PLAYING ALONG IN TIME AND KEY: CAUSAL RHYTHMIC AND HARMONIC GROUNDING FOR REAL-TIME MUSIC ACCOMPANIMENT

Kevin Bretz **Derya Soydaner** **Aske Plaat**
Leiden Institute of Advanced Computer Science (LIACS), Leiden University, The Netherlands
k.o.bretz@umail.leidenuniv.nl
{d.soydaner, a.plaat}@liacs.leidenuniv.nl

ABSTRACT

Causal music accompaniment models suffer from rhythmic drift, because generated errors are fed back as input to the following steps and accumulate until the model cannot infer its phase within the musical meter. Silent Metronome shows that providing a temporal, periodic reference signal that is separate from the input audio gives a streaming accompaniment model an explicit sense of time, more than tripling the rhythmic alignment of its output with the input audio. That signal comes from precomputed labels extracted from the dataset, which leaves open whether a live system could infer it.

This demo closes the loop and extends the method into the harmonic dimension with a chord chart conditioning signal. A small model listens to the audio and predicts the metric phase, a tracking stage denoises it and follows changes in tempo and meter, and the accompaniment model receives this causally inferred signal, with a second small model supplying the chords. Every half second the generator proposes eight candidate continuations and a learned scorer commits the one that fits best. Under a real-time protocol in which every chunk is generated before its audio has been heard, the system keeps 93 percent of the beat alignment obtained with ground-truth labels, runs on a single consumer GPU and exceeds the beat alignment of the held-out ground-truth stems on Slakh2100. Upon listening, the output remains audibly worse than the ground truth, so ongoing work targets bar-to-bar consistency and phrasing. Our demo shows the system playing along live with held-out songs, on drums, guitar, piano and bass.

1. INTRODUCTION

Silent Metronome (SiMe) [1] found that passing an explicit metrical phase signal to a streaming accompaniment model through a separate conditioning channel more than triples its beat alignment under strictly causal settings, because the signal is independent of the generated audio and cannot drift. Extending the conditioning to a ground truth chord chart also improves performance. However, those results were obtained using precomputed ground truth labels as the conditioning signal, leaving open whether the signal can be causally predicted live from the input mix accurately enough for the mechanism to survive. This demo answers this question with a strictly causal, closed-loop system (Figure 1). A beat predictor and a chord predictor read the tokens of the causal audio codec, a dynamic Bayesian network (DBN) tracker converts the noisy metrical phase prediction into a smooth tempo and phase trajectory, the generator receives clock and chords exactly as it received the oracle signals, and a learned scorer picks the best of N candidate chunks. Existing audio-domain accompaniment systems [2–4] infer rhythm implicitly and symbolic jamming agents [5, 6] work on a fixed grid, so to

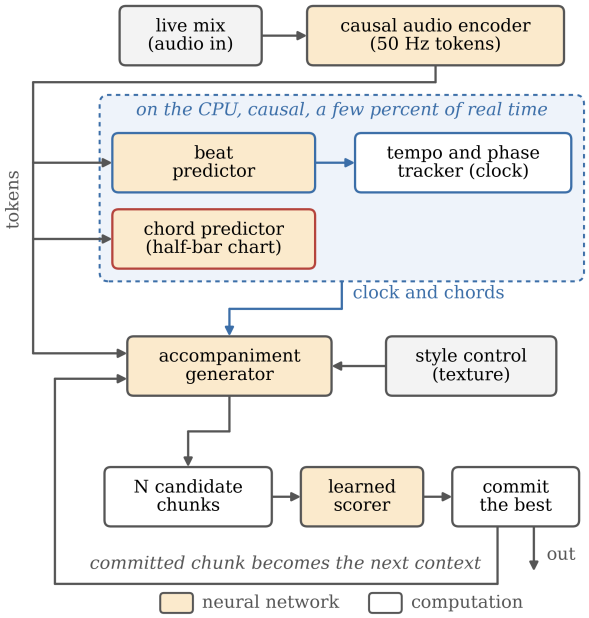


Figure 1. The live system. Shaded blocks are neural networks and white blocks are regular computations. The predictors and the clock are light enough to run on the CPU. The output chunk becomes the input of the next. Currently, the available style control is a texture reference clip.

© K. Bretz, D. Soydaner, and A. Plaat. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** K. Bretz, D. Soydaner, and A. Plaat, "Playing Along in Time and Key: Causal Rhythmic and Harmonic Grounding for Real-Time Music Accompaniment", in *Extended Abstracts for the Late-Breaking Demo Session of the 27th Int. Society for Music Information Retrieval Conf.,* Abu Dhabi, UAE, 2026.

the best of our knowledge this is the first audio-domain accompaniment system that uses an explicitly tracked causal clock.

2. SYSTEM

Generator. We keep the streaming prefix-decoder transformer of Wu *et al.* [2] exactly as in SiMe, scaled to 301 M parameters. It operates on a causal Descript Audio Codec (DAC) [7] at 50 Hz in the delay pattern of MusicGen [8] with zero look-ahead and a sliding ten-second context. Every frame receives the SiMe vector (sine and cosine of the beat and bar phase, as well as tempo and time signature), a chord label per half bar, and a texture reference, the CLAP embedding of a short clip of the desired sound, all injected through per-frame adaptive layer normalization [9]. The model is trained on Slakh2100 [10] with ground truth labels in one-second chunks and generates half-second chunks at inference. After 120k steps it is trained for another 80k steps on its own best-of-eight outputs, so that a single sample already sounds more like a selected one.

Causal predictors. Both predictors are causal transformers over the DAC codebook embeddings of the mix. The beat predictor outputs per-frame estimates of the current phase, tempo and meter, and the chord predictor outputs one of 24 major and minor triads or no-chord, plus a key, per half bar, at a frame accuracy of 0.74. At the start of a session the predictors listen for two seconds before the generator enters, which mirrors the count-in of a human band.

The clock. Regression heads alone produce a noisy signal with considerable jitter, which recreates the drift dilemma. We decode it with a DBN [11] over meter, tempo and phase, the tracker of Figure 1, which has no learned parameters and uses the observation model of madmom [12]. Forward filtering is causal, yields per-frame posteriors whose circular means are exactly the SiMe channels, runs at under one percent of real time on the CPU, and turns corrections into short glides instead of jumps. On the cold-start protocol of the SiMe paper it lifts the generator from 0.28 to 0.36 Beat-F, about half of the gap to the oracle labels at 0.43.

Streaming best-of-N selection. At each chunk boundary the generator samples $N{=}8$ candidate continuations in a single batched forward pass. A small convolutional scorer reads the codec tokens of each candidate, the mix tokens heard so far and the fed clock and chart, and predicts the beat fit, coherence and realism that the metrics below measure, with a value head for the chunks that follow. It is trained on candidates harvested from the validation split and never sees the audio under the chunk it scores. The best candidate is committed, so the output remains one causal stream.

Real-time protocol. A live system has to start each chunk early enough to finish before it must sound. We therefore generate every chunk half a second before it plays, with clock and chart extrapolated from the audio heard so far and the scorer restricted to it. The whole loop, codec, predictors, tracker, eight candidates, scoring and decoding, takes 340 ms per half-second chunk on an RTX 2080 Ti, and a desktop application plays along with songs played back into it.

System	Beat-F ↑	CoCoLa ↑	FAD ↓
Oracle clock and chords	0.521	63.57	3.13
+ best-of-8 selection	**0.781**	**64.79**	**2.76**
Predicted clock and chords	0.411	62.33	3.30
+ best-of-8 selection	0.726	63.57	2.86
Ground-truth stem	0.552	66.27	–

Table 1. Slakh2100 test split, 1,024 windows, one seed, half-second chunks. Every chunk is generated half a second before it plays, so it never hears the audio it accompanies. Oracle rows use annotated labels, predicted rows infer them from the mix.

3. RESULTS

Table 1 follows the evaluation protocol of Wu *et al.* [2] on 1,024 ten-second test windows of Slakh2100 with one stem held out as the target. Beat-F [12] compares beats found by Beat This! [13] on the accompaniment with those of the input mix and scores a window without detectable beats as zero, CoCoLa [14] measures both harmonic and percussive coherence with the input, and FAD [15] measures distributional audio quality. Retraining the same generator without the clock removes most of its beat alignment, and without the chart it loses harmonic coherence while its beat alignment stays the same, so the clock carries the rhythm and the chart the harmony. Selection adds 0.32 Beat-F and improves FAD and CoCoLa at the same time. With selection, the predicted labels keep 93 percent of the oracle beat alignment and exceed that of the held-out ground-truth stem. These numbers must be read next to the audio. By ear the accompaniment follows the beat and the chords but still sounds clearly worse than the real stem, with drum patterns that change from bar to bar and harmonic parts that follow the chart without phrasing, which none of the metrics capture.

Demonstration. The video shows the live system playing along with held-out songs in real time, first in a drums session of almost two minutes that introduces the clock, the chord chart and the candidate selection one by one before the drums come in, then on guitar, piano and bass. SiMe [1] has a project page with code and audio examples.

4. ONGOING WORK

We plan to investigate why the output often plays at regular intervals locked to the predicted metric phase, which is most noticeable for piano. We also want to give the harmonic dimension more information by predicting more than chords, and to improve the texture reference.

5. AI USAGE STATEMENT

Large language models (Anthropic Claude, used through Claude Code) assisted with writing and debugging the experiment code, orchestrating evaluation runs on the cluster, producing the figures, the demonstration video and the poster, and editing the text of this abstract. All design decisions, training choices, listening evaluations and the final text were made and verified by the authors.

6. REFERENCES

[1] K. Bretz, D. Soydaner, and A. Plaat, "Silent Metronome: Rhythmic Grounding for Live Music Accompaniment", arXiv preprint arXiv:2609.07688, 2026.

[2] Y. Wu, M. Wang, H. Lei, S. Brade, L. Blanchard, S.-L. Wu, A. Courville, and A. Huang, "Streaming generation for music accompaniment," arXiv preprint arXiv:2510.22105, 2025.

[3] M. Pasini, J. Nistal, B. Hayes, M. R. Bjare, S. Lattner, and G. Fazekas, "LiveBand: live accompaniment generation in the audio domain," arXiv preprint arXiv:2606.03803, 2026.

[4] T. Karchkhadze and S. Dubnov, "Towards real-time human-AI musical co-performance: accompaniment generation with latent diffusion models and MAX/MSP," arXiv preprint arXiv:2604.07612, 2026.

[5] Y. Wu et al., "Adaptive accompaniment with ReaLchords," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2024.

[6] A. Scarlatos et al., "ReaLJam: real-time human-AI music jamming with reinforcement learning-tuned transformers," in *Proc. ACM CHI Conf. Human Factors in Computing Systems, Extended Abstracts*, 2025.

[7] R. Kumar, P. Seetharaman, A. Luebs, I. Kumar, and K. Kumar, "High-fidelity audio compression with improved RVQGAN," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 36, 2023.

[8] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 36, 2023.

[9] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 4172–4182.

[10] E. Manilow, G. Wichern, P. Seetharaman, and J. Le Roux, "Cutting music source separation some Slakh: a dataset to study the impact of training data quality and quantity," in *Proc. IEEE Workshop Appl. Signal Process. Audio Acoust. (WASPAA)*, 2019, pp. 45–49.

[11] F. Krebs, S. Böck, and G. Widmer, "An efficient state-space model for joint tempo and meter tracking," in *Proc. Int. Soc. Music Information Retrieval Conf. (ISMIR)*, 2015.

[12] S. Böck, F. Korzeniowski, J. Schlüter, F. Krebs, and G. Widmer, "madmom: a new Python audio and music signal processing library," in *Proc. ACM Int. Conf. Multimedia*, 2016, pp. 1174–1178.

[13] F. Foscarin, J. Schlüter, and G. Widmer, "Beat This! Accurate beat tracking without DBN postprocessing," in *Proc. Int. Soc. Music Information Retrieval Conf. (ISMIR)*, 2024.

[14] R. Ciranni et al., "COCOLA: coherence-oriented contrastive learning of musical audio representations," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2025.

[15] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet audio distance: a reference-free metric for evaluating music enhancement algorithms," in *Proc. Interspeech*, 2019, pp. 2350–2354.