DEEP REGULARIZED RNNS FOR VIRTUAL ANALOG MODELING

V. Valtteri Kallinen and Lauri Juvela

Dept. of Information and Communications Engineering (DICE)
Aalto University
Espoo, Finland
valtteri.kallinen@aalto.fi

Thom Sherson

Neural DSP Technologies Oy
Helsinki, Finland
thom@neuraldsp.com

ABSTRACT

Virtual analog (VA) modeling methods seek to emulate analog audio hardware using digital signal processing (DSP). Modeling approaches fall into three broad categories: white-box methods, which use detailed device knowledge for accurate simulation; gray-box methods that use generic DSP blocks to model the system; and black-box methods, which rely solely on opaque models learned from input–output data. A category of architectures used widely in black-box modeling are recurrent neural networks (RNNs). To model device controls, the control values can be provided as conditioning input to the network. However, when the conditioning is time-varied, the models are susceptible to producing noise artifacts. Regularization of the RNN dynamics significantly reduces these artifacts, though at a loss in modeling accuracy. This paper closes the dynamics regularization quality gap by introducing deep control-conditioned LSTMs and a gammatone filterbank (GFB) loss. Experiments indicate that the proposed method achieves comparable modeling performance as unregularized baselines while avoiding the noise artifacts caused by time-varying control inputs.

1. INTRODUCTION

Virtual analog (VA) modeling seeks to digitally emulate analog audio hardware, enabling faithful reproduction of device behavior within modern signal-processing systems and workflows. Methodologically, VA modeling spans a spectrum from white-box models that fully derive their behavior from circuit schematics through gray- and black-box models in which the desired behavior is inferred or learned from measured data. Gray-box models lie between the two extremes and typically employ traditional digital signal processing (DSP) blocks (filters, nonlinear waveshapers, dynamic processors) using some prior general model of the device behavior [1]. Black-box methods, on the other hand, typically leverage conventional machine learning (ML) architectures to capture the nonlinear and dynamic characteristics of audio circuits [1]. These data-driven models are especially useful when detailed expert knowledge or the time required for analyzing electronic circuits is unavailable. In this work, we follow the black-box, ML-based line of research.

Recently, recurrent neural networks (RNNs), especially long short-term memory (LSTM) [2] and gated recurrent unit (GRU) [3] architectures, have successfully been applied to black-box virtual analog modeling applications [1, 4, 5]. Such networks are trained in a supervised manner using paired input and output audio recorded from a device. Given enough data, the models learn to generalize to new inputs and produce an output that closely matches the target device.

When used to simulate devices that have controls to adjust various processing-related parameters (such as filter cutoffs, resonances, amount of distortion, or time constants), control values are commonly recorded as static parameter metadata with the audio data [6, 7]. These can be used as extra inputs to condition the model, allowing controllability over the sound as with the real hardware. There are several proposed methods for applying control conditioning to RNNs: the simplest solution is to concatenate the control values with the audio input [6, 7, 8]. Another general approach is to directly modulate either the parameters of the model or the hidden features using an additional hyper-network [9, 10].

In general, neural networks for black-box VA modeling are trained using a time or frequency-domain loss function, or a combination of the two [11, 12]. The most frequently employed methods include the use of mean absolute error (MAE) [10], mean squared error (MSE) [9], error-to-signal ratio (ESR) [11], and multi-resolution short-time Fourier transform (MR-STFT) [10, 12] or their mappings to other scales, such as the Mel scale [13].

Time-domain loss functions are beneficial since they preserve the phase information of the target signal. Correct phase response is important in virtual analog modeling of distortion and compressor effects since these are commonly blended with the dry signal. A mismatch in phase response can result in unwanted amplification or cancellation in such cases. However, time-domain loss values map poorly to the perceptual quality of the output [14], and therefore frequency-domain loss functions are advantageous when aiming for higher perceptual accuracy. Therefore, a common approach is to combine time and frequency-domain loss functions and adjust the sum to get the desired behavior. Unfortunately, this introduces another hyperparameter, and balancing multiple loss functions can be difficult since the time and frequency-domain losses often have differing scales and convergence rates.

Recently, in [15], it was demonstrated that LSTMs and GRUs suffer from audible artifacts caused by time-varying control conditioning. To eliminate this behavior, the work proposed restricting the RNN dynamics to an asymptotically stable regime based on similar ideas in the fields of control and systems theory [16, 17, 18, 19]. However, the regularized models did not reach similar performance as their original unregularized counterparts. Additionally, the study was restricted to only a single configuration of the LSTM and GRU networks, namely models with a single 32-unit RNN layer.

This paper extends the work in [15] by introducing a deep control-conditioned architecture for VA modeling based on LSTMs. Furthermore, a gammatone filterbank (GFB) loss for model training is proposed. The results indicate that, when combined, the GFB loss and deep architecture enable the regularized models to

match the performance of MAE-trained baselines while reducing conditioning-induced noise. It is further demonstrated that regularized models benefit from additional depth compared to simply increasing the layer width.

The remainder of the paper is organized as follows: Section 2 details the regularization formulations, the network architecture, and the GFB loss. Section 3 describes the experimental setup, including tested configurations, datasets, and training procedures. Section 4 presents the empirical results. Section 5 provides discussion, and Section 6 concludes and outlines future directions. Code used for training, sound demos of the models, and training results are provided online. [1] [2] [3]

2. METHODS

This section introduces the modeling approach, including the regularization applied to recurrent layers, the control-conditioning strategy for deep LSTMs, and the GFB augmentation for time-domain loss.

2.1. Regularized RNNs

As demonstrated in [15], the key benefit of regularized RNNs is reduced control conditioning-induced noise that can happen during real-time inference when the control values are varied. This is demonstrated in Figure 1: as the controls are varied, the model produces an audio output resembling a crackling sound. In [15], limiting the model's dynamic behavior under zero audio input to an asymptotically stable regime was found to eliminate this noise.

Let $\mathbf{x}_t$ denote the current audio input, $\mathbf{h}_t$ the hidden state, and $\mathbf{p}_t$ the conditioning input at time step t. Additionally, let $\mathbf{W}_*$, $\mathbf{U}_*$, and $\mathbf{C}_*$ denote the input, hidden, and conditioning weight matrices, respectively, and let $\mathbf{b}_*$ denote the bias vectors. Finally, notating the standard sigmoid logistic function using $\sigma(\cdot)$, the hyperbolic tangent function using $\phi(\cdot)$, and the Hadamard product using $\odot$; the operation of a control-conditioned LSTM layer is governed by the following set of equations:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i\mathbf{x}_t + \mathbf{U}_i\mathbf{h}_{t-1} + \mathbf{C}_i\mathbf{p}_t + \mathbf{b}_i), \quad (1)$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_f\mathbf{x}_t + \mathbf{U}_f\mathbf{h}_{t-1} + \mathbf{C}_f\mathbf{p}_t + \mathbf{b}_f), \quad (2)$$
$$\mathbf{g}_t = \phi(\mathbf{W}_g\mathbf{x}_t + \mathbf{U}_g\mathbf{h}_{t-1} + \mathbf{C}_g\mathbf{p}_t + \mathbf{b}_g), \quad (3)$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_o\mathbf{x}_t + \mathbf{U}_o\mathbf{h}_{t-1} + \mathbf{C}_o\mathbf{p}_t + \mathbf{b}_o), \quad (4)$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t, \quad (5)$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \phi(\mathbf{c}_t). \quad (6)$$

This work employs the regularized LSTM variant from [15], where the following constraints were shown to guarantee asymptotic stability [15]:

$$\mathbf{C}_g = O, \quad \mathbf{b}_g = \mathbf{0}, \quad \|\mathbf{U}_g\|_\infty < 1, \quad \|\mathbf{f}_t + \mathbf{i}_t\|_\infty \leq 1, \quad (7)$$

where $\|\cdot\|_\infty$ denotes the induced matrix L_∞-norm. Additionally, we conduct a comparison utilizing a less strict regularization for the hidden weights of the cell gate ($\mathbf{U}_g$) by replacing the L_∞-norm with the spectral norm (induced L_2-norm) such that

$$\|\mathbf{U}_g\|_2 < 1. \quad (8)$$

[1] https://codeberg.org/rantlivelintkale/dr-rnn-va
[2] https://rantlivelintkale.codeberg.page/dr-rnn-va/
[3] https://doi.org/10.5281/zenodo.20406285

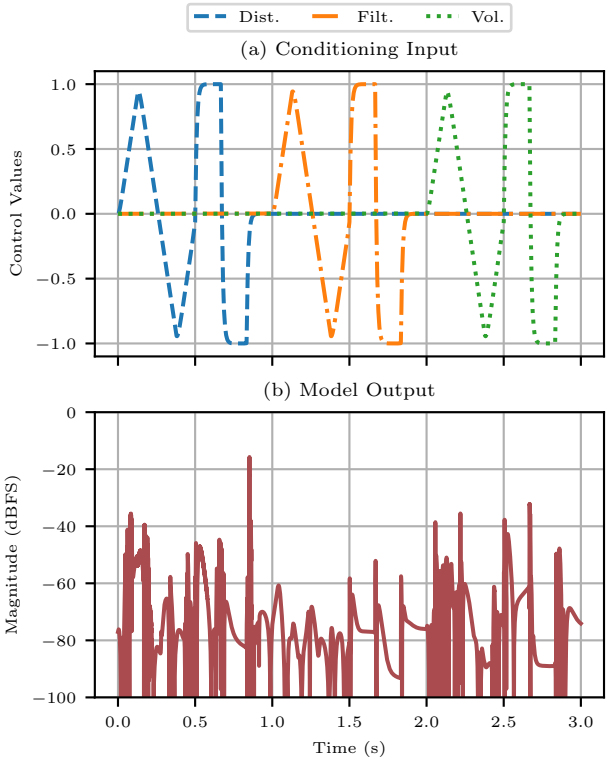

Figure 1: An example of typical control noise generated by a model under time-varying control conditioning with zero audio input. Output plotted for a 4×64 LSTM model trained on the RAT dataset. Varied control conditioning creates unwanted audible sound artifacts. Conditioning inputs are smoothed using a low-pass filter to produce values that resemble ideal conditions in a real-time use context.

While this does not formally satisfy the same stability guarantees as the L_∞-norm variant, in practice it was found to still significantly reduce control noise while allowing slightly better use of model capacity.

Equations (7, 8) were implemented via a reparametrization of the model parameters as in [20, 21]. Using reparameterizations means the model will always satisfy the constraints defined by the equations during and after training. The reparametrization for the hidden weight matrix norms was implemented as follows:

$$\mathbf{U}_g = \mathbf{U}_g \cdot \begin{cases} \frac{\tau}{\|\mathbf{U}_g\|} & \text{if } \|\mathbf{U}_g\| \geq \tau, \\ 1 & \text{otherwise,} \end{cases} \quad (9)$$

where $\tau < 1$ is a threshold to satisfy the constraint $\|\mathbf{U}_g\| < 1$. The input gate $\mathbf{f}_t$ and forget gate $\mathbf{i}_t$ were reparameterized as in [15] by mirroring forget gate parameters to the input gate, that is

$$\mathbf{U}_i = -\mathbf{U}_f, \quad \mathbf{C}_i = -\mathbf{C}_f, \quad \mathbf{b}_i = -\mathbf{b}_f, \quad (10)$$

and limiting the parameter values of $\mathbf{U}_f$, $\mathbf{C}_f$ and $\mathbf{b}_f$ such that $\|\mathbf{f}_t\|_\infty \leq 1$. Since mirroring these weights guarantees that $\mathbf{i}_t = \mathbf{1} - \mathbf{f}_t$ and thus $\|\mathbf{f}_t + \mathbf{i}_t\|_\infty \leq 1$ under zero audio input.

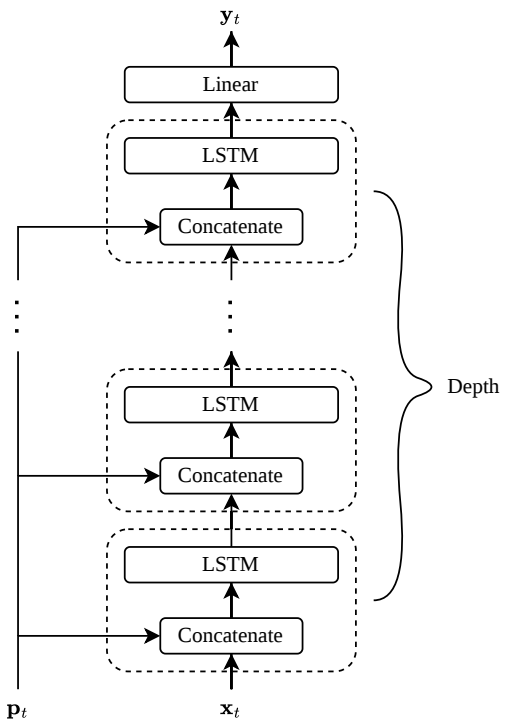

Figure 2: Deep architecture for control-conditioned LSTMs. The first layer receives as input the concatenation of audio signal $\mathbf{x}_t$ and control conditioning $\mathbf{p}_t$. Subsequent layers receive as input the concatenation of the previous layers hidden and the control conditioning $\mathbf{p}_t$. The output signal $\mathbf{y}_t$ is produced by passing the output of the final LSTM layer through a fully connected layer. Depth refers to the number of serial LSTM layers.

2.2. Deep Architecture for Concatenation-Conditioned RNNs

The deep conditioned LSTM network used in this paper is implemented as shown in Fig. 2. The network takes an audio signal $\mathbf{x}_t$ and conditioning input $\mathbf{p}_t$. The conditioning is applied by simply concatenating it as an extra input channel to each LSTM layer. This can be considered an extension of the common concatenation-based conditioning used in [6, 7, 8] to deeper networks. The architecture is not LSTM specific; for example, the LSTM layers could be replaced by GRUs. Notably, regularized deep GRU networks can be implemented using the constraints for GRUs as defined in [15].

2.3. Model Architecture and Regularization Notation

The complete models with regularization are notated as follows:

$$\mathrm{LSTM}_{\mathrm{norm}}\ \mathrm{depth} \times \mathrm{hidden\ size}, \tag{11}$$

where 'norm' is either the spectral norm (2) Eq. (8) or the infinity norm (∞) Eq. (7), depth is the number of layers, and hidden size is the number of units in each LSTM layer.

2.4. Gammatone Filterbank Extension for Time-Domain Loss

In [22], the standard time-domain loss was combined with an A-weighting filter to achieve better perceptual quality without using a frequency-domain loss function. In this paper we adopt a similar perceptually inspired approach. First, the error signal between the target and prediction is calculated as usual:

$$\mathbf{e}_t = \mathbf{y}_t - \hat{\mathbf{y}}_t. \tag{12}$$

This error signal is then weighted using a gammatone filter bank (GFB) [23], the absolute errors of the channels are summed, and the mean of the summed errors at each time step is calculated. The GFB-filtered error should provide better accuracy across multiple frequency ranges, similar to the common frequency-domain loss functions, while preserving the phase response since the loss is calculated from the time-domain error signal.

In this paper, a 32-channel filterbank is used with frequencies spread out over the ERB scale starting from a frequency of 100 Hz with subsequent frequencies being a distance of one further on the ERB scale as defined in [24]. With 32 channels, this gives us a GFB with frequencies up to $\approx 10\,\mathrm{kHz}$. Since the ERB scale as defined in [24] is only applicable for frequencies between 100 Hz and 10 kHz, we include a residual error term, which is the difference between the filter outputs and the error signal, to model the error outside the GFB. Letting C be the number of channels in the gammatone filterbank, this residual is defined as:

$$\mathrm{GFB}(\mathbf{e}_t)_{t,\mathrm{res}} = \mathbf{e}_t - \sum_{c=1}^{C} \mathrm{GFB}(\mathbf{e}_t)_{t,c}. \tag{13}$$

Letting N be the number of samples, the GFB loss is then defined as:

$$\mathcal{L}_{\mathrm{GFB}} = \frac{1}{N} \sum_{t=1}^{N} \left(|\mathrm{GFB}(\mathbf{e}_t)_{t,\mathrm{res}}| + \sum_{c=1}^{C} |\mathrm{GFB}(\mathbf{e}_t)_{t,c}| \right). \tag{14}$$

Other configurations with varying numbers of channels, with or without the residual, could also be applied. However, improved results were already observed using this 32-channel layout with the residual. Thus, this arrangement was used during this study.

2.5. Evaluation Metrics

To compare the performance of the different architectures proposed, we consider three metrics: ESR [5, 8], MR-STFT as defined and configured in [25], and the GFB loss described in the previous section.

In addition, we report the energy of the control conditioning noise generated by each model. For these control noise (CN) tests, each network was first initialized using an impulse and allowed to stabilize for one second. Then, the controls were varied, and the signal energy during the time-varying conditioning was recorded. The controls used for conditioning were varied in the same way as in Fig. 1.

3. EXPERIMENTS

This section details the experiments performed, including a description of the evaluated systems and data, the measurement and training procedures, and the model configurations used to study the effects of depth, width, and regularization.

3.1. Testing Different Network and Loss Configurations

Short training runs of 100 epochs each were conducted to investigate the way in which performance scaled with model width (number of units per RNN layer) and depth (number of sequential RNN layers). The tested configurations included all combinations of hidden sizes $d \in \{8, 16, 32, 64\}$ and depths $l \in \{1, 2, 3, 4\}$. In addition, training was performed for both unregularized and regularized LSTM variants to detect any impacts that regularization may have on this scaling. These short runs were trained and evaluated using the GFB loss defined in Eq. (13).

Based on the results of the short training runs, three configurations were chosen for longer training runs of 1000 epochs ($\approx 700\,000$ parameter updates for all datasets) to allow the models to fully converge. The configurations used were one layer of 64 units (1×64), four layers of eight units (4×8) and four layers of 64 units (4×64). Similarly, the unregularized and regularized variants were tested. Furthermore, to observe the effects of the GFB loss on performance, the models were also trained using the MAE loss as a baseline. These longer training runs were conducted using three different random number generator seeds to account for sampling bias.

3.2. Datasets and Training Details

As in [15], this paper uses the "Dataset for asymptotically stable recurrent neural networks" [26]. The training dataset consists of three devices: the ProCo Rat (RAT), Darkglass Duality Fuzz (DFZ), and Boss CS-3. The audio in the dataset consists of 1-second clips of dry electric guitar and bass guitar playing. The dataset was captured using different control combinations, and therefore it can be used to train control-conditioned models. For the ProCo Rat, the dataset has values for all three controls: distortion, filter, and volume. For the Duality Fuzz, the fuzz blend and filter controls were varied while the volume was set to the maximum position and the dry blend set to fully wet. The CS-3 dataset only has values for the attack control, while the volume and sustain controls were set to their maximum values while the tone was set to the minimum value. All audio uses a sampling rate of 48 kHz and the trained models operate under this sampling rate assumption for the input audio. The dataset is split into training and validation sets, and the validation set was used for testing as well. Further details can be found in the dataset repository.

The reason for choosing to use this dataset is the current lack of high-quality, openly available datasets for control-conditioned training. The Marshall JVM 410H dataset [27] contains too few control positions to allow a black-box model to properly generalize across different control position combinations. The pOD dataset [28], while very extensive, only includes recorded affected wet signals. Although the recorded signals were adjusted for AD/DA conversion latency, this does not account for any coloration or phase distortion caused by the passive re-amplification device used before the overdrive pedals. Moreover, all the pedals are variations of different overdrive pedals, and the controls are limited to two parameters: gain and tone.

The models and training were implemented using the PyTorch machine learning (ML) framework [29]. The Adam optimizer was used for training with a learning rate of 10^{-3} and no weight decay. As in [8], the models were trained using truncated backpropagation through time (TBPTT) with a truncation step size of 2048 samples and the training data was processed in batches of 32 samples.

4. RESULTS

This section reports objective evaluation outcomes across model configurations, including results from the short runs and extended trainings, summary tables for evaluation metrics, parameter counts, and Bark-smoothed error-residual spectra over the evaluation sets.

4.1. Results for Different Network Configurations

Fig. 3 shows the results for the network configuration test outlined in Sec. 3.1. For all architectures, increasing model capacity by increasing depth and width resulted in a reduction in evaluation loss. However, in the case of the regularized variants, a degradation in model performance for equivalent network size can be noted. This impact is slightly reduced for the proposed L_2-norm regularization, although a gap with the standard LSTM persists.

For the longer training runs, also described in Sec. 3.1, the results are gathered in Table 1. The table lists the metrics for the best models out of three runs. Observing the results for the models shown in Fig. 3 indicates that the models chosen in the first run still improved based on the evaluation GFB loss values.

4.2. Effect of Loss Function on Model Performance

A general improvement in the evaluation ESR, MR-STFT, and GFB can be seen for models trained using the GFB loss. However, sometimes an improvement in GFB does not result in an improvement in ESR or MR-STFT, as is the case for some unregularized LSTM 4×8 models. For example, the MAE-trained LSTM 4×8 RAT model has an evaluation ESR_{dB} of -15.2 while the same metric for the GFB-trained model is -13.8.

4.3. Error Signal Spectrum Comparison

The difference in modeling performance across the frequency spectrum is visualized in Fig. 4 and 5. The figures depict the Bark-smoothed spectrums of the error residual from the evaluation set for each device.

Fig. 4 shows the error residual spectrums for the L_∞ regularized models for the 1×64 and 4×8 configurations. It can be noted that the deeper and narrower 4×8 model consistently achieves lower error across the entire spectrum for all datasets. The decreasing slope towards higher frequencies can be considered a reflection of the modeled device's output spectra and does not necessarily indicate a better modeling performance at higher frequencies. However, the figures still highlight a relative improvement between the error spectrums of the models considered. For the RAT, the results are improved evenly across the spectrum, whereas for the DFZ and the CS-3 the improvements drop off towards the higher frequencies. A similar comparison was used in [22] where an improvement in the error residual around sensitive frequencies correlated with better results during a listening test. Thus, improvements in this metric should also indicate improvements in subjective results.

Fig. 5 shows the error residuals for the MAE-trained LSTM and LSTM_∞ and the GFB-trained LSTM_∞ 4×64 models. This shows the difference between the GFB loss-trained regularized model and the unregularized model, which can be considered a baseline for comparison. While the difference is modest, the GFB loss enables the L_∞-regularized model to reach similar performance as the unregularized counterpart.

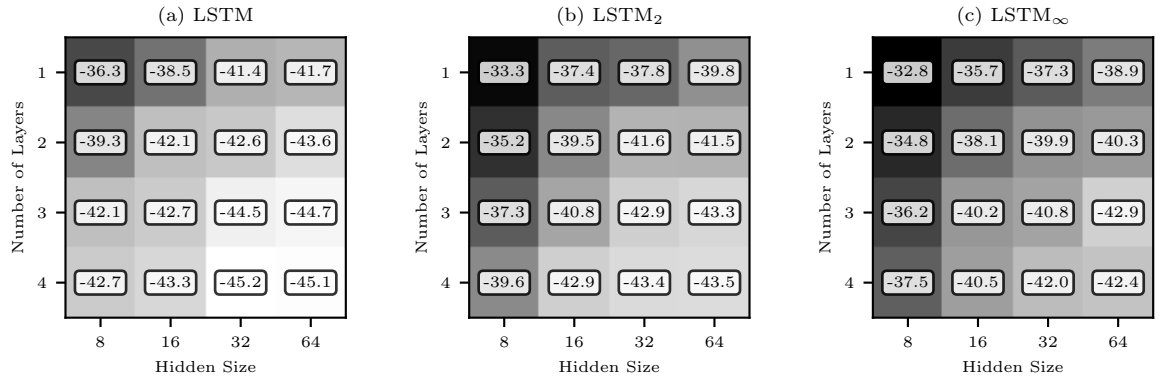

Figure 3: Training results from the short trial for different configurations of (a) LSTM, (b) LSTM$_2$ and (c) LSTM$_\infty$ on the RAT dataset. Each cell of a results matrix is a specific configuration of depth and width, for example, 4 layers of 8 LSTM units. Cell values indicate the best GFB evaluation loss on a decibel scale during a training run. Lower values are better.

Table 1: Evaluation results for the best models from 3 longer training runs (1000 epochs). Metrics were computed on the evaluation sets of each device, except for the control noise (CN) which was computed as described in 2.5. Lower values indicate better performance. A decibel value of $-\infty$ indicates a value of exactly zero on a linear scale.

			RAT				DFZ				CS-3			
Loss	Model	Config.	ESR_{dB}	MR-STFT	GFB_{dB}	CN_{dB}	ESR_{dB}	MR-STFT	GFB_{dB}	CN_{dB}	ESR_{dB}	MR-STFT	GFB_{dB}	CN_{dB}
MAE	LSTM	1×64	-17.2	0.248	-43.6	-64.2	-19.0	0.449	-31.1	-37.4	$\mathbf{-27.1}$	**0.080**	-63.7	-74.4
		4×8	-15.2	0.288	-42.4	-65.2	-15.2	0.563	-27.3	-26.9	-23.4	0.111	-59.9	-78.7
		4×64	-18.9	0.198	-45.1	-65.0	-20.3	0.442	-33.3	-33.5	-21.4	0.133	-63.9	-58.0
	LSTM$_2$	1×64	-16.4	0.343	-42.0	$-\infty$	-15.1	0.548	-26.1	$-\infty$	-20.6	0.133	-56.6	-209.1
		4×8	-16.0	0.267	-41.8	$-\infty$	-17.0	0.463	-28.9	$-\infty$	-21.3	0.118	-57.8	-220.1
		4×64	-18.5	0.211	-44.8	$-\infty$	-20.4	0.404	-32.4	$-\infty$	-25.3	0.090	-63.6	-156.8
	LSTM$_\infty$	1×64	-14.2	0.403	-39.6	$-\infty$	-12.7	0.641	-23.3	$-\infty$	-13.7	0.247	-50.6	$-\infty$
		4×8	-15.4	0.295	-41.2	$-\infty$	-17.0	0.477	-28.4	$-\infty$	-19.1	0.147	-56.2	-448.5
		4×64	-17.5	0.226	-44.0	$-\infty$	-19.8	0.423	-32.1	$-\infty$	-23.9	0.102	-62.5	$-\infty$
GFB	LSTM	1×64	-17.6	0.244	-45.2	-61.2	-18.9	**0.392**	-32.3	-40.5	-25.9	0.086	$\mathbf{-64.7}$	-86.8
		4×8	-13.8	0.341	-43.2	-57.1	-15.5	0.493	-29.8	-28.8	-22.2	0.115	-60.2	-64.1
		4×64	$\mathbf{-19.3}$	**0.194**	$\mathbf{-46.2}$	-56.1	-19.0	0.439	-33.2	-38.2	-25.6	0.098	-63.0	-58.6
	LSTM$_2$	1×64	-17.3	0.279	-43.0	$-\infty$	-15.8	0.505	-27.6	$-\infty$	-20.1	0.132	-57.0	-174.2
		4×8	-17.4	0.255	-43.4	$-\infty$	-19.1	0.431	-31.1	$-\infty$	-20.8	0.126	-57.7	-165.4
		4×64	-19.2	0.198	-45.6	$-\infty$	$\mathbf{-20.8}$	0.393	$\mathbf{-33.7}$	$-\infty$	-26.5	0.083	-64.3	-202.9
	LSTM$_\infty$	1×64	-15.7	0.337	-41.3	$-\infty$	-12.8	0.591	-24.4	$-\infty$	-13.3	0.234	-50.9	$-\infty$
		4×8	-16.0	0.283	-41.9	$-\infty$	-17.4	0.465	-29.7	$-\infty$	-19.4	0.142	-56.6	$-\infty$
		4×64	-18.0	0.219	-44.9	$-\infty$	-19.9	0.425	-32.5	$-\infty$	-23.2	0.104	-62.6	$-\infty$

4.4. Control Noise

The results of the control noise test can be seen in Table 1. The results indicate that models trained using the L_2-norm exhibit similar control noise suppression as the L_∞-norm regularized models. Since levels below -100 dBFS are lower than the noise floor for a generic audio interface, the control noise induced for the regularized models is practically zero under zero audio input. That is, the highest control noise signal energy for the regularized models is -156 dBFS.

4.5. Computational Complexity

To give a rough estimate of the computational complexity of the different configurations, Table 2 lists the number of parameters in the models trained on each dataset for the different configurations used in the longer training. The number of parameters scales quadratically with width while only scaling linearly with depth. As such, the 4×8 has the least number of parameters while the 4×64 has the most. Notably, the 4×8 models have approximately an order of magnitude fewer parameters than their 1×64 counterparts.

5. DISCUSSION

The results demonstrate a consistent trade-off between accuracy, capacity, and stability. Scaling depth and width lowers evaluation loss across architectures, confirming that larger models fit the data better; however, for equivalent sizes, both stability-regularized variants show degraded performance relative to the unregularized LSTM, with a smaller gap for the proposed L_2-norm regularization than for the L_∞-constrained model. These patterns were first observed in the short, 100-epoch runs over depth and width, and motivated the choice of three representative configurations, 1×64, 4×8, and 4×64, for extended, 1000-epoch training runs. With extended training, the selected models continued to improve, and additionally, the performance gap between unregularized and

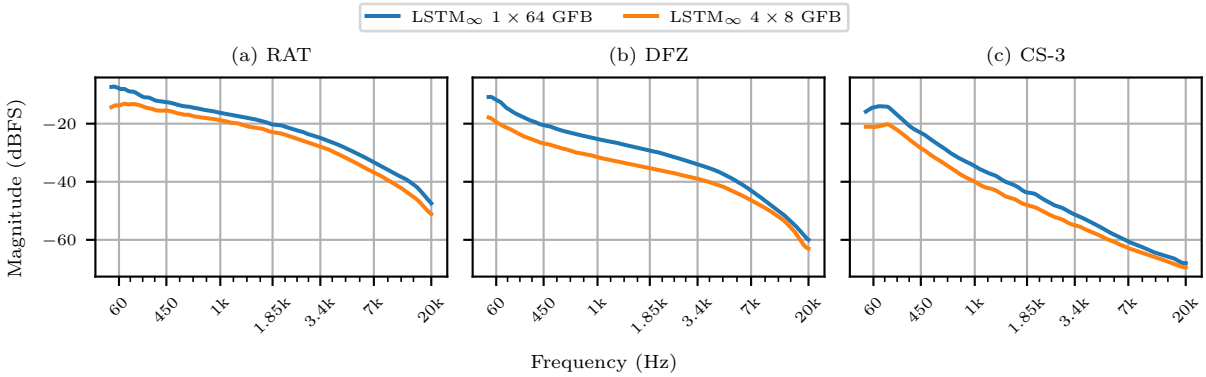

Figure 4: Bark smoothed spectrums of the error residuals over the evaluation sets for LSTM$_\infty$ 1×64 and 4×8 trained using GFB loss. Lower magnitudes indicate better performance. Deeper and narrower configurations consistently show better performance while having fewer parameters

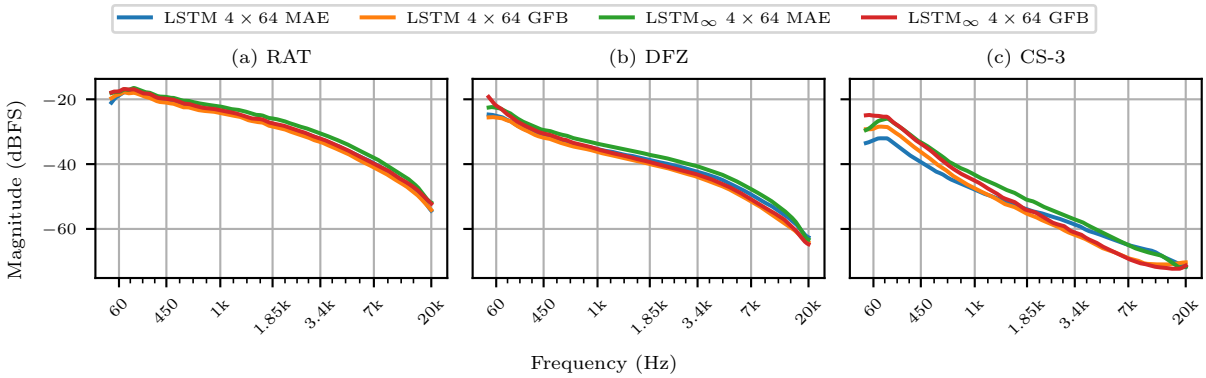

Figure 5: Bark smoothed spectrums of the error residuals over the evaluation sets for LSTM trained using the MAE loss and LSTM$_\infty$ models trained using MAE and GFB loss. Lower magnitudes indicate better performance. GFB loss helps the regularized model obtain similar performance to the unregularized baseline.

Table 2: Model parameter amounts for different models and datasets. The RAT had three controls, the DFZ two controls, and the CS-3 one control. The number of parameters affects the size of the conditioning input vector and therefore the size of the conditioning weight matrices. The number of parameters does not significantly change with the number of conditioning inputs. The number of output parameters is the same for all datasets and listed separately. The floating-point operations (FLOPS) per sample approximation does not include nonlinear activations.

	Number of Parameters				
Config.	RAT	DFZ	CS-3	Output Layer	FLOPS/sample
1×64	17 664	17 408	17 152	65	$\approx$ 18 000
4×8	2336	2208	2080	9	$\approx$ 2400
4×64	119 040	118 016	116 992	65	$\approx$ 120 000

regularized models was narrowed. For example, the GFB loss between the 4×8 RAT models for the initial run was ≈ 5.2 dB while after the longer training run, the difference was reduced to ≈ 1.3 dB.

Choice of training objective (MAE vs. GFB) affected the results in a generally consistent manner. Comparing models trained with MAE versus the GFB loss, evaluation ESR, MR-STFT, and GFB metrics generally improved with the GFB objective, although gains in GFB did not always coincide with improvements in ESR and MR-STFT (notably for some 4×8 unregularized LSTMs). Lower GFB loss values indicated better error spectra, which, given prior work in perceptual time-domain loss functions [22], should indicate subjective improvements in sound quality.

Architectural allocation of capacity also mattered under stability constraints. The results indicate that, for regularized models, deeper, narrower configurations (4×8) outperform shallower, wider ones (1×64) across devices. These findings, when combined with the way the number of parameters scales with depth versus width, suggest that regularized models benefit more from distributing capacity across depth. For example, for the LSTM$_\infty$ 1×64 and 4×8 models trained on the DFZ dataset, the ESR$_{\text{dB}}$ values were -12.8 and -17.4, respectively. Whereas for the unregularized LSTM, the values for the same configurations were -18.9 and -15.5, indicating better utilization of the added depth despite the decrease in the total number of parameters.

6. CONCLUSIONS

This work shows that deeper, concatenation-conditioned LSTMs trained with the proposed GFB loss can achieve competitive modeling performance while eliminating the control-induced crackling artifacts that afflict unregularized models during real-time parameter changes. Across configurations, increasing capacity improves accuracy, but stability constraints introduce a modest performance penalty at matched sizes; the proposed L_2-norm regularization narrows this gap relative to the proven L_∞-norm while empirically preserving the substantial control noise suppression.

Future work could pursue analytical stability characterizations for spectral-normalized recurrent layers and broaden empirical validation across additional devices and conditioning schemes. Alternatively, with the knowledge of this study, one possible avenue of research is developing alternative architectures that inherently benefit from lower levels of control conditioning-induced noise. Finally, training on time-varying control trajectories rather than only static positions could be investigated to see if these effects are simply caused by overfitting. However, currently there are no datasets with time-varying controls, and collecting such datasets presents many challenges related to the inherent difficulty in accurately actuating the controls.

7. ACKNOWLEDGMENTS

We acknowledge the computational resources provided by the Aalto Science-IT project. This research was funded by the Research Council of Finland (grant no. 371845).

8. REFERENCES

[1] M. Comunità, C. J. Steinmetz, and J. D. Reiss, "Differentiable black-box and gray-box modeling of nonlinear audio effects," *Frontiers in Signal Processing*, vol. Volume 5 - 2025, 2025.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997.

[3] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, D. Wu, M. Carpuat, X. Carreras, and E. M. Vecchi, Eds. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 103–111.

[4] M. A. Martínez Ramírez, E. Benetos, and J. D. Reiss, "Deep learning for black-box modeling of audio effects," *Applied Sciences*, vol. 10, no. 2, 2020.

[5] A. Wright, E.-P. Damskägg, L. Juvela, and V. Välimäki, "Real-time guitar amplifier emulation with deep learning," *Applied Sciences*, vol. 10, no. 3, 2020.

[6] L. Juvela, E.-P. Damskägg, A. Peussa, J. Mäkinen, T. Sherson, S. I. Mimilakis, K. Rauhanen, and A. Gotsopoulos, "End-to-end amp modeling: from data to controllable guitar amplifier models," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[7] O. Mikkonen, A. Wright, and V. Välimäki, "Sampling the user controls in neural modeling of audio devices," *EURASIP J. Audio Speech Music. Process.*, vol. 2024, p. 26, 2024.

[8] A. Wright, E.-P. Damskägg, and V. Valimaki, "Real-time black-box modelling with recurrent neural networks," in *Proceedings of the 22nd International Conference on Digital Audio Effects (DAFx2019)*, Sep. 2019, pp. 1–8.

[9] R. Simionato and S. Fasciani, "Conditioning methods for neural audio effects," in *Proceedings of the 21st Sound and Music Computing Conference*. Zenodo, Jul. 2024, pp. 411–418.

[10] Y.-T. Yeh, W.-Y. Hsiao, and Y.-H. Yang, "Hyper Recurrent Neural Network: Condition Mechanisms for Black-Box Audio Effect Modeling," in *Proceedings of the 27-th Int. Conf. on Digital Audio Effects (DAFx24)*, E. De Sena and J. Mannall, Eds., Sept. 2024, pp. 97–104.

[11] E.-P. Damskägg, L. Juvela, E. Thuillier, and V. Välimäki, "Deep learning for tube amplifier emulation," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 471–475.

[12] V. Huhtala, L. Juvela, and S. J. Schlecht, "Klann: Linearising long-term dynamics in nonlinear audio effects using koopman networks," *IEEE Signal Processing Letters*, vol. 31, pp. 1169–1173, 2024.

[13] F. Esqueda and S. Murai, "Antialiased Black-Box Modeling of Audio Distortion Circuits Using Real Linear Recurrent Units," in *Proceedings of the 28-th Int. Conf. on Digital Audio Effects (DAFx25)*, L. Gabrielli and S. Cecchi, Eds., Sept 2025, pp. 86–93.

[14] Y. Hu and P. C. Loizou, "Evaluation of objective quality measures for speech enhancement," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 229–238, 2008.

[15] V. Kallinen, "Asymptotically stable recurrent neural networks for virtual analog modelling," Master's thesis, Aalto University, Espoo, Finland, 2025.

[16] D. M. Stipanović, B. Murmann, M. Causo, A. Lekić, V. Rubies-Royo, C. J. Tomlin, E. Beigné, S. Thuries, M. Zarudniev, and S. Lesecq, "Some local stability properties of an autonomous long short-term memory neural network model," *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, 2018.

[17] S. A. Deka, D. M. Stipanović, B. Murmann, and C. J. Tomlin, "Global asymptotic stability and stabilization of long short-term memory neural networks with constant weights and biases," *Journal of Optimization Theory and Applications*, vol. 181, pp. 231 – 243, 2018.

[18] F. Bonassi, M. Farina, and R. Scattolini, "On the stability properties of gated recurrent units neural networks," *Syst. Control. Lett.*, vol. 157, p. 105049, 2020.

[19] E. Terzi, F. Bonassi, M. Farina, and R. Scattolini, "Learning model predictive control with long short-term memory networks," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 18, pp. 8877–8896, 2021.

[20] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," *ArXiv*, vol. abs/1602.07868, 2016.

[21] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," in *International Conference on Learning Representations*, 2018.

[22] A. Wright and V. Välimäki, "Perceptual loss function for neural modeling of audio systems," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 251–255.

[23] R. Patterson, I. Nimmo-Smith, J. Holdsworth, and P. Rice, "An efficient auditory filterbank based on the gammatone function," MRC Applied Psychology Unit, Cambridge, UK, Tech. Rep., 1988.

[24] B. R. Glasberg and B. C. Moore, "Derivation of auditory filter shapes from notched-noise data," *Hearing Research*, vol. 47, no. 1, pp. 103–138, 1990.

[25] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6199–6203.

[26] V. V. Kallinen, "Datasets for asymptotically stable recurrent neural networks," May 2026. [Online]. Available: https://doi.org/10.5281/zenodo.20406285

[27] Štěpán Miklánek, A. Wright, V. Välimäki, and J. Schimmel, "Marshall JVM 410H dataset," https://doi.org/10.5281/zenodo.7970723, May 2023.

[28] F. A. Dal Rì, D. Stefani, L. Turchet, and N. Conci, "Morphdrive: Latent Conditioning For Cross-Circuit Effect Modeling And A Parametric Audio Dataset Of Analog Overdrive Pedals," in *Proceedings of the 28-th Int. Conf. on Digital Audio Effects (DAFx25)*, L. Gabrielli and S. Cecchi, Eds., Sept 2025, pp. 23–30.

[29] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: an imperative style, high-performance deep learning library," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.