BRIDGE ADAPTERS: LEARNING RESTORATION BRIDGES FOR MUSIC RESTORATION

Anonymous authors
Paper under double-blind review

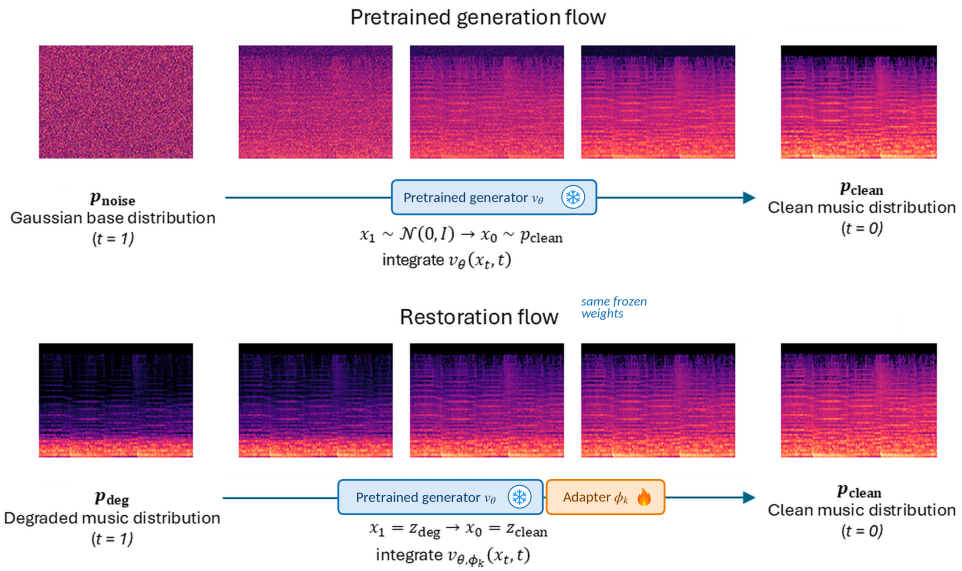

Figure 1: BRIDGE ADAPTERS reuse the frozen generator v_θ (top) with a trainable adapter ϕ_k, so that the flow starts from the degraded recording instead of noise (bottom). Spectrograms are schematic.

ABSTRACT

Pretrained music generators learn powerful priors for clean, perceptually realistic audio, but restoration requires more than generation: the output must remain faithful to a degraded recording. Existing approaches either train degraded-to-clean restoration flows from scratch, or reuse pretrained generators by conditioning their original noise-to-clean trajectory on the degraded input. We propose Bridge Adapters, a parameter-efficient alternative that changes the transport itself. Starting from a frozen foundation music generator, we train degradation-specific adapters with a degraded-to-clean flow-matching objective, converting the pretrained generative vector field into a restoration bridge while leaving the backbone unchanged. On the SonicMaster benchmark, Bridge Adapters improve over the prompted SonicMaster all-in-one restoration model in SNR, SI-SDR and FMA-Pop Fréchet Audio Distance on average over its nineteen music degradations, at the same number of sampling steps, while training only an adapter and keeping the foundation backbone frozen. Compared with conditional baselines built on the same pretrained backbone, a ControlNet and a capacity-matched conditional LoRA, adapting the transport yields substantially more faithful restorations at equal or lower sampling cost. We further show that the adapted bridges inherit the strength of the pretrained prior: within a codec-matched backbone family, a stronger backbone gives more faithful restorations, and across five music generators restoration fidelity tends to track the generation FAD of the frozen backbone. These results suggest that restoration bridges need not necessarily be learned from scratch; for music restoration, many are only an adapter away from a pretrained generative model.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

High-quality music generation remains a difficult modeling problem. Music contains structure across many time scales, from transients and high-frequency texture to rhythm, harmony, stereo image, and long-range arrangement. Recent music foundation models, such as Stable Audio and ACE-Step, have made rapid progress by learning powerful generative priors over clean, perceptually realistic music (Evans et al., 2024; Gong et al., 2025). These models are therefore natural candidates for music restoration as well: both generation and restoration require outputs that sound like high-quality music.

Restoration, however, is not ordinary generation. A generator only needs to produce plausible clean audio; a restoration system must produce clean audio that remains faithful to a particular degraded recording. This faithfulness requirement is both a constraint and a source of information. It is a constraint because the model cannot freely hallucinate any realistic musical sample: it must preserve the timing, content, instrumentation, spatial image, and musical identity of the input. But it is also a guidance signal, because the degraded recording is already highly structured and often close to the desired output. Restoration is therefore easier than generation in that it starts from an informative recording, but harder in that the output must remain tied to that recording.

This raises a central question: how should pretrained generative music models be used for restoration? A closely related line of work appears under the term *music enhancement*. Kandpal et al. (Kandpal et al., 2022) enhance consumer-grade music recordings by translating degraded mel spectrograms toward cleaner ones and vocoding the result, while Chae et al. (Chae et al., 2023) adapt time-frequency Conformers to music enhancement for noisy and reverberant recordings, including both single-stem and mixture settings. These works establish music enhancement as a music-specific counterpart to speech enhancement and highlight the importance of evaluating perceptual quality in the music domain. Our work addresses the same broad goal of improving degraded music, but studies a different mechanism: adapting frozen generative music backbones into degraded-to-clean restoration bridges.

Existing generative restoration approaches fall broadly into two families. The first family, which we call *restoration-transport methods*, learns a restoration transport directly. Flow-matching and bridge methods train a model whose trajectories start from degraded data and end at clean data. In speech, Brownian-bridge and regression-diffusion methods (Lay et al., 2023; Trachu et al., 2024) follow this principle, and UNIVERSE (Serrà et al., 2022) connects deterministic enhancement with stochastic diffusion refinement. In music, A2SB (Kong et al., 2025) learns Schrödinger bridges for high-resolution bandwidth extension and inpainting. Related ideas also appear in the image domain: Cold Diffusion (Bansal et al., 2022) shows that diffusion-like generative models can invert deterministic degradations such as blur and masking instead of relying on Gaussian noise. These methods are well aligned with the restoration problem, but they typically train a restoration model from scratch for each task or family of tasks, thereby ignoring the large clean-music priors already learned by foundation generators.

The second family, which we call *pretrained-prior methods*, reuses pretrained generative models more directly. In these methods, the degraded input is used to condition, guide, invert, or initialize a pretrained generative prior. Examples include SDEdit-style editing, posterior sampling methods for inverse problems, and ControlNet-style conditional adaptation (Meng et al., 2021; Chung et al., 2024; Zhang et al., 2023a; van Delft et al., 2024). These approaches exploit strong pretrained priors, but they often preserve the original generative transport: the model still represents a trajectory whose source is a base distribution, typically Gaussian noise, and whose target is clean data. The degraded input guides this trajectory, but does not become the source distribution of the transport.

Recent image-restoration works have begun to address the gap between these two families. IRBridge (Wang et al., 2025) observes that restoration bridges start from low-quality images whereas pretrained generative diffusion models start from noise, and derives transition equations between the two processes . This suggests that pretrained generative priors can, in principle, be incorporated into restoration bridges. However, analytic process conversions or operator-aware guidance are difficult to apply in general music restoration, where degradations are diverse, perceptual, and not always described by a tractable measurement operator.

We take a data-driven alternative. Rather than deriving an explicit conversion, we ask whether a pretrained foundation music generator can be converted into a restoration bridge by learning adapters.

A pretrained flow model defines a vector field that transports a simple base distribution, typically Gaussian noise, to clean music. Restoration requires a different transport: the target remains clean music, but the source is degraded music. Since changing the source distribution changes the vector field, conditioning a base-to-clean generator and learning a degraded-to-clean bridge are different operations.

We introduce BRIDGE ADAPTERS, a parameter-efficient method for learning this new vector field from a pretrained one. Given a frozen foundation music generator, we insert degradation-specific adapters and train the resulting network with a degraded-to-clean flow-matching objective. The adapter does not merely provide side information to the original generator. It changes the computation of the frozen backbone so that the resulting network predicts the velocity field of a new transport. Thus, our method learns a new restoration field while keeping the pretrained backbone untouched.

This design directly targets the trade-off between the two existing families of methods. Like bridge and flow-matching restoration, it starts from the degraded input and learns a degraded-to-clean transport. Like prior-based and conditioning-based methods, it reuses a powerful pretrained generator. Unlike from-scratch bridge models, it updates only a small number of parameters and unlike standard conditional control, it changes the source endpoint of the transport instead of guiding the original base-to-clean trajectory.

We evaluate BRIDGE ADAPTERS on the SonicMaster (Melechovsky et al., 2026) restoration setting, which provides paired degraded and high-quality music across nineteen degradation functions and a strong all-in-one flow-matching baseline. We also compare against a ControlNet baseline over the same pretrained backbone, allowing us to isolate the effect of changing the transport rather than merely adding a conditional branch.

- We formulate music restoration with pretrained generators as a source-endpoint adaptation problem: from the native base-to-clean generative field to a degraded-to-clean restoration field.
- We introduce BRIDGE ADAPTERS, a parameter-efficient method for converting a frozen pretrained music generator into degradation-specific restoration flows.
- We show that adapted restoration flows outperform SonicMaster, a strong from-scratch all-in-one restoration baseline, on average in paired fidelity and FMA-Pop FAD, and outperform ControlNet and capacity-matched conditional LoRA baselines in paired fidelity at equal or lower sampling cost.
- We analyze backbone strength, adapter capacity, and solver steps, showing that stronger music priors and higher-capacity adapters improve reconstruction while retaining few-step inference.

2 RELATED WORK

Music foundation models. Recent generative audio models have made high-quality music synthesis possible by learning strong priors over clean music. Stable Audio and ACE-Step are representative examples of modern music foundation models trained for perceptually realistic generation (Evans et al., 2024; Gong et al., 2025). These models motivate our work: if restoration and generation both require high-quality musical outputs, then restoration should benefit from reusing such priors.

Audio restoration with diffusion and bridges. Diffusion-based audio restoration methods differ mainly in how they use the degraded signal. The survey of Lemercier et al. (2024) distinguishes input conditioning, task-adapted diffusion, and external conditioning. Our work is closest in spirit to task-adapted diffusion and bridge methods, where the restoration process starts from the degraded signal rather than from Gaussian noise. OUVE, BBED, THUNDER, and A2SB all support the view that degraded-to-clean trajectories are natural for audio restoration (Richter & Gerkmann, 2024; Lay et al., 2023; Trachu et al., 2024; Kong et al., 2025). These methods typically train a restoration model or bridge process directly. In contrast, we ask whether a pretrained music generator can be converted into such a bridge by training only adapters on the frozen backbone.

Known-operator inverse problems. A separate line of work uses diffusion priors for inverse problems with known or parameterized degradation operators. Diffusion posterior sampling combines

a learned prior with a likelihood or measurement-consistency term (Chung et al., 2023). In audio, methods such as BABE, BABE-2, and BUDDy use parameterized restoration models for bandwidth extension, equalization, or dereverberation (Moliner et al., 2024a; 2025; 2024b). These methods are closely related in their use of clean generative priors, but they rely on an explicit operator, projection, or parameter-estimation procedure. Our setting is data-driven: we train from paired degraded–clean data grouped by degradation family, without evaluating a degradation operator at inference.

Pretrained priors for image restoration. Image restoration has extensively explored the reuse of pretrained diffusion priors. StableSR, DiffBIR, BIR-Adapter, and related methods adapt or condition large image generators for restoration tasks (Wang et al., 2024; Lin et al., 2024; Eteke et al., 2026). More directly related, IRBridge observes that restoration bridges start from low-quality images whereas pretrained diffusion models start from noise, and derives transition equations to connect these processes (Wang et al., 2025). Our work studies the analogous question for music restoration, but takes an empirical parameter-efficient route: instead of deriving a transition equation, we learn a new flow-matching field with adapters on the frozen backbone.

Adapters and conditional control. Parameter-efficient adaptation methods such as LoRA, AdaLoRA, and DoRA modify pretrained networks with a small number of trainable parameters (Hu et al., 2021; Zhang et al., 2023b; Liu et al., 2024). ControlNet freezes a pretrained diffusion model and adds trainable conditioning branches (Zhang et al., 2023a). In image restoration, adapter-based methods such as UIR-LoRA and LoRA-IR show that low-rank modules can be effective for universal or all-in-one restoration (Zhang et al., 2024; Ai et al., 2024): they specialize low-rank experts to degradations and combine or route them, including for mixed degradations. Degradation-specific low-rank experts are therefore not our contribution. The difference lies in what the adapter is trained to do: BRIDGE ADAPTERS train the pretrained generative vector field itself against a degraded-to-clean flow-matching target, so that the degraded latent becomes the source endpoint of the transport, without any conditioning branch. IRBridge addresses the same source mismatch analytically, with transition equations for a fixed pretrained model; we learn the change of transport from data with an adapter. Our capacity-matched Conditional LoRA baseline (Section 4.3) uses the same backbone, adapter, data, and solver, and only changes whether the degraded audio is the source of the transport or a conditioning signal, which isolates this design choice empirically.

All-in-one music restoration. SonicMaster is the closest full-mix music restoration baseline. It trains a single flow-matching model for music restoration and mastering across degradations involving equalization, dynamics, reverberation, amplitude, and stereo processing (Melechovsky et al., 2026). Our setting is complementary. SonicMaster learns one all-in-one model from scratch, optionally guided by text prompts. We freeze a pretrained music backbone and train small degradation-specific bridge adapters. In our main comparison, SonicMaster is given the correct degradation prompt and our method activates the corresponding adapter.

3 METHOD

This section defines the latent restoration problem, the adapterized vector field, and the training and sampling procedures used in our experiments (Figure 1; Figure 2 contrasts it with conditional and from-scratch approaches).

3.1 LATENT RESTORATION SETTING

Pretrained music generators transport Gaussian noise to clean music. Conditional approaches keep this transport and use the degraded recording only as side information, although the recording already contains much of the target: timing, content, instrumentation, and spatial cues. Bridge methods instead learn a direct degraded-to-clean transport, which is shorter and can be sampled in few steps, but they usually train the restoration model from scratch. Our hypothesis is that, for many music degradations, the degraded-to-clean field is a low-complexity adaptation of a pretrained clean-music field. BRIDGE ADAPTERS test this hypothesis: they keep the pretrained backbone frozen as a clean-music prior, move the source endpoint of the transport from Gaussian noise to the degraded latent, and learn only an adapter that predicts the corresponding restoration velocity field.

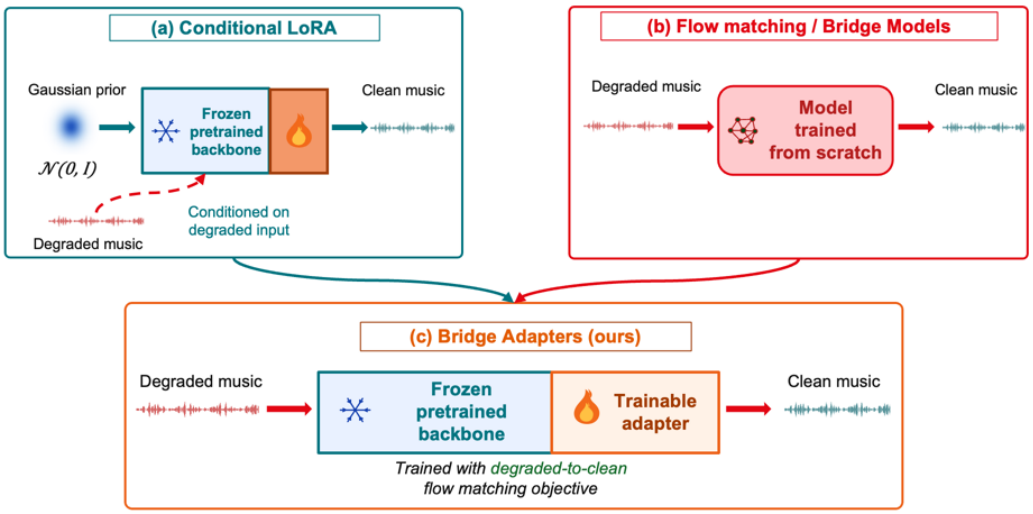

Figure 2: Overview of BRIDGE ADAPTERS. Conditional generation uses a pretrained generator but preserves its Gaussian-to-clean transport, using the degraded signal only as conditioning. Flow-matching and Schrödinger-bridge restoration learn a degraded-to-clean transport from scratch. Our method adapts the transport of a frozen pretrained music generator: the degraded recording becomes the source of the trajectory, and only an adapter is trained to predict the degraded-to-clean restoration field.

3.2 BRIDGE ADAPTERS

Let $v_\theta(z, t)$ denote the vector field of a pretrained music generator, whose native transport maps a simple base distribution, typically Gaussian noise, to the clean music distribution. For paired clean and degraded latents (z_c, z_d), restoration instead requires the transport

$$z_d \to z_c,$$

rather than the native conditional-generation path

$$z_{\text{base}} \to z_c \quad \text{conditioned on } z_d.$$

We implement this source-endpoint change by freezing the pretrained parameters and learning adapters that deform the vector field.

For each degradation family k, we insert a trainable adapter ϕ_k into the frozen backbone. The adapted network defines a new vector field

$$v_{\theta,\phi_k}(z, t),$$

where θ is frozen and only ϕ_k is optimized. The adapter is not an explicit likelihood term and not an additive correction to the final velocity. It changes the internal computation of the pretrained model, and the resulting network defines a different transport. In this sense, BRIDGE ADAPTERS learn a functional deformation

$$v_\theta \mapsto v_{\theta,\phi_k},$$

from a base-to-clean generative field to a degradation-specific degraded-to-clean restoration field. Given a paired latent example $(z_c, z_d) \sim \mathcal{D}_k$, we use the linear bridge

$$z_t = (1 - t)z_c + tz_d, \qquad t \in [0, 1],$$

so that $z_0 = z_c$ and $z_1 = z_d$. The corresponding clean-to-degraded velocity is

$$u^\star = z_d - z_c.$$

We train the adapter by flow matching:

$$\mathcal{L}_k(\phi_k) = \mathbb{E}_{(z_c,z_d)\sim\mathcal{D}_k,\, t\sim U[0,1]} \left[\|v_{\theta,\phi_k}(z_t, t) - (z_d - z_c)\|_2^2 \right]. \tag{1}$$

At inference, we encode the degraded input as z_d, select the adapter for degradation k, initialize

$$z_1 = z_d,$$

and integrate the learned ODE from $t = 1$ to $t = 0$:

$$\frac{dz_t}{dt} = v_{\theta,\phi_k}(z_t, t).$$

The restored audio is obtained by decoding z_0.

Adapter Families The bridge formulation is architecture-agnostic: any parameter-efficient module that turns the frozen backbone into an adapted field v_{θ,ϕ_k} can be used. In this work, we mainly instantiate ϕ_k with LoRA (Hu et al., 2021) and vary its rank and the layers it is applied to, to control adapter capacity. Practically, we keep every pretrained weight fixed and attach LoRA modules to selected linear projections inside the transformer backbone. Each adapted projection computes the original frozen transformation plus a learned low-rank residual update, so loading or switching a restoration behavior only requires loading the corresponding adapter weights. For targeted restoration we train one adapter ϕ_k per degradation family; for blind restoration we train a single adapter shared across all degradation families. We use the LoRA rank and the choice of target layers, e.g., attention projections only or attention plus MLP projections, as the main capacity controls.

4 Experiments

4.1 Experimental Setup

Backbone and training. Unless stated otherwise, Bridge Adapters use ACE-Step 1.5 XL as the frozen backbone (Gong et al., 2025; 2026), with one adapter per degradation family. Our main model, $\mathrm{Ours}_{\mathrm{rk128}}$, uses rank-128 LoRAs ($\alpha = 256$) on the q, k, v, and o projections of every self- and cross-attention block (167.8M trainable parameters per degradation), trained for $25{,}000$ steps (batch 8 per GPU on 4 GPUs, learning rate 10^{-4}); we also evaluate rank 32. The blind variant $\mathrm{Ours}_{\mathrm{blind}}$ is one shared rank-2048 adapter (1.41B parameters, $100{,}000$ steps) trained on all degradations, with no classifier, router, prompt, or label at inference. Inference uses 5 Euler steps unless stated otherwise. Implementation details are in Appendices D and F.

Dataset and protocol. We use the train/test split of the clean excerpts of SonicMaster (Melechovsky et al., 2026) and its 19 degradation functions (reverberation, clipping, bandwidth limitation, compression, equalization, stereo, dynamics, and amplitude). Whereas the released audio combines one to three degradations per clip, we apply each degradation function independently to every clean excerpt with the SonicMaster degradation code (random parameters, fixed seed), giving about $23{,}500$ training and 996 test pairs per family. All models are trained on the training split only and evaluated on all 996 test excerpts of each family (30 s, 44.1 kHz). For latent methods, the degraded waveform is encoded with the VAE in bfloat16, the precision used for the training latents, and restored latents are decoded in float32; decoding precision leaves SNR and SI-SDR unchanged but affects Mel distance.

Baselines. SonicMaster is an all-in-one flow-matching restoration model trained from scratch on this benchmark. We evaluate it without prompt ($\mathrm{SonicMaster}_{\mathrm{blind}}$) and with the correct degradation prompt ($\mathrm{SonicMaster}_{\mathrm{prompt}}$), the closest comparison to our degradation-specific adapters, with 5 Euler steps, the number used by our adapters, and with 10 steps. Numbers of function evaluations are only comparable within an architecture. On the same frozen backbone, we compare with two conditional baselines that keep the Gaussian-to-clean transport and use the degraded audio as conditioning: one ControlNet per degradation (4.378B trainable parameters) and a capacity-matched Conditional LoRA with the same 167.8M-parameter adapter as $\mathrm{Ours}_{\mathrm{rk128}}$ (Appendix F). For these baselines and $\mathrm{Ours}_{\mathrm{blind}}$, SNR, SI-SDR and Mel follow our protocol, whereas FAD was computed on bfloat16-decoded audio.

4.2 Metrics

We measure *paired fidelity* to each clean reference with Mel distance, SNR, and SI-SDR (Roux et al., 2018), and *distributional quality* with the Fréchet Audio Distance (Kilgour et al., 2019) against

the FMA-Pop reference distribution ($\mathrm{FAD}_{\mathrm{FMA}}$) and the Audiobox Aesthetics scores (Tjandra et al., 2025). A FAD against the clean test excerpts themselves ($\mathrm{FAD}_{\mathrm{clean}}$) is reported in the appendix only: it mainly rewards preserving the musical content, so the unrestored input obtains the best value (0.174), and we do not use it to rank restoration quality. The two axes need not agree, as in the perception–distortion trade-off (Blau & Michaeli, 2017). SI-SDR is our primary paired metric, since it best matched our listening impressions, and $\mathrm{FAD}_{\mathrm{FMA}}$ our primary perceptual metric. Definitions, additional diagnostics (STFT, LTAS, LSD), and limitations are given in Appendix C.

4.3 Results

Comparison with SonicMaster. Bridge Adapters and SonicMaster make different design choices, so the comparison should be read in two regimes. First, the degradation-specific adapter results evaluate the controlled setting where the degradation family is known and a corresponding adapter is selected. $\mathrm{SonicMaster}_{\mathrm{prompt}}$ is the closest SonicMaster variant to this controlled setting, because it also receives the degradation identity, but it remains an all-in-one model with one shared set of restoration weights. In this setting, $\mathrm{Ours}_{\mathrm{rk128}}$ improves over $\mathrm{SonicMaster}_{\mathrm{prompt}}$ on SNR, SI-SDR and $\mathrm{FAD}_{\mathrm{FMA}}$, at the same number of five sampling steps (Table 1). It also obtains higher Audiobox Aesthetics scores for content enjoyment (CE 7.34 versus 7.26), content usefulness (CU 7.62 versus 7.54) and production quality (PQ 7.86 versus 7.74; Tables 4 and 10). The SI-SDR gain is observed on 18 of the 19 degradation families and the $\mathrm{FAD}_{\mathrm{FMA}}$ gain on 19 of them. Mel distance is the exception: SonicMaster, whose outputs do not pass through our latent codec, obtains a lower Mel distance. We do not establish the cause of this gap; two plausible contributors are the frozen autoencoder, whose clean round trip alone already has a Mel distance of 0.247 (Table 13), and the sensitivity of the log-mel distance to low-level energy in near-silent time–frequency bins (Appendix C). These results show the benefit of specialization when a degradation label or external selection mechanism is available.

Blind restoration. The fair all-in-one comparison is the blind setting. Our blind adapter uses one shared set of weights for all degradations and no degradation-specific selection at inference, matching SonicMaster's all-in-one protocol more closely; it is built on ACE-Step 1.5 rather than ACE-Step 1.5 XL. It improves over $\mathrm{SonicMaster}_{\mathrm{blind}}$ on SNR, SI-SDR, and $\mathrm{FAD}_{\mathrm{FMA}}$, again with a higher Mel distance, but the gain is smaller than in the degradation-specific setting. We therefore do not interpret $\mathrm{Ours}_{\mathrm{blind}}$ as solving blind restoration; rather, it shows that a single shared bridge adapter already improves over the from-scratch all-in-one baseline without labels (we do not isolate the transport effect in the blind setting, which would require a blind Conditional LoRA), while the gap to the targeted adapters indicates that routing, mixture-of-adapter composition, or degradation estimation remains important future work.

Comparison with conditional control. We also compare against two conditional baselines built on the same frozen ACE-Step 1.5 XL backbone, which keep the Gaussian-to-clean transport and use the degraded recording only as conditioning: a ControlNet with a trainable copy of the transformer, and a capacity-matched Conditional LoRA with the same rank-128 adapter as $\mathrm{Ours}_{\mathrm{rk128}}$ (Appendix F). This comparison isolates the effect of changing the transport from the effect of reusing the pretrained prior. Both conditional baselines reach strong distributional scores, confirming the value of the shared pretrained prior: they obtain a lower $\mathrm{FAD}_{\mathrm{FMA}}$ than $\mathrm{Ours}_{\mathrm{rk128}}$. However, at the same five solver steps, $\mathrm{Ours}_{\mathrm{rk128}}$ is markedly more faithful to the paired reference, with higher SNR and SI-SDR and a lower Mel distance than both baselines, and running ControlNet for 30 steps does not close this gap. We interpret this as a trade-off between set-level and paired metrics: the conditional baselines stay closer to the distribution of clean music, whereas Bridge Adapters better preserve the specific paired target. This is why we report distributional and paired metrics separately, and use SI-SDR as the primary reconstruction metric. Overall, in our setting, adapting the transport itself is more effective than guiding the original base-to-clean flow with a conditional branch when paired fidelity and sampling efficiency are the main objectives.

Per-degradation behaviour. The average gains are not driven by a single degradation (Figure 3 in Appendix B; per-degradation tables in Appendix A). Compared with $\mathrm{SonicMaster}_{\mathrm{prompt}}$, Bridge Adapters lower $\mathrm{FAD}_{\mathrm{FMA}}$ on 19 of the 19 families and raise SI-SDR on 18 of them, by 0.2 to 1.5 dB, with the largest gains (above 1.3 dB) on vocal, volume, and stereo degradations; the only loss

Table 1: Average restoration results over the 19 SonicMaster degradations (996 test excerpts each; per-degradation values, $\mathrm{FAD}_{\mathrm{clean}}$ and Audiobox Aesthetics scores in Appendix A). The first two rows are references without restoration: the degraded input itself, and its encode–decode round trip through the ACE-Step VAE (same encoding and decoding precision as our models). SonicMaster is sampled with 5 Euler steps, the number used by our adapters, and with 10 steps. The FAD values of ControlNet, Conditional LoRA and $\mathrm{Ours}_{\mathrm{blind}}$ were computed on bfloat16-decoded audio. $\mathrm{Ours}_{\mathrm{blind}}$ uses ACE-Step 1.5; all other adapters use ACE-Step 1.5 XL.

Method	**# Steps**	**Mel** ↓	**SNR** ↑	**SI-SDR** ↑	**FAD**$_{\mathrm{FMA}}$ ↓
Degraded input	–	0.568	4.09	6.62	0.343
ACE-Step codec round trip	–	0.658	1.77	-2.35	0.272
$\mathrm{SonicMaster}_{\mathrm{blind}}$	5	0.424	2.30	-0.20	0.252
$\mathrm{SonicMaster}_{\mathrm{prompt}}$	5	**0.413**	2.80	0.30	0.245
$\mathrm{SonicMaster}_{\mathrm{blind}}$	10	0.427	2.25	-0.25	0.256
$\mathrm{SonicMaster}_{\mathrm{prompt}}$	10	0.415	2.76	0.21	0.247
ControlNet	30	0.511	2.32	-0.42	0.204
ControlNet	5	0.509	2.35	-0.54	0.204
Conditional LoRA	5	0.520	2.00	-0.89	**0.200**
$\mathrm{Ours}_{\mathrm{rk128}}$	5	0.482	**3.29**	**1.11**	0.216
$\mathrm{Ours}_{\mathrm{blind}}$	5	0.490	2.79	0.73	0.213

is on real-world effects (-3.7 dB). Severe or ambiguous degradations (small speaker, big reverb, mixture, real-world effects) remain in the low-SI-SDR region for both methods.

For the two primary metrics, we test the improvement over $\mathrm{SonicMaster}_{\mathrm{prompt}}$ family by family, since the degradation families differ strongly in difficulty (the SI-SDR of $\mathrm{Ours}_{\mathrm{rk128}}$ is 1.11 ± 5.32 dB across families). The paired improvement is 0.81 dB in SI-SDR (95% bootstrap confidence interval $[0.25, 1.16]$, exact sign-permutation test $p = 0.0055$) and 0.029 in $\mathrm{FAD}_{\mathrm{FMA}}$ ($[0.026, 0.033]$, $p < 10^{-4}$).

Unrestored input. The degraded input has a higher average SNR and SI-SDR than every method in Table 1. SI-SDR is scale-invariant, so the volume degradation, a pure gain change, scores 84 dB before restoration (2.32 dB over the other 18 families, versus 0.99 dB for $\mathrm{Ours}_{\mathrm{rk128}}$), and mild equalization, clipping, or stereo changes keep the waveform aligned with the reference, whereas generative methods resynthesize the signal. The ACE-Step codec alone lowers the input SI-SDR to -2.35 dB, the starting point of every latent method, which $\mathrm{Ours}_{\mathrm{rk128}}$ improves by about 3.5 dB; where information is destroyed (microphone, real-world effects, mud, vocal, boom), it improves SI-SDR over the input by up to 12 dB, and all methods improve over the input on perceptual metrics. Thus, SI-SDR primarily compares restoration systems to one another here; for mild mastering-style degradations, preserving the input waveform can score better than perceptually correcting it.

5 Ablation Study

5.1 Backbones, Adapter Capacity, and Portability

We applied the same degraded-to-clean Bridge formulation to five music generators with different architectures and codecs, adapting only each model's time parameterization and solver (Appendix D): Stable Audio Open (Evans et al., 2024), ACE-Step 1.5 and 1.5 XL (Gong et al., 2026), Stable Audio 3 (Evans et al., 2026), and the non-autoregressive acoustic stage of YuE2-3B (Yuan et al., 2025; Li et al., 2024) (Table 2). Moving to a new backbone requires no restoration model retraining: each degradation-specific adapter takes only a few hours to train on a single node (e.g. 5.9 h for ACE-Step 1.5 and 7.3 h for ACE-Step 1.5 XL for $25{,}000$ steps on 4 L40S GPUs, and about 1.5 h for YuE2), and the 19 adapters are independent, so they can be trained in parallel.

Within the codec-matched ACE-Step family, the stronger backbone gives more faithful restorations: ACE-Step 1.5 XL improves over ACE-Step 1.5 at equal rank. Both also improve over Stable Audio

Table 2: Bridge Adapters across pretrained backbones and adapter ranks (restoration means over the 19 degradations). Generation FAD is measured on 1000 text-to-music samples of the frozen backbone (FMA-Pop reference; Appendix B). ACE-Step and Stable Audio Open rows use 30 solver steps; YuE2 and Stable Audio 3 use their own codecs and operating points. *attention and MLP projections.

Backbone	Adapter	Gen. FAD ↓	Steps	SNR ↑	SI-SDR ↑	$\mathrm{FAD}_{\mathrm{FMA}}$ ↓
Stable Audio Open	rk32	0.535	30	2.39	-0.92	0.217
ACE-Step 1.5	rk32	0.569	30	2.61	-0.17	0.226
ACE-Step 1.5	rk128	0.569	30	2.80	0.62	0.219
ACE-Step 1.5 XL	rk32	0.488	30	2.92	0.35	0.218
ACE-Step 1.5 XL	rk128	0.488	30	3.11	0.90	0.217
YuE2-3B	rk128	0.411	16	3.88	1.78	0.264
Stable Audio 3	rk128*	0.375	5	5.12	4.18	0.213

Open, which uses a different codec and whose $\mathrm{FAD}_{\mathrm{FMA}}$ nevertheless remains competitive, suggesting that weaker priors still produce plausible music but are less able to stay close to the paired target. Capacity matters as much: rank 128 improves over rank 32 on both ACE-Step backbones, and ACE-Step 1.5 at rank 128 even surpasses XL at rank 32 in SI-SDR. Across the five backbones, the generation FAD of the frozen model (1000 text-to-music samples; Appendix B) is negatively rank-correlated with restoration SI-SDR (Spearman $\rho = -0.90$), but this cross-model comparison is exploratory, since the backbones also differ in codec, architecture, adapter configuration, and number of steps.

Number of solver steps. A few steps suffice (Table 14, Appendix B): among $\{1, 5, 15, 30\}$, five steps give the highest SI-SDR (1.11 dB) and one step the highest SNR, while adding steps beyond five does not improve the averages. We use five steps throughout; the full sweep and step-scheduling strategies are reported in the appendix.

6 Discussion, Limitations, and Conclusion

We introduced Bridge Adapters, which repurpose a frozen pretrained music generator for restoration by adapting its transport: an adapter turns the native Gaussian-to-clean field into a degraded-to-clean field whose source is the degraded latent. On SonicMaster, this is more faithful to the clean reference than a from-scratch all-in-one restoration model and than conditional baselines built on the same backbone, including a capacity-matched Conditional LoRA, with adapter-only training and few solver steps. The same bridge formulation transfers across backbones, and within the codec-matched ACE-Step family the stronger backbone yields more faithful restorations; across heterogeneous backbones, generation FAD and restoration fidelity also show a strong, though exploratory, rank association.

Parameter efficiency and adapter banks. The targeted setting uses one adapter per degradation, i.e. a bank rather than a single model: one 167.8M-parameter adapter is active per input, but the 19 adapters store 3.19B parameters, as does the Conditional LoRA bank, while a ControlNet bank stores $\approx$ 83.2B (Table 17). Our efficiency claim therefore concerns active parameters, per-degradation training cost, and sampling steps, not total storage. A single shared adapter ($\mathrm{Ours}_{\mathrm{blind}}$, 1.41B parameters) removes the bank and still improves over $\mathrm{SonicMaster}_{\mathrm{blind}}$, at lower fidelity than the targeted adapters.

Limitations. The targeted adapters assume a known degradation, and our evaluation is restricted to synthetic single degradations; blind, mixed, and real-world restoration will require routing or composition across adapters and remain open. Some degradations, such as reverberation, remain difficult, and the frozen codec limits what the latent bridge can recover. Beyond music, transport adaptation may apply to speech enhancement, image and video restoration, and source-conditioned editing.

AI USE STATEMENT

In this work, we used generative AI tools (large language models) for writing and debugging code, including training and inference launchers, evaluation and metric pipelines, and the scripts that generate the result tables; for editing the text of the paper (rephrasing, condensing, and checking consistency between the text and the reported numbers); and for identifying candidate related work. We have not used generative AI tools to generate research ideas, to design the method or the experimental protocol, or to produce data, model outputs, or experimental results; no human-subject study was conducted, so disclosures related to participants are not applicable to this work. We have reviewed all AI-assisted work. AI-assisted code was read, run, and checked by the authors; every number in the paper is computed by our evaluation scripts from the saved model outputs, and the tables are generated automatically from these outputs rather than transcribed by hand. Every related work suggested by an AI tool was checked against the original publication by the authors before being cited. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects: listening checks were performed informally by the authors only. All experiments use the publicly released SonicMaster benchmark (Melechovsky et al., 2026) and publicly released pretrained music generators, under their respective licenses, and we do not collect or release new recordings. Generative restoration can alter or invent musical content, so restored recordings should not be presented as faithful to the original performance without disclosure, in particular in archival or documentary use. Restoration tools could also be applied to copyrighted recordings; our adapters do not change the rights attached to the audio they process. The frozen backbones inherit the stylistic biases of their training data, and our distributional metric uses a pop reference set (FMA-Pop), which may favour popular Western music; results on other repertoires may differ. We are not aware of conflicts of interest or sponsorship that bear on this work.

REPRODUCIBILITY STATEMENT

The benchmark, the degradation functions, and the construction of the training and test pairs are described in Section 4.1; the bridge parameterization and conditioning of each backbone in Appendix D; the training hyperparameters and computing resources in Appendix E; the metric definitions and evaluation protocol, including the encoding and decoding precision, in Appendix C; and the conditional baselines in Appendix F. All backbones and the benchmark are publicly available, and all runs use fixed random seeds. Per-degradation results for every reported model are given in Appendix A. We will release the code, the evaluation scripts, and the trained adapters upon publication.

REFERENCES

Yuang Ai, Huaibo Huang, and Ran He. Lora-ir: Taming low-rank experts for efficient all-in-one image restoration, 2024.

Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie S. Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise, 2022.

Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. *CoRR*, abs/1711.06077, 2017.

Yunkee Chae, Junghyun Koo, Sungho Lee, and Kyogu Lee. Exploiting time-frequency conformers for music audio enhancement. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 2362–2370, 2023. doi: 10.1145/3581783.3612269.

Hyungjin Chung, Jong Chul Ye, Peyman Milanfar, and Mauricio Delbracio. Prompt-tuning latent diffusion models for inverse problems, 2023.

Hyungjin Chung, Jeongsol Kim, Michael T. Mccann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems, 2024.

Cem Eteke, Alexander Griessel, Wolfgang Kellerer, and Eckehard Steinbach. Bir-adapter: A parameter-efficient diffusion adapter for blind image restoration. *Pattern Recognition*, 179:113824, November 2026. ISSN 0031-3203. doi: 10.1016/j.patcog.2026.113824.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open, 2024.

Zach Evans, Julian D. Parker, Matthew Rice, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio 3, 2026. URL https://arxiv.org/abs/2605.17991.

Junmin Gong, Sean Zhao, Sen Wang, Shengyuan Xu, and Joe Guo. Ace-step: A step towards music generation foundation model, 2025.

Junmin Gong, Yulin Song, Wenxiao Zhao, Sen Wang, Shengyuan Xu, Jing Guo, and Xuerui Yang. Ace-step 1.5: Pushing the boundaries of open-source music generation, 2026. URL https://arxiv.org/abs/2602.00744.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685, 2021.

Nikhil Kandpal, Oriol Nieto, and Zeyu Jin. Music enhancement via image translation and vocoding. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3124–3128, 2022. doi: 10.1109/ICASSP43922.2022.9747454.

Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet audio distance: A metric for evaluating music enhancement algorithms, 2019.

Zhifeng Kong, Kevin J Shih, Weili Nie, Arash Vahdat, Sang gil Lee, Joao Felipe Santos, Ante Jukic, Rafael Valle, and Bryan Catanzaro. A2sb: Audio-to-audio schrodinger bridges, 2025.

Bunlong Lay, Simon Welker, Julius Richter, and Timo Gerkmann. Reducing the prior mismatch of stochastic differential equations for diffusion-based speech enhancement, 2023.

Jean-Marie Lemercier, Julius Richter, Simon Welker, Eloi Moliner, Vesa Välimäki, and Timo Gerkmann. Diffusion models for audio restoration: A review. *IEEE Signal Processing Magazine*, 41(6):72–84, 2024. doi: 10.1109/MSP.2024.3445871.

Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xingran Chen, Hanzhi Yin, Chenghao Xiao, Chenghua Lin, Anton Ragni, Emmanouil Benetos, Norbert Gyenge, Roger Dannenberg, Ruibo Liu, Wenhu Chen, Gus Xia, Yemin Shi, Wenhao Huang, Zili Wang, Yike Guo, and Jie Fu. Mert: Acoustic music understanding model with large-scale self-supervised training, 2024. URL https://arxiv.org/abs/2306.00107.

Xinqi Lin, Jingwen He, Ziyan Chen, Zhaoyang Lyu, Bo Dai, Fanghua Yu, Wanli Ouyang, Yu Qiao, and Chao Dong. DiffBIR: Toward blind image restoration with generative diffusion prior. In *European Conference on Computer Vision (ECCV)*, 2024.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation, 2024.

Jan Melechovsky, Ambuj Mehrish, Abhinaba Roy, and Dorien Herremans. Sonicmaster: Towards controllable all-in-one music restoration and mastering, 2026.

Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *CoRR*, abs/2108.01073, 2021.

Eloi Moliner, Filip Elvander, and Vesa Välimäki. Blind audio bandwidth extension: A diffusion-based zero-shot approach, 2024a.

Eloi Moliner, Jean-Marie Lemercier, Simon Welker, Timo Gerkmann, and Vesa Välimäki. Buddy: Single-channel blind unsupervised dereverberation with diffusion models, 2024b.

Eloi Moliner, Maija Turunen, Filip Elvander, and Vesa Välimäki. A diffusion-based generative equalizer for music restoration, 2025.

Julius Richter and Timo Gerkmann. Diffusion-based speech enhancement: Demonstration of performance and generalization. In *Audio Imagination: NeurIPS 2024 Workshop AI-Driven Speech, Music, and Sound Generation*, 2024.

Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey. SDR - half-baked or well done? *CoRR*, abs/1811.02508, 2018.

Joan Serrà, Santiago Pascual, Jordi Pons, R. Oguz Araz, and Davide Scaini. Universal speech enhancement with score-based diffusion, 2022.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, Carleigh Wood, Ann Lee, and Wei-Ning Hsu. Meta Audiobox Aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*, 2025.

Thanapat Trachu, Chawan Piansaddhayanon, and Ekapol Chuangsuwanich. Thunder : Unified regression-diffusion speech enhancement with a single reverse step using brownian bridge, 2024.

Bastien van Delft, Tommaso Martorella, and Alexandre Alahi. Code: Confident ordinary differential editing, 2024.

Hanting Wang, Tao Jin, Wang Lin, Shulei Wang, Hai Huang, Shengpeng Ji, and Zhou Zhao. Irbridge: Solving image restoration bridge with pre-trained generative diffusion models, 2025.

Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *International Journal of Computer Vision*, 132(12):5929–5949, 2024.

Ruibin Yuan, Hanfeng Lin, Shuyue Guo, Ge Zhang, Jiahao Pan, Yongyi Zang, Haohe Liu, Yiming Liang, Wenye Ma, Xingjian Du, Xinrun Du, Zhen Ye, Tianyu Zheng, Zhengxuan Jiang, Yinghao Ma, Minghao Liu, Zeyue Tian, Ziya Zhou, Liumeng Xue, Xingwei Qu, Yizhi Li, Shangda Wu, Tianhao Shen, Ziyang Ma, Jun Zhan, Chunhui Wang, Yatian Wang, Xiaowei Chi, Xinyue Zhang, Zhenzhu Yang, Xiangzhou Wang, Shansong Liu, Lingrui Mei, Peng Li, Junjie Wang, Jianwei Yu, Guojian Pang, Xu Li, Zihao Wang, Xiaohuan Zhou, Lijun Yu, Emmanouil Benetos, Yong Chen, Chenghua Lin, Xie Chen, Gus Xia, Zhaoxiang Zhang, Chao Zhang, Wenhu Chen, Xinyu Zhou, Xipeng Qiu, Roger Dannenberg, Jiaheng Liu, Jian Yang, Wenhao Huang, Wei Xue, Xu Tan, and Yike Guo. Yue: Scaling open foundation models for long-form music generation, 2025. URL https://arxiv.org/abs/2503.08638.

Cheng Zhang, Dong Gong, Jiumei He, Yu Zhu, Jinqiu Sun, and Yanning Zhang. Uir-lora: Achieving universal image restoration through multiple low-rank adaptation, 2024.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023a.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning, 2023b.

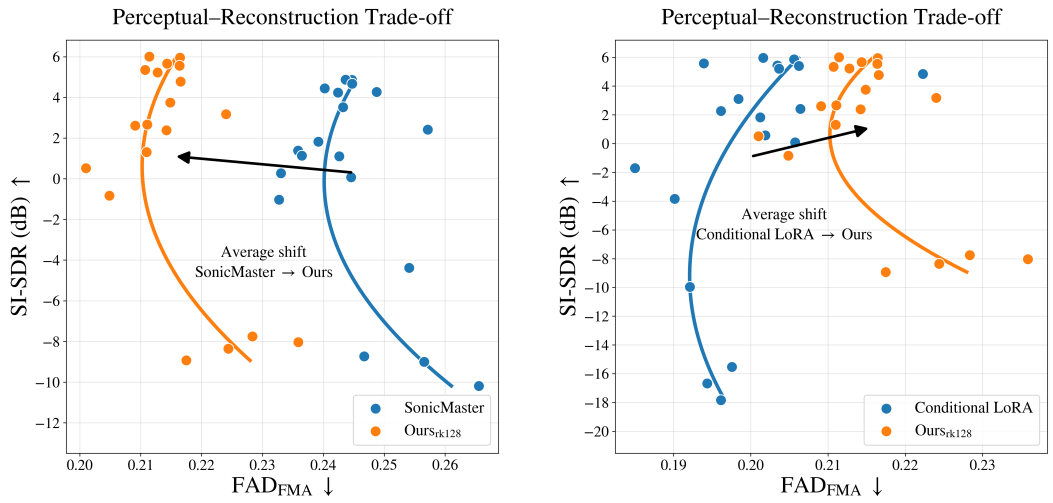

Figure 3: Perceptual–reconstruction trade-off comparison over the 19 SonicMaster degradations, with five solver steps for all systems. Each point corresponds to one degradation; the curves are quadratic fits shown as visual summaries, and the arrow shows the shift of the mean over the 19 degradations. **(a)** Compared with $\text{SonicMaster}_{\text{prompt}}$, BRIDGE ADAPTERS improve perceptual quality (FAD_{FMA}) on 19 and restoration fidelity (SI-SDR) on 18 of the 19 degradations. **(b)** Compared with the capacity-matched Conditional LoRA, BRIDGE ADAPTERS substantially improve restoration fidelity while retaining competitive perceptual quality (Conditional LoRA FAD computed on bfloat16-decoded audio).

A DETAILED METRIC TABLES

Tables 3–12 report the per-degradation values behind the averaged metrics of Tables 1, 2 and 14. All metrics are computed on the SonicMaster test split, i.e. 996 excerpts of 30 seconds per degradation family, under the evaluation protocol of Section 4.1. Each table lists all 19 families and their mean, which is the value reported in the main text. The tables are generated automatically from the metric files of each run.

B ADDITIONAL ANALYSES

B.1 PER-DEGRADATION TRADE-OFF

Figure 3 plots each SonicMaster degradation by paired fidelity (SI-SDR) and perceptual distributional quality (FAD_{FMA}).

B.2 EFFECT OF THE NUMBER OF SOLVER STEPS

Table 14 studies the effect of the number of solver steps at inference. The best average performance is obtained with very few steps, typically between one and five: one step gives the highest SNR, five steps the highest SI-SDR and the lowest Mel distance and $\text{FAD}_{\text{clean}}$. We use five steps in the main experiments. Adding steps beyond five does not improve the average results and slightly lowers SI-SDR. This is not what one would expect from integrating an exact field more accurately; a plausible explanation, which we do not test here, is that errors of the learned field accumulate along longer integrations, while the degraded-to-clean trajectory is short enough to be covered in a few steps.

Although five steps are sufficient on average, the optimal number of steps is not identical for all degradations. We hypothesize that it follows the effective length of the learned restoration bridge. Coloration, equalization, amplitude, and mild dynamics degradations usually need only a short

Table 3: Per-degradation metrics for SonicMaster blind (5 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.407	4.65	4.38	0.341	7.30	7.62	6.19	7.84	0.252
big	0.446	-0.92	-8.74	0.356	7.21	7.41	6.21	7.61	0.248
boom	0.399	1.91	-0.69	0.327	7.26	7.58	6.18	7.79	0.244
bright	0.403	4.51	4.26	0.334	7.30	7.61	6.17	7.83	0.247
clarity	0.441	4.48	4.12	0.317	7.30	7.61	6.16	7.83	0.246
clip	0.417	4.13	4.27	0.331	7.30	7.61	6.19	7.83	0.249
comp	0.417	3.30	3.29	0.337	7.29	7.60	6.19	7.81	0.250
dark	0.394	4.27	3.80	0.331	7.31	7.63	6.22	7.85	0.263
mic	0.400	1.48	-2.11	0.335	7.26	7.56	6.21	7.77	0.240
mix	0.433	-1.20	-9.19	0.361	7.25	7.48	6.17	7.67	0.257
mud	0.390	2.71	0.83	0.338	7.27	7.57	6.24	7.77	0.256
punch	0.393	4.48	3.89	0.333	7.28	7.58	6.20	7.79	0.249
real	0.463	-0.38	-5.37	0.376	7.20	7.32	6.22	7.51	0.258
small	0.430	-1.66	-10.24	0.358	7.25	7.50	6.19	7.70	0.265
stereo	0.424	2.76	2.36	0.340	7.35	7.67	6.15	7.91	0.261
vocal	0.398	2.73	1.06	0.342	7.26	7.53	6.26	7.73	0.254
volume	0.574	2.79	1.11	0.340	7.23	7.52	6.17	7.77	0.254
warm	0.400	1.92	-0.71	0.325	7.29	7.59	6.23	7.82	0.247
xband	0.422	1.80	-0.10	0.336	7.28	7.58	6.20	7.79	0.248
Mean	**0.424**	**2.30**	**-0.20**	**0.340**	**7.27**	**7.56**	**6.20**	**7.77**	**0.252**

Table 4: Per-degradation metrics for SonicMaster prompt (5 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.378	5.70	4.86	0.329	7.28	7.60	6.17	7.82	0.245
big	0.440	-0.78	-8.73	0.356	7.21	7.42	6.20	7.61	0.247
boom	0.380	3.29	1.38	0.331	7.27	7.57	6.17	7.78	0.236
bright	0.380	5.72	4.87	0.333	7.28	7.59	6.17	7.81	0.244
clarity	0.433	5.20	4.27	0.302	7.29	7.60	6.18	7.82	0.249
clip	0.408	4.90	4.67	0.332	7.27	7.58	6.19	7.81	0.245
comp	0.423	3.26	3.52	0.335	7.28	7.58	6.18	7.79	0.243
dark	0.386	4.88	4.44	0.334	7.28	7.59	6.18	7.80	0.240
mic	0.397	1.88	-1.03	0.335	7.25	7.54	6.19	7.74	0.233
mix	0.429	-1.10	-9.00	0.361	7.25	7.49	6.17	7.67	0.257
mud	0.384	3.01	1.10	0.339	7.24	7.53	6.22	7.72	0.243
punch	0.375	5.31	4.24	0.335	7.28	7.59	6.18	7.80	0.242
real	0.457	0.02	-4.38	0.375	7.18	7.30	6.20	7.48	0.254
small	0.424	-1.63	-10.18	0.358	7.26	7.52	6.19	7.72	0.266
stereo	0.404	3.53	2.41	0.339	7.31	7.64	6.14	7.86	0.257
vocal	0.397	2.89	1.14	0.344	7.22	7.49	6.24	7.67	0.236
volume	0.550	2.65	1.82	0.325	7.21	7.50	6.17	7.72	0.239
warm	0.382	2.60	0.27	0.331	7.26	7.55	6.18	7.77	0.233
xband	0.418	1.94	0.08	0.337	7.28	7.57	6.21	7.77	0.245
Mean	**0.413**	**2.80**	**0.30**	**0.338**	**7.26**	**7.54**	**6.19**	**7.74**	**0.245**

Table 5: Per-degradation metrics for Stable Audio rk32 (30 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.491	5.67	4.82	0.398	7.28	7.58	6.16	7.79	0.208
big	0.537	-0.98	-8.81	0.474	6.99	7.05	6.25	7.29	0.243
boom	0.481	2.88	1.14	0.399	7.23	7.50	6.18	7.72	0.202
bright	0.485	5.62	4.83	0.397	7.28	7.57	6.16	7.78	0.208
clarity	0.529	5.08	4.20	0.413	7.27	7.53	6.19	7.76	0.218
clip	0.507	4.17	4.59	0.401	7.24	7.51	6.18	7.72	0.209
comp	0.511	3.39	3.00	0.405	7.25	7.53	6.18	7.75	0.200
dark	0.500	4.37	4.42	0.398	7.26	7.55	6.17	7.76	0.205
mic	0.515	0.19	-5.78	0.420	7.13	7.35	6.26	7.55	0.198
mix	0.530	-1.13	-9.63	0.456	6.99	7.07	6.23	7.27	0.225
mud	0.503	1.99	-0.11	0.428	7.13	7.31	6.28	7.51	0.211
punch	0.474	5.42	4.28	0.397	7.26	7.55	6.16	7.76	0.201
real	0.574	-2.35	-17.35	0.559	6.55	6.59	6.35	6.90	0.306
small	0.518	-1.06	-10.41	0.452	7.05	7.20	6.25	7.41	0.228
stereo	0.497	3.54	2.49	0.410	7.27	7.53	6.20	7.76	0.217
vocal	0.492	2.83	0.66	0.440	7.11	7.26	6.30	7.47	0.223
volume	0.544	2.40	1.85	0.429	7.11	7.34	6.27	7.56	0.223
warm	0.484	2.18	-0.43	0.412	7.18	7.41	6.22	7.64	0.199
xband	0.529	1.21	-1.15	0.417	7.16	7.39	6.23	7.60	0.200
Mean	**0.510**	**2.39**	**-0.92**	**0.427**	**7.14**	**7.36**	**6.22**	**7.58**	**0.217**

Table 6: Per-degradation metrics for ACE-Step 1.5 rk32 (30 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.485	5.84	6.04	0.270	7.41	7.73	6.19	7.99	0.223
big	0.516	-0.91	-8.75	0.361	7.01	7.11	6.25	7.38	0.247
boom	0.463	3.21	1.67	0.294	7.39	7.68	6.22	7.94	0.210
bright	0.469	6.19	5.96	0.274	7.40	7.71	6.19	7.96	0.218
clarity	0.510	4.63	4.18	0.281	7.39	7.70	6.20	7.95	0.240
clip	0.480	5.10	5.53	0.300	7.39	7.66	6.22	7.93	0.214
comp	0.511	3.08	3.71	0.298	7.39	7.70	6.21	7.97	0.214
dark	0.468	5.16	5.33	0.296	7.40	7.70	6.21	7.96	0.210
mic	0.507	0.30	-3.66	0.304	7.35	7.61	6.28	7.84	0.200
mix	0.554	-1.72	-11.65	0.479	6.69	6.75	5.84	7.00	0.326
mud	0.469	3.21	1.43	0.316	7.36	7.61	6.28	7.86	0.214
punch	0.486	4.85	3.77	0.314	7.23	7.56	6.25	7.77	0.209
real	0.556	-1.95	-13.45	0.413	6.74	6.81	6.42	7.17	0.278
small	0.502	-1.32	-10.56	0.346	7.10	7.31	6.19	7.53	0.232
stereo	0.476	3.78	3.12	0.311	7.40	7.70	6.25	7.97	0.224
vocal	0.474	3.42	2.07	0.320	7.34	7.57	6.31	7.82	0.215
volume	0.544	2.90	1.89	0.301	6.86	7.29	6.29	7.61	0.212
warm	0.468	2.25	0.05	0.303	7.39	7.63	6.28	7.90	0.207
xband	0.510	1.60	0.04	0.299	7.34	7.60	6.29	7.84	0.199
Mean	**0.497**	**2.61**	**-0.17**	**0.320**	**7.24**	**7.50**	**6.23**	**7.76**	**0.226**

Table 7: Per-degradation metrics for ACE-Step 1.5 rk128 (30 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.462	6.48	5.94	0.293	7.40	7.72	6.18	7.98	0.215
big	0.511	-1.01	-8.14	0.341	7.16	7.34	6.26	7.56	0.233
boom	0.463	3.40	1.94	0.294	7.39	7.69	6.22	7.94	0.210
bright	0.483	5.91	6.04	0.262	7.40	7.71	6.19	7.96	0.221
clarity	0.497	5.86	5.32	0.282	7.38	7.67	6.22	7.93	0.221
clip	0.491	4.56	5.62	0.296	7.38	7.67	6.22	7.93	0.214
comp	0.503	3.47	4.17	0.292	7.39	7.70	6.22	7.96	0.214
dark	0.466	5.36	5.34	0.295	7.40	7.70	6.20	7.96	0.210
mic	0.494	1.04	-2.02	0.305	7.36	7.62	6.28	7.86	0.206
mix	0.501	-1.03	-8.71	0.332	7.21	7.43	6.26	7.63	0.226
mud	0.471	3.27	1.80	0.308	7.36	7.62	6.27	7.87	0.212
punch	0.483	5.02	3.75	0.308	7.22	7.55	6.25	7.76	0.203
real	0.551	-1.62	-9.43	0.372	7.03	7.10	6.44	7.36	0.258
small	0.508	-1.65	-9.15	0.323	7.23	7.46	6.26	7.67	0.224
stereo	0.475	4.04	3.22	0.304	7.39	7.70	6.24	7.96	0.220
vocal	0.472	3.75	2.49	0.313	7.35	7.59	6.29	7.83	0.215
volume	0.547	1.82	2.66	0.312	7.26	7.51	6.34	7.78	0.242
warm	0.462	2.67	0.63	0.300	7.39	7.66	6.25	7.92	0.209
xband	0.502	1.75	0.31	0.301	7.35	7.61	6.27	7.85	0.202
Mean	**0.492**	**2.80**	**0.62**	**0.307**	**7.32**	**7.58**	**6.26**	**7.83**	**0.219**

Table 8: Per-degradation metrics for ACE-Step 1.5 XL rk32 (30 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.468	6.20	5.94	0.287	7.40	7.72	6.19	7.98	0.219
big	0.499	-0.49	-7.92	0.350	7.15	7.30	6.23	7.54	0.236
boom	0.463	3.57	2.17	0.298	7.39	7.69	6.21	7.95	0.211
bright	0.464	6.37	5.93	0.286	7.40	7.72	6.18	7.97	0.216
clarity	0.499	5.91	5.45	0.283	7.38	7.67	6.21	7.93	0.223
clip	0.482	4.88	5.54	0.300	7.39	7.67	6.22	7.92	0.214
comp	0.500	3.63	4.21	0.293	7.40	7.71	6.22	7.96	0.212
dark	0.466	5.23	5.35	0.295	7.40	7.70	6.21	7.96	0.212
mic	0.495	0.54	-3.58	0.306	7.35	7.62	6.26	7.86	0.202
mix	0.496	-0.82	-8.67	0.340	7.20	7.41	6.25	7.62	0.228
mud	0.471	3.37	1.84	0.311	7.36	7.62	6.28	7.87	0.215
punch	0.482	5.01	3.77	0.309	7.22	7.55	6.25	7.76	0.204
real	0.536	-1.49	-13.04	0.395	6.86	6.94	6.42	7.25	0.257
small	0.500	-1.23	-9.54	0.322	7.20	7.41	6.24	7.62	0.217
stereo	0.473	4.01	3.40	0.309	7.39	7.70	6.24	7.96	0.222
vocal	0.480	3.38	2.22	0.316	7.34	7.58	6.30	7.82	0.214
volume	0.514	3.38	2.82	0.307	7.27	7.55	6.31	7.83	0.231
warm	0.474	2.34	0.78	0.301	7.39	7.65	6.25	7.91	0.211
xband	0.504	1.75	-0.09	0.303	7.34	7.60	6.26	7.86	0.198
Mean	**0.488**	**2.92**	**0.35**	**0.311**	**7.31**	**7.57**	**6.25**	**7.82**	**0.218**

Table 9: Per-degradation metrics for ACE-Step 1.5 XL rk128 (30 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.465	6.33	5.91	0.290	7.40	7.72	6.18	7.98	0.217
big	0.501	-0.66	-7.82	0.331	7.20	7.38	6.23	7.60	0.227
boom	0.462	3.82	2.56	0.295	7.39	7.70	6.21	7.95	0.212
bright	0.465	6.44	5.95	0.286	7.40	7.70	6.19	7.96	0.212
clarity	0.494	5.99	5.47	0.301	7.39	7.68	6.21	7.94	0.221
clip	0.480	4.94	5.62	0.295	7.39	7.68	6.21	7.93	0.214
comp	0.497	3.76	4.67	0.290	7.39	7.70	6.21	7.95	0.217
dark	0.469	5.10	5.31	0.294	7.40	7.70	6.20	7.96	0.211
mic	0.493	0.93	-2.09	0.302	7.35	7.63	6.27	7.87	0.207
mix	0.499	-1.09	-8.42	0.330	7.26	7.48	6.27	7.68	0.226
mud	0.467	3.67	2.18	0.308	7.37	7.64	6.25	7.89	0.215
punch	0.460	5.87	5.20	0.292	7.39	7.69	6.19	7.95	0.210
real	0.526	-0.95	-9.36	0.379	7.06	7.16	6.41	7.44	0.251
small	0.494	-1.11	-8.99	0.317	7.26	7.49	6.25	7.69	0.215
stereo	0.467	4.31	3.67	0.306	7.40	7.70	6.23	7.96	0.216
vocal	0.482	3.50	2.51	0.309	7.34	7.59	6.29	7.83	0.214
volume	0.506	3.60	3.08	0.305	7.34	7.60	6.30	7.86	0.231
warm	0.470	2.66	1.14	0.298	7.39	7.67	6.23	7.92	0.211
xband	0.499	2.02	0.43	0.303	7.34	7.60	6.26	7.84	0.201
Mean	**0.484**	**3.11**	**0.90**	**0.307**	**7.34**	**7.61**	**6.24**	**7.85**	**0.217**

Table 10: Per-degradation metrics for $\text{Ours}_{\text{rk128}}$ (ACE-Step 1.5 XL rk128, 5 steps). Lower is better for Mel, $\text{FAD}_{\text{clean}}$ and FAD_{FMA}; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
airy	0.465	6.33	5.95	0.287	7.40	7.72	6.18	7.98	0.216
big	0.494	-0.41	-7.75	0.333	7.22	7.44	6.19	7.65	0.228
boom	0.461	3.95	2.67	0.294	7.38	7.69	6.20	7.95	0.211
bright	0.465	6.49	6.00	0.284	7.39	7.70	6.18	7.96	0.211
clarity	0.491	6.09	5.55	0.285	7.38	7.68	6.19	7.94	0.216
clip	0.482	4.90	5.67	0.292	7.39	7.68	6.20	7.94	0.214
comp	0.493	3.94	4.77	0.290	7.39	7.70	6.21	7.95	0.217
dark	0.465	5.20	5.35	0.294	7.40	7.70	6.20	7.96	0.211
mic	0.491	1.58	-0.84	0.300	7.33	7.63	6.25	7.86	0.205
mix	0.496	-0.93	-8.35	0.329	7.27	7.51	6.24	7.69	0.224
mud	0.466	3.81	2.38	0.307	7.37	7.66	6.23	7.90	0.214
punch	0.459	5.82	5.23	0.294	7.40	7.70	6.19	7.96	0.213
real	0.520	-0.22	-8.03	0.368	7.11	7.22	6.35	7.42	0.236
small	0.488	-0.86	-8.93	0.322	7.26	7.52	6.24	7.72	0.217
stereo	0.465	4.41	3.75	0.306	7.40	7.71	6.22	7.97	0.215
vocal	0.481	3.62	2.61	0.307	7.34	7.62	6.26	7.85	0.209
volume	0.501	3.83	3.18	0.295	7.34	7.61	6.27	7.85	0.224
warm	0.471	2.75	1.31	0.299	7.38	7.67	6.22	7.92	0.211
xband	0.497	2.14	0.52	0.302	7.34	7.61	6.25	7.85	0.201
Mean	**0.482**	**3.29**	**1.11**	**0.305**	**7.34**	**7.62**	**6.22**	**7.86**	**0.216**

Table 11: Per-degradation metrics for ControlNet (ACE-Step 1.5 XL, 30 steps). FAD values use bfloat16-decoded audio. Lower is better for Mel, $\mathrm{FAD}_{\mathrm{clean}}$ and $\mathrm{FAD}_{\mathrm{FMA}}$; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\mathrm{FAD}_{\mathrm{clean}}$ ↓	$\mathrm{FAD}_{\mathrm{FMA}}$ ↓
airy	0.466	6.12	5.90	0.294	0.203
big	0.538	-2.06	-15.06	0.307	0.212
boom	0.470	3.26	2.26	0.294	0.205
bright	0.466	6.34	5.78	0.287	0.208
clarity	0.521	5.38	5.04	0.290	0.217
clip	0.504	3.89	5.16	0.291	0.204
comp	0.498	3.91	4.91	0.290	0.205
dark	0.498	4.00	5.44	0.292	0.199
mic	0.530	1.18	-0.52	0.298	0.196
mix	0.540	-2.19	-15.59	0.298	0.202
mud	0.485	3.09	2.38	0.293	0.206
punch	0.461	5.74	5.22	0.291	0.207
real	0.591	-1.60	-8.49	0.300	0.201
small	0.533	-2.43	-16.94	0.295	0.200
stereo	0.493	3.08	2.06	0.292	0.204
vocal	0.500	2.82	2.52	0.294	0.202
volume	0.573	1.55	2.61	0.290	0.202
warm	0.484	1.68	0.83	0.295	0.204
xband	0.563	0.42	-1.45	0.306	0.193
Mean	**0.511**	**2.32**	**-0.42**	**0.295**	**0.204**

Table 12: Per-degradation metrics for $\mathrm{Ours}_{\mathrm{blind}}$ (ACE-Step 1.5 rk2048, 5 steps). FAD values use bfloat16-decoded audio. Lower is better for Mel, $\mathrm{FAD}_{\mathrm{clean}}$ and $\mathrm{FAD}_{\mathrm{FMA}}$; higher is better for the other metrics.

Degradation	Mel ↓	SNR ↑	SI-SDR ↑	$\mathrm{FAD}_{\mathrm{clean}}$ ↓	$\mathrm{FAD}_{\mathrm{FMA}}$ ↓
airy	0.475	5.71	5.68	0.300	0.205
big	0.504	-0.43	-7.70	0.320	0.230
boom	0.469	2.75	0.76	0.292	0.207
bright	0.476	5.77	5.80	0.288	0.203
clarity	0.498	5.70	5.28	0.317	0.207
clip	0.508	3.67	5.61	0.293	0.216
comp	0.506	3.29	4.31	0.294	0.213
dark	0.481	4.74	4.85	0.289	0.222
mic	0.481	1.50	-1.98	0.300	0.202
mix	0.497	-0.92	-8.71	0.325	0.222
mud	0.475	3.23	1.97	0.308	0.211
punch	0.466	5.52	5.09	0.291	0.206
real	0.526	-0.67	-7.30	0.361	0.235
small	0.495	-1.10	-9.03	0.315	0.224
stereo	0.482	3.73	3.37	0.301	0.212
vocal	0.472	3.75	2.57	0.308	0.206
volume	0.533	2.46	2.75	0.313	0.219
warm	0.473	2.36	0.20	0.293	0.206
xband	0.498	1.99	0.40	0.298	0.197
Mean	**0.490**	**2.79**	**0.73**	**0.306**	**0.213**

Table 13: Clean reconstruction metrics for the pretrained autoencoders, computed between the clean test waveforms and their encode–decode round trip with the autoencoder in float32. Lower is better for LSD, Mel, and STFT; higher is better for SI-SDR and SNR.

Autoencoder	LSD ↓	Mel ↓	SI-SDR ↑	SNR ↑	STFT ↓
ACE-Step VAE	0.568	0.247	5.88	6.61	4788.29
Stable Audio VAE	0.683	0.371	4.93	5.88	5171.13

Table 14: Effect of the number of ODE solver steps at inference for $\text{Ours}_{\text{rk128}}$, averaged over the 19 degradations. All rows use the same trained adapters.

Method	# Steps	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
$\text{Ours}_{\text{rk128}}$	30	0.484	3.11	0.90	0.307	**7.34**	7.61	**6.24**	7.85	0.217
$\text{Ours}_{\text{rk128}}$	15	0.484	3.14	0.95	0.306	**7.34**	7.61	**6.24**	**7.86**	0.217
$\text{Ours}_{\text{rk128}}$	5	**0.482**	3.29	**1.11**	**0.305**	**7.34**	**7.62**	6.22	**7.86**	0.216
$\text{Ours}_{\text{rk128}}$	1	0.485	**3.54**	0.87	0.323	7.26	7.55	6.13	7.74	**0.213**

Table 15: Generation quality of each frozen base model (text-to-music from Gaussian noise, 1000 samples of 10 s prompted with SonicMaster test tags, peak-normalized) versus the restoration quality of BRIDGE ADAPTERS trained on it (means over the 19 degradations). Stable Audio 3 and YuE2 use their own codecs and the operating points indicated.

	Generation			Restoration			
Backbone	FAD ↓	PQ ↑	Adapter	SI-SDR ↑	Mel ↓	PQ ↑	FAD_{FMA} ↓
Stable Audio Open	0.535	7.24	rk32, 30 steps	-0.92	0.510	7.58	0.217
ACE-Step 1.5	0.569	7.87	rk128, 30 steps	0.62	0.492	7.83	0.219
ACE-Step 1.5 XL	0.488	8.02	rk128, 30 steps	0.90	0.484	7.85	0.217
YuE2-3B	0.411	7.58	rk128, 16 steps	1.78	0.294	7.87	0.264
Stable Audio 3	0.375	7.86	rk128 attn+MLP, 5 steps	4.18	0.594	7.77	0.213

trajectory because the degraded latent is already close to the clean target. More destructive or ambiguous degradations, such as microphone, small-speaker, mixture, and real-world effects, can benefit from additional solver steps because the bridge must reconstruct missing spectral content or undo stronger nonlinear changes. This suggests two simple scheduling strategies. In the targeted setting, where the degradation identity is known, one can select a per-degradation step budget from a small validation sweep and store it together with the corresponding adapter; for example, choose the smallest budget in $\{1, 5, 15, 30\}$ whose validation SI-SDR is within a tolerance of the best budget for that degradation. In the blind setting, a practical alternative is adaptive early stopping: run the solver until the relative latent update $\|x_{t+\Delta t} - x_t\|_2 / \|x_t\|_2$, or a cheap spectral-change proxy such as the change in log-mel energy between consecutive decoded estimates, falls below a threshold, with a maximum budget for severe inputs. We do not use such dynamic scheduling in the main results in order to keep the comparison simple and fixed-budget; all main BRIDGE ADAPTERS results use five steps. Appendix F evaluates such a calibration-based lookup scheduler.

B.3 GENERATION QUALITY VERSUS RESTORATION QUALITY

Generation quality of the frozen base model. We sample $1{,}000$ excerpts of 10 s from Gaussian noise with the text-to-music pipeline of each model, prompted with tag descriptions built from the genres, instrument tags, and vocal attributes of the SonicMaster test tracks, with no lyrics and the default sampler of each model. All outputs are peak-normalized to -1 dBFS before evaluation. We report FAD against the FMA-Pop reference set and the Audiobox Aesthetics production quality (PQ).

Restoration quality of its Bridge Adapters. We report the restoration metrics of the adapters trained on each backbone, averaged over the 19 degradations. Stable Audio 3 and YuE2 use their own codecs and the operating points indicated in Table 15.

In this small and heterogeneous set of backbones, generation FAD is negatively rank-correlated with restoration SI-SDR (Spearman $\rho = -0.90$; Table 15): a lower generation FAD goes with a higher restoration SI-SDR. The two models with the lowest generation FAD, Stable Audio 3 and YuE2, also give the most faithful restorations, followed by ACE-Step 1.5 XL; the only inversion concerns the two models with the highest generation FAD, Stable Audio Open and ACE-Step 1.5. Other generation metrics do not order the backbones in the same way (e.g. generation PQ), the correlation is weaker for the distributional metrics of the restored audio (0.30 between generation FAD and restoration FAD_{FMA}), and Mel distances are not comparable across codecs. We therefore consider this cross-model comparison exploratory: the backbones differ in codec, architecture, adapter rank and target layers, and number of solver steps, and five models are too few for a statistical test. The

Table 16: Comparison of pretrained backbones and adapter ranks, averaged over the 19 degradations. All rows use 30 inference steps.

Method	Mel ↓	SNR ↑	SI-SDR ↑	$\text{FAD}_{\text{clean}}$ ↓	CE ↑	CU ↑	PC ↑	PQ ↑	FAD_{FMA} ↓
Stable Audio – rk32	0.510	2.39	-0.92	0.427	7.14	7.36	6.22	7.58	**0.217**
ACE-Step 1.5 – rk32	0.497	2.61	-0.17	0.320	7.24	7.50	6.23	7.76	0.226
ACE-Step 1.5 – rk128	0.492	2.80	0.62	**0.307**	7.32	7.58	**6.26**	7.83	0.219
ACE-Step 1.5 XL – rk32	0.488	2.92	0.35	0.311	7.31	7.57	6.25	7.82	0.218
ACE-Step 1.5 XL – rk128	**0.484**	**3.11**	**0.90**	**0.307**	**7.34**	**7.61**	6.24	**7.85**	**0.217**

Table 17: Parameter and compute accounting of the adaptation methods built on the frozen ACE-Step backbones, with their restoration results at five solver steps (means over the 19 degradations; all columns except the last two are independent of the degradation family). "Stored" counts the trainable parameters needed to cover the 19 SonicMaster degradations. The FAD_{FMA} of ControlNet, Conditional LoRA and $\text{Ours}_{\text{blind}}$ was computed on bfloat16-decoded audio.

Method	Backbone	Active params.	Stored params.	Training steps	SI-SDR ↑	FAD_{FMA} ↓
ControlNet	ACE-Step 1.5 XL	4.378B	83.2B	$19 \times 25{,}000$	-0.54	0.204
Conditional LoRA	ACE-Step 1.5 XL	167.8M	3.19B	$19 \times 25{,}000$	-0.89	0.200
$\text{Ours}_{\text{rk128}}$	ACE-Step 1.5 XL	167.8M	3.19B	$19 \times 25{,}000$	1.11	0.216
$\text{Ours}_{\text{blind}}$	ACE-Step 1.5	1.41B	1.41B	100,000	0.73	0.213

cleaner evidence is the codec-matched comparison of Section 5.1 (full metrics in Table 16), where ACE-Step 1.5 XL gives more faithful restorations than ACE-Step 1.5 at equal adapter rank. Both results are consistent with our hypothesis that a BRIDGE ADAPTER reuses the pretrained prior.

B.4 PARAMETER ACCOUNTING

C METRIC DETAILS

We evaluate restoration with two complementary families of metrics: reference-based reconstruction metrics and reference-free or distributional perceptual metrics. The former measure closeness to the paired clean target, while the latter measure whether the restored audio resembles high-quality music as a distribution. Neither family is sufficient on its own.

C.1 RECONSTRUCTION METRICS

Mel distance. Mel distance measures the discrepancy between the mel-spectrogram of the restored audio and that of the clean reference. Given a restored waveform $\hat{x}$ and clean reference x at 44.1 kHz, we compute magnitude mel-spectrograms $M(\cdot)$ with 80 mel bands, $n_{\text{fft}} = 2048$ and hop 512, per channel, and report the root-mean-square difference of their base-10 logarithms, floored at 10^{-5}:

$$d_{\text{mel}}(\hat{x}, x) = \sqrt{\text{mean}_{f,t}\Big(\log_{10}\max\big(M(\hat{x}), 10^{-5}\big) - \log_{10}\max\big(M(x), 10^{-5}\big)\Big)^2},$$

averaged over channels. Because of the logarithm, the metric is sensitive to low-level energy in time–frequency bins where the reference is near silent; for instance, lower-precision decoding of the same latents increases it without changing SNR or SI-SDR. Mel distance is useful because it compares audio in a perceptually motivated time–frequency representation. However, it is still a pointwise reference metric: small phase, alignment, or amplitude differences can affect the score, and a lower mel distance does not always imply better perceived quality.

Multi-resolution STFT distance. We report a multi-resolution STFT distance between restored and clean waveforms. For a set of FFT/window configurations $\mathcal{R}$, this metric compares spectral magnitudes across several time–frequency resolutions:

$$d_{\text{STFT}}(\hat{x}, x) = \frac{1}{|\mathcal{R}|}\sum_{r\in\mathcal{R}} \big\| |\text{STFT}_r(\hat{x})| - |\text{STFT}_r(x)| \big\|.$$

Using multiple resolutions makes the metric sensitive to both short transients and longer harmonic structure. As with mel distance, STFT distance remains a paired reconstruction metric and may penalize perceptually acceptable alternatives.

SNR. The signal-to-noise ratio measures the energy of the clean reference relative to the residual error:

$$\mathrm{SNR}(\hat{x}, x) = 10 \log_{10} \frac{\|x\|_2^2}{\|x - \hat{x}\|_2^2}.$$

Higher SNR indicates better waveform-level reconstruction. SNR is simple and interpretable, but it is sensitive to gain differences and does not correlate perfectly with perceived audio quality, especially for music.

SI-SDR. Scale-invariant signal-to-distortion ratio removes global scale before measuring distortion (Roux et al., 2018). Let

$$x_{\mathrm{target}} = \frac{\langle \hat{x}, x \rangle}{\|x\|_2^2} x, \qquad e = \hat{x} - x_{\mathrm{target}}.$$

Then

$$\mathrm{SI\text{-}SDR}(\hat{x}, x) = 10 \log_{10} \frac{\|x_{\mathrm{target}}\|_2^2}{\|e\|_2^2}.$$

SI-SDR is less sensitive to global loudness mismatch than SNR and is widely used in speech separation and enhancement. For music restoration, it remains useful as a fidelity metric but may under-reward perceptually plausible restorations that differ slightly from the reference.

LTAS distance. Long-term average spectrum (LTAS) compares the average spectral envelope of the restored signal and clean reference over time. If $S_x(f, t)$ denotes the magnitude spectrum, the LTAS is

$$\mathrm{LTAS}_x(f) = \frac{1}{T} \sum_{t=1}^{T} S_x(f, t).$$

We then compute a distance between $\mathrm{LTAS}_{\hat{x}}$ and LTAS_x. LTAS is useful for detecting global spectral coloration, such as overly dark, bright, or bandwidth-limited restorations. However, it averages over time and is therefore insensitive to temporal artifacts, transient smearing, and local musical errors.

LSD. The log-spectral distance (LSD) compares the log-magnitude spectra of the restored audio and the clean reference frame by frame. With $S_x(k, t) = |\mathrm{STFT}(x)|$ computed with a Hann window ($n_{\mathrm{fft}} = 2048$, hop 512),

$$\mathrm{LSD}(\hat{x}, x) = \frac{1}{T} \sum_{t=1}^{T} \sqrt{\frac{1}{K} \sum_{k=1}^{K} \big(\log_{10}(S_{\hat{x}}(k, t) + \epsilon) - \log_{10}(S_x(k, t) + \epsilon)\big)^2}, \qquad \epsilon = 10^{-8}.$$

LSD is sensitive to spectral coloration and missing bandwidth, but, being computed on log magnitudes, it is also strongly affected by differences in near-silent time–frequency bins and ignores phase. It should therefore be interpreted together with STFT distance and with perceptual metrics such as FAD.

Clean-set FAD ($\mathrm{FAD}_{\mathrm{clean}}$). Fréchet Audio Distance (FAD) compares two sets of audio samples by embedding them with a pretrained audio model and computing the Fréchet distance between Gaussian fits to the embedding distributions (Kilgour et al., 2019). For $\mathrm{FAD}_{\mathrm{clean}}$, we compute FAD between the restored set and its corresponding clean-reference set. If (μ_r, Σ_r) and (μ_c, Σ_c) are the embedding mean and covariance of the restored and clean sets, then

$$\mathrm{FAD} = \|\mu_r - \mu_c\|_2^2 + \mathrm{Tr}\left(\Sigma_r + \Sigma_c - 2(\Sigma_r \Sigma_c)^{1/2}\right).$$

Lower FAD indicates that the restored set is closer to the clean set in the embedding space. Unlike mel distance, SNR, or SI-SDR, FAD is distributional and not pairwise. It is therefore useful for measuring aggregate realism and distributional alignment, but it does not guarantee that each restored sample is faithful to its paired reference.

C.2 Perceptual Metrics

FMA-Pop FAD ($\mathrm{FAD}_{\mathrm{FMA}}$). We also report a reference-distribution FAD, denoted $\mathrm{FAD}_{\mathrm{FMA}}$, where restored samples are compared against an external high-quality music reference distribution. In our experiments, this reference is FMA-Pop. This metric measures whether the restored audio resembles realistic music globally, rather than whether it matches a specific paired target. It is therefore complementary to reconstruction metrics. A model may obtain good $\mathrm{FAD}_{\mathrm{FMA}}$ while changing the content of the input; for this reason, we always report it together with paired fidelity metrics.

Audiobox Aesthetics. We use the Meta Audiobox Aesthetics toolbox to estimate perceptual quality along multiple dimensions, including content enjoyment, content usefulness, production quality, and production complexity (Tjandra et al., 2025). These scores provide a learned estimate of subjective audio quality across speech, music, and sound. They are useful for detecting improvements that may not be captured by pointwise spectral distances. However, as learned no-reference metrics, they may inherit biases from their training data and should not be treated as a substitute for listening tests.

C.3 Interpreting the Trade-off

Restoration metrics should be interpreted as a trade-off rather than as a single scalar ranking. Paired metrics such as Mel distance, STFT distance, SNR, and SI-SDR measure faithfulness to the reference. Perceptual and distributional metrics such as $\mathrm{FAD}_{\mathrm{FMA}}$ and Audiobox Aesthetics measure whether the restored signal sounds like high-quality music. Improving one family does not necessarily improve the other. This is the audio analogue of the perception–distortion trade-off (Blau & Michaeli, 2017): a restoration system should ideally move the Pareto frontier, improving realism without sacrificing fidelity to the degraded input.

C.4 Latent Restoration Setting

Let (x_c, x_d) be a paired clean/degraded waveform and let E, D be a frozen audio encoder and decoder respectively. Latent Restoration aims to reconstruct the clean latent $z_c = E(x_c)$ given the degraded latent $z_d = E(x_d)$. The restored latent $\hat{z}_c$ is then decoded back into a waveform $\hat{x}_c$. Since D is fixed, the codec constrains the space of outputs and discards information, and the clean autoencoding output $D(E(x_c))$ provides a practical codec reference point. It is not a strict bound: a restored latent $\hat{z}_c \neq E(x_c)$ may decode closer to x_c than the clean round trip. We report the reconstruction metrics of the autoencoders in Table 13.

The codec also limits restoration of very quiet inputs. The `volume` degradation peak-normalizes the clean excerpt and scales it by $a \in \{0.001, 0.003, 0.01, 0.05\}$. An encode–decode round trip of the degraded signal alone, scored against that same signal, reaches an SI-SDR of -15.3 dB at $a = 0.001$ and -6.2 dB at $a = 0.003$, against 1.7 dB at $a = 0.05$ and -0.4 to 2.5 dB for equalization and clipping degradations: below about -50 dBFS, the decoded signal is dominated by the codec's own noise floor. Consistently, at $a = 0.001$ the latent of the degraded input is closer to the codec's silence latent than to the latent of the clean excerpt (cosine similarity 0.70 versus 0.43). The same analysis shows that the input must be encoded in the precision used for the training latents: when the quiet inputs are encoded in float32 instead of bfloat16, the `volume` adapter no longer restores the signal level.

D Bridge Parameterization and Conditioning per Backbone

All backbones are used without any text, lyrics, or semantic conditioning: the bridge is defined only by the latent pair (z_c, z_d). The bridge time is sampled uniformly, $t \sim \mathcal{U}[0, 1]$, without timestep shift unless stated otherwise, and the same interpolation and solver conventions are used at training and inference.

ACE-Step 1.5 and ACE-Step 1.5 XL. The frozen DiT decoder receives null cross-attention states (a single zero token), a zero latent-context slot with an all-ones context mask, and no caption or lyric encoder output. In the code, the bridge is written with reversed time: $x_s = (1 - s)z_d + sz_c$ with

$s = 1 - t$, the decoder receives the timestep $1 - s = t$, and the network output is regressed onto $z_c - z_d$. This is the same path and the same field as Equation (1) up to the sign convention of the time variable. Inference integrates this field with explicit Euler steps from the degraded latent; with a single step, the output is $z_d + v(z_d, 1)$.

Stable Audio Open. The same reversed-time convention is used. The diffusion transformer, natively a v-prediction model, receives the bridge time as its timestep, null cross-attention conditioning and the duration conditioning of a 30-second clip; its raw output is regressed directly onto the bridge velocity, so the adapter also absorbs the change of parameterization.

Stable Audio 3. Its native objective is rectified flow, so we use the convention of Equation (1) directly: $x_\sigma = (1 - \sigma)z_c + \sigma z_d$, target $z_d - z_c$, and Euler integration from $\sigma = 1$ to $\sigma = 0$, with the text prompt removed and no inpainting context.

YuE2-3B. We adapt the non-autoregressive acoustic stage only, with an empty token sequence (no style, lyrics, or score tokens). The bridge follows the reversed-time convention, the bridge time is passed through the model's native logit mapping and timestep shift, and inference uses a 16-step midpoint solver.

E Training Parameters and Computing Resources

E.1 Computing Resources

Each adapter was trained on one node with 4 GPUs (NVIDIA L40S or H100).

E.2 LoRAs Training Parameters

We trained restoration LoRAs from three backbone baselines: ACE-Step 1.5, ACE-Step 1.5-XL, and Stable Audio. Unless otherwise stated, all runs used the same optimization setup: Adam-style training through `torchrun`, learning rate 1×10^{-4}, 100 warmup steps, gradient accumulation of 1, gradient clipping at 0.5, LoRA rank 32 (or 128), LoRA alpha twice the rank, 25,000 training steps, validation on 500 examples, logging every 10 steps, and random seed 42.

Parameter	ACE-Step 1.5	ACE-Step 1.5-XL	Stable Audio
Base model	`acestep-v15-sft`	`acestep-v15-xl-sft`	`stable_audio_2_0`
Batch size	32	32	8
LoRA rank	32	32	32
LoRA alpha	64	64	64
Learning rate	1×10^{-4}	1×10^{-4}	1×10^{-4}
Warmup steps	100	100	100
Gradient accumulation	1	1	1
Gradient clipping	0.5	0.5	0.5
Maximum steps	25,000	25,000	25,000
Validation interval	1,000	2,500	2,500
Checkpoint interval	12,500	12,500	12,500
Validation sampler steps	30	30	30
Data loader workers	4	4	4
Prefetch factor	4	4	4
Persistent workers	yes	yes	yes
LoRA target preset	attention only	attention only	all default trainable LoRA layers
Adapter type	LoRA	LoRA	LoRA

Table 18: Default training parameters used for the restoration LoRAs derived from each baseline.

For the ACE-Step 1.5 and ACE-Step 1.5-XL baselines, we used the attention-only target-module preset provided by our training pipeline. For Stable Audio, we used the default target layers exposed by the Stable Audio LoRA training script. In all cases, training used one node with 4 GPUs.

F Additional Experimental Evidence and Clarifications

This section collects additional experiments, implementation details, and clarifications. It describes the capacity-matched Conditional LoRA and ControlNet baselines, reports an NFE sweep for the conditional baselines, analyzes calibration-based NFE selection, and clarifies the specialist and universal deployment settings. Unless stated otherwise, all averages are over the 19 degradation families and follow the evaluation protocol of Section 4.1.

F.1 Training protocol and adapter configurations

All reported $\mathrm{Ours}_{\mathrm{rk128}}$ adapters were trained for $25{,}000$ steps. The $\mathrm{Ours}_{\mathrm{rk128}}$ configuration uses a frozen ACE-Step 1.5 XL backbone, LoRA rank 128, alpha 256, dropout 0.1, and LoRAs on the Q, K, V, and output projections of every self- and cross-attention block. It is optimized with AdamW at a learning rate of 10^{-4}, with 100 warm-up steps, gradient clipping at 0.5, a per-GPU batch size of 8 on 4 GPUs, uniform bridge-time sampling, and Euler inference.

ACE-Step 1.5 XL contains 32 transformer layers with hidden width $2{,}560$, 32 query heads, 8 key/value heads, and head dimension 128. Its attention projections map Q: $2{,}560 \to 4{,}096$, K/V: $2{,}560 \to 1{,}024$, and O: $4{,}096 \to 2{,}560$. These are full pretrained matrices; rank 128 refers only to the LoRA residual update. Inserting rank-128 LoRAs into Q, K, V, and O in every self- and cross-attention block gives 167.8M trainable parameters.

The universal, or blind, model is a single shared degradation-agnostic adapter, not a router or ensemble. One attention LoRA is trained jointly on all 19 degradations without labels, prompts, classifiers, routing, or a mixture of experts. It is built on the ACE-Step 1.5 backbone (not XL) and uses rank $2{,}048$, alpha $4{,}096$, 1.41B trainable parameters, and $100{,}000$ training steps. We therefore distinguish this universal setting from the specialist setting throughout the discussion below.

F.2 Capacity-matched conditioning control

The Conditional LoRA baseline isolates changing the transport source from adding LoRA capacity or providing the degraded recording to the model. It uses the same frozen ACE-Step 1.5 XL backbone, rank-128 LoRA architecture, 167.8M trainable parameters, paired data, training budget, and inference solver as the Bridge Adapter. The two systems differ only in how the degraded audio enters the transport:

- **Conditional LoRA** starts from Gaussian noise, preserves the Gaussian-to-clean objective, and supplies the degraded waveform—encoded by the frozen ACE VAE—through ACE-Step's native audio-to-audio aligned latent-context channels at every solver step. No new conditioning network is introduced.
- **Bridge Adapter** starts from the degraded latent and learns degraded-to-clean transport. The native latent-context slot receives ACE-Step's learned silence latent.

Thus, this comparison changes whether the degraded audio is side information or the source of the trajectory while holding the backbone, adapter architecture, trainable capacity, data, and solver fixed. The conditional baselines were evaluated with the same codec and bfloat16 input encoding as our models; their SNR and SI-SDR are therefore directly comparable, and their Mel distance is computed on the float32 decoding of the same latents, as for our models. Their FAD values were computed on bfloat16-decoded audio, and their Audiobox Aesthetics scores are not available.

At matched five NFE (Table 1), Bridge Adapters improve over Conditional LoRA by 2.00 dB in SI-SDR, 1.28 dB in SNR, and 0.039 in Mel distance, while Conditional LoRA obtains a lower $\mathrm{FAD}_{\mathrm{FMA}}$. The matched comparison therefore supports the conclusion that changing the transport source substantially improves paired restoration fidelity at equal compute, at the cost of a small loss in distributional perceptual quality.

F.3 NFE sweep and conditional-baseline efficiency

Table 19 rules out insufficient sampling as the explanation for the conditional-baseline gap. Increasing the sampling budget improves the paired fidelity of both conditional baselines, but does not recover that of the five-step bridge. Even at 30 NFE, Conditional LoRA remains worse than the five-NFE

Table 19: NFE sweep for the conditional baselines and $\text{Ours}_{\text{rk128}}$, averaged over the 19 degradations. Mel uses float32-decoded audio; it is not available (–) for Conditional LoRA at 1 and 30 NFE. Baseline FAD values use bfloat16-decoded audio.

Method	**# Steps**	**Mel** ↓	**SNR** ↑	**SI-SDR** ↑	**FAD**$_{\text{clean}}$ ↓	**FAD**$_{\text{FMA}}$ ↓
$\text{Ours}_{\text{rk128}}$	1	0.485	**3.54**	0.87	0.323	0.213
$\text{Ours}_{\text{rk128}}$	5	**0.482**	3.29	**1.11**	0.305	0.216
$\text{Ours}_{\text{rk128}}$	30	0.484	3.11	0.90	0.307	0.217
ControlNet	1	0.524	2.00	-1.36	0.330	0.219
ControlNet	5	0.509	2.35	-0.54	0.297	0.204
ControlNet	30	0.511	2.32	-0.42	**0.295**	0.204
Conditional LoRA	1	–	1.70	-1.64	0.339	0.220
Conditional LoRA	5	0.520	2.00	-0.89	0.301	**0.200**
Conditional LoRA	30	–	2.14	-0.51	0.298	**0.200**

Table 20: Accounting and deployment semantics for the controlled comparisons. Parameter counts report the active trainable parameters for one input.

Method	Backbone	Initial state	Use of degraded audio	Deployment	Trainable params.	Training steps
$\text{Ours}_{\text{rk128}}$	ACE-Step 1.5 XL	Degraded latent	Transport source	Specialist	167.8M	25,000
Conditional LoRA	ACE-Step 1.5 XL	Gaussian	Native latent context	Specialist	167.8M	25,000
ControlNet	ACE-Step 1.5 XL	Gaussian	Copied control branch	Specialist	4.378B	25,000
$\text{Ours}_{\text{blind}}$	ACE-Step 1.5	Degraded latent	Transport source	One shared adapter	1.41B	100,000

bridge by 1.62 dB in SI-SDR and 1.14 dB in SNR; ControlNet remains behind by 1.52 dB and 0.96 dB, respectively, and by 0.030 in Mel distance.

F.4 CONTROLNET IMPLEMENTATION

The ControlNet comparison freezes ACE-Step 1.5 XL and creates a trainable, pretrained-weight copy of its decoder trunk. At every function evaluation, both branches process the current Gaussian-flow state, while the copied branch additionally receives the degraded waveform, encoded by the frozen ACE VAE, through ACE-Step's aligned latent-context interface. Zero-initialized linear projections inject the residual from each copied block into the corresponding frozen block, making the initial system exactly equivalent to the pretrained generator. The system retains the original Gaussian-to-clean rectified-flow objective: sampling begins from Gaussian noise and the degraded recording is used only as conditioning.

One ControlNet is trained per degradation for $25{,}000$ steps with AdamW, a learning rate of 10^{-5}, weight decay 10^{-2}, 200 warm-up steps, gradient clipping at 1.0, and an effective batch size of 16. Its copied branch and residual projections contain 4.378B trainable parameters, $26.1\times$ the 167.8M parameters of the rank-128 Bridge Adapter. The ControlNet result is therefore not capacity matched; it tests whether a much larger condition-only branch over the same pretrained prior recovers the benefits of changing the transport source. The capacity-matched Conditional LoRA comparison above provides the stricter attribution test.

F.5 UNCERTAINTY PROTOCOL AND INTERPRETATION OF FAD

SonicMaster provides separate training and test splits. All adapters and baselines are trained only on paired examples from the training split; all reported results use the held-out test set of 996 excerpts per degradation family, and no test recording is used for training. The main text reports degradation-level uncertainty: means, standard deviations and 95% bootstrap confidence intervals over the 19 families, together with paired bootstrap intervals and exact paired sign-permutation tests for the family-level improvement over SonicMaster-Prompt. Because the degradation families differ strongly in difficulty, the intervals of the means are wide, whereas the paired family-level improvements are consistent.

Table 21: Calibration-selected NFE on 476 evaluation sources disjoint from the 520 calibration sources. The confidence intervals are computed from paired per-source differences between the lookup and fixed-NFE policies.

Policy	Mean NFE	SI-SDR ↑	SNR ↑	Mel ↓	LSD ↓
Fixed NFE 5	5.00	1.283	3.339	0.5625	1.0350
Calibration-selected lookup	**3.21**	**1.309**	**3.482**	**0.5602**	**1.0301**
Mean difference	–	+0.0257	+0.1432	−0.00226	−0.00485
95% CI	–	[0.0083, 0.0432]	[0.1280, 0.1583]	[−0.00255, −0.00197]	[−0.00522, −0.00448]

FAD is a set-level distributional metric and does not preserve one-to-one pairing. $\mathrm{FAD}_{\mathrm{clean}}$ is computed, for each degradation family, between restorations of 996 distinct held-out 30-second excerpts and the corresponding 996 clean excerpts, totaling approximately 8.3 hours. It is therefore not computed from one song split into chunks. Although the clean reference sets correspond one-to-one to the restored tracks, FAD discards this pairing and compares only their aggregate embedding distributions. It can also be sensitive to finite-sample estimation. Its uncertainty is therefore only assessed across degradation families. We interpret it alongside genuinely paired metrics such as SI-SDR, SNR, and Mel distance. This explains how a model can improve set-level alignment while remaining less faithful to a particular clean recording. $\mathrm{FAD}_{\mathrm{FMA}}$, computed against an external high-quality music set, instead probes the perceptual-quality axis.

F.6 Calibration-based adaptive NFE

The Mel-optimal NFE is positively correlated with the average degraded-to-clean latent distance, suggesting that stronger degradations can benefit from longer integration. To test a non-oracle policy, we split the 996 sources into 520 calibration and 476 disjoint evaluation sources. For each degradation, the calibration split selects one NFE from $\{1, 5, 10\}$; the resulting lookup is then frozen before evaluation. The policy selects NFE 1 for 11 degradations, NFE 5 for six, and NFE 10 for two. This analysis was run in the rebuttal evaluation pipeline, with the VAE decoding in bfloat16 and over all 19 families, so its absolute Mel and LSD values are not comparable with Table 1; only the paired differences between the two policies are of interest here.

The lookup reduces mean NFE by 35.8% while improving all four metrics, and every paired confidence interval excludes zero. Each specialist adapter defines a distinct vector field, so its solver budget can be calibrated together with its weights. This associates each learned transport with its calibrated numerical budget without requiring an additional predictor at inference.

F.7 Specialization, routing, and the universal model

At the system level, the specialist Bridge Adapter bank can be viewed as a hard-routed mixture of LoRA experts over one shared frozen backbone. Adapter selection plays the same operational role as selecting the corresponding degradation prompt for SonicMaster. Only one 167.8M-parameter adapter is active for an input, while the complete bank of 19 adapters stores 3.19B adapter parameters. The comparison to SonicMaster-Prompt is therefore protocol matched rather than capacity matched: both receive the correct degradation identity at inference. SonicMaster-Prompt is the most directly matched from-scratch baseline available because it targets full-mix music restoration, uses the same benchmark and degradation taxonomy, and receives the correct degradation identity.

Table 22 contextualizes the effect of targeting a known degradation and the effect of the Bridge system. It is not a causal decomposition: the blind and targeted Bridge adapters also differ in backbone and capacity, and we do not train a blind Conditional LoRA. Every entry is the improvement of the second system over the first; positive values always indicate an improvement. For Mel and $\mathrm{FAD}_{\mathrm{FMA}}$, the difference is first minus second; for SNR and SI-SDR, it is second minus first.

The universal $\mathrm{Ours}_{\mathrm{blind}}$ model improves SNR by 0.49 dB, SI-SDR by 0.93 dB and $\mathrm{FAD}_{\mathrm{FMA}}$ from 0.252 to 0.213 over SonicMaster-Blind at the same five steps, but has a higher Mel distance. Targeting adds further paired fidelity in both the SonicMaster and the Bridge systems, but does not by itself explain the core Bridge advantage.

Table 22: Targeting and Bridge comparisons over the 19 degradations (not a causal decomposition). Positive values indicate that the second system improves over the first. $\mathrm{Ours}_{\mathrm{blind}}$ uses ACE-Step 1.5 and $\mathrm{Ours}_{\mathrm{rk128}}$ ACE-Step 1.5 XL, so the second row also includes a backbone change; the $\mathrm{FAD}_{\mathrm{FMA}}$ of $\mathrm{Ours}_{\mathrm{blind}}$ was computed on bfloat16-decoded audio.

Effect	First system	Second system	Δ Mel	Δ SNR	Δ SI-SDR	Δ $\mathrm{FAD}_{\mathrm{FMA}}$
Targeting	$\mathrm{SonicMaster}_{\mathrm{blind}}$	$\mathrm{SonicMaster}_{\mathrm{prompt}}$	$+0.011$	$+0.50$	$+0.50$	$+0.007$
Targeting	$\mathrm{Ours}_{\mathrm{blind}}$	$\mathrm{Ours}_{\mathrm{rk128}}$	$+0.009$	$+0.50$	$+0.37$	-0.003
Bridge, blind	$\mathrm{SonicMaster}_{\mathrm{blind}}$	$\mathrm{Ours}_{\mathrm{blind}}$	-0.067	$+0.49$	$+0.93$	$+0.039$
Bridge, targeted	$\mathrm{SonicMaster}_{\mathrm{prompt}}$	$\mathrm{Ours}_{\mathrm{rk128}}$	-0.069	$+0.48$	$+0.81$	$+0.029$

F.8 Scope, limitations, and future directions

The study remains restricted to SonicMaster's paired, synthetically generated degradations. This controlled setting enables exact fidelity evaluation over 19 corruption families, but does not establish robustness to unknown mixtures, archival damage, or real recording pipelines. Gains are also heterogeneous across degradation families, so aggregate results should be read together with the per-degradation tables. The gap between the universal and specialist adapters is an important limitation: one shared transport does not yet capture the full diversity of restoration trajectories. Finally, every method is bounded by its frozen audio codec and cannot reconstruct information absent from that representation.

Learned routing, mixtures or compositions of adapters, and training on real or mixed degradations are therefore important next steps. An explicitly routed SonicMaster mixture-of-experts model could also improve over text-only conditioning and would provide a useful future baseline. Extending Bridge Adapters to source-level restoration is another natural direction. These extensions would move the method beyond the current controlled, known-degradation setting while retaining the modular pretrained prior.