Conditioning PARCNet with TFiLM for Robust Packet Loss Concealment

Filippo Daniotti
Department of Computer Science and Engineering
University of Trento
Trento, Italy
filippo.daniotti@unitn.it

Luca Turchet
Department of Computer Science and Engineering
University of Trento
Trento, Italy
luca.turchet@unitn.it

***Abstract*—Real-time applications, such as Networked Music Performance (NMP), typically employ best-effort protocols to minimize latency; however, this may cause buffer underruns at the receiver side. Packet Loss Concealment (PLC) techniques are employed to cope with this issue. While a few PLC algorithms have been proposed over the years, they usually assume that the past buffer always features valid packets, which is usually not the case in real life scenarios. In this technical report, we present a novel method to address the increasing divergence of the next prediction when the past buffer features packets that are prediction themselves. We iteratively train several instances of our model, and for each iteration an increasing number of packets in the buffer is dropped and concealed with a surrogate model, ensuring robust concealments even when the past buffer is lossy. Additionally, we employ use Temporal Feature-Wise Linear Modulation (TFiLM) as a conditioning strategy to leverage the positional information of concealed packets in the buffer.**

***Index Terms*—Networked Music Performance, Packet Loss Concealment, Deep Learning, Convolutional Neural Network**

I. Introduction

The widespread adoption of the Internet has driven the development of a wide array of interactive multimedia applications, including those requiring real-time interaction. While Voice over IP (VoIP) applications for video conferencing are now well-established, Networked Music Performance (NMP) applications—allowing geographically-separated musicians to perform together in real-time—have experienced limited adoption. Among the challenges hindering NMP adoption, latency is particularly demanding. As indicated by studies on psychoacoustics, end-to-end latency should not exceed 20-30 ms [1]. This constraint imposes strict requirements on application design, such as addressing packet loss or excessive delay.

To account for missing network packets, NMP applications typically implement a jitter buffer at the receiver side, but the possibility of buffer under-runs has to be considered, as they produce annoying artifacts in the audio playback. To address this problem at the receiver side, applications feature Packet Loss Concealment (PLC) algorithms. Over time, many PLC techniques have been proposed, ranging from naïve data-filler to more refined techniques that can interpolate the missing information based on past samples [2] [3] [4].

Recent advancements in Deep Learning (DL) have led to its adoption in many audio-related tasks. While many attempts have been made to adopt deep learning-based methods to develop PLC algorithms tailored for speech signals [5] [6] [7], there has been limited exploration of such algorithms for musical audio streams. Typically, PLC models use a buffer of past packets as input to obtain a prediction of the following packet. However, packet losses usually come in burst of multiple consecutive lost packets. Consequently, a DL model often relies on a buffer composed largely of previously predicted packets. This recursive use of predictions amplifies error accumulation, causing subsequent predictions to deviate increasingly from the ground truth.

In this technical report, we define our entry for the IEEE-IS² 2025 PLC Challenge presenting a method to train a DL PLC model robust to bursts of consecutive lost packets. Specifically, we introduce an iterative training approach featuring a surrogate model that simulates packet predictions during data loading. Additionally, we apply a conditioning strategy that enforces the model's awareness of the positional context of lost packets during inference.

II. Method

A. Challenge specifications

As from the specifications of the challenge, the setup features a collection of 16-bits audio tracks sampled at 44.1 kHz. For each track, a packet trace in the form of binary mask is provided, where each digit represent whether a 512 samples packet is lost. The traces were sampled from two different subsets:

- *subset 1* features bursts of at most 6 packets;
- *subset 2* features bursts of at most 16 packets.

The traces were sampled from subset 1 with high probability, and from subset 2 with low probability.

Given the above specification, any DL model running on the blind test set will likely have to perform the inference pass on consecutive packets, or a few valid packets away from the next lost packet. This is detrimental to the performances of the model, as the packets in the past buffer are typically assumed to be all valid at training time, which does not represent what happens in the real scenario and in the challenge. Additionally, the more previously concealed packets are in the buffer at a given time t, the more the prediction for packet at time $t+1$ will diverge from the ground truth. Hence, DL PLC models

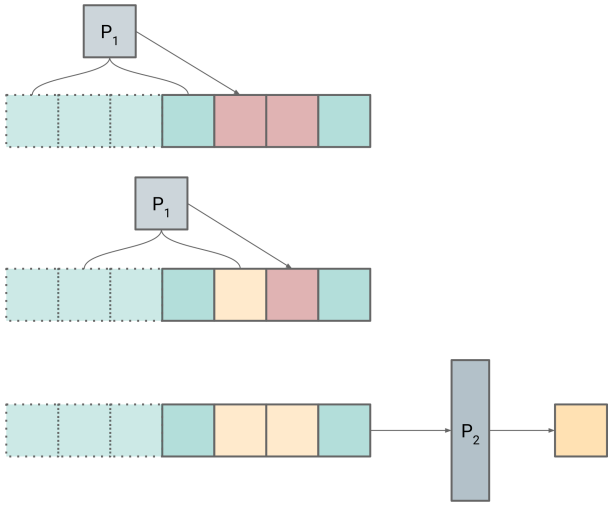

Fig. 1. Detail of the pre-processing phase of the past buffer of a datapoint during the training of model P_2. Squares with solid borders represent packets in the past buffer. It is assumed that the binary mask has already been generated and applied to the past buffer. The surrogate model P_1 slides through the past buffer and produces concealments for each of the lossy packets. In this picture, the number of packets in the past buffer k has been limited to 4 for the sake of conciseness.

should be robust to the presence of previously concealed packets in their input buffer.

In the following, we present a method to train a robust PLC model. We use PARCNet [8] as a backbone model. Specifically, we use the most recent revision of it, which is the baseline for the challenge[1].

B. Iterative training with surrogate model

As from the challenge specifications, our goal was to train a model robust to past buffers with up to N previously concealed packets.

Hence, we train N instances of the our model iteratively. Each instance P_n is trained with n concealed packets within the context buffer, and the model P_0 is the baseline PARCNet model. At data loading time, for each example in the dataset a binary mask sampled from a uniform distribution with n ones and $k - n$ zeros, where n is the number of lost packets in the mask and k is the total length of the past buffer. Then, the past buffer is zeroed according to the binary mask and the lossy buffer is concealed running the PARCNet algorithm using a *surrogate model*, that is the model trained at the previous iteration P_{n-1}. The process is illustrated in Figure 1. By training N models progressively increasing the number of previously concealed packets in the buffer, we allow each P_n model to adapt to longer bursts.

We used $k = 8$ as past buffer size as from the baseline PARCNet model. A high-level description of the training framework is presented in Algorithm 1.

[1]https://github.com/polimi-ispl/2024-music-plc-challenge/tree/main/parcnet-is2

Algorithm 1: Iterative training with surrogate model

Input: Baseline model: P_0
Input: Number of lossy packets: N
Output: Trained models: $[P_1, \ldots, P_N]$
for $n \leftarrow 1$ **to** N **do**
  // Excerpt of the regular PARCNet training procedure
  repeat
    ...
    // x_i is the past, y_i is the ground truth
    x_i, $y_i \leftarrow$ loadDatapoint();
    x_i, $y_i \leftarrow$ getLossyPacketsMask(m_i);
    $\tilde{x}_i \leftarrow$ getLossyPastFromMask(x_i, m_i);
    $\tilde{x}_i \leftarrow$ fillLossyPast($\tilde{x}_i$, P_{n-1});
    restOfTrainingLoop($\tilde{x}_i$, y_i, P_m)
  until *convergence*;
return $[P_1, \ldots, P_N]$

C. TFiLM conditioning

In order to exploit the information of which packets within the past buffer are valid and which are previously concealed, we employ Temporal Feature-Wise Linear Modulation (TFiLM) [9]. TFiLM applies an affine transformation to the feature activation maps within the hidden layers of the model. The parameters of the affine transformation γ_i and β_i are inferred by a Recurrent Neural Network (RNN) from a conditioning sequence, that is the binary mask of lossy packets. The RNN predicts B pairs of γ_i and β_i parameters, where each pair $i, 0 < i < B$ of parameters is applied to the i-th layer of the NN. The affine transformations embed the temporal information of the position of the previously concealed packets in the past buffer and ultimately force the model to rely on valid packets.

We define a TFiLM generator NN with a Gated Recurrent Unit (GRU) layer with 128 hidden units, followed by two parallel fully-connected layers for the γ_i and β_i parameters. The TFiLM generator runs before the main body of the NN of PARCNet and is trained jointly with it. Hence, the number of parameters of the model is increased to 700k with respect to the 416k of baseline PARCNet.

D. Dataset

We train our models on the Medley Solos DB dataset [10], which contains close to 18 hours of solo instrument recordings. This collection spans a diverse range of eight instruments: clarinet, distorted electric guitar, female vocals, flute, piano, tenor saxophone, trumpet, and violin. The Medley Solos DB consists of 21,571 audio clips stored as PCM wave files, each sampled at 44.1 kHz with a single (mono) channel and a bit depth of 32 bits. Every clip has a uniform length of 2,972 milliseconds, which corresponds to 65,536 discrete-time samples.

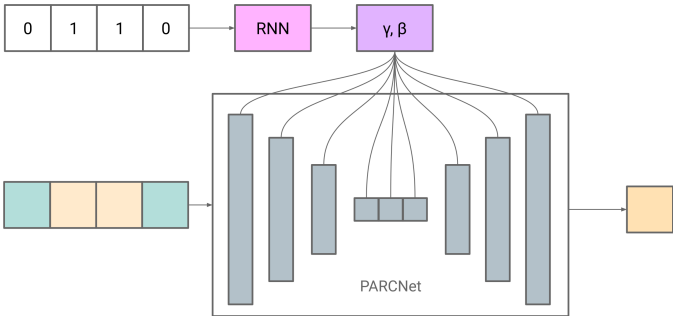

Fig. 2. High-level illustration of the PARCNet architecture enriched with TFiLM. On the bottom left, we have the past buffer - green represent valid, yellow represent concealed - fed to the main body of the PARCNet architecture. On the top left, we have the binary mask of previously concealed packets within the past buffer, which is fed to the RNN to obtain the conditioning parameters γ_i and β_i. The i-th pair of conditioning parameters is applied to the feature activation maps of the i-th block of all the the Dilated Residual Block of the Encoder and Decoder and GLU blocks of the Bottleneck.

III. Experiments

As from the iterative training procedure described, we trained $N = 2$ models, where each model P_n is an instance of PARCNet enriched with TFiLM. We trained each model until a plateau in the validation loss is reached using Early Stopping. All hyperparameters are the same as Baseline PARCNet. All the training experiments were run on a 64 bit Linux machine running Ubuntu 22.04.3 LTS. The machine was equipped with an Intel(R) Core(TM) i9-10940X 3.30GHz CPU with 2 threads per core and 14 cores, 200GB of RAM and two GPUs, namely an NVIDIA GeForce RTX 4090 and an NVIDIA GeForce RTX 3090, both featuring 24GB of dedicated VRAM. All the experiments were performed with Python 3.11.5 and CUDA 12.2. All the random generators were fed with the same seed (42).

IV. Conclusions

In this technical report we described our entry to the IEEE-IS² 2025 PLC Challenge. The method features an iterative training procedure, where the buffer of past packets is altered so that progressively more and more packets are concealed by a surrogate model. Additionally, the model is conditioned with TFiLM, in order to exploit the position of corrupted packets in the buffer to give more importance to valid packets. Adding the TFiLM module inevitably has a negative impact on the inference time, possibly undermining real-time capabilities of the algorithm. Many solutions from the domain of TinyML can be employed - e.g., network pruning, post-training weight quantization and knowledge distillation - howere, their applicability to the model is left for future studies.

References

[1] C. Rottondi, C. Chafe, C. Allocchio, and A. Sarti, "An overview on networked music performance technologies," vol. 4, pp. 8823–8843.

[2] M. Fink and U. Zölzer, "Low-delay error concealment with low computational overhead for audio over ip applications," in *International Conference on Digital Audio Effects*, 2014.

[3] G. Zhang and W. B. Kleijn, "Autoregressive model-based speech packet-loss concealment," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4797–4800, IEEE.

[4] M. Sacchetto, Y. Huang, A. Bianco, and C. Rottondi, "Using autoregressive models for real-time packet loss concealment in networked music performance applications," in *Proceedings of the 17th International Audio Mostly Conference*, AM '22, (New York, NY, USA), p. 203–210, Association for Computing Machinery, 2022.

[5] N. L. Westhausen and B. T. Meyer, "tplcnet: Real-time deep packet loss concealment in the time domain using a short temporal context," 2022.

[6] J. Wang, Y. Guan, C. Zheng, R. Peng, and X. Li, "A temporal-spectral generative adversarial network based end-to-end packet loss concealment for wideband speech transmission," *The Journal of the Acoustical Society of America*, vol. 150, pp. 2577–2588, 10 2021.

[7] J.-M. Valin, A. Mustafa, C. Montgomery, T. B. Terriberry, M. Klingbeil, P. Smaragdis, and A. Krishnaswamy, "Real-time packet loss concealment with mixed generative and predictive model," 2022.

[8] A. I. Mezza, M. Amerena, A. Bernardini, and A. Sarti, "Hybrid packet loss concealment for real-time networked music applications," vol. 5, pp. 266–273.

[9] S. Birnbaum, V. Kuleshov, S. Z. Enam, P. W. Koh, and S. Ermon, *Temporal FiLM: capturing long-range sequence dependencies with feature-wise modulation*. Red Hook, NY, USA: Curran Associates Inc., 2019.

[10] V. Lostanlen, C.-E. Cella, R. Bittner, and S. Essid, "Medley-solos-db: a cross-collection dataset for musical instrument recognition," Sept. 2019.