Proceedings of the 29th International Conference on Digital Audio Effects (DAFx26), Cambridge, MA, USA, 1–4 September 2026

FOURIER NEURAL OPERATORS FOR SAMPLE-RATE-INDEPENDENT VIRTUAL ANALOG MODELING

Oliviero Massi, Alessandro Ilic Mezza, and Alberto Bernardini

Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano
Piazza Leonardo Da Vinci 32, 20133 Milano, Italy
oliviero.massi@polimi.it | alessandroilic.mezza@polimi.it | alberto.bernardini@polimi.it

ABSTRACT

Neural networks that operate directly on time-domain signals are widely used for virtual analog (VA) modeling. A key limitation of these models is their dependence on the sampling rate used during training, which becomes implicitly encoded in the learned parameters, so that changing it generally alters the realized dynamics. Although architectural modifications to recurrent neural networks have been proposed to enable sample-rate independent operation, these approaches are inherently tailored to upsampling and do not accommodate downsampling scenarios. In this manuscript, we present a VA modeling framework based on Fourier Neural Operators (FNOs) adapted to process fixed-duration audio frames. The proposed formulation defines the learned mapping over a fixed temporal support and evaluates it on uniform grids of different densities, so that a model trained at a single sampling rate can be applied at unseen sampling resolutions. Numerical results on a nonlinear transistor circuit show that the proposed model achieves competitive accuracy in upsampling scenarios while remaining directly applicable to downsampling, unlike a sample-rate independent baseline recurrent architecture.

1. INTRODUCTION

Machine learning, and deep learning in particular, has become one of the most active research directions in virtual analog (VA) modeling, i.e., the digital emulation of analog audio circuits and devices [1]. Within this landscape, neural networks have emerged as the most prominent example of *closed-box* methods: they learn the input-output behavior of a target system directly from measurements, typically with little or no explicit knowledge of the underlying circuit topology or constitutive relations [2]. *Glass-box* methods, however, continue to play a central role in VA, relying on the discrete-time solution of the algebraic or ordinary differential equations characterizing a circuit schematic, through frameworks such as state-space methods [3], port-Hamiltonian formulations [4], and wave digital filters [5]. The growing prominence of machine learning has also reshaped this side of the field, encouraging hybrid strategies in which physically grounded models are combined with trainable components and gradient-based optimization procedures [6–10].

Among closed-box neural architectures for VA, sample-wise recurrent neural networks (RNNs) have been especially influential, owing to their low algorithmic latency and their ability to encode temporal dependencies through an internal state [11]. A common limitation of this class of architectures, however, is that sample-rate independence is not obtained natively: the discrete-time step associated with the training data is absorbed into the model architecture and learned parameters, so changing the sampling rate does not simply refine the grid, but alters the dynamics the model realizes. For this reason, recent research has explored remedies, including integer and fractional delay-based modifications for integer and non-integer oversampling [12], and explicit resampling stages wrapped around the network in order to adapt the processing rate to that expected by the model [13].

A class of models whose formulation is independent of the sampling grid is that of neural operators, which learn mappings between function spaces rather than between fixed-dimensional vectors [14, 15]. As such, the same learned operator can be evaluated on uniform discretizations of the same underlying signal at different sampling densities. This makes neural operators a natural candidate for fixed-duration audio processing, where signals acquired at different sampling frequencies may be interpreted as different samplings of the same time interval. Among them, the Fourier Neural Operator (FNO) is particularly attractive for sample-rate-independent VA modeling. Originally introduced for operator learning in partial differential equations [14], FNOs employ Fourier-domain processing as a natural mechanism for decoupling the learned mapping from a particular sampling grid.

The relevance of Fourier representations to learning across different sampling frequencies is not unique to operator learning. In related audio signal processing applications, it has been shown that time-frequency and convolutional deep networks can be transferred to different sampling frequencies when resolution is defined in physical units rather than in samples [16, 17]. In the VA literature, Fourier and time-frequency approaches have likewise been explored for frame-wise neural modeling of modulation effects [18]. Related ideas have also appeared in physical modeling, where FNO-inspired layers have been incorporated into RNNs as fast convolutional blocks [19]. In that case, however, such layers are used for efficient recurrent modeling rather than for learning across sampling frequencies.

In this manuscript, we investigate VA modeling with FNOs adapted to process fixed-duration frames. In this setting, the learned mapping is defined over a fixed temporal support and can be evaluated on uniform grids of different densities, so that a model trained at a given sampling rate may be applied at inference time on different sampling grids over the same time interval. At the architectural level, the FNO parameterizes the linear operator in the Fourier domain while acting on a reduced set of modes, thereby providing a compressed representation of the underlying linear transformation.

By relying on FFT and IFFT operations, this formulation can be implemented efficiently on modern computing hardware [20]. At the same time, the nonlinearity is applied in the time domain after the inverse transform. This distinguishes the proposed approach from architectures in which transfer across sampling frequencies is achieved through time-frequency processing or specially designed convolutional layers [16, 17]. For VA modeling, this is a relevant structural feature, as it preserves a separation between linear dynamic processing and static nonlinear behavior that is consistent with block-oriented descriptions of nonlinear audio systems. This design principle shares similarities with state-space-inspired neural networks, where linear filtering and nonlinear blocks are interleaved [21–23], while also suggesting an analogy with cascaded traditional Wiener–Hammerstein models [24].

The remainder of the manuscript is organized as follows. Section 2 reviews the FNO formulation from the literature. Section 3 presents its adaptation to VA modeling, including the proposed analysis–synthesis windowing for fixed-duration frame processing and overlap–add reconstruction. Section 4 introduces the considered VA case study and the related dataset, together with the corresponding modeling and training setup. Section 5 reports numerical results and compares the proposed method against a sample-rate independent RNN baseline in both upsampling and downsampling scenarios, the latter being a setting for which the baseline is not directly applicable. Finally, Section 6 concludes the manuscript.

2. FOURIER NEURAL OPERATOR

An operator is a mapping between function spaces. More formally, let $\mathcal{U}$ and $\mathcal{Y}$ denote two function spaces defined on a bounded domain $D \subset \mathbb{R}^d$. An operator

$$\mathcal{G} : \mathcal{U} \to \mathcal{Y}\,, \tag{1}$$

associates to each input function $u \in \mathcal{U}$ an output function $y = \mathcal{G}(u) \in \mathcal{Y}$ [25]. Several audio systems can be described as operators, since they map continuous input scalar functions to continuous output scalar functions over a temporal domain; examples include gain stages, filters and equalizers, dynamic-range processors, nonlinear waveshapers, and time-domain effects such as delay, reverberation, chorus, or flanging.

From this perspective, operator learning consists in approximating a target operator $\mathcal{G}$ from data by means of a parametric model $\mathcal{G}_{\boldsymbol{\theta}}$, with parameters $\boldsymbol{\theta}$, such that $\mathcal{G}_{\boldsymbol{\theta}}(u) \approx \mathcal{G}(u)$ for functions u drawn from a distribution of interest [14, 15]. Given a cost functional $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, the corresponding learning problem can be written as [26]

$$\min_{\boldsymbol{\theta}} \mathbb{E}_u[\mathcal{L}(\mathcal{G}_{\boldsymbol{\theta}}(u), \mathcal{G}(u))]\,. \tag{2}$$

Neural operators address this problem by constructing parametric mappings between function spaces rather than between fixed-dimensional vectors [14, 15, 25]. For the sake of simplicity, we limit the discussion to one-dimensional functions, for which $D \subset \mathbb{R}$. Denoting by $z_\ell : D \to \mathbb{R}^{c_\ell}$ the latent function at layer ℓ, a neural operator layer is written as

$$z_{\ell+1}(x) = \sigma(W z_\ell(x) + (\mathcal{K} z_\ell)(x))\,, \quad x \in D\,, \tag{3}$$

where W is a pointwise linear transformation, $\mathcal{K}$ is a linear operator, and σ is a pointwise nonlinearity. By stacking multiple layers of this form, together with suitable input and output transformations [14], it is possible to obtain a nonlinear approximation of the target operator $\mathcal{G}$.

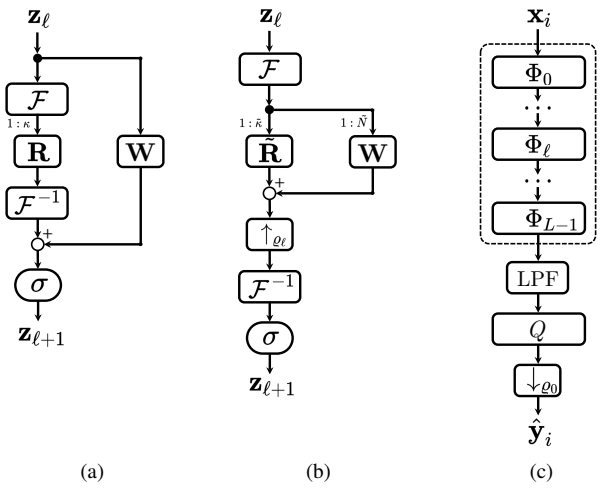


Figure 1: Block diagrams of the considered Fourier-based architectures: (a) original FNO layer; (b) modified Fourier layer proposed for VA modeling; (c) proposed frame-level architecture, comprising a stack of modified Fourier layers Φ_ℓ, brickwall lowpass filtering, channelwise multilayer perceptron Q, and final downsampling $\downarrow_{\varrho_0}$.

The Fourier Neural Operator (FNO) specializes this construction by parameterizing the linear operator $\mathcal{K}$ in the Fourier domain [14, 15]. For a given uniform discretization of the domain, let $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the Fourier and inverse Fourier transforms, implemented through the FFT and its inverse, respectively. The corresponding Fourier layer, represented in Figure 1a, is written as

$$\mathbf{z}_{\ell+1} = \sigma(\mathbf{W}\mathbf{z}_\ell + \mathcal{F}^{-1}(\mathbf{R}\,\mathcal{F}_{1:\kappa}(\mathbf{z}_\ell)))\,, \tag{4}$$

where the skip branch $\mathbf{W} \in \mathbb{R}^{c_{\ell+1} \times c_\ell}$ is a learned real-valued mixing matrix applied across channels to individual entries of $\mathbf{z}_\ell \in \mathbb{R}^{c_\ell \times M_\ell}$, and $\mathbf{R} \in \mathbb{C}^{c_{\ell+1} \times c_\ell \times \kappa}$ is a learned complex-valued linear transformation acting only on the κ lowest-frequency coefficients $\mathcal{F}_{1:\kappa}(\mathbf{z}_\ell)$ extracted from $\mathcal{F}(\mathbf{z}_\ell) \in \mathbb{C}^{c_\ell \times M_\ell}$, with c_ℓ denoting the number of latent channels at layer ℓ. In practice, this means that only $\kappa \leq M_\ell$ complex Fourier modes are retained and transformed by the spectral weights in $\mathbf{R}$, while the remaining high-frequency modes are set to zero before applying $\mathcal{F}^{-1}$.

Since κ is independent of the number of FFT points, the size of $\mathcal{F}_{1:\kappa}(\cdot)$, hence the shape of $\mathbf{R}$, remains constant regardless of the specific temporal grid. Therefore, the same learned operator can be evaluated on different uniform discretizations of the domain. In the one-dimensional audio setting considered in this work, this property translates directly into sample-rate invariance.

3. FNO FOR VIRTUAL ANALOG MODELING

In this section, we adapt the FNO formulation of Section 2 to accommodate the requirements of VA modeling. This adaptation involves three elements: a frame-based formulation over fixed temporal support, a modified Fourier layer, and a full analysis–synthesis scheme for processing arbitrary-length signals.

3.1. Frame-Based Formulation

For audio applications, the domain $D \subset \mathbb{R}$ may be taken as a fixed time interval $D = [0, T]$, so that both the input and output of the operator are continuous signals defined over the same temporal support. Under this formulation, changing the sampling rate only modifies the discretization used to represent $u(t)$ and $y(t)$ over D.

Let $D_N = \{t_n\}_{n=0}^{N-1}$ denote a uniform N-point discretization of D, with sample times $t_n = nT_\mathrm{s}$, sampling period $T_\mathrm{s} = 1/f_\mathrm{s}$, and $N = \lceil f_\mathrm{s} T \rceil$. Hence,

$$\begin{aligned} \mathbf{u} &= [u(t_0), \ldots, u(t_{N-1})]^\top \in \mathbb{R}^N , \\ \mathbf{y} &= [y(t_0), \ldots, y(t_{N-1})]^\top \in \mathbb{R}^N , \end{aligned} \tag{5}$$

denote the corresponding sampled observations of the input and output functions, respectively. Note that since N depends on f_s whereas T is fixed, different sampling rates yield different uniform discretizations of the same temporal interval.

In practice, however, audio signals are typically much longer than any practical choice of T. The input–output mapping is therefore not learned directly over the full signal support. Rather, to make the operator-learning formulation of Section 2 applicable, short-time frames are obtained by applying shifts and tapered windows to the continuous-time signals $u(t)$ and $y(t)$. The FNO is then trained to approximate the resulting frame-wise operator. Accordingly, for each index i, we define $\mathbf{u}_i$ and $\mathbf{y}_i$ as the discretization over D_N of

$$\begin{aligned} u_i(t) &= w_\mathrm{a}(t) u(t + \tau_i) , \\ y_i(t) &= w_\mathrm{s}(t) y(t + \tau_i) , \end{aligned} \tag{6}$$

where $t \in D$, τ_i is a time shift, and w_a and w_s denote tapered analysis and synthesis windows, respectively.

This choice of applying tapered windows serves two purposes. First, it makes the operator-learning formulation of Section 2 directly applicable to audio signals, whose support is now fixed by construction. Second, the use of tapered windows mitigates boundary discontinuities induced by the periodic extension implicitly assumed by the Fourier layer [14]. The network therefore learns to output windowed frames, which are later recombined through overlap–add synthesis. The frame duration T (in seconds) is therefore a key design parameter, as it determines the domain over which the learned operator acts and, consequently, the time span of the system dynamics that can be captured within a single frame. Unlike recurrent architectures, which retain a hidden state across samples, the FNO does not propagate information from one frame to the next. As a result, the model can only exploit temporal dependencies contained within each individual frame.

3.2. Modified Fourier Layer

Because in (4) the nonlinearity σ is applied in the time domain, each layer may generate harmonic components above the Nyquist frequency associated with the input discretization. In VA applications, this issue is commonly mitigated by upsampling the model input by a factor greater than one and downsampling the output by the same factor. While this can be achieved in several ways [13], FNOs provide an elegant and efficient means of performing sample-rate conversion by virtue of operating natively in the frequency domain [27].

Specifically, within an FNO, upsampling and downsampling can be implemented by zero-padding and truncating the frequency-domain representation prior to the inverse FFT, respectively, with the former being equivalent to ideal sinc interpolation in the time domain [27]. From an architectural viewpoint, efficiently incorporating upsampling into the Fourier layer described in Section 2 requires only a minor modification of the skip-connection branch: by linearity of the Fourier transform, $\mathcal{F}(\mathbf{W}\mathbf{z}_\ell) = \mathbf{W}\mathcal{F}(\mathbf{z}_\ell)$, so that (4) can be rewritten as

$$\mathbf{z}_{\ell+1} = \sigma(\mathcal{F}^{-1}(\uparrow_{\varrho_\ell} (\mathbf{W}\, \mathcal{F}(\mathbf{z}_\ell) + \mathbf{R}\, \mathcal{F}_{1:\kappa}(\mathbf{z}_\ell)))) . \tag{7}$$

Here, $\uparrow_{\varrho_\ell}$ denotes the operator that maps an in-band spectrum to its zero-padded representation; if $\varrho_\ell = 1$, no upsampling occurs. Notably, ϱ_ℓ can be chosen independently at training and inference time. In particular, during inference it may be freely adjusted according to the operating sampling rate.

Since audio signals are real, their Fourier coefficients satisfy Hermitian symmetry. Therefore, only half the bins carry unique information, and both the skip branch and spectral convolution can be implemented by processing only the non-redundant nonnegative-frequency modes, with the IFFT implicitly restoring Hermitian symmetry before returning to the time domain. This means that the spectral weights in $\mathbf{R}$ need only be applied to the lowest $\tilde{\kappa} = \lfloor \kappa/2 \rfloor + 1$ Fourier modes, which reduces the number of trainable parameters in $\tilde{\mathbf{R}} \in \mathbb{C}^{c_{\ell+1} \times c_\ell \times \tilde{\kappa}}$ and halves the computational cost of the operation. Similarly, we apply $\mathbf{W}$ only to the $\tilde{N} = \lfloor N/2 \rfloor + 1 < M_\ell$ Fourier modes in the original passband, so that the skip branch remains bounded to the Nyquist limit of the input resolution. The resulting modified Fourier layer $\mathbf{z}_{\ell+1} = \Phi_\ell(\mathbf{z}_\ell)$ is defined by

$$\Phi_\ell(\mathbf{z}_\ell) = \sigma(\mathcal{F}^{-1}(\uparrow_{\varrho_\ell} (\mathbf{W}\, \mathcal{F}_{1:\tilde{N}}(\mathbf{z}_\ell) + \tilde{\mathbf{R}}\, \mathcal{F}_{1:\tilde{\kappa}}(\mathbf{z}_\ell)))) , \tag{8}$$

where only $\mathcal{F}$ (computed once per layer), $\mathcal{F}^{-1}$, and σ operate at the upsampled dimension M_ℓ.[1]

In the proposed skip branch formulation, each nonnegative-frequency complex bin contributes two real matrix-vector products, making its flop count proportional to N. In contrast, $\mathbf{W}\mathbf{z}_\ell$ in (4) requires M_ℓ matrix multiplications, one for each time step at the higher rate, while also propagating harmonics generated above the original Nyquist frequency, thereby increasing the risk of aliasing due to folded spectral images as depth increases. Consequently, implementing the skip connection in the frequency domain yields a net gain in terms of flops equal to ϱ_ℓ.

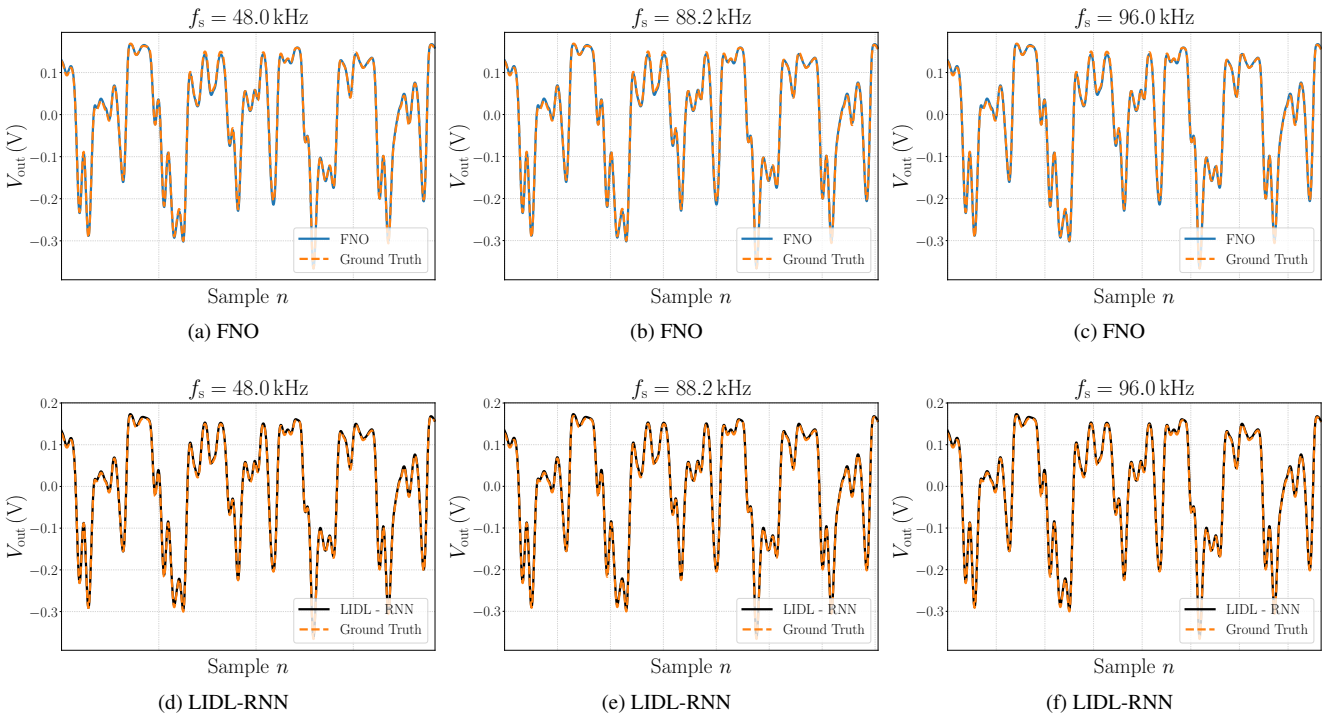
3.3. Full Architecture and Output Reconstruction

At the frame level, the model implements the mapping $\mathbf{u}_i \mapsto \hat{\mathbf{y}}_i$, where i denotes the frame index. If the first modified Fourier block $\Phi_0 : \mathbb{R}^N \to \mathbb{R}^{M_1}$ applies internal upsampling by a factor ϱ_0 while the remaining use $\varrho_\ell = 1$, $\ell = 1, \ldots, L-1$, then latent representations

$$\mathbf{z}_1 = \Phi_0(\mathbf{u}_i) , \quad \mathbf{z}_{\ell+1} = \Phi_\ell(\mathbf{z}_\ell) , \tag{9}$$

are sampled on a refined grid of size $M_\ell = \varrho_0 N$.

[1] A forward pass through the ℓth layer of the modified FNO consists of (i) one c_ℓ-channel FFT with complexity $O(c_\ell M_\ell \log M_\ell)$; (ii) two modal truncations (tensor views), with complexity $O(1)$; (iii) $\tilde{\kappa}$ complex matrix–vector multiplications and $2\tilde{N}$ real matrix–vector multiplications, each with complexity $\Theta(c_{\ell+1} c_\ell)$; (iv) one padding operation ($\uparrow_{\varrho_\ell}$), with complexity $O(c_{\ell+1}(\tilde{N} + \tilde{\kappa}))$ due to writing the spectral convolution outputs into a preallocated zero tensor; (v) $M_\ell\, c_{\ell+1}$ pointwise evaluations of the activation function σ; (vi) one $c_{\ell+1}$-channel IFFT with complexity $O(c_{\ell+1} M_\ell \log M_\ell)$.



(a) FNO (b) FNO (c) FNO

(d) LIDL-RNN (e) LIDL-RNN (f) LIDL-RNN

Figure 2: Time-domain validation on representative guitar audio excerpt. Subfigures (a)–(c) show the predictions of the proposed FNO model at $f_s = 48\,\mathrm{kHz}$, $88.2\,\mathrm{kHz}$, and $96\,\mathrm{kHz}$, respectively, whereas subfigures (d)–(f) report the corresponding predictions of the LIDL-RNN baseline. In each case, the predicted waveform is compared with the ground truth.

After the stack of modified Fourier blocks, the resulting high-resolution latent representation is low-pass filtered by a brick-wall filter, realized in the frequency domain by zeroing out the Fourier coefficients above the Nyquist limit of the original input resolution. The filtered representation is then projected by a channel-wise multilayer perceptron Q, whose output is downsampled back to the original temporal resolution through $\downarrow_{\varrho_0}$, yielding the predicted frame $\hat{\mathbf{y}}_i \in \mathbb{R}^N$. A schematic view of the complete frame-level architecture is shown in Figure 1c.

At the signal level, the frame-wise predictions are recombined through overlap–add synthesis. Although the network is trained to predict already windowed frames, an additional synthesis window is applied during reconstruction in order to suppress edge discontinuities at frame boundaries. In practice, this double windowing is compensated by choosing the analysis and synthesis windows as square-root versions of a common prototype window satisfying the COLA condition [28].

At inference time, the hop size determines the window scaling factor and, more importantly, the processing rate of the model, and therefore its effective throughput. In this sense, the proposed architecture is inherently buffered, since predictions are produced over finite-duration frames rather than through a purely sample-wise inference. Smaller values of τ_i increase the rate of the output updates, at the cost of a larger number of overlapping frame evaluations. Algorithmic latency can thus be made arbitrarily small by initializing the first frame $\mathbf{u}_0$ with zeros and filling it as new samples become available, such that the resulting approximation error vanishes once the input buffer is fully populated with valid data. In the limiting case $\tau_i = iT_s$, the model approaches a sample-wise operating regime while still retaining its frame-based formulation. In such a highly overlapped setting, recursive spectral update methods such as the sliding discrete Fourier transform (DFT) may offer computational advantages over repeated FFT evaluations on adjacent frames [29].

4. CASE STUDY

As a case study, we consider the input stage of the Big Muff Pi circuit, already adopted in [10] as a benchmark for nonlinear audio-circuit modeling. This stage features a single npn 2N5089 bipolar junction transistor (BJT), whose nonlinear behavior becomes particularly relevant as the input amplitude is increased and the device is progressively driven toward saturation.

A dataset is obtained by numerically simulating in Mathworks Simscape the Big Muff Pi input stage, following the same data-generation procedure adopted in [10]. The circuit is excited by clean guitar excerpts and exponential sweeps at multiple amplitudes, so as to cover both weakly and strongly nonlinear operating conditions. The resulting dataset has a total duration of approximately 130 s. In the present study, all signals are considered at a training sampling frequency of $f_s = 48$ kHz. Of the total amount of data, 90% is used for training, while the remaining 10% is reserved for evaluation.

From the full-length signals, we extract fixed-duration frames of length $T = 20$ ms using the root-Hann window. At the training sampling frequency, each frame therefore contains $N = 960$ samples. A 75% overlap is employed, corresponding to a constant hop

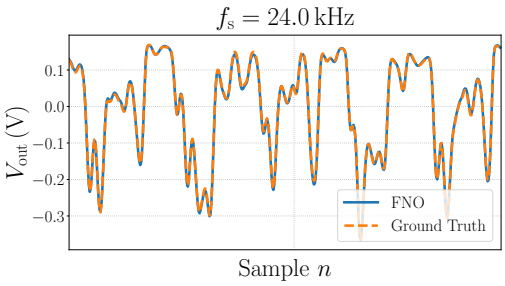


Figure 3: Time-domain validation on representative guitar audio excerpt of the proposed FNO model at the downsampled sampling frequency $f_s = 24$ kHz. The predicted waveform is compared with the corresponding ground truth.

length of $T/4$, i.e., 5 ms. Each input frame $\mathbf{u}_i$ is paired with the corresponding target frame $\mathbf{y}_i$, consistently with the formulation in Section 3.1. The resulting frame pairs are randomly shuffled and grouped into mini-batches of 128 elements.

Implemented in Python using PyTorch, the proposed model consists of $L = 3$ modified Fourier layers, each with $c_\ell = 8$ latent channels, $\ell = 0, \dots, L-1$. The number of retained real Fourier modes is set to $\bar{\kappa} = 64$, and the final projection is implemented by a two-layer channelwise multilayer perceptron Q with hidden dimension 8. All nonlinear activation functions in the architecture are chosen as SiLU. The resulting model has 17,785 trainable parameters. During training, the internal upsampling factor is set to $\varrho_0 = 2$, in order to reduce the generation of spurious aliased harmonics in the nonlinear stages.

The model is trained for 1000 epochs by minimizing the normalized mean squared error (NMSE),

$$\mathcal{L} = \frac{\|\mathbf{y}_i - \hat{\mathbf{y}}_i\|_2^2}{\|\mathbf{y}_i\|_2^2}, \tag{10}$$

where $\mathbf{y}_i$ and $\hat{\mathbf{y}}_i$ denote the ith ground truth and predicted sampled output frames, respectively. Optimization is performed with Adam [30], using the default coefficients $\beta_1 = 0.9$ and $\beta_2 = 0.999$, an initial learning rate of 10^{-3}, which is halved every 200 epochs. Training is carried out on a machine with a single NVIDIA Titan X GPU with 12 GB of memory.

In the next section, we evaluate the proposed model when trained at a single sampling frequency and tested at unseen frame resolutions corresponding to different sampling rates.

5. NUMERICAL RESULTS

Evaluation is carried out on the held-out portion of the dataset, consisting of guitar excerpts not used during training. To assess the ability of the model to generalize across unseen sampling resolutions, the evaluation signals are considered not only at the training sampling frequency (48 kHz), but also after resampling to 88.2 kHz, 96 kHz, and 24 kHz. These correspond, respectively, to a non-integer upsampling factor, an integer upsampling factor, and a downsampling factor with respect to the training resolution. In all cases, resampling of the evaluation signals is performed using the method discussed in [27].

Table 1: Validation metrics at the considered sampling rates f_s. MAE and SC values are reported in units of $\times 10^{-3}$ and $\times 10^{-2}$, respectively.

Metric	Model	24 kHz	48 kHz	88.2 kHz	96 kHz
MAE	LIDL-RNN	N/A	2.173	2.319	2.333
	FNO (Ours)	2.894	2.897	2.898	2.898
SC	LIDL-RNN	N/A	3.852	3.920	3.932
	FNO (Ours)	2.932	3.237	3.408	3.431

5.1. Sample-Rate-Independent RNN Baseline

As a baseline, we consider the sample-rate-independent recurrent neural architectures proposed by Carson et al. for neural audio effect processing [12]. In that framework, a recurrent model trained at a given sampling frequency is adapted at inference time to operate at a different one by reinterpreting the unit-sample delay in the hidden-state recursion as a delay on the new sampling grid. For integer upsampling factors, the modified recurrence can be implemented with a standard delay line. For non-integer factors, the required hidden state is approximated through fractional-delay interpolation. In the present work, we consider the linear-interpolation delay-line (LIDL) variant. As noted by Carson et al., this construction applies to upsampling, but not to downsampling, where it leads to an implicit non-causal delay relation [12].

The baseline architecture is composed of a Gated Recurrent Unit (GRU) layer with 64 hidden units, followed by a single fully connected linear layer. The hidden dimension is chosen so that the overall number of trainable parameters (i.e., 12,929) is of the same order of magnitude as that of the proposed FNO model (17,785). The network is trained on the same dataset used for the FNO, at 48 kHz, using sequences with a duration of 0.1 s extracted every 25 ms. The loss function is again the NMSE defined in (10). Training is performed for 1000 epochs with Adam, using an initial learning rate of 10^{-3}. At inference time, the model is made sample-rate independent through the LIDL hidden state interpolation strategy discussed above.

5.2. Model Validation

We first compare the predicted output signals of the proposed FNO model and of the LIDL-RNN baseline against the corresponding ground truth in the time domain. Figure 2 reports representative waveform excerpts at the training sampling frequency 48 kHz, at the non-integer upsampled frequency 88.2 kHz, and the integer upsampled frequency 96 kHz. Subfigures (a)–(c) show the predictions obtained with the proposed FNO model, while subfigures (d)–(f) report the corresponding outputs of the LIDL-RNN baseline. In all evaluations reported in this section, the proposed FNO model is used with inference-time internal upsampling factor set to $\varrho_0 = 1$, i.e., no upsampling, corresponding to the simplest deployment configuration.

To quantify the discrepancy between full-length model predictions and reference signals at the different sampling rates, we consider two validation metrics: mean absolute error (MAE), which measures agreement in the time domain, and spectral convergence (SC) [10], which provides a complementary assessment in the time-frequency domain. The results are reported in Table 1.

The two architectures achieve a comparable performance in

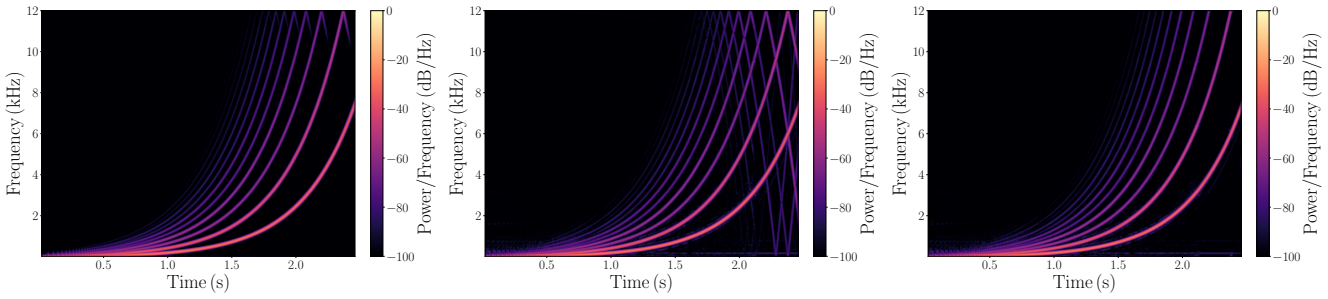


Figure 4: Time-frequency analysis of the proposed FNO model in the downsampled $f_s = 24$ kHz scenario. The ground truth in (a) is compared with the corresponding model predictions obtained for (b) $\varrho_0 = 1$, and (c) $\varrho_0 = 2$.

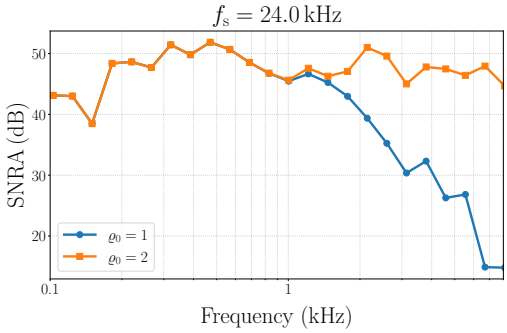


Figure 5: SNRA curves at $f_s = 24$ kHz for inference-time internal upsampling factors $\varrho_0 = 1$ and $\varrho_0 = 2$. Higher values indicate reduced aliasing-related spectral energy.

both integer and non-integer upsampling scenarios, with MAE generally favoring the LIDL-RNN baseline, whereas the proposed FNO model consistently achieves slightly better SC values.

By contrast, the sample-rate independent LIDL-RNN architecture is not directly applicable in the downsampling case, whereas the proposed FNO-based architecture can still operate directly on downsampled input signals. A corresponding prediction at 24 kHz is therefore reported separately in Figure 3. The metrics obtained in this case should be read against the values in Table 1; in particular, its numerical values remains of the same order as those measured at 48 kHz, 88.2 kHz, and 96 kHz, indicating that the proposed formulation preserves a comparable level of accuracy also below the training sampling frequency.

5.3. Time-Frequency Analysis of Internal Upsampling

To complement the time-domain validation, we consider a qualitative time-frequency analysis of the proposed FNO in the downsampled 24 kHz condition. This setting is of particular interest because, as discussed in Section 3.2, the lower Nyquist limit would increase the tendency of newly generated harmonics to fold back into the available bandwidth. This is a suitable scenario in which to inspect the role of internal upsampling.

For this analysis, we consider the same FNO model trained at the reference $f_s = 48$ kHz. As input, we use successive 20 ms windows extracted from a 2.5 s exponential sine sweep from 100 Hz to 8 kHz, with an amplitude not included in the training dataset. The corresponding ground truth output is obtained by suitably resampling at 24 kHz the Simscape simulation output of the reference circuit. The spectrogram of this signal is then compared with those of the FNO predictions obtained using two different inference-time values of the internal upsampling factor, $\varrho_0 \in \{1, 2\}$.

Figure 4 reports the resulting time-frequency representations. Figure 4a shows the ground-truth output, while Figures 4b and 4c report the corresponding FNO predictions obtained with $\varrho_0 = 1$ and $\varrho_0 = 2$, respectively. A visual inspection shows that, for $\varrho_0 = 1$, the upper-frequency region near the end of the sweep contains a more evident pattern of spurious components folding back toward lower frequencies. By contrast, when $\varrho_0 = 2$ is used at inference time, this effect is no longer visible and the harmonic trajectories remain more clearly separated.

The qualitative spectrogram analysis is further supported by an aliasing measure, namely the signal-to-aliasing-noise ratio (SNRA), computed following [12]. The model is probed with sine waves of duration 3 s and amplitude 0.35 V, using one test signal for each frequency. After the prediction is obtained, the window-boundary regions are discarded and the central 1 s segment is used for the analysis. From this segment, an ideal bandlimited harmonic reference is synthesized by retaining the expected non-aliased harmonic components. The SNRA is then obtained by measuring the energy ratio between the retained bandlimited harmonic components and the remaining spectral residual. Larger SNRA values indicate that less energy is present outside the non-aliased harmonic structure, and therefore correspond to a lower level of aliasing-related distortion.

Figure 5 shows the resulting SNRA curves obtained when the FNO is operated at an external sampling rate of $f_s = 24$ kHz; the curves correspond to two different inference-time values of the internal upsampling factor, $\varrho_0 \in \{1, 2\}$. Consistently with the sweep range used in Figure 4, the test frequencies are distributed approximately logarithmically from 100 Hz to 8 kHz. The results provide a quantitative counterpart to the visual differences observed in Figures 4b and 4c, respectively. The two settings behave similarly at lower test frequencies, whereas $\varrho_0 = 2$ yields higher SNRA values in the upper part of the range, in agreement with the reduced fold-back patterns observed in Figure 4c.

Taken together, these results suggest that enabling internal upsampling at inference time can be beneficial when the proposed FNO is evaluated below the training sampling frequency. This im-

provement comes at the cost of increased computational complexity, since the FFT/IFFT operations and the time-domain nonlinear stages point-wise nonlinearities act on an expanded internal grid. At the same time, the learned spectral processing itself remains confined to the retained real modes in the $\tilde{\mathbf{R}}$ branch and to the original passband in the $\mathbf{W}$ branch, both of which are independent of the upsampling factor.

6. CONCLUSION

In this manuscript, we presented a VA modeling framework based on FNOs, specifically adapted to operate on fixed-duration audio frames. The proposed formulation adopts an operator learning perspective, defining the learned input-output mapping over a fixed temporal support. This allows a model trained at a single sampling rate to be evaluated on uniform grids of different densities, corresponding to unseen sampling rates. As a result, the same trained model can be applied not only in upsampling scenarios, but also in downsampling conditions, which are not architecturally addressed by existing sample-rate independent recurrent neural network approaches.

From a methodological standpoint, we introduced a modified Fourier layer tailored to the real-valued audio signal processing setting and embedded it within a frame-based analysis–synthesis framework that enables the processing of arbitrary length signals through overlap–add reconstruction. The proposed approach was evaluated for the modeling of a nonlinear BJT-based circuit and compared against a sample-rate-independent LIDL-RNN baseline.

The results showed that the FNO-based model achieves comparable accuracy in integer and non-integer upsampling scenarios, while remaining directly applicable in downsampling conditions without requiring architectural adaptations. By contrast, the LIDL-RNN baseline is not directly applicable for processing signals sampled below the training sampling rate, and would instead require the input signal to be upsampled before processing.

The competitive performance observed in the present study warrants further investigation in this class of models for VA applications. Future work includes a systematic analysis of how performance scales with model size, particularly as the number of trainable parameters is reduced, as well as extending the experimental validation of the proposed framework to a broader range of audio circuital systems and operating conditions. It would also be of interest to investigate how applying internal upsampling for aliasing reduction at training time modifies the learned FNO weights. Finally, exploring hybrid architectures that combine operator learning principles with physically informed structures may provide a promising path toward more accurate and robust VA models.

7. REFERENCES

[1] V. Välimäki, S. Bilbao, J. Smith, J. Abel, J. Pakarinen, and D. Berners, *Virtual analog effects*, 2nd ed. United Kingdom: John Wiley & Sons, 2011, pp. 473–522.

[2] T. Vanhatalo, P. Legrand, M. Desainte-Catherine, P. Hanna, A. Brusco, G. Pille, and Y. Bayle, "A review of neural network-based emulation of guitar amplifiers," *Applied Sciences*, vol. 12, no. 12, 2022.

[3] G. Borin, G. De Poli, and D. Rocchesso, "Elimination of delay-free loops in discrete-time models of nonlinear acoustic systems," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 5, pp. 597–605, 2000.

[4] A. Falaize-Skrzek and T. Hélie, "Simulation of an analog circuit of a wah pedal: a port-Hamiltonian approach," in *Proceedings of the 135th Audio Engineering Society (AES) Convention*. New York, USA: Audio Engineering Society, 2013.

[5] G. De Sanctis and A. Sarti, "Virtual analog modeling in the wave-digital domain," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 4, pp. 715–727, 2010.

[6] F. Esqueda, B. Kuznetsov, and J. D. Parker, "Differentiable white-box virtual analog modeling," in *Proceedings of the 24th International Conference on Digital Audio Effects (DAFx20in21)*. Vienna, Austria: IEEE, 2021, pp. 41–48.

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

[8] J. Chowdhury and C. J. Clarke, "Emulating diode circuits with differentiable wave digital filters," in *Proceedings of the 19th Sound and Music Computing Conference*. Saint-Étienne, France: SMC Network, 2022, pp. 2–9.

[9] O. Massi, A. I. Mezza, R. Giampiccolo, and A. Bernardini, "Deep learning-based wave digital modeling of rate-dependent hysteretic nonlinearities for virtual analog applications," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2023, no. 12, 2023.

[10] ——, "Training neural models of nonlinear multi-port elements within wave digital structures through discrete-time simulation," in *Proceedings of the 28th International Conference on Digital Audio Effects (DAFx25)*. Ancona, Italy: Università Politecnica delle Marche, September 2025, pp. 94–101.

[11] A. Wright, E. Damskägg, V. Välimäki *et al.*, "Real-time black-box modelling with recurrent neural networks," in *Proceedings of the 22nd International Conference on Digital Audio Effects (DAFx-19)*. Birmingham, UK: University of Birmingham, 2019.

[12] A. Carson, A. Wright, J. Chowdhury, V. Välimäki, and S. Bilbao, "Sample rate independent recurrent neural networks for audio effects processing," in *Proceedings of the 27th International Conference on Digital Audio Effects (DAFx24)*. Guildford, UK: University of Surrey, 2024, pp. 17–24.

[13] A. Carson, V. Välimäki, A. Wright, and S. Bilbao, "Resampling filter design for multirate neural audio effect processing," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 2163–2174, 2025.

[14] Z. Li, N. B. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," in *Proceedings of the 9th International Conference on Learning Representations (ICLR 2021)*, Online, 2021.

[15] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces with applications to PDEs," *Journal of Machine Learning Research*, vol. 24, no. 89, pp. 1–97, 2023.

[16] J. Paulus and M. Torcoli, "Sampling frequency independent dialogue separation," in *Proceedings of the 30th European Signal Processing Conference (EUSIPCO)*, 2022, pp. 160–164.

[17] K. Saito, T. Nakamura, K. Yatabe, and H. Saruwatari, "Sampling-frequency-independent convolutional layer and its application to audio source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2928–2943, 2022.

[18] A. Carson, C. Valentini-Botinhao, S. King, and S. Bilbao, "Differentiable grey-box modelling of phaser effects using frame-based spectral processing," in *Proceedings of the 26th International Conference on Digital Audio Effects (DAFx23)*. Copenhagen, Denmark: Aalborg University, September 2023, pp. 219–226.

[19] J. D. Parker, S. J. Schlecht, R. Rabenstein, and M. Schäfer, "Physical modeling using recurrent neural networks with fast convolutional layers," in *Proceedings of the 25th International Conference on Digital Audio Effects (DAFx20in22)*. Vienna, Austria: Vienna University of Music and Performing Arts, 2022, pp. 138–145.

[20] V. Välimäki, R. Salmi, S. Bilbao, S. J. Schlecht, and D. Ziccarelli, "Zero-phase sound via giant FFT," in *Proceedings of the 28th International Conference on Digital Audio Effects (DAFx25)*. Ancona, Italy: Università Politecnica delle Marche, September 2025, pp. 290–297.

[21] R. Simionato and S. Fasciani, "Comparative study of state-based neural networks for virtual analog audio effects modeling," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2025, no. 30, July 2025.

[22] F. Esqueda and S. Murai, "Antialiased black-box modeling of audio distortion circuits using real linear recurrent units," in *Proceedings of the 28th International Conference on Digital Audio Effects (DAFx25)*. Ancona, Italy: Università Politecnica delle Marche, September 2025, pp. 86–93.

[23] R. Diaz, C. De La Vega Martin, and M. Sandler, "Towards efficient modelling of string dynamics: A comparison of state space and Koopman based deep learning methods," in *Proceedings of the 27th International Conference on Digital Audio Effects (DAFx24)*. Guildford, UK: University of Surrey, 2024, pp. 200–207.

[24] F. Eichas, S. Möller, and U. Zölzer, "Block-oriented modeling of distortion audio effects using iterative minimization," in *Proceedings of the 18th International Conference on Digital Audio Effects (DAFx-15)*. Trondheim, Norway: Norwegian University of Science and Technology, 2015, pp. 243–248.

[25] A. Anandkumar, K. Azizzadenesheli, K. Bhattacharya, N. Kovachki, Z. Li, B. Liu, and A. Stuart, "Neural operator: Graph kernel network for partial differential equations," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020.

[26] V. R. Vapnik, "Information science and statistics," in *The Nature of Statistical Learning Theory*, 2nd ed. NY, USA: Springer New York, 1999.

[27] V. Välimäki and S. Bilbao, "Giant FFTs for sample rate conversion," *Journal of the Audio Engineering Society*, vol. 71, pp. 88–99, March 2023.

[28] J. O. Smith III, *Spectral Audio Signal Processing*. W3K Publishing, 2006. [Online]. Available: http://ccrma.stanford.edu/~jos/sasp/

[29] E. Jacobsen and R. Lyons, "The sliding DFT," *IEEE Signal Processing Magazine*, vol. 20, no. 2, pp. 74–80, 2003.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 7-9 2015.