Enhancing Music Packet Loss Concealment with Generative Spectrogram Inpainting

Carlo Aironi*, Leonardo Gabrielli*, Samuele Cornell† and Stefano Squartini*
*Università Politecnica delle Marche, Italy †Carnegie Mellon University, USA
Email: c.aironi@staff.univpm.it, (l.gabrielli, s.squartini)@univpm.it, samuele.cornell@ieee.org

***Abstract*—This technical report addresses the issue of packet loss concealment (PLC) in Networked Music Performance (NMP) by proposing a novel method that leverages Linear Predictive Coding (LPC) combined with a bin2bin Generative Adversarial Network (GAN). Unlike existing methods such as PARCnet, which estimate the LPC error using an ANN, our approach uses the LPC-generated audio to condition a generative bin2bin GAN model for spectrogram inpainting. Experimental results show that the method significantly improves the corrupted audio quality mitigating the negative impact of packet loss and providing a robust solution for real-time audio transmission at a relatively low computational cost.**

***Index Terms*—Packet Loss Concealment, Generative Adversarial Network, Linear Predictive Coding**

I. Introduction

Networked Music Performance (NMP) relies on high-quality audio transmission with minimal latency to provide a seamless and immersive experience. However, the inherent unpredictability of network conditions often leads to packet loss, which can severely degrade audio quality and disrupt the performance. Addressing Packet Loss Concealment (PLC) in NMP is thus critical for maintaining the integrity and quality of the transmitted audio.

Traditional PLC methods, such as interpolation and repetition of lost packets, are often insufficient for the high-quality demands of NMP. More advanced techniques have been investigated since the 1990s [1], but they still face limitations in maintaining audio fidelity and minimizing latency.

In this technical report, we propose a novel method for the PLC challenge organized at the 5th IEEE International Symposium on the Internet of Sounds. It combines Linear Predictive Coding (LPC) with a bin2bin [2] Generative Adversarial Network (GAN). The baseline method proposed by the organizers is a slightly modified PARCnet [3], which is also based on a linear predictor (LP) and an artificial neural network (ANN). In PARCnet, the goal of the ANN is to estimate the LPC error, in order to correct the residuals in the time domain. In our approach, instead, a bin2bin GAN is employed to generate audio conditioned by the initial guess signal provided by the LP model. This method leverages the predictive capabilities of LPC and the generative strength of GANs to reconstruct lost packets with high fidelity.

II. Method

The goal of the challenge is to conceal one or more lost audio packets of very short duration (512 samples at 44100 Hz sampling rate). We assume that the defect locations are indicated by a binary mask, known a priori. The proposed concealment algorithm is illustrated in Figure 1. As soon as a loss occurs, the system must provide a plausible reconstruction in a fully causal setting, without any lookahead. Assuming that all previous packets are reliable (even if some were reconstructed in previous steps), the system initially performs an inpainting hypothesis using the LP, which is adapted to infer $p = 128$ coefficients from a context of 7 previous contiguous frames (3584 samples, i.e. 81.3 ms). The LP is required to produce a segment twice the length of the gap, i.e., 1024 samples. This strategy provides a prediction of future samples, useful for the subsequent potential crossfade step, and facilitates the bin2bin network operation by placing the corrupted samples slightly further away from the context boundary.

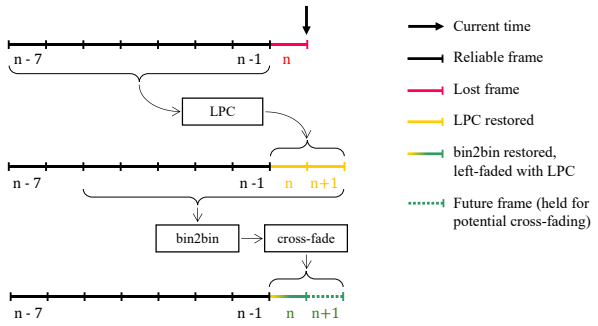

Fig. 1. Illustration of the proposed inpainting method.

A. bin2bin Model

In the proposed bin2bin approach, the generator architecture leverages the U-Net design, incorporating skip-connections between homogeneous layers. The U-Net consists of a convolutional encoder that downsamples the input spectrogram and a decoder that upsamples the latent representation using 2d transposed convolutions. Two models have been proposed with different size, named *full* and *lite* for simplicity. The latter is designed to run in real-time on a CPU, while the former requires a GPU to offload the computational power. The number of convolutional kernels increases in the encoder section as the network deepens. Specifically, for each of the 7 convolutional blocks, the number of kernels is [64, 128, 256, 512, 512, 512, 512] in the *full* setting, while they were reduced

to [16, 32, 64, 128, 128, 128, 128] in the *lite* setting; all kernels are sized 4×4 bins. At the bottleneck, an additional conv2d layer processes the signal. Following, the decoder section has a mirrored structure, and employs the same number of kernels for the 2d transposed convolution operations. The only differences between the encoder and decoder are the activation functions used (Leaky-ReLU in the encoder and ReLU in the decoder) and the inclusion of two dropout layers in the inner upsample blocks. The generator G accepts an STFT magnitude signal of size $1 \times 256 \times 256$, (channels, frequency bins, time frames). During inference, the magnitude STFT reconstructed by the generator is combined with the original STFT phase spectrogram (predicted by the LP) to reconstruct the audio signal in the time domain.

The GAN Discriminator is a typical classification network, starting with an initial convolutional layer followed by a series of CNN blocks. Each CNN block features a convolutional layer with reflection padding, batch normalization, and a LeakyReLU activation function. The network concludes with a fully connected layer that outputs a scalar value, representing the Discriminator's assessment of the input data.

B. Loss criteria

The discriminator is trained using a least-square conditional loss function to make the training more stable and alleviate the vanishing gradient problem [4]. The objective functions for the joint conditional and least squares GAN (referred to as LSCGAN) are defined as follows:

$$\min_D \mathcal{L}_{LSCGAN}(D) = \frac{1}{2}\mathbb{E}_{x,c}\left[(D(x|c)-1)^2\right] + \\ + \frac{1}{2}\mathbb{E}_{z,c}\left[(D(G(z)|c))^2\right] \quad (1)$$

$$\min_G \mathcal{L}_{LSCGAN}(G) = \frac{1}{2}\mathbb{E}_{z,c}\left[(D(G(z)|c)-1)^2\right] \quad (2)$$

The generator model is trained by combining the GAN objective with two conventional pixel-wise losses between the generated source spectrogram reconstruction and the expected target spectrogram. Based on our previous studies, we have found it beneficial to use loss functions that relate to the perceptual quality of the audio signal. These include the log-STFT magnitude loss ($\mathcal{L}_{mag}$) and Spectral Convergence loss ($\mathcal{L}_{sc}$), defined as follows:

$$\mathcal{L}_{sc}\left(S,\hat{S}\right) = \frac{\sqrt{\sum_{t,f}\left(|S_{t,f}| - |\hat{S}_{t,f}|\right)^2}}{\sqrt{\sum_{t,f}|S_{t,f}|^2}} \quad (3)$$

$$\mathcal{L}_{mag}\left(S,\hat{S}\right) = \frac{\sum_{t,f}|\log|S_{t,f}| - \log|\hat{S}_{t,f}||}{T \cdot N} \quad (4)$$

where $|S_{t,f}|$ and $|\hat{S}_{t,f}|$ represent the STFT magnitude vector of the target and the generated signal respectively, at time t, while T and N denote the number of time and frequency bins. The spectral loss is given a weight of 10, while the adversarial is given a weight of 1.

C. DNN training protocol

The training procedure involves processing chunks of 4096 samples, which equate to approximately 93 ms at the sampling rate of 44100 Hz, and correspond to 8 quantized gaps. As the first step, the audio context, randomly extracted from the dataset, is corrupted by inserting zeros at the end for a duration of 1024 samples, then a coarse reconstruction is performed with the LP. The resulting audio segment is then transformed into the time-frequency domain and fed into the bin2bin generator network, which refines the reconstruction acting as a conditional-GAN (cGAN) [5]. The reliable portion of spectrogram, preceding the gap, serves as the conditioning signal, along with the rough inpainted bins provided by the LP.

We followed a common practice in training generative networks, which consists in balancing the evolution of training by iterating n_G times the generator weights update, for every one of D. We used the value n_G = 10. The model was trained for an arbitrary number of epochs. Due to the lack of an objective metric that directly correlates with the quality of the reconstruction, we opted to systematically evaluate the training progress by listening to the generated audio samples. Based on these evaluations, we determined our criteria for selecting the best network checkpoint. Finally, we used the Adam optimizer for both the generator and the discriminator, with a learning rate of 0.0002, progressively decreased to half, with a cosine profile, and a batch size of 16.

D. Datasets

To ensure robust network generalization during the training phase, we used an ensemble of three music signals collections. The first, as recommended by the challenge organizers, is Medley Solos DB [6], comprising nearly 18 hours of solo instrument recordings across a taxonomy of eight instruments: clarinet, distorted electric guitar, female singer, flute, piano, tenor saxophone, trumpet, and violin. Additionally, to enable the network to learn an extended frequency range for each musical instrument timbre, we included the GoodSounds collection [7], which contains monophonic recordings of both sustained notes and scales. Finally, we augmented the training set by generating 45 additional hours of synthetic audio clips from MIDI sequences. For this purpose, we selected nine representative instruments from the categories of keyboards, strings and woodwinds, using the GeneralUser GS soundfont.[1]. The MIDI sequences were sourced from the MAESTRO dataset [8] and synthesized using FluidSynth [2].

III. Experiments

According to the challenge guidelines, the test set was inpainted using the proposed technique. Processing times are provided in Table I. These were measured on an Intel i7-6850K (3.6 GHz) processor from 2016 which, in benchmarks, has close performances to an Intel i5-10400F. We also show

[1]https://schristiancollins.com/generaluser.php

[2]https://www.fluidsynth.org

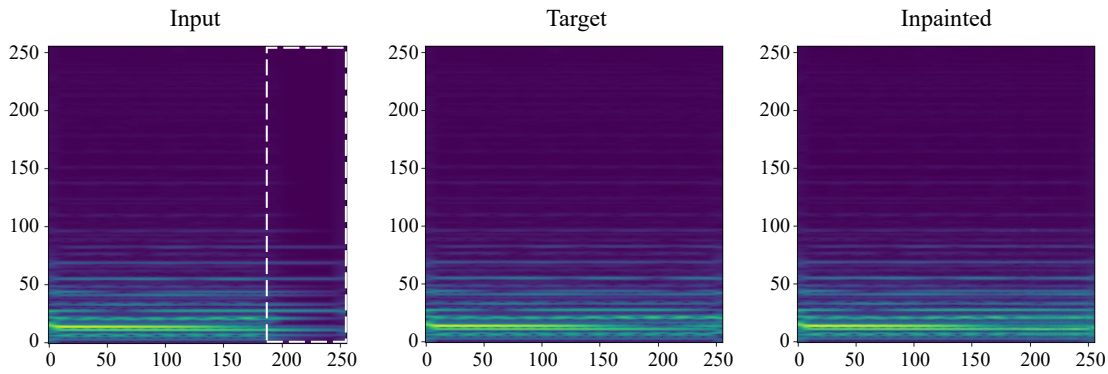

Fig. 2. Magnitude spectrograms of an example reconstruction. Left: LPC inpainted spectrogram, fed into the bin2bin. White dotted line indicates gap displacement. Center: target spectrogram. Right: bin2bin refined spectrogram. The axes of the plots indicate the frequency bin and the frame index.

GPU time when running on a Nvidia Titan X 12 GB. In the CPU configuration, operations involving numpy arrays were accelerated with numba compiler; this allowed us to achieve realtime operation for the *lite* model, which can process a single gap in less than the allowed stride time (i.e. 11.6 ms).

Finally, the learnable parameter count for the considered architectures amounts to 54.4 million, for the *full* configuration and 3.4 million for the *lite* configuration.

TABLE I
FORWARD PROCESS TIME REQUIRED FOR INPAINTING A 512-SAMPLE GAP, EQUIVALENT TO 11.6 MS AT 44.1 KHZ SAMPLE RATE.

	CPU runtime		GPU runtime	
Model	*lite*	*full*	*lite*	*full*
Bin2bin forward	8.1 ms	54.0 ms	1.8 ms	1.8 ms
LPC	0.75 ms	0.75 ms	0.75 ms	0.75 ms
cross-fade	1.0 ms	1.0 ms	1.0 ms	1.0 ms
Total	9.85 ms	55.75 ms	3.55 ms	3.55 ms

IV. CONCLUSIONS

This technical report described an approach to PLC based on LPC and a bin2bin GAN. By combining the robustness of LPC with the generative capabilities of GANs, our proposed method will hopefully increase the current benchmark for PLC in NMP, to ensure higher quality, real-time audio transmission despite network-induced packet loss.

REFERENCES

[1] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, no. 5, pp. 40–48, 1998.

[2] C. Aironi, S. Cornell, L. Serafini, and S. Squartini, "A Time-Frequency Generative Adversarial Based Method for Audio Packet Loss Concealment," in *31st European Signal Processing Conference (EUSIPCO)*, 2023, pp. 121–125.

[3] A. I. Mezza, M. Amerena, A. Bernardini, and A. Sarti, "Hybrid packet loss concealment for real-time networked music applications," *IEEE Open Journal of Signal Processing*, vol. 5, pp. 266–273, 2024.

[4] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "On the Effectiveness of Least Squares Generative Adversarial Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 12, pp. 2947–2960, dec 2019.

[5] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," *arXiv*, 2014.

[6] V. Lostanlen, C.-E. Cella, R. Bittner, and S. Essid, "Medley-solos-DB: a cross-collection dataset for musical instrument recognition," Sep. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3464194

[7] G. Bandiera, O. Romani Picas, H. Tokuda, W. Hariya, K. Oishi, and X. Serra, "Good-sounds.org: a framework to explore goodness in instrumental sounds," 08 2016.

[8] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C.-Z. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck, "Enabling factorized piano music modeling and generation with the MAESTRO dataset," in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=r1lYRjC9F7