Live Music Diffusion Models: Efficient Fine-Tuning and Post-Training of Interactive Diffusion Music Generators

Zachary Novack[*] UC San Diego
Stephen Brade[*] MIT
Haven Kim UC San Diego
Hugo Flores García Adobe
Nithya Shikarpur MIT
Chinmay Talegaonkar UC San Diego
Julian McAuley UC San Diego
Taylor Berg-Kirkpatrick UC San Diego
Cheng-Zhi Anna Huang MIT

Abstract

Interactive streaming music generation promises the use of generative models for live performance and co-creation that is impossible with offline models. However, SOTA models exist in the discrete-AR regime, requiring industrial levels of compute for both training and inference. In this work, we investigate whether audio diffusion models, with their wide support in the open-source community but non-streaming bidirectional nature, can be repurposed efficiently into interactive models accessible on consumer hardware. By taking a critical look at the modern pipeline for block-wise outpainting diffusion, we identify critical inefficiencies during inference that result in strictly worse computational efficiency than their discrete-AR counterparts. We propose **Live Music Diffusion Models** (LMDMs), a simple modification of the generative diffusion process that recovers, and then outperforms, the inference complexity of the discrete Live Music Models (LMMs) through block-wise KV Caching. Unlike LMMs, LMDMs further enable stable post-training alignment through our novel ARC-Forcing paradigm, reducing error accumulation without any explicit RL or reward models. We demonstrate the application of LMDMs in a number of creative domains, including text-conditioned generation, sketch-based music synthesis, and jamming. We finally show how LMDMs can be used as a generative instrument in a real artist-AI collaboration, utilizing LMDMs as a "generative delay" to transform musicians' improvisation live for variable timbral effects while running locally on a consumer gaming laptop.

1 Introduction

Generative music models have rapidly advanced, promising full song generation with high realism and control over musical attributes (Agostinelli et al., 2023; Copet et al., 2023; Evans et al., 2024b; Novack et al., 2025c; Yuan et al., 2025). In parallel, there has been growing interest in *live interaction*: treating models as *instruments* or *co-musicians* to be played with in real time, merging the study of musical agents Assayag et al. (2006); Tatar & Pasquier (2019); Lee & Pasquier (2025); Lewis (2000a) with state-of-the-art deep music generation. Particulary, the recent Live Music Models (LMMs) (Team et al., 2025) showing unprecedented quality while generating comprehensive musical content with live textual controls at a fixed delay. However, LMMs and other strong systems built on discrete

[*]equal contribution, correspondence to `znovack@ucsd.edu, brade@mit.edu`

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

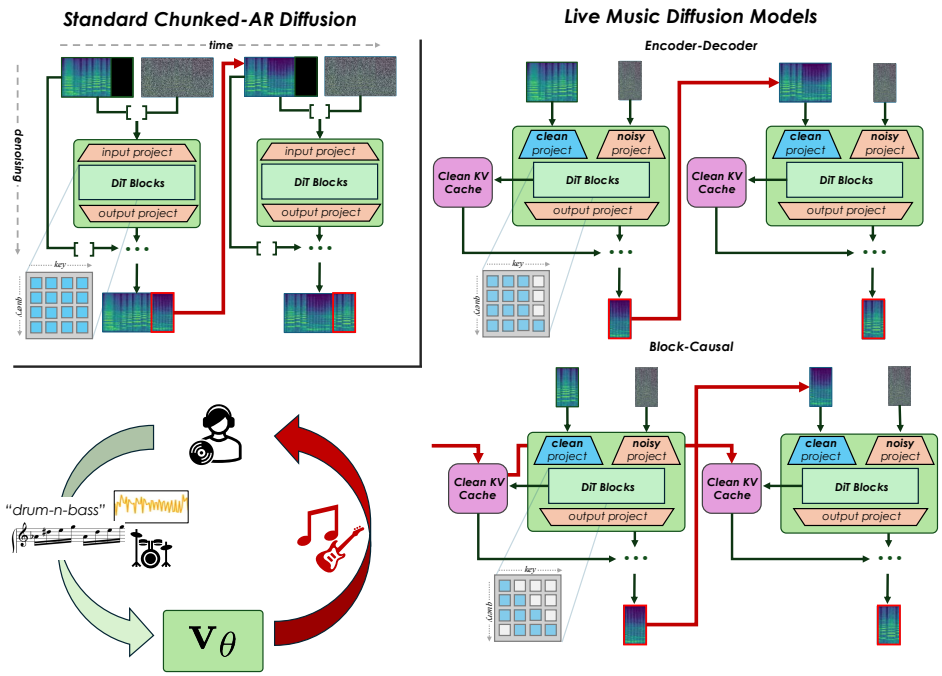


Figure 1: **Live Music Diffusion Models**. Standard block-AR diffusion (top left) concatenates clean context with noisy states over all frames with full bidirectional attention, with no way to cache the context encoding. LMDMs (right side) route the clean context and noisy target frames through separate projections and utilize custom attention masks to ensure clean context encoding is not impacted by the target frames, enabling KV-Caching over diffusion steps (in Enc-Dec, top-right) and over time (in Block-Causal, bottom right). LMDMs enable live interactive musical co-creation on consumer-grade hardware, modulating an array of possible controls in real time (bottom left).

autoregression (AR) have billions of parameters: LMMs alone require over 40 GB of VRAM, making local inference on consumer hardware impractical.

In contrast, diffusion models (Ho et al., 2020) offer a potential solution. Diffusion-based approaches enjoy better data-efficiency than discrete-AR methods (Prabhudesai et al., 2025), and a wealth of open-source music models exist that are performant yet much smaller than strong LMs (Novack et al., 2025a; Evans et al., 2024b; Chen et al., 2024b; Liu et al., 2024) and have seen adoption by musicians (RoyalCities, 2026; Carr & Zukowski, 2024; Fitzgerald et al., 2025). Critically, diffusion has shown capacity for fine-grained control, from gestural sketch conditioning (Flores García et al., 2025) to pitch, dynamics, and melodic controls (Tsai et al., 2025; Novack et al., 2024), that have no clean analogue in discrete-AR systems. However, such approaches are inherently not streamable given diffusion's full bidirectional attention across time, and limited attempts to bridge this gap (Novack et al., 2025b) cannot leverage the inference efficiency of discrete-AR methods (e.g. KV-Caching).

In this work, we repurpose open-source audio diffusion models into interactive streaming models on consumer hardware as **Live Music Diffusion Models**[2] (LMDMs). By analyzing block-diffusion *outpainting*, we find that a simple routing mechanism between clean history and noisy present blocks, combined with dedicated attention masking, enables noise-wise KV Caching and recovers the *exact* inference complexity of encoder-decoder LMMs. A further block-causal variant achieves strictly faster complexity with full temporal KV caching. This is done *purely through standard finetuning*, bypassing from-scratch training and completing in under 8 GPU hours. Second, as LMDM inference is *fully differentiable* (unlike discrete-AR sampling), we combine the ARC framework (Novack et al., 2025a) with Self-Forcing (Huang et al., 2025) into our novel *ARC-Forcing* recipe, providing global adversarial supervision on multi-block rollouts to reduce error accumulation and accelerate sampling without any RL or pretrained reward models. Third, we explore the full controllability of offline diffusion across text-conditioned generation with on-the-fly prompt transitions (Team et al., 2025), localized sketch controls (Flores García et al., 2025), and interactive accompaniment (Wu et al.,

[2]Audio examples are available at `https://stephenbrade.github.io/lmdm-public/`.

2025c). Finally, we demonstrate that streamability, controllability, and long-horizon stability together make LMDMs viable as generative *instruments*. We build a real-time system via ONNX export and C++/JUCE, deploying sketch-conditioned LMDMs as a *generative delay* on a consumer gaming laptop. We put this system in front of talented musicians from an institutional fellowship program, and are actively using LMDMs in a live musical performance. In summary, our contributions are:

1. We introduce **Live Music Diffusion Models**, a simple modification to diffusion models that enables KV-Caching over diffusion steps and time through standard finetuning.
2. We propose **ARC-Forcing**, an RL-free adversarial post-training recipe providing global supervision on multi-block rollouts without reward models.
3. We bring the **full controllability** of offline diffusion, including text, sketch, and accompaniment controls, into the near real-time streaming regime.
4. We **deploy LMDMs as a generative instrument** with real musicians in collaborative sessions and live performances on consumer hardware.

2 Background

2.1 Flow Matching

In this work, we primarily focus on the Flow Matching with Optimal Transport path (Esser et al., 2024; Liu et al., 2022) (also commonly referred as Rectified Flow) generative modeling paradigm, given its success in audio generative models (Novack et al., 2025a; Tal et al., 2025; Lan et al., 2024) and its general equivalence with diffusion-based approaches (Gao et al., 2024). Given a stereo audio sequence $\mathbf{a} \in \mathbb{R}^{Lf_s \times 2}$, we first compress it into a compact, C-channel VAE latent representation $\mathbf{x} \in \mathbb{R}^{T \times C}$, where each $\mathbf{x}_t$ denotes the tth latent time frame of $\mathbf{x}$[3]. In flow matching, we define a forward corruption process that interpolates our sample with some amount of gaussian noise up to a noise level k: $\mathbf{x}^{(k)} = (1-k) \cdot \mathbf{x} + k \cdot \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I})$, which we can write as shorthand as sampling $\mathbf{x}^{(k)} \sim q(\mathbf{x}^{(k)} \mid \mathbf{x}, k)$. The goal of flow matching is to learn the reverse of this process, transferring pure gaussian noise ($k = 1$) into our data distribution ($k = 0$). We can view the forward process as an ordinary differential equaiton of the form: $\mathrm{d}\mathbf{x}^{(k)}/\mathrm{d}k = \boldsymbol{\varepsilon} - \mathbf{x} := \mathbf{v}$. Thus, if we can learn a proper noise-conditioned velocity network $\mathbf{v}_\theta$ to approximate this velocity, we can solve the ODE in reverse using any normal solver (e.g. Euler, Heun, RK4). We can learn this velocity model by drawing samples from the forward corruption process and regressing our model against the marginal velocity at that point: $\mathbb{E}_{\mathbf{x}\sim\mathcal{X}, k\sim p(k), \mathbf{x}^{(k)}\sim q(\mathbf{x}^{(k)}|\mathbf{x},k)} \left[\|\mathbf{v}_\theta(\mathbf{x}^{(k)}, k) - \mathbf{v}\|_2^2\right]$. As the generative process for flow matching denotes a iterative procedure over *noise levels* (from high to low) rather than any temporal axis, most flow models utilize full *bidirectional* attention across the temporal dimension, generating the entire $\mathbf{x}^{(0)}$ sequence together. We augment $\mathbf{v}_\theta$ with extra conditions $\mathbf{c}$ such as text prompts, and can sample with classifier-free guidance (CFG) as $\mathbf{v}^w_\theta(\mathbf{x}^{(k)}, k, \mathbf{c}) = \mathbf{v}_\theta(\mathbf{x}^{(k)}, k, \varnothing) - w(\mathbf{v}^w_\theta(\mathbf{x}^{(k)}, k, \mathbf{c}) - \mathbf{v}^w_\theta(\mathbf{x}^{(k)}, k, \varnothing))$ for some weight $w > 1$.

2.2 Block-Autoregressive Outpainting

In this work, we broadly consider the setting of Team et al. (2025), that is, *block*-based autoregressive generation. Given some past s frames of (latent) audio context, the goal is to learn a generative model over the next o frames: $p_\theta(\mathbf{x}_{s:s+o} \mid \mathbf{x}_{1:s}, \mathbf{c})$. After generating the target o-length "block", the model slides its context o frames in the future (forgetting the furthest history block while encoding the newly generated block as context) and continues generation. LMMs parameterize p_θ as a T5 (Raffel et al., 2023)-like encoder-decoder network: the encoder fuses the past history and global conditions into a single embedding that the decoder then conditions on (through cross-attention) to generate the next block, where a temporal decoder decodes autoregressively over time on the first codebook and a depth decoder decodes autoregressively over codebook levels in tandem.

Some past work has considered diffusion-based block-wise outpainting (Karchkhadze & Dubnov, 2026; Novack et al., 2025b). In such setups, the flow models conditioning $\mathbf{c}$ is augmented with some amount of preceding audio context $\mathbf{x}^{\text{clean}} := [\mathbf{x}^{(0)}_{1:s}, \mathbf{0}_{s:T}]$ (i.e. the remaining $o = T - s$ frames are set to 0). In the case of Diffusion Tranformer (DiT)-based models, this is normally applied through

[3]We use t to denote time, rather than noise level to be in line with existing streaming music literature (Wu et al., 2025c).

channel concatenation. Fine-tuning thus proceeds nearly identically to normal flow matching: one samples $\mathbf{x}^{(k)} \sim q(\mathbf{x}^{(k)} \mid \mathbf{x}, k)$, appends $\mathbf{x}^{\text{clean}}$ to $\mathbf{x}^{(k)}$, and predicts the velocity. Once trained, inference is modified such that the generated iterate $\mathbf{x}^{(k)}$ always aligns with the ground truth $\mathbf{x}^{(0)}_{1:s}$ over the first s frames, resetting these such frames to $\mathbf{x}^{(k)}_{1:s} \sim q(\mathbf{x}^{(k)}_{1:s} \mid \mathbf{x}^{(0)}_{1:s}, k)$ at each step. The context window then slides (as in LMMs) over one block and inference continues.

3 Live Music Diffusion Models

In this section, we show how to transform standard offline diffusion models into **Live Music Diffusion Models** (LMDMs). First, in Sec. 3.1, we show that a simple routing mechanism, combined with a matching attention mask, can enable both noise-wise and block-wise KV-caching. Then, in Sec. 3.2, we show how our pipeline enables the use of RL-Free rollout adversarial post-training.

3.1 Routing Clean Context for Efficient KV Caching

The key difference separating current block-wise streaming diffusion models from their LMM equivalents is the recurrent diffusion denoising process over the full latent context. First consider the inference efficiency for the encoder-decoder LMMs. If we let $\mathcal{E}^{\text{LMM}}_{1:s}, \mathcal{D}^{\text{LMM}}_{t}$ denote the overall latency of a *single* forward pass for the encoding of s frames of context and decoding of a single tth frame of output respectively, then the overall latency is $O(\mathcal{E}^{\text{LMM}}_{1:s} + \sum_{t=s}^{s+o} \mathcal{D}^{\text{LMM}}_{t})$, where the primary bottleneck stems from the o iterative decoder calls. In contrast, for normal block-based diffusion, the latency is $O((\mathcal{E}^{\text{Diff}} + \mathcal{D}^{\text{Diff}})_{1:T} \cdot K)$: block-based AR diffusion introduces a global dependency on the number of diffusion steps K, which even when reduced (Novack et al., 2025b), does not alleviate the fact that the process of "encoding/decoding" is fused into a single operation that is run every diffusion step, rather than the efficient single-pass context encoding that LMMs use.

We can see that the issue is that standard diffusion is trained with full *bidirectional* context over *noisy* states, even when the model is also provided clean context through channel concatenation. Formally, for noisy states $\mathbf{x}^{(k)}$ and clean context $\mathbf{x}^{\text{clean}}$, we can write the *initial* hidden state of the DiT (i.e. before any attention blocks) under channel concatenation as:

$$\mathbf{h}^{\text{init},k} = \mathbf{W}^{\text{init}}[\mathbf{x}^{(k)}, \mathbf{x}^{\text{clean}}]^\top = \mathbf{A}\mathbf{x}^{(k)} + \mathbf{B}\mathbf{x}^{\text{clean}}, \tag{1}$$

where $\mathbf{W}^{\text{init}} \in \mathbb{R}^{H \times 2C} = [\mathbf{A}, \mathbf{B}]$ is the input projection of the model. However, we know from Sec. 2.2 that $\mathbf{x}^{\text{clean}}$ is all 0s after the first s context frames, thus giving us that:

$$\mathbf{h}^{\text{init},k}_{s:T} = \mathbf{A}\mathbf{x}^{(k)}_{s:T} \qquad \mathbf{h}^{\text{init},k}_{1:s} = \mathbf{A}\mathbf{x}^{(k)}_{1:s} + \mathbf{B}\mathbf{x}^{(0)}_{1:s} \tag{2}$$

This exposes the two key problems: the part of our initial hidden state that "encodes" past context is mixed with variable noise levels, and is allowed to attend *forward* in time to the target output frames, making our encoded context change at every sampling step. To alleviate this fact, we propose a simple solution: First, as we already know a priori which frames are context vs. target, we can implement a simple routing mask $\mathbf{r} := [\mathbf{0}_{1:s}, \mathbf{1}_{s:T}]$ (i.e. a mask separating context from generation), which will be combined with our noisy latent before projecting into the model. This then gives us:

$$\mathbf{h}^{\text{init},k} = \mathbf{W}^{\text{init}}[\mathbf{r} \odot \mathbf{x}^{(k)}, \mathbf{x}^{\text{clean}}]^\top \tag{3}$$

$$\implies \mathbf{h}^{\text{init},k}_{s:T} = \mathbf{A}\mathbf{x}^{(k)}_{s:T} \qquad \mathbf{h}^{\text{init},k}_{1:s} = \mathbf{B}\mathbf{x}^{(0)}_{1:s}, \tag{4}$$

and thus $\mathbf{h}^{\text{init},k}_{1:s}$ is the same for all possible noise levels k and is only a function of the context.

This however does still not fix the fact that our "encoding" representations attend to the future, thus changing our encoding through each DiT block. To fix this and achieve our target computational complexities, we propose two attention mask variants for LMDMs:

Encoder-Decoder LMDMs: Here, we restrict every attention operation inside the DiT such that the first s frames can only attend to each other and *not* the last o frames (i.e. where we aim to generate the next block of audio lantents), while the output last o frames can attend to themselves and all proceeding frames. This asymmetric attention pattern fully decouples the encoding of past context from the decoding of the next block. Because of this, we can now use KV-Caching (Pope et al., 2023) over successive diffusion sampling steps: given a block of clean context $\mathbf{x}^{(0)}_{1:s}$, we can first

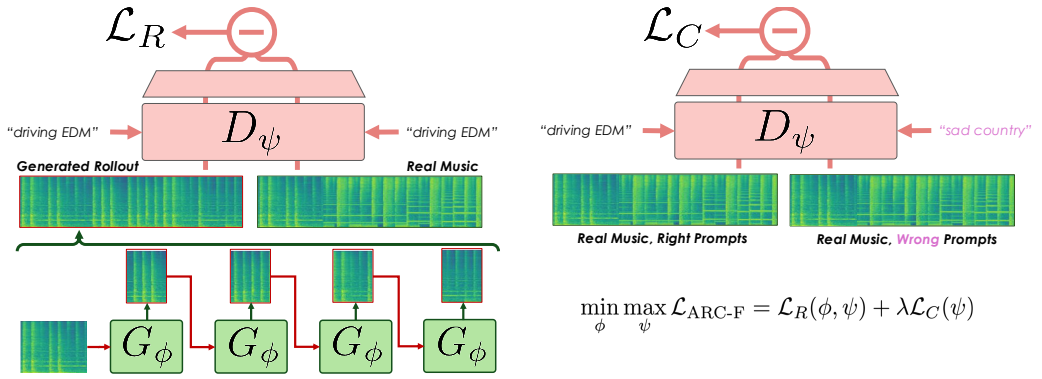


Figure 2: ARC-Forcing. G_ϕ is post-trained by generating AR rollouts across time with KV Caching, and then passing in the rollouts along with real music (with the same starting context and conditions) into the bidirectional $D\psi$, which uses a relativistic objective. D_ψ is also trained with an auxillary contrastive loss on real music with matching vs. mismatched captions to encourage text following.

pass this through our DiT and cache Key/Value states for each transformer block, and then perform every step of diffusion denoising for the target block using these states without recomputation. This yields an inference complexity of $O(\mathcal{E}_{1:s}^{\text{LMDM}} + \mathcal{D}_{s:T}^{\text{LMDM}} \cdot K)$, achieving the same complexity class as LMMs, with a *single* encoding pass over clean context and an iterative decoding process for the next block. We term this as "Encoder-Decoder" LMDMs as it follows much of the same process of classical Encoder-Decoder LLMs (and LMMs), where the explicit encoded representation produced by a separate encoder module is replaced by an implicit encoding through the KV-Cache.

Block-Causal LMDMs: While Enc-Dec LMDMs enable KV-Caching as a function of *diffusion step*, there is no temporal KV-Caching possible, as with a fixed s-frame window the context changes every time we finish generating a new block and add it to the context. To enable KV-Caching over noise level and time, we can modify the attention mask further: by introducing a block-causal depedency over o-sized blocks even within the first s frames, this enforces that frames of context can only attend to past context (or within their block). Thus, after generating the newest block, only this block must be cached before proceeding with the next generated block. This exposes the inference complexity (after a warmup period) of $O(\mathcal{E}_{s-o:s}^{\text{LMDM}} + \mathcal{D}_{s:T}^{\text{LMDM}} \cdot K)$, achieving a strictly better complexity that LMMs by reducing the amount of context needing the be encoded at every new block.

We display the core architectural differences and attention masks in Fig. 1. Note that because the modifications only change the initial projection and attention patterns, training can proceed with the standard flow matching loss. Additionally, since embedding past context only requires learning the new parameter $\mathbf{B}$ and injecting it into the initial hidden state, turning normal diffusion models into LMDMs can be easily done with practically no architectural modifications.

3.2 Rollout Post-training through ARC-Forcing

Though the above routing mechanism and attention pattern enable block-wise AR diffusion with efficient KV-Caching, they do not fix one key issue: error accumulation, which is a known issue in discrete-AR (Wu et al., 2025c) and continuous AR models (Pasini et al., 2024; Saito et al., 2025; Huang et al., 2025). As training only supervises the model on a single block level, it does not match the inference scenario where errors may compound. This thus motivates the need for an explicit *post-training* phase, where one can potentially supervise the model on its own rollouts. However, such rollout post-training phases in AR music generation have both required brittle RL-based pipelines (Wu et al., 2025b,a) and explicitly trained reward models (Wu et al., 2025c) to do so.

In this work, we show that we can avoid both such issues. First, as diffusion sampling is itself a fully differentiable operation (in contrast to discrete-AR regimes, where inference involves sampling from the output codebook probabilities), we can avoid any need for explicit RL to post-train our model and instead rely on the recent *Self-Forcing* paradigm (Huang et al., 2025) from video generation. Then, to avoid the need of an explicit reward model, we can instead purely supervise the post-training process as a fully adversarial one, drawing from the recent Adversarial Relativistic Contrastive (ARC) method (Novack et al., 2025a) and functionally learn our reward model as we post-train our generative model.

We denote the adaptation of these two approaches to streaming music generation as **ARC-Forcing**. Formally, we aim to post-train our flow model $\mathbf{v}_\theta$ into a few-step generator G_ϕ, thus increasing inference speed, but *also* to improve full AR rollouts with global supervision. To do so, we follow Huang et al. (2025) by generating B-block rollouts from G_ϕ, using KV-Caching (either noise-wise or both noise and time-wise) to maintain efficient training. To further accelerate post-training, we use a stochastically chosen $k \sim U[2, K_{\text{max}}]$ steps for each generated block, only propagating gradients on the final step. For each batch item, G_ϕ receives either s frames of existing context (mirroring scenarios during long-form rollouts), a null context with probability p_{uncond} (mimicking the start of generation), or a partially null context where only the first n frames are null with p_{partial}.

Instead of using an explicit reward model (which is not readily available) or using the original model itself as a "teacher" (which requires the teacher itself be of high enough quality), we instead use a noise-aware discriminator D_ψ with *full bidirectional* context as our source of global supervision. D_ψ is initialized from our base diffusion model, and thus can take in text conditions. Given our generated rollout $\widehat{\mathbf{x}}$ and ground truth music (with matching controls and starting context) $\mathbf{x}$, we add noise to both at the same level and pass both into the discriminator. We then use the *relativistic* loss $\mathcal{L}_R$:

$$\mathbb{E}_{\mathbf{x},\mathbf{c}\sim\mathcal{X},\widehat{\mathbf{x}}\sim G_\phi(\mathbf{x}_{1:s},\mathbf{c}),k\sim p(k)}\left[f\left(D_\psi(q(\widehat{\mathbf{x}}^{(k)} \mid \widehat{\mathbf{x}},k),\mathbf{c},k) - D_\psi(q(\mathbf{x}^{(k)} \mid \mathbf{x},k),\mathbf{c},k)\right)\right], \quad (5)$$

where f is the softplus function. By supervising the model on long music-rollout pairs, we not only avoid degenerate solutions common in other GAN objectives (Huang et al., 2024) but also provide learning signal on the full rollout. In order to ensure that D_ψ does not overfit to high-frequency features and can encourage good text following, we train D_ψ with an auxilary contrastive objective $\mathcal{L}_C$, using a relativistic objective on real music with the correct vs. incorrect *prompts*:

$$\mathbb{E}_{\mathbf{x},\mathbf{c}\sim\mathcal{X},k\sim p(k)}\left[f\left(D_\psi(q(\mathbf{x}^{(k)} \mid \mathbf{x},k),\mathcal{P}(\mathbf{c}),k) - D_\psi(q(\mathbf{x}^{(k)} \mid \mathbf{x},k),\mathbf{c},k)\right)\right], \quad (6)$$

where $\mathcal{P}$ is a random batch-wise permutation matching up each music sample with a random prompt. The discriminator is then trained with a combination of $\mathcal{L}_C$ and $\mathcal{L}_R$, with a weight λ determining the strength of the contrastive term (in our work, set to 1). Unlike in previous works (Novack et al., 2025a,b) that used ARC in the offline setting, we found one critical change needed for adapting it to rollout-based long-form post-training: as the context window for D_ψ is decoupled and is considerably larger than the context window for G_ϕ (i.e. $\approx$ 30s for D_ψ vs. $\approx$ 10 for G_ϕ), initializing D_ψ directly from the LMDM $\mathbf{v}_\theta$ resulted in a weak discriminator that quickly destabilized post-training. To remedy this, we found that warmstarting the backbone diffusion model of the discriminator D_ψ on longer audio segments for a few thousand iterations stabilized long rollout post-training.

4 The Live Music Design Space

Despite the growing interest in live interactive music models controlled with text (Team et al., 2025), sketch based controls for pitch and other features (Novack et al., 2025b), or audio models that can jam (Wu et al., 2025c), each of these works require bespoke architectures. Inspired by (Kim et al., 2026), in this section we unify these broad tasks and show that the all can be realized under the flexible design framework of LMDMs. Notably, as we wish to model $p_\theta(\mathbf{x}_{s:s+o} \mid \mathbf{x}_{1:s}, \mathbf{c})$, there are no restrictions on the form that $\mathbf{c}$ can take.

We can delineate conditioning signals for these models across two axes: (1) whether controls are global ($\mathbf{c} \in \mathbb{R}$) or local ($\mathbf{c} \in \mathbb{R}^T$), and (2) whether they are *instrument*-like or *accompaniment*-like. In the former case, controls describe *features* of the target output and interaction is determined purely by how quickly the model can synthesize the current block of controls. In the latter case, controls describe *conditions* that provide context to the model but arrive at a strict external time schedule, and thus interaction must balance reactivity to the controls with practical latency. We thus define many previous interactive paradigms within this framework:

Global Text-Conditioning. In the case which is most comparable to Team et al. (2025), $\mathbf{c}$ is simply a global text prompt with no temporal access. Though inherently non-temporal, we classify text prompts as instrument-like due to the use of *prompt transitions*, where such global conditions are modulated between different prompts in real time.

Instrument-like Sketch Controls. In the case most similar to (Novack et al., 2025b), $\mathbf{c} \in \mathbb{R}^T$ are instrument-like local conditions which gives the model detailed local information about what must be

generated in future. Specifically, we explore top-k CQT conditions (similar to Tsai et al. (2025)) and a loudness condition which mirrors Flores García et al. (2025) and Novack et al. (2025b).

Accompanying Stem Generation. This case explores when the condition is a separate accompanying audio stream, akin to the live jamming setting studied in Wu et al. (2025c). Unlike the sketch and text-conditioned settings above, the accompaniment task is purely *accompaniment-like*: the model has no explicit description of what the future should sound like, and must instead infer a musically compatible continuation from the causal history of the single input stem and its own prior outputs, while also dealing with a fixed future visibility $t_f < 0$ (i.e. the model only receives signal from the other stem up to some cutoff frame before the target block) to compensate for system latency.

5 Experiments and Results

Method	D-NFE	Blocks	Sampler	TTFF	w/Priming?	FD	KD	CLAP
Magenta RealTime	800†	24	-	≈ 4	✗	72.14	**0.47**	0.35
Stable Audio Open	100	1	DPM++	10.35	✗	96.51	0.55	**0.41**
MusicGen-Large	2.4K	1	-	10.81	✗	190.47	0.52	0.31
LMDM (ED)	50	21	Euler	0.132	✗	61.06	1.14	0.20
LMDM (ED)+AF	8	21	Ping-Pong	0.037	✗	**35.88**	0.74	0.29
LMDM (BC)	50	21	Euler	0.128	✗	64.87	1.20	0.20
LMDM (BC)+AF	2	21	Ping-Pong	0.019	✗	47.26	0.91	0.23
LMDM (ED)	50	21	Euler	0.132	✓	35.35	0.62	0.23
LMDM (ED)+AF	8	21	Ping-Pong	0.037	✓	**29.00**	**0.35**	**0.32**
LMDM (BC)	50	21	Euler	0.128	✓	47.13	0.74	0.24
LMDM (BC)+AF	2	21	Ping-Pong	0.019	✓	35.45	0.53	0.23

Table 1: Global Results on Text-Conditioned LMDMs. †Magenta-RTs NFE's cane be broken down to 50 calls for the temporal transformer and $15 \cdot 50$ calls of the lightweight depth transformer.

In this section, we evaluate our LMDMs on an array of creative musical tasks. First, we compare against existing text-conditioned generation systems in both standard generation and prompt transitions, as well investigate the roll ARC-Forcing plays in stabilizing long-form generation. Then we investigate how LMDMs perform for accompaniment generation as a function of the future visibility. Finally, we assess sketch-conditioned generation through offline evaluation and preliminary observations of live use by musicians. Full details can be found in App. A.

5.1 Text-Conditioned LMDMs

Global evaluation. Here, following Team et al. (2025) we report standard global musical metrics (FD/KL for quality, CLAP score for text adherence), as well as metrics for latency (D-NFE, TTFF). We report results both with and without ground truth audio priming. In Tab. 1 we find that despite having half the parameters and trained using nearly 100x less data than LMMs, LMDMs achieve strong FD scores and low generation latency, while KL and text-adherence results are mixed relative to the baselines. We further find that in Enc-Dec LMDMs generally outperform Block-Causal ones. This suggests that the ability to adapt the full contextual encoding as new music is added to the context is beneficial for performance.

Listening evaluation. In our listening study, 19 participants compared pairs of 15 s generations for the same prompt on audio quality, prompt adherence, and temporal coherence (Table 2). Here, we find that LMDM was statistically comparable with MusicGen-Large and SAO on all criterion despite the efficiency gains, and was only worse than MagentaRT on audio quality.

Per-window evaluation. After this, we next evaluated how LMDMs perform on such global metrics as a function of time, generating up to 2 minutes of content. Each metric is calculated on a local sliding window of audio context (see App.A for more details). In Fig. 3, the benefits of ARC-Forcing are on display: without ARC-Forcing, in both setups nearly every metric gradually degrades as a function of time, while ARC-forcing significantly mitigates autoregressive drift. As Enc-Dec models outperform Block-Causal models in this setting, we focus on this model class for the rest of this work.

LMDM versus	Quality	Prompt adherence	Temporal coherence
Magenta RealTime	-0.87^*	-0.53	-0.50
MusicGen-Large	$+0.21$	$+0.16$	$+0.13$
Stable Audio Open	-0.18	$+0.18$	0.00

Table 2: Mean relative ratings of LMDM on 7-point likert scale listening study ($n = 19$). *Significant under paired Wilcoxon tests. Full results and protocol appear in Appendix C.

Configuration	FD-OpenL3 ↓	KL-PaSST ↓	CLAP ↑
ED, no post-training (8 steps)	116.66	1.4962	0.1480
ARC only (no rollout)	103.89	0.7660	0.1683
Without contrastive loss	66.65	1.0197	0.2670
Without discriminator warm start	283.37	3.6661	−0.1159
Self-Forcing-style GAN (+R1/R2)	148.91	1.3043	0.1967
Full ARC-Forcing	**35.88**	**0.7400**	**0.2900**
ED, no post-training (50 steps)	61.06	1.1447	0.2046

Table 3: ARC-Forcing component ablations for text-conditioned Enc-Dec LMDMs. See Appendix B.1 for the tested variants.

ARC-Forcing ablations. In Table 3, we then analyzed which components of the ARC-Forcing recipe were most integral to its performance. We find that our full setup performs the best, with notably that removing discriminator warm-starting severely degrades generation and removing the contrastive objective worsens FD and KL despite a smaller CLAP change.

Prompt transition evaluation. Following Team et al. (2025), we then test the ability of LMDMs to perform prompt transitions, where one prompt is cross-faded with another in over-time. We first found that standard ARC-Forced sampling (which omits CFG) was not enough to break out of the strong condition from past generations, and normal CFG resulted in severe oversaturation artifacts. To remedy this, we found two solutions: (1) whenever one prompt crosses a dominant weight over the other prompt, we drop out the first $d = 180$ frames of context, reducing the signal of past audio, and (2) we adapt CFG++ (Chung et al., 2024) framework to the standard distilled "ping-pong" sampler, allowing for increasing text importance without flying off-manifold (see App. H). In Fig 4, we observe prompt transitions with these modifications. Additional tests using the original LMM tag pairs also show transitions, while adapting tags into captions improves tag-referenced adherence (Appendix D).

5.2 Accompaniment LMDMs

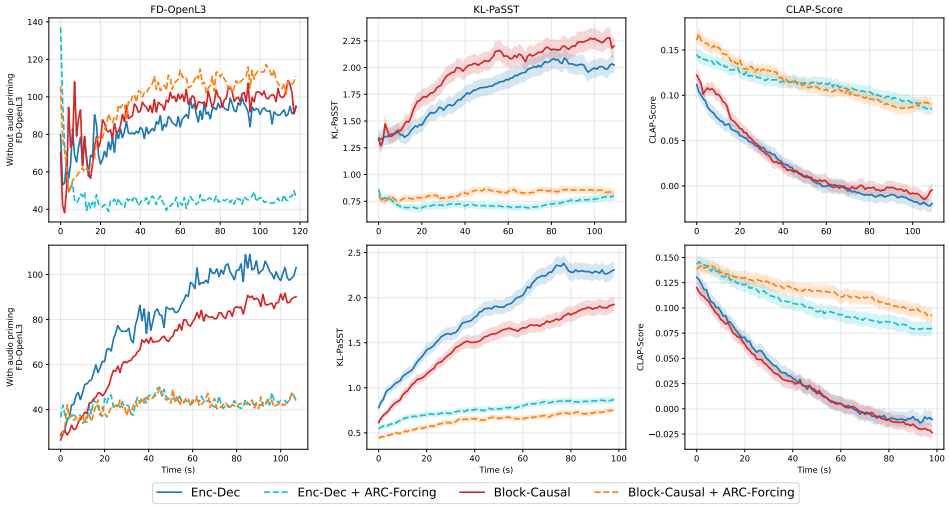


Figure 3: Global Text-Conditioned metrics over time. In both Enc-Dec and Block-Causal LMDMs, ARC-Forcing strongly reduces error accumulation and the gradual degradation of metrics over time.

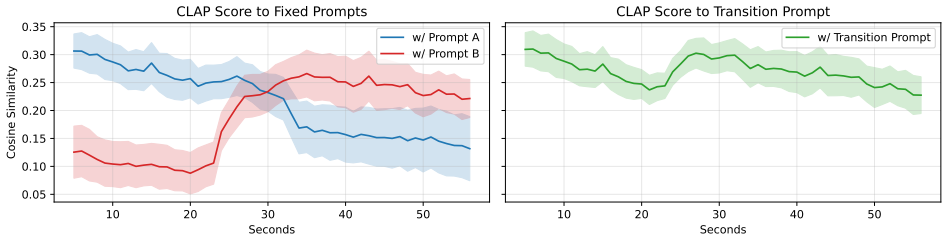


Figure 4: Prompt Transitions using Enc-Dec LMDMs.

Method	D-NFE	Blocks	Sampler	+AF?	FD↓	KL↓	CLAP↑	Mel↑	Rhy↑	Dyn↑
LMDM (ED)	50	5	Euler	✗	101.01	1.52	**0.23**	0.26	0.45	0.46
LMDM (ED)	8	5	Ping-Pong	✓	181.79	1.24	0.14	0.27	0.45	0.45
LMDM (ED-U230)	50	24	Euler	✗	126.41	1.70	0.23	0.18	0.42	0.28
LMDM (ED-U230)	8	24	Ping-Pong	✓	162.38	1.32	0.15	0.21	0.42	0.38
LMDM (Bidir)	50	1	Euler	✗	**78.51**	**1.23**	0.19	**0.33**	**0.48**	**0.57**

Table 4: Distributional and control-following metrics on MUSDB18 test stems (ED and Bidir variants) for the sketch conditioned models with Control evaluation metrics from Tsai et al. (2025).

Given our best Enc-Dec setting from the text-conditioned experiments, we next test LMDMs for stem-conditioned accompaniment. In this accompaniment-like case, we mainly investigate how LMDMs perform with different amounts of future visibility: in settings where $t_f \geq 0$, the model can condition on stem audio that directly corresponds the the target block but leave no room for generative latency to stream, while $t_f < 0$ reduce the amount of stem context the model can see to account for modeling latency and real-time interaction. Here we are primarily concerned with the CoCoLA score (Ciranni et al., 2025), which measures inter-stem alignment. In Fig. 5, we show the relative CoCoLA score (normalized between the average score for ground-truth pairs and fully random pairings) for Enc-Dec ARC-Forced LMDMs with t_f ranging from approx. -2 to 2 seconds. Here we find that while reducing t_f predictably reduces performance, the model does not perform close to random in the $t_f < 0$ case (in contrast to the models from (Wu et al., 2025c)), showing how ARC-Forcing effectively can help mitigate the lack of signal at each block from the stem.

5.3 Sketch-Conditioned LMDMs

We leverage the Enc-Dec setting to trained LMDMs on Jamendo with sketch like controls with details in I. We evaluate the model on 11s rollouts with sketch controls extracted from Vocal, Bass, Drums, and other stems from the MusDB test set ($N = 200$) and shuffled captions from MusicCaps (Agostinelli et al., 2023); all generations use no audio prefixes. We choose this rollout strategy to mirror a real time live interaction. In all, we observe comparable control following with respect to an offline bidirectional model with manageable performance degradation for smaller block-sizes.

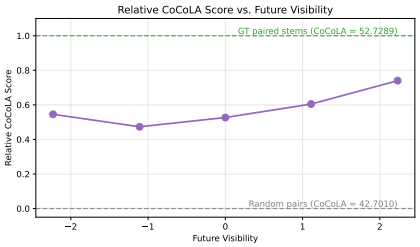


Figure 5: Relative CoCoLA score for Accompaniment LMDMs at variable t_f.

5.4 Preliminary Live Musician Observations with Sketch-Conditioned LMDMs

We deploy sketch-conditioned LMDMs as a real-time generative delay effect: the system takes the last block of a musician's audio, computes sketch controls, and schedules the generated block at a fixed offset into the future. We manage latency by exporting models via ONNX and embedding forward passes in a custom C++/JUCE application (7).

Functionality. The LMDM operates as a block-processing audio effect with block size B seconds: it buffers B seconds of input, computes sketch features, runs inference in τ_θ seconds, and writes a B-second output block, inducing a fixed delay of $\Delta = B + \tau_\theta$ where $\tau_\theta < B$ must hold for gapless playback. On an RTX 4090 Laptop GPU, the eight-step, 10-latent Jamendo model uses $B = 0.464$ s blocks with mean inference time 367 ms, giving $\Delta \approx 0.83$ s. Deployment profiling appears in Appendix D.3.

Participants and Setup. We recruited three instrumentalists from a fellowship program at our institution: a saxophonist (P1), a guitarist (P2), and a cellist (P3). P1 and P2 each tried the self-forced Jamendo LMDM as a generative delay ($\sim$1 s delay) and a foley-like LMDM finetuned on FSD50k (Fonseca et al., 2021) ($\sim$3 s delay), with two conditioning modalities for the Jamendo model: sketch controls from the musician's solo signal, or from a mix of the musician with a drum loop to encourage rhythmic consistency. With P3, we collaborated on a composed performance built around an LMDM finetuned on humpback whale songs (Sayigh et al., 2016), culminating in a public concert (one of two scheduled dates completed at time of writing). Two musicians completed a demonstration, hands-on session, and semi-structured interview; the third was interviewed after extended performance preparation. Videos are provided in supplementary material.

Responsiveness and Musical Dialogue. Participants described moments of musical dialogue with the system. P2 noted that the Jamendo model *"both follows you and accurately throws out new ideas,"* adding that *"even if you stay relatively static, it'll reference your playing while adding something different."* P3 discovered a similar dialogic quality in the whale model, describing the challenge of deciding *"when to initiate, when to answer, and when to simply play my own line and trust something interesting might follow."* A highlight was building toward an emotional peak in their own playing and hearing the whale responses follow.

Timbral Exploration. The musicians also valued the timbral range LMDMs afforded beyond their instruments. P1 discovered that the Jamendo model would shift from bright synth-like mimicry of their saxophone in upper registers to deep bass tones when he played below a certain range. P3 found the whale model compelling for its *"constrained unpredictability"*: she learned to expect two call types whose pitch neighborhood was similar but rarely exact, and gradually developed strategies for triggering responses that fit each musical moment. The foley models expanded this palette further, as P1 used a "wind chime" prompt to complement an improvisation of extended saxophone techniques.

Challenges. During live use, musicians observed a generic EDM texture despite genre prompts. Follow-up text-only tests support a caption-to-tag distribution mismatch: CLAP falls from 0.2952 with full captions to 0.2330 with genre words, while the ONNX comparison shows no degradation in adherence. These tests do not fully isolate the cause in live sketch-conditioned use (Appendix D). The foley models followed text prompts well but struggled with CQT control, likely because much of FSD50k lacks strong fundamental frequencies, making the top-k CQT uninformative for this domain.

6 Conclusion

We introduce Live Music Diffusion Models, showing that open-source audio diffusion models can be repurposed into interactive streaming generators through simple routing and attention masking modifications that enable KV-Caching over both diffusion steps and time. Our ARC-Forcing post-training recipe provides stable, RL-free global supervision on multi-block rollouts, significantly mitigating error accumulation. LMDMs achieve competitive quality with drastically reduced latency, running on consumer hardware at a fraction of the parameter count and training cost of discrete-AR alternatives. Deployment as a generative delay demonstrated live use on consumer hardware, with preliminary musician observations illustrating possibilities for timbral exploration and musical dialogue. However, our musician studies consistently highlighted that further latency reduction could encourage more flexible interactions, and achieving sub-second block sizes will likely require advances in causal audio codecs and architectural efficiency beyond the modifications presented here.

Acknowledgements and Funding Disclosure

We'd like to thank Petros Karypis for their help on debugging a pesky RoPE implementation, Shih-Lun Wu for paper feedback, and Suwan Kim, Valerie K. Chen, Sebastian Franjou, Matthew Michalek for helping demo our sketch-conditioned system. The present work was supported by start up academic funding and general institutional support.

References

Agostinelli, A., Denk, T. I., Borsos, Z., Engel, J., Verzetti, M., Caillon, A., Huang, Q., Jansen, A., Roberts, A., Tagliasacchi, M., et al. MusicLM: Generating music from text. *arXiv:2301.11325*, 2023.

Assayag, G., Bloch, G., Chemillier, M., Cont, A., and Dubnov, S. OMax brothers: A dynamic topology of agents for improvisation learning. In *Proceedings of the 1st ACM Workshop on Audio and Music Computing Multimedia (AMCMM)*, pp. 125–132, 2006.

Ball, P. J., Bauer, J., Belletti, F., Brownfield, B., Ephrat, A., Fruchter, S., Gupta, A., Holsheimer, K., Holynski, A., Hron, J., Kaplanis, C., Limont, M., McGill, M., Oliveira, Y., Parker-Holder, J., Perbet, F., Scully, G., Shar, J., Spencer, S., Tov, O., Villegas, R., Wang, E., Yung, J., Baetu, C., Berbel, J., Bridson, D., Bruce, J., Buttimore, G., Chakera, S., Chandra, B., Collins, P., Cullum, A., Damoc, B., Dasagi, V., Gazeau, M., Gbadamosi, C., Han, W., Hirst, E., Kachra, A., Kerley, L., Kjems, K., Knoepfel, E., Koriakin, V., Lo, J., Lu, C., Mehring, Z., Moufarek, A., Nandwani, H., Oliveira, V., Pardo, F., Park, J., Pierson, A., Poole, B., Ran, H., Salimans, T., Sanchez, M., Saprykin, I., Shen, A., Sidhwani, S., Smith, D., Stanton, J., Tomlinson, H., Vijaykumar, D., Wang, L., Wingfield, P., Wong, N., Xu, K., Yew, C., Young, N., Zubov, V., Eck, D., Erhan, D., Kavukcuoglu, K., Hassabis, D., Gharamani, Z., Hadsell, R., van den Oord, A., Mosseri, I., Bolton, A., Singh, S., and Rocktäschel, T. Genie 3: A new frontier for world models. 2025.

Bogdanov, D., Won, M., Tovstogan, P., Porter, A., and Serra, X. The mtg-jamendo dataset for automatic music tagging. In *Machine Learning for Music Discovery Workshop, International Conference on Machine Learning (ICML 2019)*, Long Beach, CA, United States, 2019. URL `http://hdl.handle.net/10230/42015`.

Brade, S., Ma, T., Blanchard, L., Lecamwasam, K., Salcedo, C. M., Kim, S., Naseck, P., Li, A., R Michalek, M., Franjou, S., and Huang, A. Agents in concert: A case-study of bringing ai to the stage in practice. In *Proceedings of the 31st International Conference on Intelligent User Interfaces*, IUI '26, pp. 1340–1361, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400719844. doi: 10.1145/3742413.3789104. URL `https://doi.org/10.1145/3742413.3789104`.

Cachay, S. R., Aittala, M., Kreis, K., Brenowitz, N., Vahdat, A., Mardani, M., and Yu, R. Elucidated rolling diffusion models for probabilistic forecasting of complex dynamics. *arXiv preprint arXiv:2506.20024*, 2025.

Caillon, A. and Esling, P. RAVE: A variational autoencoder for fast and high-quality neural audio synthesis. *arXiv preprint arXiv:2111.05011*, 2021.

Carr, C. and Zukowski, Z. Prompt jockeys (2024) - the rise of djing with a neural network, 2024. URL `https://www.youtube.com/watch?v=_fpnAHoRSqU`.

Chen, B., Martí Monsó, D., Du, Y., Simchowitz, M., Tedrake, R., and Sitzmann, V. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024a.

Chen, K., Wu, Y., Liu, H., Nezhurina, M., Berg-Kirkpatrick, T., and Dubnov, S. MusicLDM: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. In *ICASSP*, 2024b.

Chung, H., Kim, J., Park, G. Y., Nam, H., and Ye, J. C. CFG++: Manifold-constrained classifier free guidance for diffusion models. *arXiv:2406.08070*, 2024.

Ciranni, R., Mariani, G., Mancusi, M., Postolache, E., Fabbro, G., Rodolà, E., and Cosmo, L. Cocola: Coherence-oriented contrastive learning of musical audio representations. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., and Défossez, A. Simple and controllable music generation. In *NeurIPS*, 2023.

Cramer, A. L., Wu, H.-H., Salamon, J., and Bello, J. P. Look, listen, and learn more: Design choices for deep audio embeddings. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, pp. 3852–3856, 2019.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024.

Evans, Z., Parker, J., Carr, C., Zukowski, Z., Taylor, J., and Pons, J. Long-form music generation with latent diffusion. *arXiv:2404.10301*, 2024a.

Evans, Z., Parker, J. D., Carr, C., Zukowski, Z., Taylor, J., and Pons, J. Stable audio open. *arXiv:2407.14358*, 2024b.

Fitzgerald, J., Moore, G. R. D., Shirken, B., Glass, P., Zananiri, E., Novack, Z., McAuley, J., Berg-Kirkpatrick, T., and Tierney, M. Beyond the vivid unknown. `https://competitionimmersive.festival-cannes.com/en/selection/beyond-the-vivid-unknown/`, 2025.

Flores García, H., Nieto, O., Salamon, J., Pardo, B., and Seetharaman, P. Sketch2Sound: Controllable audio generation via time-varying signals and sonic imitations. In *ICASSP 2025 – IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025.

Fonseca, E., Favory, X., Pons, J., Font, F., and Serra, X. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:829–852, 2021.

Forsgren, S. and Martiros, H. Riffusion: Stable diffusion for real-time music generation, 2022. URL `https://riffusion.com/about`.

Gao, R., Hoogeboom, E., Heek, J., Bortoli, V. D., Murphy, K. P., and Salimans, T. Diffusion meets flow matching: Two sides of the same coin. 2024. URL `https://diffusionflow.github.io/`.

Garcia, H. F., Seetharaman, P., Kumar, R., and Pardo, B. VampNet: Music generation via masked acoustic token modeling. In *ISMIR*, 2023.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. In *NeurIPS Workshop on Deep Gen. Models and Downstream Applications*, 2021.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Huang, N., Gokaslan, A., Kuleshov, V., and Tompkin, J. The gan is dead; long live the gan! a modern baseline gan. In *ICML Workshop on Structured Probabilistic Inference and Generative Modeling*, 2024.

Huang, X., Li, Z., He, G., Zhou, M., and Shechtman, E. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Karchkhadze, T. and Dubnov, S. Towards real-time human-ai musical co-performance: Accompaniment generation with latent diffusion models and max/msp. *arXiv preprint arXiv:2604.07612*, 2026.

Kim, D., Lai, C.-H., Liao, W.-H., Murata, N., Takida, Y., Uesaka, T., He, Y., Mitsufuji, Y., and Ermon, S. Consistency trajectory models: Learning probability flow ODE trajectory of diffusion. In *ICLR*, 2023.

Kim, Y., Lee, S.-J., and Donahue, C. Amuse: Human-ai collaborative songwriting with multimodal inspirations. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 2025.

Kim, Y., Brade, S., Wang, A., Zhou, D., Kim, H., Wang, B., Lee, S.-J., Flores Garcia, H. F., Huang, A., and Donahue, C. A design space for live music agents. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, pp. 1–36, 2026.

Koutini, K., Schlüter, J., Eghbal-Zadeh, H., and Widmer, G. Efficient training of audio transformers with patchout. *arXiv preprint arXiv:2110.05069*, 2021.

Krol, S. J., Llano Rodriguez, M. T., and Loor Paredes, M. J. Exploring the Needs of Practising Musicians in Co-Creative AI Through Co-Design. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2025.

Lan, G. L., Shi, B., Ni, Z., Srinivasan, S., Kumar, A., Ellis, B., Kant, D., Nagaraja, V., Chang, E., Hsu, W.-N., et al. High fidelity text-guided music editing via single-stage flow matching. *arXiv:2407.03648*, 2024.

Lee, K. J. M. and Pasquier, P. Musical agent systems: MACAT and MACataRT. *arXiv preprint arXiv:2502.00023*, 2025. Presented at the Creativity and Generative AI Workshop, NeurIPS 2024.

Lewis, G. E. Too many notes: Computers, complexity and culture in Voyager. *Leonardo Music Journal*, 10:33–39, 2000a.

Lewis, G. E. Too Many Notes: Computers, Complexity and Culture in Voyager. *Leonardo Music Journal*, 10:33–39, 2000b.

Li, T., Tian, Y., Li, H., Deng, M., and He, K. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024.

Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W., and Plumbley, M. D. AudioLDM: Text-to-audio generation with latent diffusion models. In *ICML*, 2023.

Liu, H., Yuan, Y., Liu, X., Mei, X., Kong, Q., Tian, Q., Wang, Y., Wang, W., Wang, Y., and Plumbley, M. D. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *TASLP*, 2024.

Liu, X., Gong, C., and Liu, Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv:2209.03003*, 2022.

Manco, I., Weck, B., Doh, S., Won, M., Zhang, Y., Bodganov, D., Wu, Y., Chen, K., Tovstogan, P., Benetos, E., et al. The song describer dataset: a corpus of audio captions for music-and-language evaluation. *arXiv:2311.10057*, 2023.

Manilow, E., Wichern, G., Seetharaman, P., and Le Roux, J. Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity. In *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 45–49. IEEE, 2019.

Nistal, J., Pasini, M., and Lattner, S. Improving musical accompaniment co-creation via diffusion transformers. *arXiv:2410.23005*, 2024.

Novack, Z., McAuley, J., Berg-Kirkpatrick, T., and Bryan, N. J. DITTO: Diffusion inference-time T-optimization for music generation. In *ICML*, 2024.

Novack, Z., Evans, Z., Zukowski, Z., Taylor, J., Carr, C., Parker, J., Al-Sinan, A., Iodice, G. M., McAuley, J., Berg-Kirkpatrick, T., and Pons, J. Fast text-to-audio generation with adversarial post-training. In *WASPAA*, 2025a.

Novack, Z., Saito, K., Zhong, Z., Shibuya, T., Cui, S., McAuley, J., Berg-Kirkpatrick, T., Simon, C., Takahashi, S., and Mitsufuji, Y. Flashfoley: Fast interactive sketch2audio generation. In *39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI for Music*, 2025b.

Novack, Z., Zhu, G., Casebeer, J., McAuley, J., Berg-Kirkpatrick, T., and Bryan, N. J. Presto! distilling steps and layers for accelerating music generation. In *ICLR*, 2025c.

Pachet, F. The Continuator: Musical Interaction With Style. *Journal of New Music Research*, 32: 333–341, 2003.

Pasini, M., Nistal, J., Lattner, S., and Fazekas, G. Continuous autoregressive models with noise augmentation avoid error accumulation. *arXiv preprint arXiv:2411.18447*, 2024.

Pope, R., Douglas, S., Chowdhery, A., Devlin, J., Bradbury, J., Heek, J., Xiao, K., Agrawal, S., and Dean, J. Efficiently scaling transformer inference. *Proceedings of machine learning and systems*, 5:606–624, 2023.

Prabhudesai, M., Wu, M., Zadeh, A., Fragkiadaki, K., and Pathak, D. Diffusion beats autoregressive in data-constrained settings. *arXiv preprint arXiv:2507.15857*, 2025.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023. URL `https://arxiv.org/abs/1910.10683`.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Rouard, S., Orsini, M., Roebel, A., Zeghidour, N., and Défossez, A. Continuous audio language models. *arXiv preprint arXiv:2509.06926*, 2025.

Rowe, R. *Interactive Music Systems: Machine Listening and Composing*. MIT Press, Cambridge, MA, 1993.

RoyalCities. Foundation-1. `https://huggingface.co/RoyalCities/Foundation-1`, 2026.

Saito, K., Tanke, J., Simon, C., Ishii, M., Shimada, K., Novack, Z., Zhong, Z., Hayakawa, A., Shibuya, T., and Mitsufuji, Y. Soundreactor: Frame-level online video-to-audio generation, 2025. URL `https://arxiv.org/abs/2510.02110`.

Sayigh, L., Daher, M. A., Allen, J., Gordon, H., Joyce, K., Stuhlmann, C., and Tyack, P. The watkins marine mammal sound database: an online, freely accessible resource. In *Proceedings of Meetings on Acoustics*, volume 27, pp. 040013. Acoustical Society of America, 2016.

Song, K., Chen, B., Simchowitz, M., Du, Y., Tedrake, R., and Sitzmann, V. History-guided video diffusion. *arXiv preprint arXiv:2502.06764*, 2025.

Song, Y., Dhariwal, P., Chen, M., and Sutskever, I. Consistency models. In *ICML*, 2023.

Tal, O., Kreuk, F., and Adi, Y. Auto-regressive vs flow-matching: a comparative study of modeling paradigms for text-to-music generation. *arXiv preprint arXiv:2506.08570*, 2025.

Tatar, K. and Pasquier, P. MASOM: A musical agent architecture based on self-organizing maps, affective computing, and variable Markov models. In *Proceedings of the 5th International Workshop on Musical Metacreation (MUME)*, 2017.

Tatar, K. and Pasquier, P. Musical agents: A typology and state of the art towards musical metacreation. *Journal of New Music Research*, 48(1):56–105, 2019.

Team, L., Caillon, A., McWilliams, B., Tarakajian, C., Simon, I., Manco, I., Engel, J., Constant, N., Li, P., Denk, T. I., et al. Live music models. *arXiv preprint arXiv:2508.04651*, 2025.

Thelle, N. J. W. and Pasquier, P. Spire Muse: A virtual musical partner for creative brainstorming. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, 2021.

Tsai, F.-D., Wu, S.-L., Lee, W., Yang, S.-P., Chen, B.-R., Cheng, H.-C., and Yang, Y.-H. Musecontrollite: Multifunctional music generation with lightweight conditioners. *arXiv preprint arXiv:2506.18729*, 2025.

Wu, S.-L., Donahue, C., Watanabe, S., and Bryan, N. J. Music ControlNet: Multiple time-varying controls for music generation. *TASLP*, 2024.

Wu, Y., Chen, K., Zhang, T., Hui, Y., Berg-Kirkpatrick, T., and Dubnov, S. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP*, 2023.

Wu, Y., Brade, S., Ma, A. T., Fowler, T.-J., Yang, E., Banar, B., Courville, A., Jaques, N., and Huang, C.-Z. A. Generative adversarial post-training mitigates reward hacking in live human-ai music interaction. *arXiv preprint arXiv:2511.17879*, 2025a.

Wu, Y., Cooijmans, T., Kastner, K., Roberts, A., Simon, I., Scarlatos, A., Donahue, C., Tarakajian, C., Omidshafiei, S., Courville, A., et al. Adaptive accompaniment with realchords. *arXiv preprint arXiv:2506.14723*, 2025b.

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

Yin, T., Zhang, Q., Zhang, R., Freeman, W. T., Durand, F., Shechtman, E., and Huang, X. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22963–22974, 2025.

Yuan, R., Lin, H., Guo, S., Zhang, G., Pan, J., Zang, Y., Liu, H., Liang, Y., Ma, W., Du, X., et al. Yue: Scaling open foundation models for long-form music generation. *arXiv:2503.08638*, 2025.

Contributions

Zachary Novack – Project Lead, Algorithmic Methodology, Text- and Stem-Conditioned Model Development

Stephen Brade – Project Co-Lead, Sketch-Conditioned Model Development, On-Device Model Wrangling, Live API Development, Artist Collaboration

Haven Kim – Data collection and Pre-processing, Evaluation Design and Development, Project Ideation

Hugo Flores García – Lead Live API Creation and Development, Project Ideation, Artist Collaboration

Nithya Shikarpur – Live API Development, Artist Collaboration, Project Ideation

Chinmay Talegaonkar – Algorithmic Methodology, Project Ideation

Julian McAuley – Project Support and Advising

Taylor Berg-Kirkpatrick – Project Support and Advising

Cheng-Zhi Anna Huang – Project Support and Advising

A Experimental and Evaluation Protocol

All models are finetuned from the base version of Stable Audio Open Small (Novack et al., 2025a), a 340M parameter DiT originally trained on $\approx$ 12s of latent audio. For text-conditioned generation, models are trained on the MTG-Jamendo dataset (Bogdanov et al., 2019), excluding samples from Song Describer. We train all variants on a fixed length of 240 latent frames with a target generation block size of 48. Models are first finetuned from SAO-Small with the context routing and attention mask for 10k iterations with a batch size of 256, taking approximately 8 GPU hours. ARC-Forcing then proceeds for 18k iterations with a batch size of 80. During ARC-Forcing, we perform 12 block rollouts from the model. In the initial finetuning phase, we set p_{uncond} and p_{partial} to 0.2 and 0.3 respectively, while in ARF-Forcing we shift them to 0.5 and 0.1 respectively to improve non-primed generation. For the ARC-Forcing discriminator, we finetune SAO-Small on 768 sequence lengths for 10k steps. When ARC-Forced, LMDMs do not use CFG unless otherwise stated, while non-ARC-Forced LMDMs use a CFG of 7.

We report results for two inference settings on 47 s clips, following Team et al. (2025): *audio-primed generation*, where the model is given a caption and the first s frames of the corresponding ground-truth track as a prefix, and *text-only generation*, where only the caption is provided. We refer to these settings as *primed* and *text-only* hereafter.

Following prior works (Evans et al., 2024b; Team et al., 2025), we report three metrics: Fréchet Distance over OpenL3 embeddings (FD-OpenL3) (Cramer et al., 2019), KL divergence over PaSST logits (KL-PaSST) (Koutini et al., 2021), and CLAP score (Wu et al., 2023). All three metrics are computed using the toolkit released with Saito et al. (2025). Following prior works (Evans et al., 2024b; Team et al., 2025), all reference and conditioning come from the Song Describer Dataset (SDD) (Manco et al., 2023); FD-OpenL3 statistics are aggregated over its 706 unique ground-truth tracks. For latency, we report the number of function evaluations for the decoding process of a single chunk (D-NFE), as well as the Time to First Frame (TTFF), i.e., the wallclock time until the first audio is output, calculated on an NVIDIA 6000 Pro Blackwell GPU. We compare against the SOTA Magenta-RealTime, as well as Stable Audio Open (Evans et al., 2024b) and MusicGen-Large (Copet et al., 2023). We sweep the number of inference steps for ARC-Forced LMDMs exponentially from 1 to 8 (i.e. 1,2,4,8).

To observe drift over time, each generation is produced at the same length as its corresponding ground-truth track, and the same three metrics are recomputed inside sliding windows whose size is determined by each backbone's receptive field: FD-OpenL3 with a 1 s window and 1 s hop (matching OpenL3's 1 s training clip), and KL-PaSST and CLAP score with a 10 s window and 1 s hop (matching PaSST's `max_model_window` and CLAP's input length). We use 8 sampling steps for both ARC-Forced models in this experiment.

Following Live Music Models (Team et al., 2025), we evaluate prompt transitions on 128 pairs. Because the pairs from Team et al. (2025) are short genre/instrument tags (e.g., *Accordion* $\rightarrow$ *Ambient*) that do not match the caption-style conditioning distribution our models were trained on, we use pairs drawn from 256 prompts at random from SDD (Manco et al., 2023). The full list of 256 prompts is provided in Appendix K. Note here we use CFG++ (Chung et al., 2024) with a weight of 0.7.

For the accompaniment experiments, following Wu et al. (2025c), we finetune and post-train Enc-Dec LMDMs on the Slakh MIDI dataset of synthesized stems (Manilow et al., 2019), where stems from the same piece are randomly sampled as context and target. In this setup, ARC-Forcing only occurs for 8k steps due to observed faster convergence. We consider 5 models at varying future visibilities in intervals of 24 latents (roughly 1.1s) from 2.2 to -2.2. All other hyperparameters match the text-conditioned case. In this setup, we replace the text conditions with the midi program name for the target stem (e.g. "electric bass").

B Additional ARC-Forcing and Efficiency Experiments

B.1 ARC-Forcing Component Ablations

Table 3 reports text-conditioned Enc-Dec comparisons at eight sampling steps, with the 50-step non-post-trained model included only as a reference. ARC-only training supervises one generated block without autoregressive rollouts. The no-contrastive variant removes the discriminator's auxiliary contrastive objective. The no-warm-start variant initializes the discriminator directly from the base SAO-Small flow-matching checkpoint rather than first adapting it to longer sequences. This variant rapidly destabilized during training, with generator loss increasing to approximately twenty times that of full ARC-Forcing. The Self-Forcing-style GAN variant uses a relativistic objective with R1/R2 penalties (Huang et al., 2024).

B.2 Efficiency Protocol and Context-Length Sweeps

We compare bidirectional outpainting, Enc-Dec, and Block-Causal at matched shapes and batch size one, without CFG, using the Ping-Pong sampler on the same RTX 6000 Pro Blackwell GPU. Each context block contains 48 latent frames. All variants use CUDA Graphs; Enc-Dec and Block-Causal additionally use static, fixed-address KV caches updated in place. We prefill the context to simulate steady state, perform ten warmup iterations through block generation and VAE decoding, and average full block latency over fifty runs.

Steps	Context blocks	Bidir		Enc-Dec		Block-Causal	
		ms	MB	ms	MB	ms	MB
8	2	35.19	1910	37.61	2034	30.15	2034
8	4	37.62	1911	37.21	2044	30.06	2043
8	8	40.06	1912	37.33	2064	31.16	2062
8	12	41.75	1915	36.52	2093	31.16	2090
50	2	150.59	1911	131.53	2035	126.81	2034
50	4	157.72	1912	132.49	2044	128.37	2044
50	8	168.57	1914	136.17	2064	129.72	2062
50	12	196.84	1917	137.60	2093	129.89	2090

Table 5: Steady-state latency and peak memory versus context length.

Bidirectional latency increases with context length, while LMDM latency changes comparatively little over the tested range, at a modest cache-memory cost. At the main four-block setting, Enc-Dec is approximately tied with Bidir at eight steps and faster at fifty. These small-model, short-sequence results should be distinguished from asymptotic complexity.

C Blind Listening Evaluation

Nineteen participants were recruited through an open call on an internal research-institution forum. Participants reported their formal musical training: eight had at least ten years, three had four to

nine years, three had one to three years, and five had none. Nine used in-ear headphones, nine used over-ear headphones, and one used speakers.

Participants compared two 15 s generations conditioned on the same prompt, rating preference along audio quality, prompt adherence, and temporal coherence on a seven-point likert scale. The study compared LMDM, Stable Audio Open (Evans et al., 2024b), MusicGen-Large (Copet et al., 2023), and Magenta RealTime (Team et al., 2025).

The reported analysis first applied a Kruskal–Wallis omnibus test over per-participant mean comparison ratings, with a Friedman test for matched samples, followed by paired Wilcoxon signed-rank tests. The six model-pair comparisons per criterion used a Bonferroni significance threshold of $0.05/6$. Table 6 gives the reported pairwise p-values and mean ratings; the listed p-values are compared with this corrected threshold. Positive means favor the first-listed model.

	Quality		Prompt adherence		Temporal coherence	
Comparison	Mean	p	Mean	p	Mean	p
LMDM / MagentaRT	-0.87	0.0031^*	-0.53	0.1031	-0.50	0.0904
LMDM / MusicGen	$+0.21$	0.5682	$+0.16$	0.7029	$+0.13$	0.7318
LMDM / SAO	-0.18	0.6458	$+0.18$	0.6769	0.00	0.9426
MagentaRT / MusicGen	$+0.79$	0.0048^*	$+1.03$	0.0039^*	$+0.89$	0.0161
MagentaRT / SAO	$+0.55$	0.0622	$+0.87$	0.0035^*	$+0.68$	0.0417
MusicGen / SAO	$+0.08$	0.8632	$+0.63$	0.0592	$+0.50$	0.0671

Table 6: Full listening-study pairwise comparisons. $^*p < 0.05/6$. MusicGen denotes MusicGen-Large.

D Prompt Distribution and Deployment Diagnostics

D.1 Tag-to-Caption Adaptation and Prompt Transitions

We additionally evaluate the original short genre/instrument tag pairs from Appendix G of Live Music Models (Team et al., 2025). To adapt the conditioning distribution (as our LMDMs were trained on natural language), tags can be expanded manually or with an LLM into short or verbose captions; for example, "Afrobeat" becomes "An afrobeat music track," or a longer description of its percussion, bass, horns, and guitar. We then evaluate similarity both to the actual caption input and to its originating tags; this allows us to compare adaptation strategies and assess overall performance when considering prompt adaptation as part of the model pipeline.

Fig. 6 reports average CLAP similarity to the source and target prompts over five evaluation windows, with crossover at 30 s. Notably, *all* adaptation styles show a shift toward the target prompt. Under evaluating CLAP similarity with respect to the underlying tags, caption adaptation increases adherence relative to direct tag conditioning, likely due to aligning the input distribution to the one seen during training.



Figure 6: Caption

D.2 Prompt Following and ONNX Check

In a text-only diagnostic, CLAP was 0.2952 with full captions, 0.2330 with genre words alone, and 0.1115 without a prompt. We report a separate PyTorch-versus-ONNX comparison in Table 7. It does

not show an adherence loss from ONNX export. Together these findings support a caption-to-tag distribution mismatch as an explanation for reduced live prompt adherence, but they do not isolate all factors in the live sketch-conditioned setting.

Metric	PyTorch reference	ONNX rollout
FD-OpenL3 ↓	34.62	35.46
KL-PaSST ↓	0.4576	0.4561
CLAP ↑	0.2949	0.3209

Table 7: Export-pipeline diagnostic using SDD prompts.

D.3 Consumer-Laptop Deployment Profile

We profile the sketch-conditioned, ARC-Forced model with eight Ping-Pong steps on an NVIDIA GeForce RTX 4090 Laptop GPU (16376 MiB VRAM; driver 580.159.03), using the ONNX Runtime CUDA execution provider in fp32 inside the JUCE application. Blocks contain ten latents ($B = 0.464$ s). CFG over text and sketch controls uses DiT batch size three, giving 24 DiT batch-items per generated block.

Measurement	Value
Mean / median inference time	367 / 361.7 ms
Inference standard deviation	10.7 ms
Minimum / maximum inference time	349.0 / 399.2 ms
p95 / p99 inference time	384.6 / 391.4 ms
Mean end-to-end delay $B + \tau_\theta$	0.83 s
Mean / p99 real-time factor τ_θ / B	0.790 / 0.843
Peak host RSS / GPU memory	1929 / 6010 MB
Sketch extraction (RMS/CQT)	99.4 ± 7.5 ms
DiT sampling	198.9 ± 3.2 ms
VAE decoding	48.3 ± 1.3 ms

Table 8: Laptop deployment measurements for one sketch-conditioned configuration. The listed stages are a partial breakdown of total inference time.

E Procedure for Preliminary Musician Observations

Three musicians were recruited by email from a university fellowship program for accomplished performers. Two completed a structured session with a ten-minute demonstration, fifty minutes of hands-on interaction, and a ten-minute semi-structured interview. The third had extensive prior interaction while preparing a performance and completed an interview only. Interviews focused on successes and challenges encountered while playing with LMDM. Sessions were audio/video recorded and interviews transcribed. We located moments participants themselves identified as compelling or problematic and selected corresponding video segments for the reported observations and project materials. We did not apply a coding scheme or formal thematic analysis.

F Planned Release

We plan to release training and inference code for text-, sketch-, and accompaniment-conditioned LMDMs, data preprocessing and evaluation code, checkpoints for the main evaluated models, the JUCE/ONNX application and export scripts, and experiment prompts and generated samples under a permissive open-source license.

G Related Work

G.1 Interactive and Controllable Music Generation

LMDMs build on a sustained line of interactive music systems and musical agents that have pursued real-time machine listening and musical response since well before deep generative modeling (Rowe, 1993; Tatar & Pasquier, 2019). Early call-and-response agents such as Voyager (Lewis, 2000b) and the Continuator (Pachet, 2003) established much of the interaction paradigm we adopt, and a sustained line of corpus-based and concatenative systems has realized closely related behaviors: the OMax family resequences segments of a corpus matched to a musician's recent input (Assayag et al., 2006); MASOM (Tatar & Pasquier, 2017) and MACAT/MACataRT (Lee & Pasquier, 2025) build real-time agents from self-organizing maps, factor-oracle temporal models, and audio mosaicing; and Spire Muse (Thelle & Pasquier, 2021) introduces a mirroring mode that is a direct precedent for the buffer-and-respond behavior we term a *generative delay*. These systems have real strengths that a generative model does not inherit automatically: fidelity to a chosen corpus, little or no training cost, and a highly interpretable relationship between what a musician plays and what the agent returns. Our aim is to explore a different region of the same design space rather than to displace them. LMDMs, alternatively, can be prompted and reach timbres and styles through natural language rather than curated corpora; they are also amenable to the inference and post-training strategies native to diffusion, as ARC-Forcing demonstrates. The additional training cost of this approach is born once, as a musician needs only a consumer laptop at performance time.

In the landscape of deep generative music modeling, most systems prioritize one-shot generation, mapping control modalities to fixed-length compositions. This includes high-fidelity text-to-music models (Evans et al., 2024b; Forsgren & Martiros, 2022; Yuan et al., 2025) and controllable offline systems utilizing dynamics, melody, music stems, or gestural sketches (Novack et al., 2024; Wu et al., 2024; Flores García et al., 2025; Nistal et al., 2024). However, this offline paradigm remains disconnected from musical traditions centered on real-time adaptation and interaction (Krol et al., 2025; Kim et al., 2025; Brade et al., 2026), creating a workflow incompatibility for many practicing musicians. Historically, technologists and musicians have bridged gaps between technology and tradition like these by adapting expansive creative technologies to be simultaneously more accessible and more usable for musicians. For example, from the miniaturization of synthesizers made them portable and affordable while adapting their control interfaces from patch cables to keyboards allowed them to be more easily integrated into musical traditions that leveraged piano.

This trajectory continues with the creation of more efficient neural architectures and inference paradigms more ammenable to musical interaction. Models like RAVE (Caillon & Esling, 2021) which accepts audio as an input and performs real-time timbre transfer on consumer hardware exemplifies interactivity and efficiency. VampNet (Garcia et al., 2023) allows musicians to create generative loops, providing a generative paradigm analogous to loop pedals. Recent streaming attempts like FlashFoley (Novack et al., 2025b) leverage voice as a control modality to shape generated audio. The state-of-the-art model, Live Music Models (LMMs) (Team et al., 2025), brings text-controllable high-quality music generation to the near real-time setting. In this work, we push the envelope by bringing high-quality music generation to consumer-grade hardware while simultaneously introducing controls that let musicians interface with these models through their instruments, bridging the gap between progress and tradition. To bring high-quality interactive music generation to consumer-grade hardware with the inference efficiency of discrete AR models, we introduce block-wise KV caching and an ARC-Forcing post-training paradigm inspired by the need for rollout-based stability (Wu et al., 2025c).

G.2 Autoregressive Diffusion

Many early works in the diffusion literature focused on static image generation (Ho et al., 2020; Ho & Salimans, 2021; Rombach et al., 2022; Esser et al., 2024), which was the inspiration for the state of fixed-length diffusion-based music generation (Evans et al., 2024a; Forsgren & Martiros, 2022; Chen et al., 2024b; Liu et al., 2023). Recently however, interest has grown in *video* generation, and in particular autoregressive video generation, both from the lens of increasing inference efficiency (Yin et al., 2025) and for creating interactive world models (Ball et al., 2025). Initial diffusion-based video generation focused on approaches with bidirectional attention but folding in the noise schedule as a function of time (with future frames noisier than sooner ones), such as Diffusion Forcing (Chen et al., 2024a) and its variants (Song et al., 2025; Cachay et al., 2025). Later works expanded this to fully causal diffusion, where frames would be generated purely on a history of clean frames (Yin et al., 2025). These have culminated with the recent *Self-Forcing* paradigm (Huang et al., 2025), which post-trains fully causal video diffusion models on real rollouts from the model, using

distribution matching approaches to provide exact global losses that accelerate sampling and reduce error accumulation over time.

However, this direction remains largely unexplored for diffusion-based music generation. Recent continuous-AR approaches (Pasini et al., 2024; Rouard et al., 2025; Saito et al., 2025), based on the fully autoregressive continuous language-model formulation of Li et al. (2024), do not study rollout-based post-training, typically require multi-billion-parameter models for strong performance, and are architecturally distinct from standard diffusion systems. Meanwhile, controllable and interactive diffusion models (Novack et al., 2025b; Karchkhadze & Dubnov, 2026) still rely on bidirectional block-wise outpainting, limiting their efficiency relative to discrete autoregressive models. In this work, we show that targeted modifications can make diffusion-based generation competitive with, and even more efficient than, the current state of the art for interactive streaming inference. We further extend Self-Forcing with Adversarial Relativistic Contrastive (ARC) post-training (Novack et al., 2025a) to support stable minute-long rollouts.

H Derivation of Ping-Pong++ (P4) Solver

Consider the standard stochastic solver used by few-step "consistency-style" (i.e. trained to output $\mathbf{x}^{(0)}$ from any $\mathbf{x}^{(k)}$, as opposed to arbitrary $\mathbf{x}^{(s)}$ as in Kim et al. (2023)) Song et al. (2023); Novack et al. (2025c,a), often called the ping-pong sampler. It's update rule (in flow-matching notation) can be written as:

$$\mathbf{x}^{(k_{i-1})} = (1 - k_{i-1})\mathbf{x}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}) + k_{i-1}\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{7}$$

where $\mathbf{x}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}) = \mathbf{x}^{(k_i)} - k_i\mathbf{v}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c})$ (i.e. we stay in "v-prediction") following Novack et al. (2025a) to ease convergence in adversarial post-training, and $\mathbf{v}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}) = \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \varnothing) + w(\mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}) - \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \varnothing))$ for some guidance weight $w > 1$.

The core of CFG++ Chung et al. (2024) is to formulate sampling where the *denoising* process maximizes adherence to the text prompt, while keeping the *renoising* process unconditional. We can reformulate our pingpong sampler into this denoising-renoising framework as:

$$\mathbf{x}^{(k_{i-1})} = \underbrace{\mathbf{x}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c})}_{\text{denoising}} + \underbrace{k_{i-1}(\boldsymbol{\varepsilon} - \mathbf{x}^{w}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}))}_{\text{renoising}} \tag{8}$$

To work the ping-pong sampler into a CFG++-style sampler, we modify Eq. 8 to use the *unconditional* velocity for the renoising process, forming:

$$\mathbf{x}^{(k_{i-1})} = \mathbf{x}^{\lambda}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \mathbf{c}) + k_{i-1}(\boldsymbol{\varepsilon} - \mathbf{x}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \varnothing)), \tag{9}$$

where $\mathbf{x}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \varnothing) = \mathbf{x}^{(k_i)} - k_i\mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}^{(k_i)}, k_i, \varnothing)$, and $\lambda \in [0, 1]$. This modified P4 sampler is able to tune inference-time text strength much more stably than with the standard ping-pong sampler with normal CFG, as the standard implicit $k_{i-1}\mathbf{x}^{w}_{\boldsymbol{\theta}}(x^{(k_i)}, k_i, \mathbf{c})$ causes noticeable sonic artifacts in the few step regime.

I Catologue of Sketch Models Trained

J Interface Design

As shown in Figure 7, the interface provides the main interaction components of the application.

K Prompt Transition Pairs

Table 10 lists the 128 prompt pairs (A, B) used for the prompt transition evaluation in Section 5. Both endpoints are drawn from the Song Describer Dataset (SDD) Manco et al. (2023) captions, sampled from a pool of 256 distinct captions.

Table 9: Overview of sketch-based encoder-decoder models trained across datasets and configurations. Datasets: FSD50k (Fonseca et al., 2021); ≈48 minutes of humpback whale song (Sayigh et al., 2016); Jamendo.

Method	Architecture	Dataset	Block Size	+AF?	Eff. BS	Steps
LMDM (ED)	Enc-Dec	FSD50k	208/47	✗	128	120k
LMDM (ED)	Enc-Dec	Humpback whale	208/47	✗	128	10k
LMDM (ED)	Enc-Dec	Jamendo	192/48	✗	128	130k
LMDM (ED)	Enc-Dec	Jamendo	192/48	✓	288	4.3k
LMDM (ED)	Enc-Dec	Jamendo	230/10	✗	128	140k
LMDM (ED)	Enc-Dec	Jamendo	230/10	✓	288	3.0k
LMDM (Bidir)	Bidirectional	Jamendo	240/–	✗	128	120k



Figure 7: User interface of the system built in JUCE leveraged in the user studies and performances.

Table 10: Prompt transition pairs used in the cross-prompt continuity evaluation. Each row contains a source prompt A and a target prompt B.

#	Prompt A	Prompt B
1	Driving, energetic and positive rock song (male voice) perfect for sport or action.	An alternative rock piece with piano base, drums and male vocal which often does falsettos.
2	Haunting expansive sound as if you are in space	Joyful Christmas song or children's song featuring a bell melody.
3	Bright synth swooshes in the opening followed by a prominent acoustic guitar and a happy-go-lucky bass-line and beat creating a light and breezy feeling evoking a shopping mall or tourist cafe.	Fast-paced heartsick synthwave techno with crisp melodies and repetitive drums
4	Pop rock song with male lead singer containing slight dissonant passages between the lead guitar and the supporting piano chords.	Emotional and intimate lovely French song featuring acoustic guitars and soft male vocals

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
5	The song as a catchy brass riff and percussions and an upbeat guitar in the background, with a Spanish lyric sung by two singers, one male and one female singers.	The relaxed melody and slow tempo make this song a combination of romantic and peaceful piece
6	Electronic rock high-energy song with vocals but no reverb that draws you in with instrumentation changes, syncopation, and panning.	A funky rock song so high paced and dynamic that it makes one dizzy.
7	Heavy metal song with folk influences, drum and distorted guitars are sustained by a French speaking male voice and a synth lead, in some point the music stops and there is a rain sample.	Calming and tense classical music played only on piano with strong variance in change of dynamics (velocity)
8	A strings orchestra and piano combine in this waltz to give a fantasy feeling.	Country song with acoustic guitar and singing along with slide guitar for embellishments that can be listened to while relaxing at home or just driving
9	Creepy electronic track gives a sensation of suspense and intrigue.	A romantic song which features a couple who just reconciled after going through individual hardships or challenge in their relationship
10	A typical punk rock song of the 2000s, sung by a male voice with a positive and energetic attitude.	A peaceful piano piece for relaxing by the fireplace.
11	A slow swingy song with a male vocal, bluesy clean guitar and smooth drums.	It is an instrumental piano song, with a moody relaxing and gentle tone
12	Instrumental piece with some baroque-era musical arrangements that could well be used in the opening credits of a period film.	Serene, but slightly tense piano piece played at a moderate tempo
13	an uplifting jazz song that makes your head shake	Psychedelic rock with raggae rhythm featuring electric guitar with phaser effects, bass and drums but no vocals
14	A warm and slow paced song that has an R&B base invites to get closer.	Cheerful French love song in a reggae style, and with chorus in English.
15	Piano ternary piece with a repetitive and dynamic pattern that modulate overtime with some 3-for-2 parts	Its an ambient song, with electronic elements like a synth, delayed piano, sine wave oscillators, no vocal, can come under the background, mysterious vibe, relaxing as well
16	Electropop track with lots of synths and a female singer with an eastern european accent, making you want to dance at the club.	Neo-soul song with nylon string guitar and female vocals in French
17	Mid tempo blues swing song that one might find in a casual bar setting for people to dance to	This type of electronic music can be used in a dance club since it builds up fairly quickly and has a dance beat
18	Ominous 2010s hip-hop beat with FM-style bass and discordant sound design and non-english rapping.	Nervous gypsy instrumental song with jazzy acoustic guitars.
19	Electronic instrumental that has a consistent beat and a melody that at times has a descending pattern which can be used for a soundtrack	Slow tempo pop song using only piano and vocals with sad lyrics
20	Progressive electronic song with an intro of African percussions.	Positive instrumental pop song with a strong rhythm and brass section.
21	Upbeat rock guitar song with a punk feel about love	Trance experimental electronic track with a weird sounding vocal sampled lead, hollow drums and a 8-bit sounding beat.
22	This son is pumping with heavy percussion layers and an energetic female vocal.	This is an electronic track with classic snare-roll build up, four-on-the-floor drums and syncopated synth melodies
23	Groovy instrumental funk rock track with occasional guitar solos that give you a feeling of longing.	A pop-rock love song from the 80's that conveys a quiet sense of positivity.
24	Country drinking song with a ragtime or blusey vibe, featuring a female voice.	Electronic music with a sci fi feel to it like an event is about to start which is mostly based on various percussion as compared to a melody

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
25	This track starts with some effect making it look old and vintage, before the effect is removed and the singer starts singing louder, supported by drums, bass and electric guitar	This energetic rock song starts with a drumstick countdown and has a catchy guitar riff and male vocalist.
26	Calming instrumental played on bass violin and flute that can be put in the background while doing some work or study	A catchy electropop track featuring a male vocalist with a unique, drawling delivery style and occasional autotune that adds a cool, quirky edge to the energetic, synthesized soundscape, while the catchy pop chorus is sung by a female vocalist.
27	A guitar folk song with husky male voice	Gentle and dreamy love song with acoustic guitars, strings and male vocals.
28	This is an experimental electronic songs, with noisy synths and a male voice speaking a slavic language.	A sweet and fun track featuring slide guitar and a male vocal.
29	A country tune with a male lead voice and some backing vocals, played with several guitars, some of them using slide.	A harsh instrumental rock song which sounds a bit artificial and dull.
30	French folk, singer-song writer piece featuring guitar and voice.	Ambient instrumental piece with a sense of birth from nothing.
31	Mysterious piece of jazz music for a typical film noir scene, played on piano, bass (and with a trumpet solo section).	Country-tinged, piano driven pop with a female lead singer
32	This is a calm swing music with drums, synth and trumpet responding to each other and makes you want to bounce.	Quivering male voice over solo acoustic guitar, indie folk rock, playful.
33	initially calm and slightly solemn solo piano with pronounced quick double attacks on some notes making it more lively later on	Sounds like a vaporwave song with 80s-style synths, a steady, driving beat with hand percussion and nature sounds in the background.
34	Slow tempo pop song using only vocals and piano with lyrics around love	Progressive rock song with a driving synth base and slurred high pitched male voice.
35	Elegant and pure sounds that convey a sense of order and cleanliness	A lively and happy ska song featuring energetic brass and children vocals.
36	This song starts with an ambient pad intro with a hip-hop influenced beat dropping halfway through.	This is a very fast and dynamic gipsy track with trumpets and voices, that makes you want to jump.
37	Filmic uplifiting non-vocal orchestral piece which builds into a synth and guitar based track	Rock song with a heavy guitar and bass riff played mostly on down strokes with an energizing beat supplemented with vocals
38	An instrumental Christmas carol with a trumpet that leads the melody and a choir of "ahhs".	Pop song that is initially sung by a male voice and then a female voice which carries most of the melody that has an upbeat tempo but sad style
39	An alternative rock song with a male lead singer, drums and an electric guitar with a little distorsion.	A twisty nice melody song by a slide electric guitar on top of acoustic chords later accompanied with a ukelele.
40	A nasal synth line opens the song before a choir of women singers give an emphatic lyrical performance about antagonism and hurt; a deep synthetic bass runs through the track and the drums get more active as the song continues.	Christmas carol with only instruments feels like in a Disney land fill with toys.
41	A soothing track with a mellow synth sound enriched progressively by an upbeat drum machine and robot-like vocalizations.	energetic ska track with driving guitars and drums and brass featuring raw and scratchy vocals
42	this song starts with an emotional piano melody and then drops into a kind of DnB track with drum break samples and a thumping bass, keeping the piano and later on it has a drum break, whe a violin is added.	Classic video instruction background song with major chord and ukelele.
43	An epic soundtrack with drums and choirs that conveys a sense of tension, defiance and danger.	A happy Latin song with a strong rhythmic and percussive component that invites you to dance and enjoy.

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
44	Dancefloor, edm influenced instrumental, driven by a piano house progression layered with trance saws	This instrumental song is so calm that makes me feel sleepy
45	A smooth blues song gets exciting as the vocal joins and lets us through his meandering melody.	Energetic bluesy song with a harmonica and horn section in musical dialogue.
46	A loud rock/metal song in French with guitars and drums	Instrumental with mostly piano that has a melancholic feel which can be heard alone
47	An ambient track perfect for meditations and focus tasks.	A fun song with guitar, drums, brass instruments and a male vocal.
48	Instrumental rock song that begins with an ambient intro and then progresses through various sections of electric guitars that intensify over time.	Groovy rhythm with the guitar especially when the bass kicks in, un-noticeable absence of loud snare and kick, poppy and catchy
49	Elegant and fragile piece of orchestral instrumental music for a film soundtrack set in the Middle Ages or in an Asian country, with a melody played by a violin accompanied by plucked strings.	This is a experimental piece or sound effect with quiet noises of running water and someone whispering.
50	Upbeat acoustic guitar song with country style singing and vocals along with harmonica plus whistling at the end	Rock style track with simple bassline, heavy guitars, and a male rock vocalist.
51	A traditional heavy metal riff intro with a transition to grunge-like verse.	90s hip hop with a moody synth which gives a very ominous but danceable vibe to it.
52	Classical symphony, sounding like someone is doing something hastily	A blues song with crisp modern production, featuring punchy drums, lots of slide guitar and a passionate male singer.
53	Jungle or forest fauna sounds followed by floaty ethnic flute solo on top of some hopeful synthesizer harmonies.	Adventurous and curiosity inspiring soundtrack consisting of bells, flutes, strings, and choir.
54	Smooth chill electronic music featuring calming flute and techno beats.	Fast tempo percussion with an energetic beat and a vocal melody
55	Childish and innocent orchestral piece of music that conveys joy and happiness and gives a sense of beginning.	Upbeat song that has a humming riff accompanied by guitar licks that can be used for a casual listening setting
56	Happy optimistic instrumental song with whistles, xylophone and ukulele that can be used to clap and dance along to	Strummed acoustic guitar and female vocal on an uplifting pop song performed by real musicians.
57	A folk song with a bitter-sweet acoustic guitar and a pleasant sounding male vocalist, that turns into a melancholic duet with a female voice.	A mellow and joyful piece of classical music for solo piano
58	A Latin American song with accordion and trumpets and a swaying, but heavy feel.	This is pop song with drums, synth, electirc guitar and some lyrics, with a classical chord progression and a memorable melody.
59	Instrumental mainly focused on clean electric guitar sound riffs and melody which might be fun to listen to on the background	Classical piano piece in major key with a slow start, played beautifully without other instruments, making you close your eyes and find inner peace.
60	A medium-paced EDM-style song with an intro by a synthesizer, features some ambient sound and simple drum.	A gentle folk song with an acoustic guitar and a male voice that conveys a sense of fragility and hope.
61	Slow-paced guitar ballad with clean guitar solos interspersed throughout the composition	Indie folk-rock song sung by a young male vocal with things to say.
62	A French folk track backed by acoustic guitar	Guitar heavy folk song with accompanying male vocalist talking with introspection, not a particular happy song
63	A whimsical string arrangement that feels like bouncing through some woods on an adventure in a video game, with a triumphant finish.	Fast tempo electronic music with a melody that repeats consistently
64	Driving rock song with an energetic chorus, featuring heavily distorted guitars and male vocals.	A dreamy and ethereal piece of electronic music with a Spanish guitar.
65	Hip hop track with a subtle reggaeton feel, with a male rapper and a female soul choir in the background.	Innocent and playful classical music for orchestra featuring wind musical instruments such as oboes or fagots.

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
66	This is an instrumental track based on electronic samples that can be used before the start of a movie to give a doomsday or post apocalypse feel	A folksy ditty featuring a warbling woman narrating a story over a variety of jamming, Western-sounding instruments, which evokes a nostalgic sense of older country, bluesy rock.
67	A positive tune with a surprising mix of instruments: acoustic guitar, tabla and sitar	a rock genre song with fast tempo guitar riffs and drums with a processing voice singing
68	Pop song with claps, piano chords and vocals	This song has a murky, underwater sound with panicked vocals
69	This smooth jazz track which features a sleepy English vocal is pleasant to listen and perfect for nightdreaming.	depressive music with only guitar and a sad voice for guys using drugs would be ideal for a rainy day to be even more sad
70	A rap song with two male voices and sounds in the background that loosely resemble a siren.	Electro dance song to play in the pub to cheer up the crowd.
71	A very energetic and bright funk rock song which features a noticeably solid guitar licks.	Pop love song that would accompany perfectly a morning run.
72	The song start with a dark guitar riff but quickly turns into French EDM mixed with downtempo, with a strong beat and vocal samples and pads morphing through the song.	French-language song with a jazzy, late-night vibe featuring a male duet.
73	One cannot avoid moving the feet and neck listening to this fast and loopy brazilian tune.	Portuguese-spoken ska track with usual melody and instruments.
74	1990s techno with cliche minor key progression and tacky synthetic saxophone lead	The song lyrics is in Spanish and has a salsa rhythm that makes you want to have fun and dance
75	Wonderfully emotional soundtrack with a violin carrying the main melody.	A ska song with all the usual suspects except for a slow and kind of melancholic intro.
76	A ballad song with an acoustic guitar and a male voice singing french words.	energetic rock song about love with guitar and piano
77	A wobbly funky track with a foreground bass line and rhythms made with vocal just makes one feel alright.	Grand, ambitious movie music with powerful, march-like drum beats, as well as brass melody.
78	A positive and enthusiastic pop-rock song from the 80s featuring reverbered guitars.	A frantic rock intro with bass and drums turns into a happy and energetic ska punk song with a male vocalist.
79	Piano ballad that could be used in a ballet, accompanied by percussion and a female vocalist with backing harmonies	Introspective, raw rock song with organic acoustic guitars and a raspy voice
80	Lush vocals on well-reverbed guitars give a nice sensation at first but then makes the listener ask for more.	cheerful happy music played on a piano for relaxing
81	Male vocalist with a raspy voice singing over melancholic piano chords and drums increasing in intensity, with a slighty dissonant chorus featuring distorted guitars.	slow and arpeggiated solo piano with a boatload of reverb
82	a sinister medieval-sounding piece with a synthesized flute or organ and strong downbeats at a marching pace.	Instrumental on acoustic and electric guitar with a pleasant feel that can be played in a cafe
83	A pleasant instrumental folk song with acoustic guitars and violins that conveys a sense of peace and happiness.	A thrilling instrumental track featuring a series of stringed instruments being gradually built into the song, culminating in eastern european vibe section.
84	Upbeat hip-hop with trap vibes and bubbly bass synth.	2000's style urban pop song with synthesised pad and drums.
85	This a dark artsy metal piece featuring plenty of sound effects.	This sentimental rock track features a clean electric guitar with heavy echoes and troubled male vocals.
86	A driving french song featuring fast guitars playing.	8-bit melody brings one back to the arcade saloons while keeping the desire to dance.
87	A fuzzy and dampened electronic piece which progressively brightens throughout the track.	This track featuring a solo piano gives a sense of safety, determinedness and vulnerability.
88	Upbeat fast tempo with a blues rock feel that one can dance	Acoustic guitar that overtakes the female vocal line) starts the song, as a rimshot-heavy drum groove emerges; warbly vocals bring in the chorus, which is nostalgic and reminiscent.

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
89	A lively and fun ska song in Spanish with energetic brass sections.	Classical solo piano music, romantic period. Complex harmonic textures, the pièce reminds of a joyful but melancholic moment, with cold weather outside and the bite of winter at the door, sitting around a fireplace in the living room.
90	Energetic pop rock track that will surely keep the dancefloor moving in a live concert.	Upbeat latin/salsa song with Spanish lyrics, accordion, and flamenco guitar suitable for a warm and hectic dance floor
91	This is an alternative rock song with slow tempo and guitar, male vocals and drums.	Synth-y, spacey art-pop track with a catchy beat
92	It's a folk song with country vibe, major chord, and rock drum kit.	Indie song with a synth effect riff along with vocals
93	A french song that opens with an answering machine sample and then evolves into a alt rock track with frenetic guitar split by a accordion-like synth track	A driving indie rock song sung in Spanish by a male vocalist.
94	This song makes you feel like entering to a haunted mansion or magical village.	A catchy bassline with a drum rhythm sequence introduces the song, followed by a simple chord cadence played on a piano, with some vocals with no lyrics.
95	Electronic disco music about love that builds up quickly and has a dance beat to it	A rock song that transitions between calming humming melody and regular rock singing giving a change of feel throughout the song
96	Male hip-hop track with a tribal vibe featuring african percussion.	Typical energetic and positive EDM song from the early 10s that could be played in a discotheque on a summer night.
97	A fun upbeat drums section playing with a keyboard and some electronic horn type noises, feels like the automatic song on a keyboard that you'd play in music lessons	An harmonised male vocal sample intro, a downtempo electronic beat, a male sung chorus riff followed by a male rapped verse.
98	A pop style synthesized instrumental track with mellow, chorus-like top line, and a prominent drum beat.	A sad pop song using piano and vocals that has lyrics around someone they love
99	Instrumental jazz ensemble with a bossa nova rhythmic vibe that conveys a sense of positivity.	Live recording of a french blues music that starts with a male voice talking alone, and then is followed by a fast-paced blues music with drums, trumpets, harmonica and electric guitar
100	Latin chill song with a bosa nova flavour, accompanied by a melodic trumpet and sung by a male voice.	The rap song has a catchy riff with percussion based on claps and beats but there is a vocal rap on top that goes into a chorus that one can sing along to
101	Electronic music that can be listened to in a passive setting	An instrumental world fusion track with prominent reggae elements.
102	Upbeat commercial-sounding song with a simple drum rhythm, guitar chords and a simple but spacey-sounding synth melody	Futuristic experimental song with piano over a drum and bass rhythm, with an accordion added towards the middle of the song.
103	A ukulele instrumental track sounding like children's song, with a persistent clap percussion and some different riffs on the same theme played in sequence by a glockenspiel, a synth and a slide guitar.	Rap song full of pop elements and sound effects, quite classical and simple dispite off the complex post production
104	Repetitive, sad feeling pop song with a piano intro, finger snapping and female vocal samples.	A musical call to be in the here and now with this deeply spiritual piece.
105	Heavy distorted lead guitar with sparsely supporting piano and distant sounding drums in the background	Melodic and heartfelt piano ballad performed by a female voice accompanied by piano.
106	A whimsy and unserious piano melody, underpinned by consistent snares with occasional offbeat hihats, with a woman singing in a sometimes childish and sometimes sultry voice	a piano ballad with a female singer evoking nostalgia and disillusioned love

Continued on next page

Table 10 – *Continued from previous page*

#	Prompt A	Prompt B
107	Repetitive dancefloor track with loud synths and a simple beat	Gentle folk song with male vocals and arpeggiated acoustic guitar.
108	An romantic song with only piano and vocals that is contemplative yet relaxing	This song starts with a strange french conversation and chain sounds and then gives way to a very heavy and intense synth metal section.
109	EDM fast-paced non-vocal track with light four-on-the-floor beat predominantly based around wobble saw bass line and synth melody	This track is hispanic genre, with spanish lyrics, it has a sort of romantic, love theme, with trumpets, guitar and drums.
110	arythmic piano accompanied by some catchy lyrics sung by an emotionally charged man.	This track starts creating an ambiental atmosphere on which a female singer starts singing, evoking a misterious and sinister character.
111	This track sounds like the backing track to an informative video about a campaign or a new product.	A luminous and moving instrumental piece for piano.
112	This midi electronic instrumental has a hopeful and optimistic vibe with melodies that mostly repeat for 4 bars	This song starts with a piano that can be heard from far but then gets exciting with guitar and a fast-paced male vocal.
113	Rising sweling violin instrumental that elicites calm floating ephemeral emotion.	Rock song starting with guitar line, with a catchy refrain.
114	A dark and slow electronic track makes one think of scary dungeons filled with slimy skulls.	Two electric guitars in conversation with each other, one with a wua-wua effect and the other with a strong delay effect.
115	Warm repeating melody which seems to be a folk music playing by a band where they greets new visitor.	Contemporary trendy optimistic indie pop, with dirty drums, happy guitar comping and synthesizer solo
116	Typical guitar-folk song with no lyrics, but a choir of voices being used for giving texture.	Instrumental song with a sad but intense romantic Latin vibe, the melody is carried by a piano.
117	Orchestral music with a slow and steady pace that sounds like a soundtrack to a nature movie or documentary.	A modern country song with romantic themed lyrics.
118	A track with brit rock essence accompanied with pan flute melody.	Instrumental ambient track with an 80s chillout vibe, featuring bongos and a piano melody.
119	Synthetic orchestral music with hopeful, yearning harmonies, fast pizzicato motifs and woodwind lead	This generic pop/ rock song features a man singing with a nasal voice, with a tired and boring beat behind it.
120	upbeat synthesized guitar pop instrumental track, with consistent tempo and rhythm	A retro-futurist drum machine groove drenched in bubbly synthetic sound effects and a hint of an acid bassline.
121	Instrumental song played by an ensemble of cheerful acoustic guitars, giving the feeling that all is well and nothing bad is going to happen.	Unnerving mix of electronic and rock music that has a gloomy vocal feel and turns into screaming in the chorus
122	Intriguing, slowly progressing electronic tune perfectly match for an indie platform game	Ambient song with duduk and oriental drums perfect for starting a trip to east.
123	Pop guitar strumming with a raspy whispery low female voice	Alternative / experimental rock song with male vocals an a futuristic dreamy vibe.
124	Calm sitar and Indian tabla with dramatic synthetic strings background	A joyful classical track performed on a grand piano.
125	EDM pop song with an energetic an positive mood.	A cinematic piece with a very bright piano, later joined by a drum machine, hand clapping and a violin synth sound which makes the track sound a little bit more oriental.
126	bluesy guitar with a slow repetitive rythm in a smoky room in latin america	Agitated West Coast rock with brief bass solo and British influence
127	Eurodance pop track, with a simple rough synth chord progression and a straightforward drum beat punctuated by a female voice sample.	This is an energetic and positive rock song with guitars, keyboard, drums and a male vocal.
128	A delightful chord progression on an acoustic guitar later accompanied by an ukelele and harmonica.	Indie rock most likely a 4 piece band that someone can listen to while driving

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction list four contributions: LMDMs through routing and attention masking to enable KV-caching, ARC-Forcing post-training, bringing offline diffusion controls to streaming, and deploying LMDMs as a generative instrument with real musicians. All of these are substantiated in Secs. 3-5 with experimental results and a live deployment.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Sec. 5.4 ("Challenges") and our conclusion for our discussion of limitations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: No formal theoretical results exist in the paper. All results rest on empirical analysis.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper describes all modeling design decisions, from the base model, to the routing and masking modifications (Sec. 3.1) and the ARC-Forcing post-training procedure (Sec. 3.2), as well as evaluation metrics and protocols (Sec. 5 and App. A), the datasets used (Jamendo, Slakh, MusDB, FSD50k), and key hyperparameters (number of diffusion steps, block sizes, etc.). Full experimental details are deferred to the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: Code will be eventually released but is not included in the present submission. All data comes from publicly available sources, but we are not allowed to actively host them.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Key experimental details are provided throughout the paper, with full details in the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: 95% Confidence Interval bounds are provided for the main over time evaluations (Figures 3 and 4).

   Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details of inference and training compute resources are described both in the main experiments section as well as the appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research involves only uses publicly available datasets. The human study with musicians was conducted through with informed consent implied through voluntary participation and recorded interviews.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses broader impacts in our conclusion.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The model described in the paper has no high-risk of misuse, as it lacks the ability to directly engage in large scale copyright infringement in the way large Lyrics2Song models do.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites the original publications for all datasets used and all baseline models (Stable Audio Open Small, MusicGen, Magenta RealTime). All models and datasets used support licenses that allow the present experiments.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We provide a comprehensive demo page of example generations and anonymized user interactions in the paper.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: Sec. 5.4 describes the musician interaction study in detail. Videos are provided in supplementary material.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [Yes]

Justification: We attained institutional approval for our user study, which posed minimal risk.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core methodology of this paper (flow matching, DiT architectures, ARC-Forcing post-training) does not involve LLMs as a component. The method operates on continuous audio latent representations using diffusion/flow-matching models, not large language models.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.