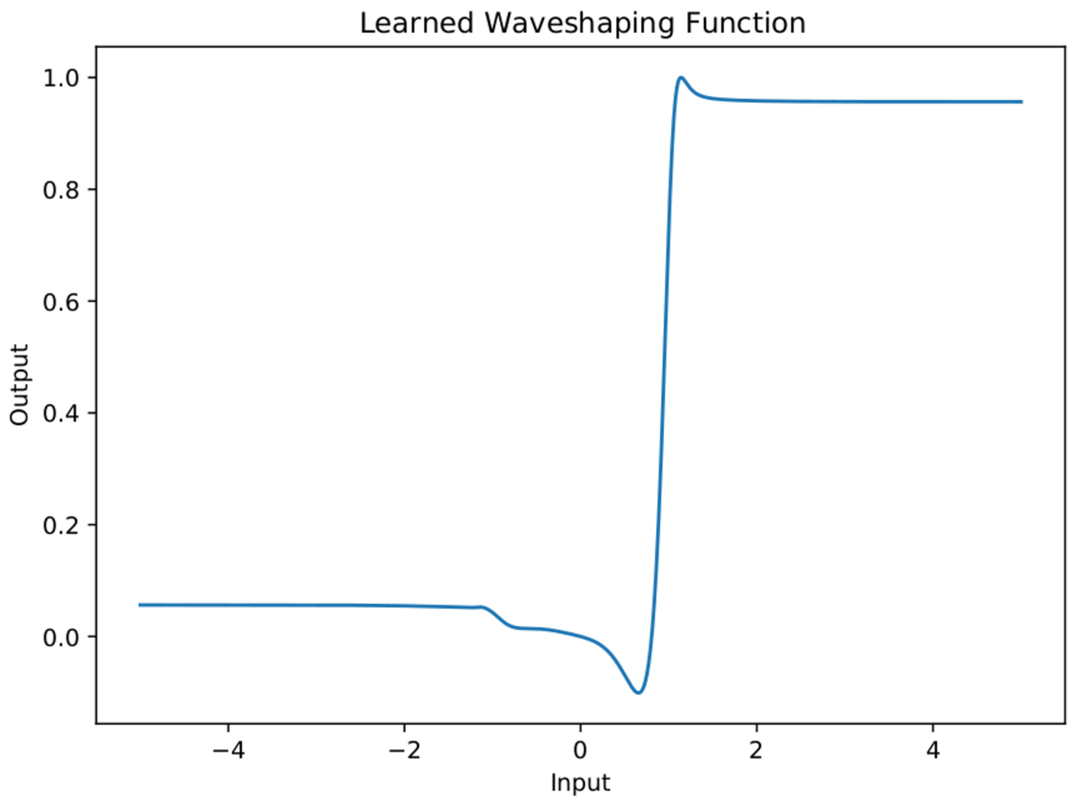
$$a_0 = -\sum_{c=1}^{n} a_c \tanh(c \times b_{DC})$$

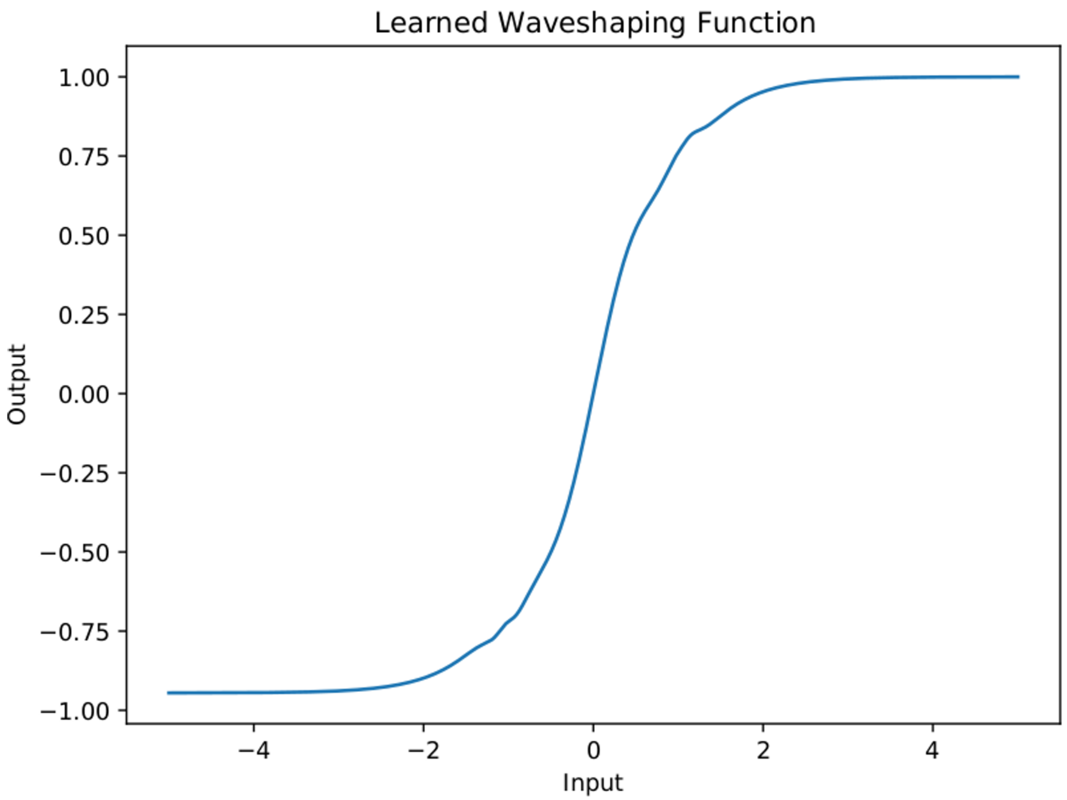




Powtanh family

$$f(x) = a_1 \tanh(x) + a_2 \tanh(x^2) + \ldots + a_n \tanh(x^n)$$





Evaluation - Reverse Engineering Distortion Effects

- Stochastic gradient descent updates the W-H model parameters to fit a dry/wet audio pair

- Parameters include:
    - Attenuations for pre-emphasis and de-emphasis filters
    - Gain
    - All coefficients in the waveshaper
    - Volume

- Loss is calculated by passing a dry audio sample through the estimated W-H model and measuring the multiscale spectrogram loss between the estimated and target audio

Results

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

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

Results

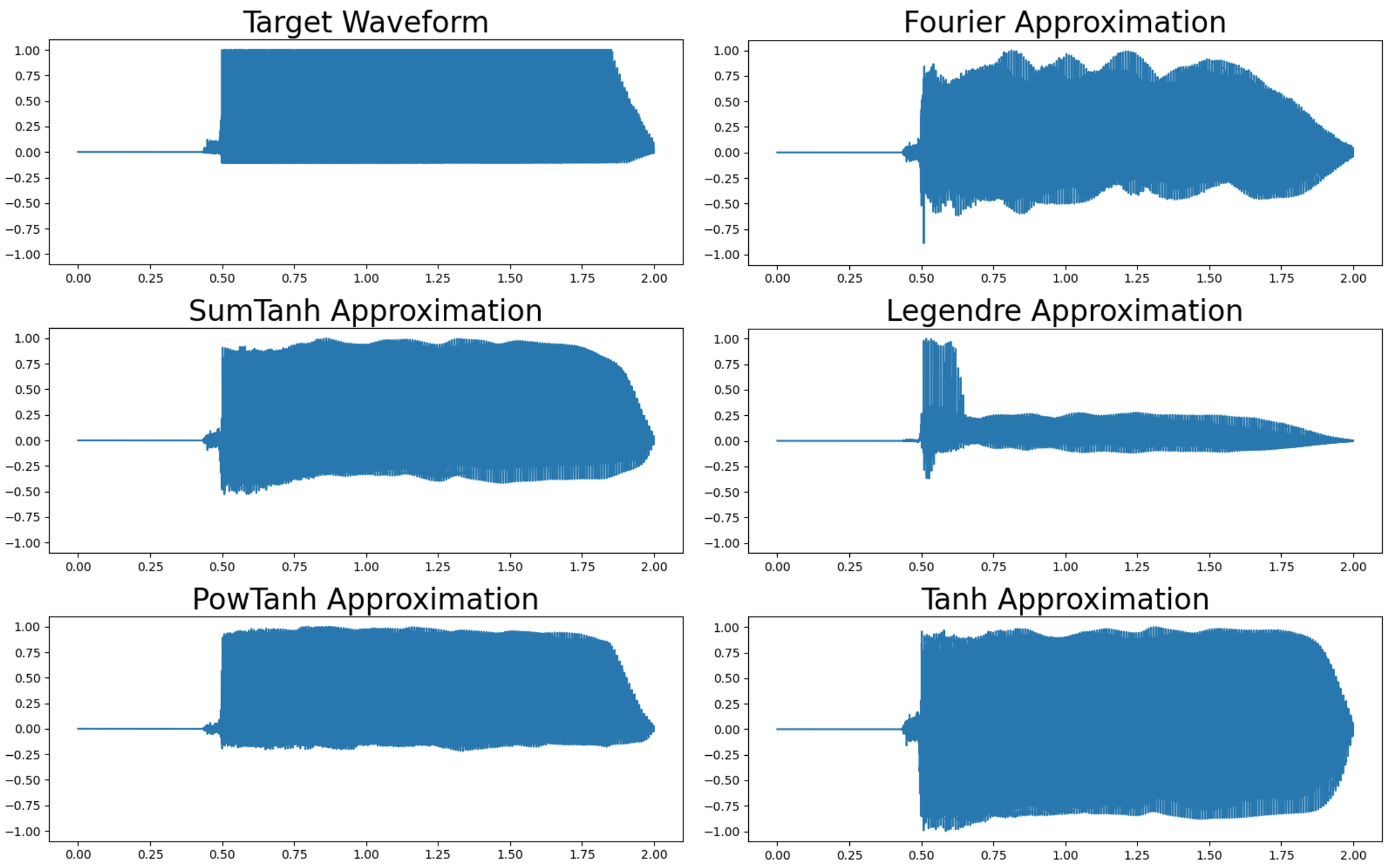
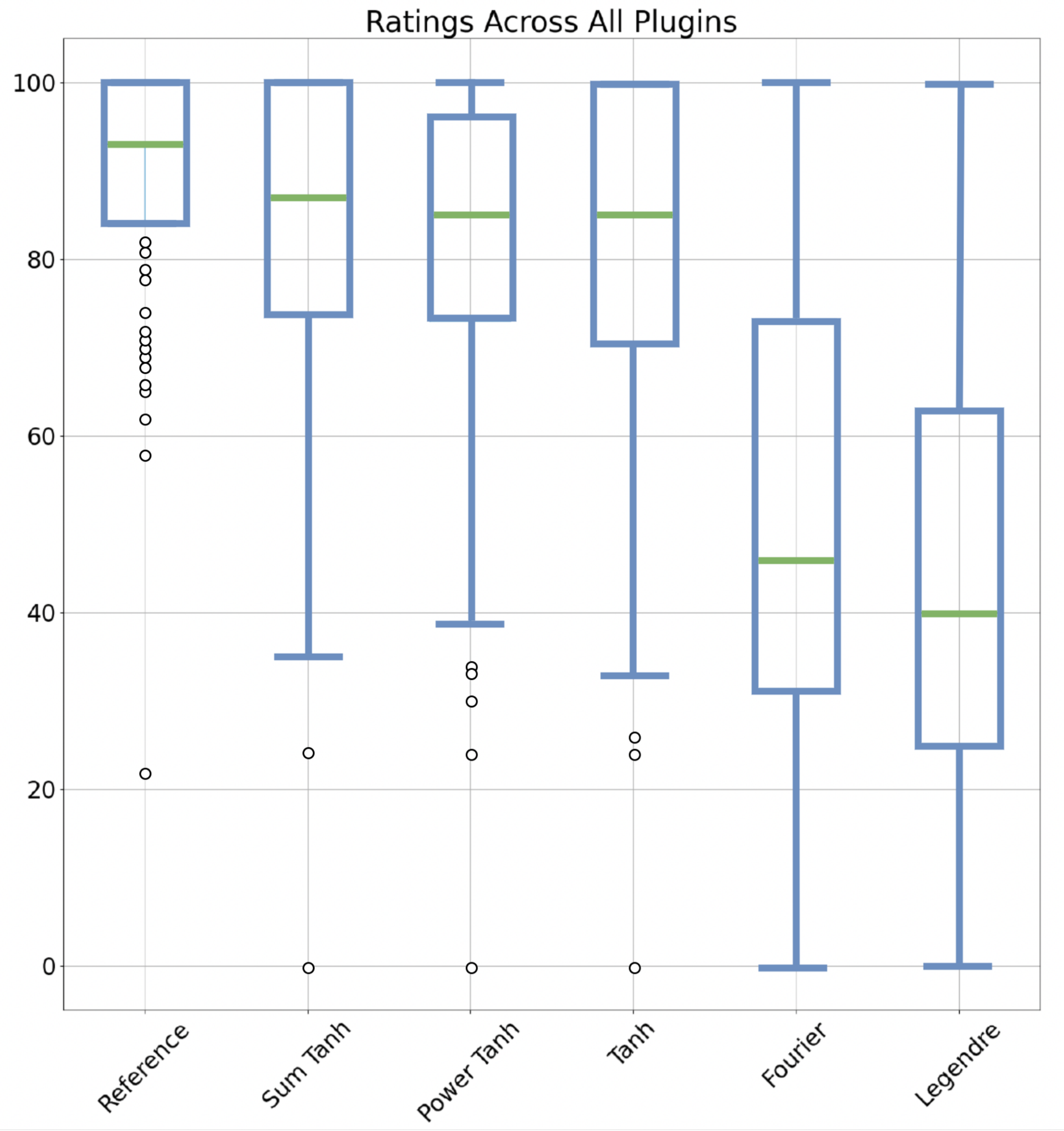


Fig. 2: Example of estimated waveforms for each waveshaping model on a VTB example

Results



Ratings Across All Plugins
100
80
60
40
20
0
Reference
Sum Tanh
Power Tanh
Tanh
Fourier
Legendre

Future Work

- Implement explicit antialiasing
- Model hysteresis for distortion effects with memory
- Implement signal-dependent conditioning for full effect modelling

Thank You

jtcolonel.github.io/DistRevEng

@JTColonel

mcomunita.github.io

@marcomunita

eecs.qmul.ac.uk/~josh/

@IntelSoundEng

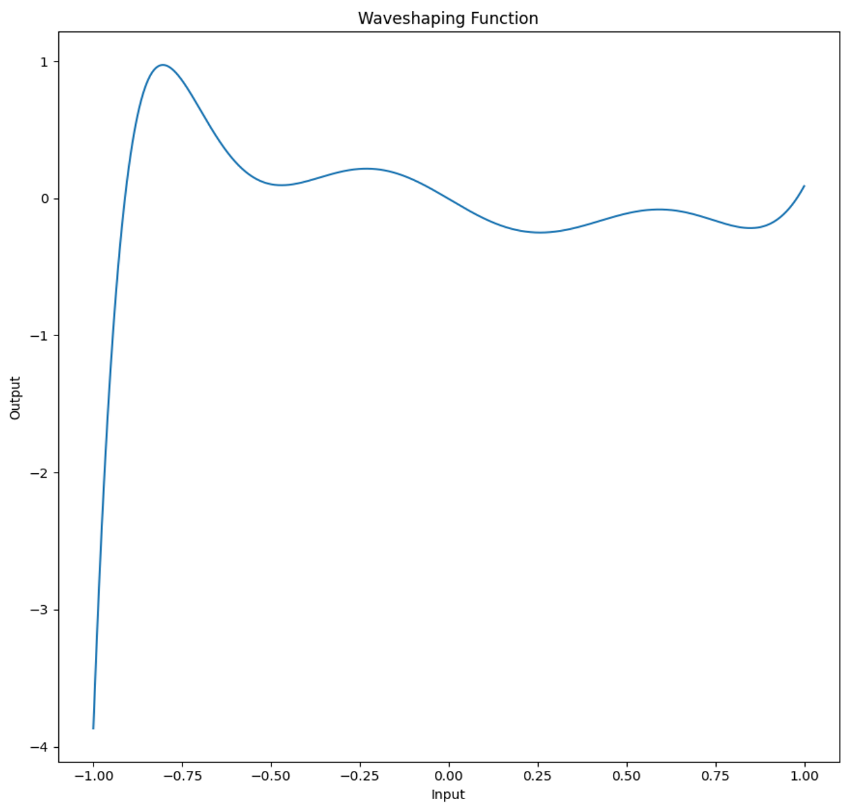
Queen Mary
University of London

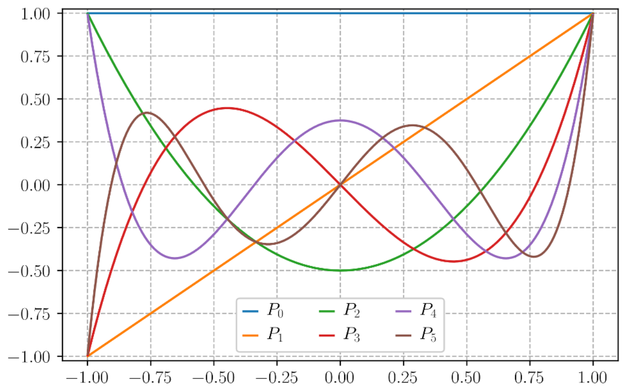
AES
NEW YORK
AUDIO ENGINEERING CONVENTION

YAMAHA

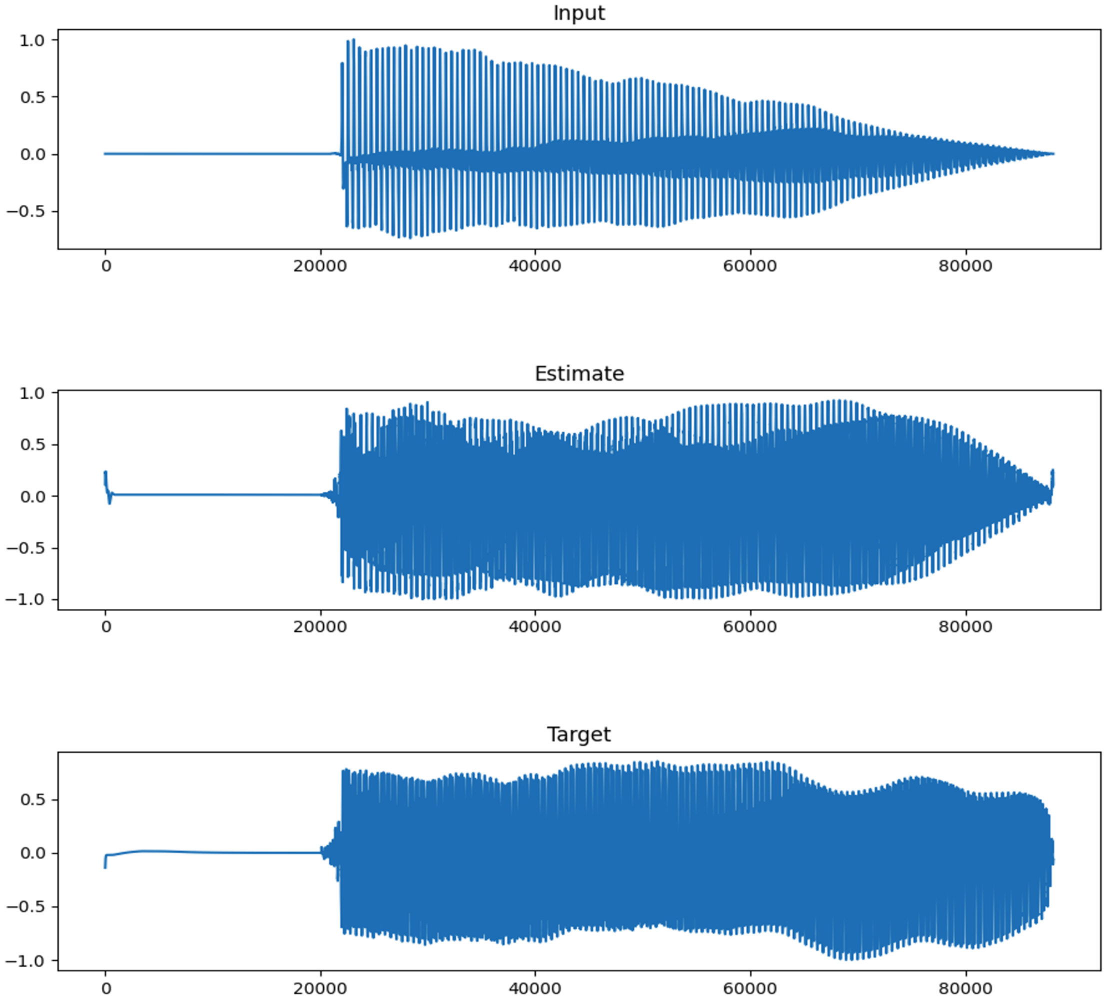
Appendix - Example Legendre Model

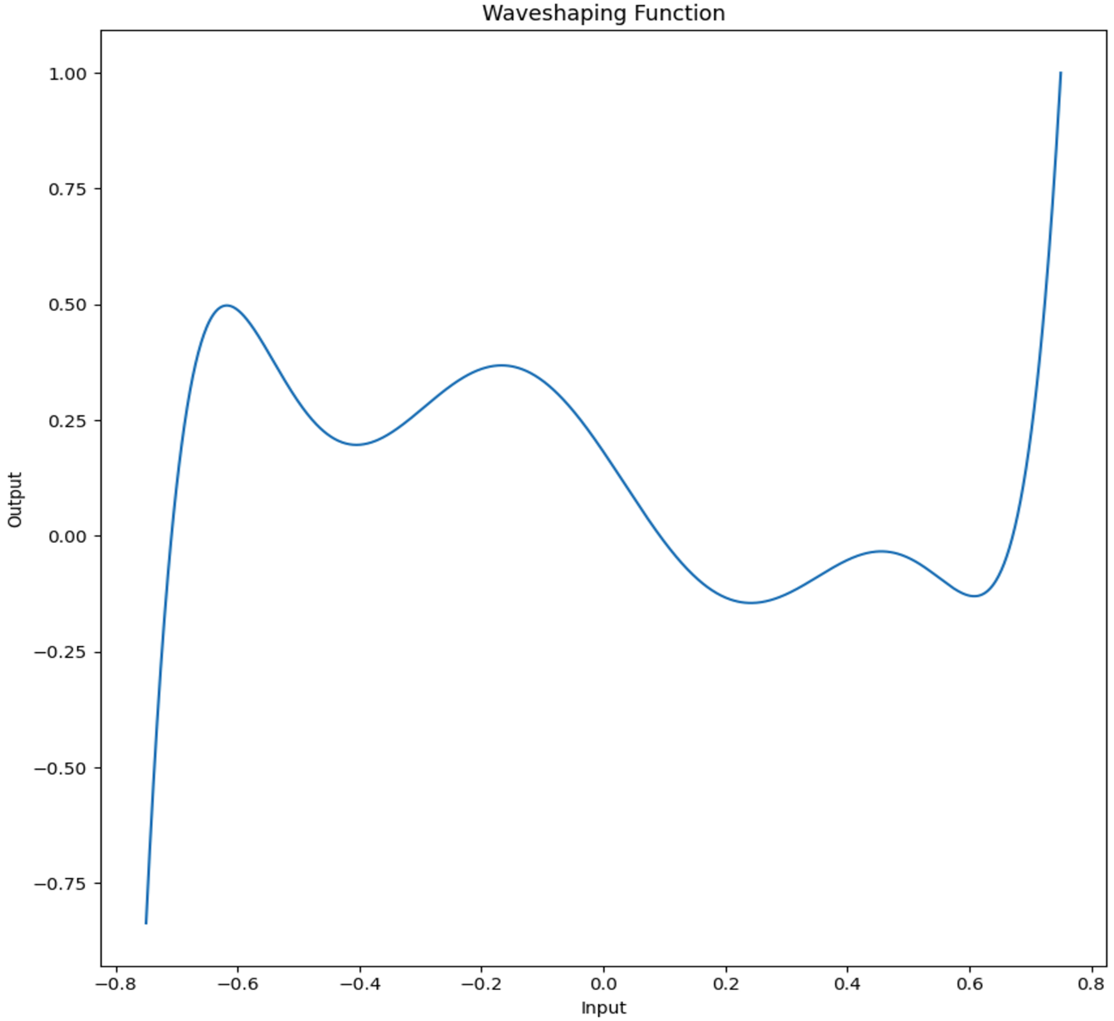
- Complete and orthogonal polynomials (ortho on [-1,1])
- System learns weighted sum of first N polynomials
- Signal normalized to [-0.9,0.9] before waveshaping to avoid overshoot





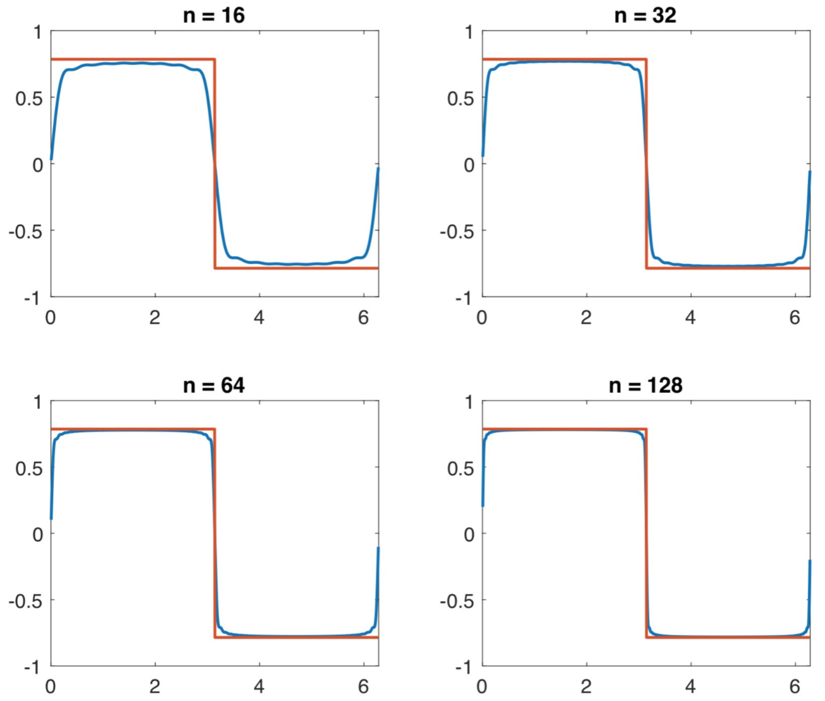
Appendix - Example Legendre Model

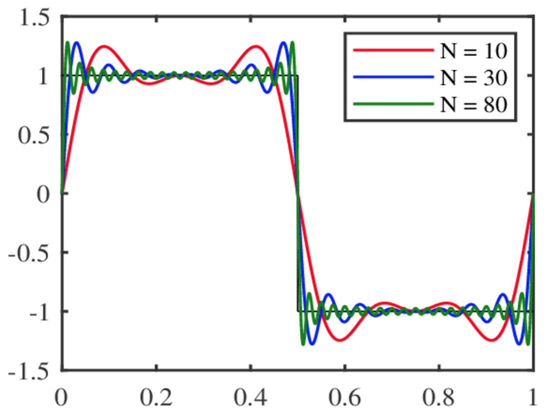




Appendix - Example Fourier Model

- Complete and orthogonal sines and cosines (ortho on [-π,π])
- System learns weighted sum of first N sinusoids
- Signal normalized to [-0.9π,0.9π] before waveshaping to avoid overshoot/zeroing





Appendix - Example Fourier Model