AUDIO-TO-AUDIO VIA DIFFUSION WARM INITIALIZATION

Cristóbal Andrade and Sebastian J. Schlecht

Multimedia Communications and Signal Processing,
Friedrich-Alexander-Universität Erlangen-Nürnberg
Erlangen, Germany
{cristobal.andrade, sebastian.schlecht}@fau.de

ABSTRACT

In this paper, we propose diffusion warm initialization as a simple yet effective approach for a range of audio-to-audio transformation tasks. To illustrate the generality of the approach, we demonstrate its use in timbre transfer, MIDI-to-Real synthesis, and multiple audio enhancement tasks. We conduct a detailed empirical analysis on timbre transfer to investigate the role of the initialization time t_{init}. The effect of t_{init} is evaluated using pitch-based Jaccard Distance and Fréchet Audio Distance to quantify faithfulness to the input signal and alignment with the target distribution. Our results provide practical guidance for selecting t_{init} and show that, once properly chosen, a single pretrained diffusion model combined with warm initialization can support multiple transformation objectives without task-specific training or conditioning. Despite its simplicity, this approach already achieves competitive results when compared with more complex pipelines designed specifically for these tasks. We further observe that warm initialization does not necessarily require explicit noise injection, as the guide signal itself can often serve as a valid initialization state for the backward diffusion process. Together, these findings show that warm initialization provides a simple and effective framework that serves as a fundamental building block for more complex audio transformation pipelines.

1. INTRODUCTION

Diffusion models are a class of generative models that have gained significant attention in recent years, with applications ranging from image synthesis [1] and audio generation [2] to protein structure design [3]. They operate by progressively corrupting data samples through a forward process that adds noise in small increments, while a neural network is trained to approximate the reverse process and remove this noise step by step [4]. In this way, the model learns to transform samples drawn from a Gaussian distribution, which is simple to sample from, into samples from a complex data distribution.

Warm initialization *hijacks* the generative process by starting the reverse diffusion from a guide signal $\mathbf{x}^{(\text{g})}$, typically with noise added according to the chosen starting diffusion time, rather than from pure noise. As a result, the reverse diffusion process modifies an existing signal instead of synthesizing one entirely from scratch [5, 6, 7]. Such a mechanism is particularly suitable for audio-to-audio transformations, where certain characteristics of the input signal must be preserved while others are modified.

Copyright: © 2026 Cristóbal Andrade et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, adaptation, and reproduction in any medium, provided the original author and source are credited.

In this work, we show that warm initialization provides a simple yet effective method for a wide range of audio transformation tasks. To illustrate its versatility, we present examples including timbre transfer, MIDI-to-Real Synthesis and audio enhancement. In this context, audio transformations via warm initialization can be interpreted as moving a sample $\mathbf{x}^{(\text{g})}$ from a low-probability region toward a higher-probability region under the data distribution p_0. Audio examples are provided on the companion website[1].

This work further explores the idea that unconditional generation with diffusion models can be repurposed for multiple tasks without task specific retraining [8]. In this context, we argue that warm initialization plays an important role, as it allows the reverse diffusion process to start in a neighborhood of the desired portion of the data distribution rather than from pure noise. By initializing the process with a meaningful guide signal $\mathbf{x}^{(\text{g})}$, the generation begins closer to signals that already contain relevant structure, enabling the model to modify selected characteristics while preserving others. We therefore propose warm initialization as a fundamental step in audio transformation pipelines.

Selecting the initialization time t_{init} in warm initialization remains an open problem with limited theoretical grounding. Prior work such as AudioLDM [9] employs warm initialization for audio manipulation but does not provide guidance on how to select the initialization timestep. In practice, this parameter is typically selected through manual tuning, balancing heuristic trade offs between realism and faithfulness [5]. Realism refers to how closely the output resembles a valid sample from the target data distribution, while faithfulness refers to how well the transformation preserves the required properties of the guide signal. In this work, we provide a systematic analysis of the realism–faithfulness tradeoff in audio warm initialization, introducing complementary audio-domain metrics to identify a suitable operating point for t_{init}, an aspect absent from prior work on audio warm initialization.

We also provide empirical evidence that explicit noise corruption of the guide signal $\mathbf{x}^{(\text{g})}$ is not strictly necessary for warm initialization. Our experiments show that using the guide signal directly as the starting point of the reverse diffusion can be sufficient, and that additional noise injection may introduce detrimental artifacts, such as generative hallucinations.

This paper is organized as follows. Section 2 introduces diffusion denoising models and the concept of warm initialization. Section 3 describes the proposed framework and the metrics used to analyze and tune the initialization time. Section 4 presents experimental results and example applications. Section 5 discusses limitations and directions for future work. Finally, Section 6 provides concluding remarks.

[1] cristobalandrade.github.io/Audio-to-Audio-via-Diffusion-Warm-Initialization

2. DIFFUSION WARM INITIALIZATION

2.1. Diffusion Models

Diffusion models are a class of deep generative models that achieve high sample quality with strong mode coverage [10]. They learn to generate data by learning to transform a simple-to-sample distribution p_T, typically Gaussian noise $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, to the data distribution p_0 [4].

Training comprises a forward and a reverse process. The forward process defines a discrete-time Markov chain defined over $t \in [0, T]$ that progressively corrupts a data sample $\mathbf{x}_0 \sim p_0$ according to a predefined noise schedule. Specifically, the transitions $q(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ are Gaussian and chosen such that each latent variable admits the closed-form representation $\mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\varepsilon}$, with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, where α_t and σ_t are time-dependent coefficients controlling the signal and noise contributions, respectively. As t increases, α_t decreases while σ_t increases [4, 11]. This construction induces a sequence of intermediate marginals p_t, with $\mathbf{x}_t \sim p_t$ that gradually transform the data distribution into Gaussian noise.

In the reverse process, a neural network f_θ is trained to approximate the reverse-time transition kernel of the forward Markov chain, namely the posterior distribution $q(\mathbf{x}_{t-1} \mid \mathbf{x}_t)$. In practice, this is achieved by learning a parameterization of the transition, commonly through prediction of the injected noise $\boldsymbol{\varepsilon}$ or the score function $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$. These two parameterizations are mathematically linked via Tweedie's formula [12]. By approximating the posterior transitions, the learned model progressively transforms samples from noise toward the data manifold.

After training, generation is performed by sampling $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and iteratively applying the learned reverse transitions, transporting the sample through the intermediate distributions p_t toward p_0. The final output of this process is denoted by $\hat{\mathbf{x}}_0$, corresponding to a generated sample from the model distribution approximating p_0. Under this formulation, if samples $\mathbf{x}_t \sim p_t$ were available, the reverse diffusion process could be initialized directly from these samples, thereby generating outputs that lie on p_0.

2.2. Warm Initialization

Warm initialization is a technique in diffusion models in which the backward process is initialized from an intermediate diffusion timestep $t_{\text{init}} \in [0, T]$. In this work, we parameterize this choice using $\tau_{\text{init}} \in [0, 1]$, which denotes the fraction of the reverse diffusion trajectory that is skipped. The reverse process is initialized with a guide signal $\mathbf{x}^{(\text{g})}$ from a distribution $p_{\text{g}} \approx p_{t_{\text{init}}}$ that lies closer to the data distribution p_0 than a pure noise distribution p_T [13]. It is commonly employed to accelerate sampling by truncating the reverse trajectory, or to incorporate structured guidance in enhancement tasks [14, 15], where the generation process should only modify degraded components while preserving the remaining characteristics of the input signal.

Warm initialization can be expressed as:

$$\mathbf{x}^{(\text{g})}_{t_{\text{init}}} \sim \mathcal{N}(\alpha_{t_{\text{init}}} \mathbf{x}^{(\text{g})}, \sigma^2_{t_{\text{init}}} \mathbf{I}) \tag{1}$$

Two standard formulations are commonly found in the literature: Variance Exploding (VE) and Variance Preserving (VP). In the VE formulation, $\alpha_t = 1$ for all t, and σ_T is selected sufficiently large such that the terminal marginal p_T approaches $\mathcal{N}(0, \sigma_T^2 I)$. In contrast, in VP case the coefficient satisfy $\alpha_t^2 + \sigma_t^2 = 1$ for all t, with σ_t decreasing to 0 as t approaches 0, resulting in $p_T = \mathcal{N}(\mathbf{0}, \mathbf{I})$. In this work, we use VE as it has demonstrated good empirical performance in prior studies [5].

Algorithm 1 Diffusion Warm Initialization

Require: Input signal guide $\mathbf{x}^{(\text{g})}$, initialization fraction $\tau_{\text{init}} \in [0, 1]$, desired noise portion $\lambda \in [0, 1]$, diffusion noise schedule σ_t, diffusion denoiser f_θ, number of inference steps T, guidance scale ω

$t_{\text{init}} \leftarrow \lceil (1 - \tau_{\text{init}}) \cdot T \rceil$
Sample $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
$\mathbf{x}_{t_{\text{init}}} \leftarrow \mathbf{x}^{(\text{g})} + \lambda \, \sigma_{t_{\text{init}}} \boldsymbol{\varepsilon}$
for $t \leftarrow t_{\text{init}}$ **down to** 1 **do**
    $\mathbf{x}_{t-1} \leftarrow$ Diffusion Pipeline Step$(\mathbf{x}_t, t; f_\theta, \omega)$
end for
return $\mathbf{x}_0$

The central assumption underlying warm initialization is that the injected Gaussian noise $\mathcal{N}(\mathbf{0}, \sigma^2_{t_{\text{init}}} \mathbf{I})$ should be chosen such that the noised guide $\mathbf{x}^{(\text{g})}_{t_{\text{init}}}$ lies in a high-density region of an intermediate marginal of the reverse diffusion process [5]. In our experiments, we show that explicit addition of Gaussian noise to the guide signal is not always necessary, as $\mathbf{x}^{(\text{g})}$ can already be treated as a sample from an intermediate marginal p_t, allowing initialization without additional noise injection.

To the best of our knowledge, there is a lack of theoretical evidence for systematically selecting the initialization time t_{init}. Existing studies provide partial insights into the role of initialization but do not offer practical guidance. For example, [16] shows that the number of required reverse diffusion steps decreases as the initialization approaches the data distribution p_0, motivating starting closer to the data manifold. Similarly, image editing methods such as SDEdit [5] observe that the discrepancy between the guide signal and the generated output decreases as the initialization point moves closer to the end of the reverse diffusion process.

In practice, t_{init} is typically selected heuristically through manual tuning. A commonly discussed heuristic is the trade off between realism and faithfulness. Earlier initialization encourages stronger projection onto the learned data manifold, often producing more realistic outputs but increasing deviation from the guide signal. Conversely, later initialization preserves structural properties of $\mathbf{x}^{(\text{g})}$ while limiting the extent of model driven modification [5, 14]. In the following sections, we examine this behavior empirically and provide guidance for identifying this operating regime in the audio domain.

3. AUDIO-TO-AUDIO VIA WARM INITIALIZATION

In this section, we present the general framework of warm initialization for audio-to-audio tasks. We also introduce two metrics designed to assess the trade-off between realism and faithfulness and to help identify a suitable initialization time t_{init}.

3.1. Framework

Algorithm 1 provides a unified formulation of diffusion warm initialization as a modification of the standard diffusion sampling procedure for audio transformation tasks, where the reverse trajectory is initialized from a guide signal $\mathbf{x}^{(\text{g})}$.

Figure 1 illustrates the difference between a standard unconditional generative pipeline and a warm initialization pipeline in the context of timbre transfer. In this setting, the target data distribution p_0 corresponds to piano recordings. In the unconditional case, generation starts from Gaussian noise and progressively converges toward samples from p_0. In contrast, warm initialization starts from a guide signal $\mathbf{x}^{(\mathrm{g})}$, here an oboe recording, and applies reverse diffusion to transform its timbre while preserving its musical structure.

For the selected t_{init} in Figure 1 the melody and temporal structure of the oboe signal are retained, while its spectral characteristics evolve toward those of a piano. This can be observed in the spectrogram, where the harmonic content shifts from an oboe-like distribution to a piano-like distribution. Importantly, the model is not conditioned on this specific oboe sample and may not have seen the underlying melody during training, yet it is able to map the signal toward the piano distribution p_0 through reverse diffusion.

This example highlights how warm initialization enables controlled transformations: instead of generating from noise, the model starts from a structured input and selectively modifies its characteristics. A more detailed analysis of timbre transfer using warm initialization is provided in Section 4.

The choice of the initialization time t_{init} determines how strongly the diffusion process modifies the guide signal. Figure 2 shows the spectrograms of the outputs $\hat{\mathbf{x}}_0(\mathbf{x}^{(g)}, t_{\text{init}})$ obtained via reverse diffusion for different values of t_{init}, using $T = 100$ inference steps. Earlier initialization results in the model modifying the signal more aggressively, producing sounds that better match the target distribution but preserve less of the guide's structure. Later initialization preserves more characteristics of the input spectrogram while limiting the extent of the modification[1].

The parameter λ in Algorithm 1 controls the portion of the noise schedule used for initialization. In the following sections, we show empirically that adding Gaussian noise is not always necessary, and that $\lambda = 0$ often yields better results. This suggests that the guide signal $\mathbf{x}^{(\mathrm{g})}$ may already lie close to an intermediate marginal p_t, allowing the reverse diffusion process to be initialized without additional noise. Increasing λ can be detrimental, as additional noise injection tends to introduce hallucinated artifacts in the generated signal.

For this framework, we use the Stable Audio Open diffusion model [2]. We use text conditioning to steer generation toward the desired target distribution. Conditioning is implemented via classifier-free guidance (CFG), where the denoiser combines unconditional and text-conditional predictions, and a guidance scale ω controls the strength of this signal [17]. For instance, in timbre transfer experiments targeting piano, we employ the prompt *"Grand Piano, Chords and Melodies"*. The reverse diffusion is initialized from the guide signal $\mathbf{x}^{(\mathrm{g})}$, which defines the starting point of the trajectory, while the text prompt guides the subsequent evolution toward the target distribution. The guidance scale ω is included in Algorithm 1.

When applied in the context of warm initialization, we observe that the model requires a substantially higher guidance scale ω than the training setting, where $\omega = 7$ was used. In practice, this corresponds to increasing the CFG strength, thereby amplifying the conditional signal and improving fidelity to the target distribution [17]. This limitation would not arise in a model trained exclusively on the target domain, such as a piano-only generator, where alignment with the target distribution is intrinsic.

3.2. Metrics

To assess the trade-off between alignment with the target distribution and preservation of the characteristic structure of the input signal, we employ two complementary metrics: Fréchet Audio Distance (FAD), which measures perceptual similarity in a learned embedding space, and Jaccard distance over pitch sets, for quantifying melodic consistency.

3.2.1. Fréchet Audio Distance (FAD)

The Fréchet Audio Distance (FAD) measures how close a reference and a test set of audio samples are in an embedding space and it is often used as a proxy for generation quality [18]. For its computation both the reference and test audio embeddings are modeled with multivariate normal distributions. Let μ_r and μ_t denote the means, and Σ_r and Σ_t the covariance matrices, of the reference and test distributions, respectively. The FAD between two gaussian is then given by:

$$\mathrm{FAD} = \|\mu_r - \mu_t\|^2 + \mathrm{tr}\left(\Sigma_r + \Sigma_t - 2\sqrt{\Sigma_r \Sigma_t}\right) \tag{2}$$

In this work we use the toolbox provided by [19] using the *LAION-CLAP* embedding as it is used in Stable Audio [2].

3.2.2. Jaccard Distance (JD)

To assess the faithfulness of the generated output with respect to the guide signal, we compute the Jaccard Distance (JD)

$$\mathrm{JD}(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|} \tag{3}$$

where A and B are sets of pitch values extracted from the guide signal $\mathbf{x}^{(g)}$ and the generated output $\hat{x}_0$ respectively. The pitch sets are obtained using the mono pitch between the reference and test melodies using the mono pitch algorithm based on MELODIA [20] as implemented in the Essentia library [21]. Lower JD values indicate higher melodic similarity between the generated output and the guide signal $\mathbf{x}^{(\mathrm{g})}$.

4. APPLICATIONS

In this section, we demonstrate the versatility of warm initialization across several audio-to-audio tasks, including timbre transfer, MIDI-to-real synthesis, and audio enhancement. As the main experimental study, we analyze timbre transfer in detail to investigate the role of the initialization time t_{init} and its effect on the trade-off between faithfulness to the input signal and alignment with the target distribution. The remaining applications illustrate how the same framework can be applied to different tasks without task-specific modifications.

4.1. Timbre Transfer

Timbre denotes the perceptual properties that distinguish two instruments playing the same note at the same intensity, beyond pitch and amplitude. Timbre transfer refers to altering the instrument identity of an audio signal while preserving melody, rhythm, and other musical structure.

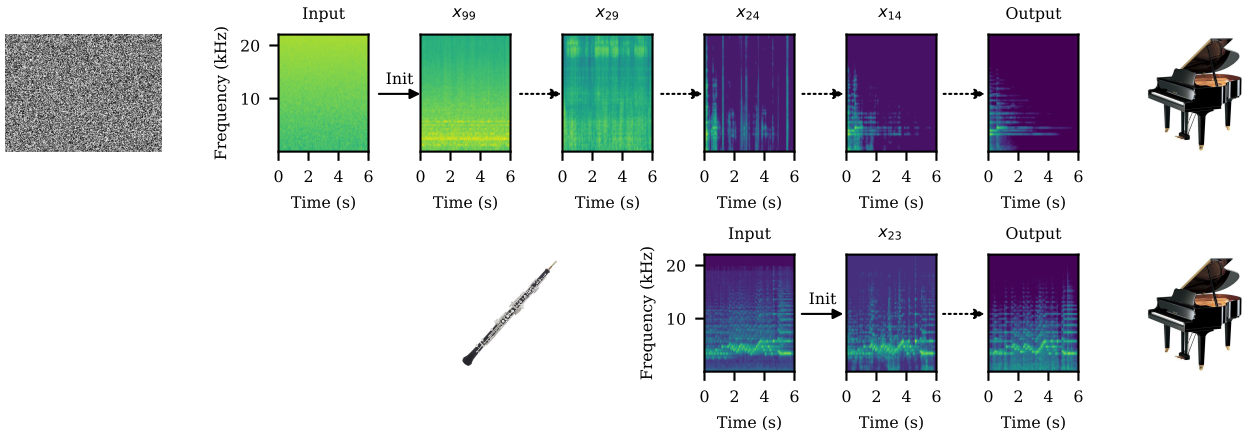

Figure 1: Diffusion generation compared with warm initialization. Dashed arrows indicate reverse diffusion steps. Top: Standard generation from Gaussian noise gradually produces piano audio. Bottom: Warm initialization from $\mathbf{x}^{(g)}$ (Oboe) preserves the melody while the diffusion process shifts the timbre toward the piano distribution.

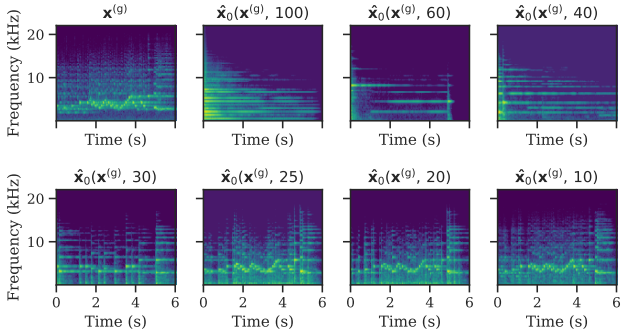

Figure 2: Spectrograms of oboe-to-piano timbre transfer for different initialization times t_{init}. A diffusion schedule with $T = 100$ steps is used, with reverse diffusion starting at t_{init}. Smaller t_{init} leads to stronger deviation from the input, while larger values preserve more of the original melodic structure.

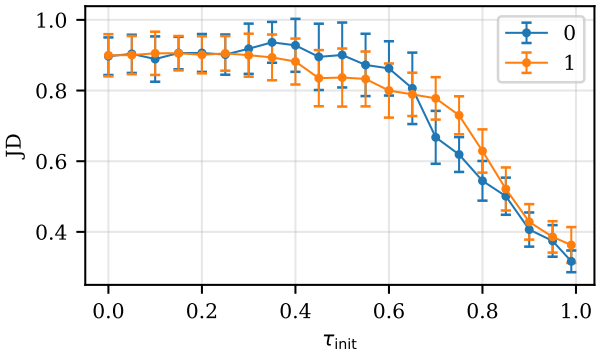

Figure 3: JD as a function of τ_{init} for oboe-to-piano timbre transfer for $\lambda = 0$ and $\lambda = 1$. $\tau_{\text{init}} = 0$ denotes that the model performs all reverse steps, while $\tau_{\text{init}} = 1$ denotes that no reverse steps are performed. Lower values indicate greater melodic similarity to the guide signal $\mathbf{x}^{(g)}$.

Several deep learning approaches have been proposed for this task. These include VAE-based methods [22, 23, 24], DDSP-based approaches [25], and diffusion-based models. Diffusion approaches either condition generation on a target timbre [26, 27, 28, 29, 30] or bridge separately trained source and target models through forward–reverse diffusion schemes [31]. In [7], warm initialization is presented as a *naive* baseline, where t_{init} is selected by training a classifier and choosing the time step at which the input is classified as noise with 50% probability. Similarly, AudioLDM [9] proposes warm initialization in the latent domain for style transfer, but does not provide guidance on how to select t_{init}.

As a case study, we perform timbre transfer from oboe to piano using Algorithm 1 with $\lambda \in \{0, 1\}$, where $\mathbf{x}^{(g)}$ is an oboe sample obtained from MUSOPEN [32]. The diffusion model is run for $T = 100$ inference steps with a guidance scale of $\omega = 30$. To assess the effect of the initialization time t_{init}, we compute the JD between $\mathbf{x}^{(g)}$ and 50 samples generated via warm initialization for each configuration, and report the mean and standard deviation. In addition, we compute FAD between a reference set of 100 unconditionally generated piano, produced using the default guidance scale $\omega = 7$, and 100 samples generated via warm initialization. The corresponding results are shown in Figures 3 and 4, respectively. The same analysis using FAD and JD is provided in the Appendix Section 8 for a string to clarinet timbre transfer experiment.

Figure 3 shows that, for both values of λ, the melodic distance to the guide signal $\mathbf{x}^{(g)}$ is initially high and decreases substantially once $\tau_{\text{init}} \gtrsim 0.65$. In contrast, Figure 4 indicates that later initialization leads to outputs that deviate further from the target piano distribution, as reflected by increasing FAD. We observe that the FAD is greater than zero at $\tau_{\text{init}} = 0$. This residual is a consequence of the high guidance scale used for the transfer, which differs from the scale applied during the unconditional generation of the reference samples.

For $\lambda = 1$, the JD remains consistently higher for $t_{\text{init}} > 0.65$, while FAD values are largely comparable between $\lambda = 0$ and

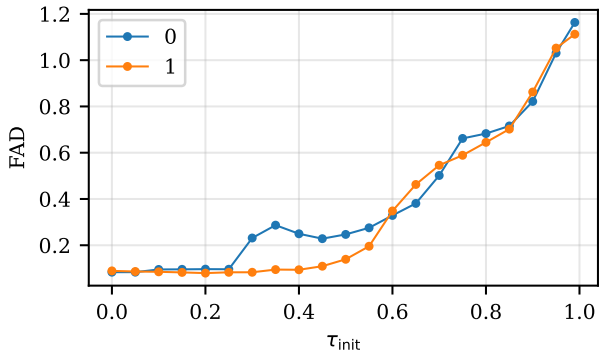

Figure 4: FAD as a function of τ_{init} for oboe-to-piano timbre transfer for $\lambda = 0$ and $\lambda = 1$. $\tau_{\text{init}} = 0$ denotes that the model performs all reverse steps, while $\tau_{\text{init}} = 1$ denotes that no reverse steps are performed. Lower values indicate closer alignment with the piano reference distribution.

$\lambda = 1$. Informal listening suggests that the increased melodic deviation corresponds to stronger generative hallucinations, where the model introduces musical events not present in the guide signal. This indicates that stronger noise injection amplifies hallucinated content without substantially improving alignment with the piano distribution. Listening suggests the presence of a perceptual sweet spot around $\tau_{\text{init}} \approx 0.8$, indicating that a Jaccard distance below 0.6 and FAD below 0.7 may serve as a practical rule of thumb for selecting τ_{init}. It is worth noting that the location of the sweet spot may vary depending on the diffusion model used for generation.

Audio examples illustrating the effect of varying τ_{init} and the influence of λ within the identified sweet spot are provided on the companion website. In addition to the oboe-to-piano experiments, we include further timbre transfer cases for comparison with prior work: string-to-clarinet in relation to [26, 27], synth-to-violin compared to [7], and violin-to-flute in comparison with [31]. Thereby we set $\lambda = 0$. These examples indicate that our method performs competitively despite its simpler formulation.

We also include further exploratory examples in which different input signals are transformed to piano. These include humming, nature recordings, and computer mouse clicks, which produce toy piano–like sounds, isolated piano notes, and plucked piano–string–like transients.

4.2. MIDI-to-Real

MIDI-to-Real synthesis refers to transforming audio rendered from symbolic MIDI into realistic recordings that capture timbral detail and characteristics of human performance. Several approaches found in literature build on Differentiable Digital Signal Processing (DDSP) [33]. For example, MIDI-DDSP employs a three-stage pipeline that maps MIDI notes to expressive performance features and subsequently synthesizes audio using a DDSP-based renderer [34]. Castellon et al. [35] similarly train a neural network to predict synthesis parameters from MIDI, which are then used to drive a DDSP synthesizer.

More recently, diffusion-based approaches have also been explored. Take et al. [36] propose a diffusion-based refinement method that operates on rendered audio in a manner related to our approach, although practical guidance for selecting the initialization time t_{init} is not discussed.

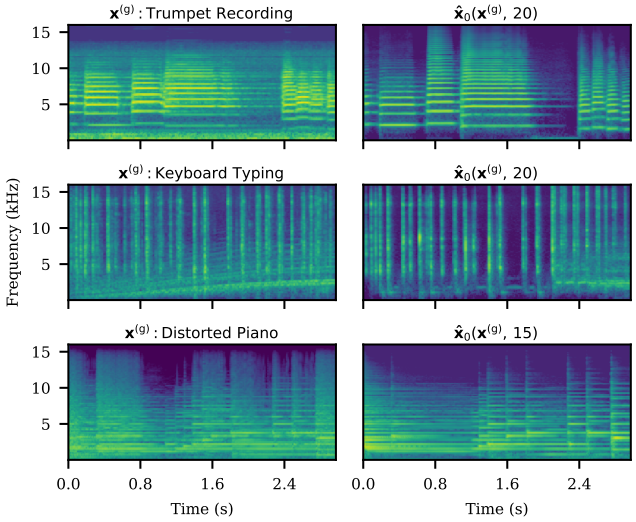

Figure 5: Spectrograms of enhanced audio signals. Left: degraded guide signals $\mathbf{x}^{(g)}$. Right: corresponding outputs obtained via warm initialization.

In our experiments, we treat MIDI-to-Real synthesis as an audio-to-audio refinement task and apply warm initialization with $\lambda = 0$. Audio examples comparing our method with prior work are provided on the companion website. From these examples, we observe that the strength of the transformation depends on the quality of the initial MIDI rendering. Despite its simplicity, the proposed framework is able to produce realistic outputs and achieves performance comparable to more complex pipelines designed specifically for this task.

4.3. Audio Enhancement

Audio enhancement refers to the task of improving the quality of recorded audio signals, including speech, music, and impulse responses. Degradations may arise from environmental factors such as background noise or interference, as well as from limitations or suboptimal configurations of the recording hardware and acquisition system. Common enhancement tasks include dereverberation [14], declipping [8, 37] and denoising[38].

Historically, audio enhancement has been addressed using classical signal processing techniques [38, 39], and more recently through deep learning approaches [40, 41]. In particular, diffusion-based methods have shown promising results for restoration and enhancement tasks [14, 42, 43].

In our experiments, we apply warm initialization with $\lambda = 0$ to several enhancement scenarios, including declipping, denoising, and source interference suppression. Figure 5 shows a few examples, including a noisy trumpet recording, keyboard typing with a siren in the background, and a distorted piano signal. Audio examples comparing our approach with prior work are provided on the companion website. These examples show that the proposed framework can be applied to a range of enhancement tasks while remaining conceptually simple.

5. DISCUSSION

While warm initialization is a versatile method, it presents certain limitations. One limitation we encountered during experimentation is that the method shows reduced performance when the guide signal contains human voice, as in humming-to-piano or sketch-to-sound scenarios, where the resulting outputs exhibit lower quality[1]. This suggests that the model fails to consistently move such inputs toward high-density regions of the target distribution during reverse diffusion. Further investigation is required to determine whether this behavior is inherent to the model or a consequence of the initialization strategy.

Another limitation arises in source separation tasks. The method often fails to remove unwanted components; instead of suppressing them through reverse diffusion, these components tend to persist or are altered into more artificial or distorted versions. This behavior is particularly evident for highly percussive elements, such as drums, which are not effectively suppressed by the framework. This provides a clear case where additional mechanisms on top of warm initialization may be required to achieve selective suppression.

A further practical consideration concerns the sensitivity of the method to its hyperparameters. The guidance scale ω, the noise injection parameter λ, and the initialization time τ_{init} all influence the output. In our experiments, the sweet spot around $\tau_{\text{init}} \approx 0.8$ generalizes across both timbre transfer experiments. However, these values are specific to Stable Audio Open. Different diffusion models have different probability paths and noise schedules, causing the operating regime for τ_{init} to shift accordingly. The metric-based framework proposed in Section 3.2 provides a practical tool for retuning these parameters in such cases.

More broadly, the approach relies on empirical tuning of the initialization time t_{init} and lacks a theoretical framework relating initialization to the structure of the data and marginal manifolds. These limitations highlight the need for further theoretical investigation of warm initialization.

6. CONCLUSION

In this work, we investigated diffusion warm initialization as a simple framework for audio-to-audio tasks. We showed that warm initialization enables a single pretrained diffusion model to support a range of transformations, including timbre transfer, MIDI-to-real synthesis, and audio enhancement, without task-specific retraining or conditioning.

A detailed empirical study on timbre transfer examined the role of the initialization time t_{init}. Using Fréchet Audio Distance and pitch-based Jaccard distance, we analyzed the trade-off between alignment with the target distribution and preservation of the guide signal. The results reveal the presence of a practical operating regime for t_{init}, providing empirical guidance for selecting this parameter in audio transformation tasks.

We further observed that explicit corruption of the guide signal with Gaussian noise is not always required. In many cases, the guide signal itself can already lie close to an intermediate diffusion state, allowing the reverse process to start directly from the input signal. Additional noise injection may instead introduce hallucinated content and reduce fidelity to the guide.

Our findings suggest that warm initialization constitutes a simple but effective mechanism for adapting pretrained diffusion models to audio-to-audio tasks. Future work may investigate the theoretical properties of this mechanism and explore how the latent structure of diffusion models influences the transformations induced by warm initialization.

7. REFERENCES

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 10 684–10 695.

[2] Z. Evans, J. D. Parker, C. Carr, Z. Zukowski, J. Taylor, and J. Pons, "Stable Audio Open," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Hyderabad, India, 2025.

[3] J. Watson, D. Juergens, N. Bennett, B. Trippe, J. Yim, H. Eisenach, W. Ahern, A. Borst, R. Ragotte, L. Milles, B. Wicky, N. Hanikel, S. Pellock, A. Courbet, W. Sheffler, J. Wang, P. Venkatesh, I. Sappington, S. Torres, and D. Baker, "De novo design of protein structure and function with rfdiffusion," *Nature*, vol. 620, 07 2023.

[4] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 6840–6851.

[5] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon, "SDEdit: Guided image synthesis and editing with stochastic differential equations," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022.

[6] B. Kawar, S. Zada, O. Lang, O. Tov, H. Chang, T. Dekel, I. Mosseri, and M. Irani, "Imagic: Text-based real image editing with diffusion models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2023, pp. 6007–6017.

[7] C. H. Lee, J. Nistal, S. Lattner, M. Pasini, and G. Fazekas, "Diffusion timbre transfer via mutual information guided inpainting," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Barcelona, Spain, 2026.

[8] E. Moliner, J. Lehtinen, and V. Välimäki, "Solving audio inverse problems with a diffusion model," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Rhodes Island, Greece, 2023, pp. 1–5.

[9] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "AudioLDM: Text-to-audio generation with latent diffusion models," in *Proc. Int. Conf. Mach. Learn. (ICML)*, ser. Proc. Mach. Learn. Res., vol. 202. PMLR, 2023, pp. 21 450–21 474.

[10] Z. Xiao, K. Kreis, and A. Vahdat, "Tackling the generative learning trilemma with denoising diffusion GANs," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 6680–6692.

[11] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021.

[12] B. Efron, "Tweedie's formula and selection bias," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1602–1614, 2011.

[13] J. Scholz and R. E. Turner, "Warm starts accelerate conditional diffusion," *arXiv preprint arXiv:2507.09212*, Sep. 2025.

[14] E. Moliner, J.-M. Lemercier, S. Welker, T. Gerkmann, and V. Välimäki, "BUDDy: Single-channel blind unsupervised dereverberation with diffusion models," in *Proc. IEEE Int. Workshop Acoust. Signal Enhancement (IWAENC)*, Aalborg, Denmark, 2024, pp. 136–140.

[15] E. Moliner, F. Elvander, and V. Välimäki, "Blind audio bandwidth extension: A diffusion-based zero-shot approach," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 32, pp. 5092–5105, 2024.

[16] H. Chung, B. Sim, and J. C. Ye, "Come-Closer-Diffuse-Faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022, pp. 12 413–12 422.

[17] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *Proc. NeurIPS Workshop Deep Generative Models Downstream Appl.*, 2021.

[18] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms," in *Proc. Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, Graz, Austria, 2019, pp. 2350–2354.

[19] A. Gui, H. Gamper, S. Braun, and D. Emmanouilidou, "Adapting Fréchet audio distance for generative music evaluation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Seoul, Korea, 2024, pp. 995–999.

[20] J. Salamon and E. Gomez, "Melody extraction from polyphonic music signals using pitch contour characteristics," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 6, pp. 1759–1770, 2012.

[21] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. Zapata, and X. Serra, "ESSENTIA: an open-source library for sound and music analysis," in *Proceedings of the 21st ACM International Conference on Multimedia*, ser. MM '13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 855–858. [Online]. Available: https://doi.org/10.1145/2502081.2502229

[22] P. Esling, A. Chemla-Romeu-Santos, and A. Bitton, "Generative timbre spaces: Regularizing variational auto-encoders with perceptual metrics," in *Proc. Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, Paris, France, 2018, pp. 438–445.

[23] A. Caillon and P. Esling, "RAVE: A variational autoencoder for fast and high-quality neural audio synthesis," in *Proc. Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, Online, 2021, pp. 292–299.

[24] J. T. Colonel and S. Keene, "Conditioning autoencoder latent spaces for real-time timbre interpolation and synthesis," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2020, pp. 1–7.

[25] M. Carney, C. Li, E. Toh, N. Zada, P. Yu, and J. Engel, "Tone transfer: In-browser interactive neural audio synthesis," in *Joint Proceedings of the ACM IUI 2021 Workshops*, ser. CEUR Workshop Proceedings, vol. 2903. CEUR-WS.org, 2021.

[26] T. Baoueb, X. Bie, H. Janati, and G. Richard, "WaveTransfer: A flexible end-to-end multi-instrument timbre transfer with diffusion," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process. (MLSP)*, London, UK, 2024.

[27] L. Comanducci, F. Antonacci, and A. Sarti, "Timbre transfer using image-to-image denoising diffusion implicit models," in *Proc. Int. Soc. Music Inf. Retr. Conf. (ISMIR)*, Milan, Italy, 2023, pp. 257–263.

[28] H. F. García, O. Nieto, J. Salamon, B. Pardo, and P. Seetharaman, "Sketch2Sound: Controllable audio generation via time-varying signals and sonic imitations," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Hyderabad, India, 2025.

[29] S. Li, Y. Zhang, F. Tang, C. Ma, W. Dong, and C. Xu, "Music style transfer with time-varying inversion of diffusion models," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 4, 2024, pp. 3859–3867.

[30] H. Huang, Y. Wang, L. Li, and J. Lin, "Musical timbre style transfer with diffusion model," *PeerJ Computer Science*, vol. 10, p. e2194, 2024.

[31] M. Mancusi, Y. Halychanskyi, K. W. Cheuk, E. Moliner, C.-H. Lai, S. Uhlich, J. Koo, M. A. Martínez-Ramírez, W.-H. Liao, G. Fabbro, and Y. Mitsufuji, "Latent diffusion bridges for unsupervised musical audio timbre transfer," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Hyderabad, India, 2025.

[32] Musopen, "Musopen: Free sheet music and recordings," 2024. [Online]. Available: https://musopen.org

[33] J. Engel, L. Hantrakul, C. Gu, and A. Roberts, "DDSP: Differentiable digital signal processing," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.

[34] Y. Wu, E. Manilow, Y. Deng, R. Swavely, K. Kastner, T. Cooijmans, A. Courville, C.-Z. A. Huang, and J. Engel, "MIDI-DDSP: Detailed control of musical performance via hierarchical modeling," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022.

[35] R. Castellon, C. Donahue, and P. Liang, "Towards realistic MIDI instrument synthesizers," in *Proceedings of the NeurIPS Workshop on Machine Learning for Creativity and Design*, 2020.

[36] O. Take and T. Akama, "Annotation-free midi-to-audio synthesis via concatenative synthesis and generative refinement," *arXiv preprint arXiv:2410.16785*, 2025.

[37] K. Y. Lee, N. Meyer-Kahlen, V. Välimäki, and S. J. Schlecht, "Solving room impulse response inverse problems using flow matching with analytic wiener denoiser," *arXiv preprint arXiv:2602.00652*, 2026.

[38] S. J. Godsill and P. J. W. Rayner, *Digital Audio Restoration: A Statistical Model-Based Approach*. New York: Springer, 1998.

[39] H. Attias, J. Platt, A. Acero, and L. Deng, "Speech denoising and dereverberation using probabilistic models," in *Advances in Neural Information Processing Systems*, T. Leen, T. Dietterich, and V. Tresp, Eds., vol. 13. MIT Press, 2000.

[40] C. Yu, R. E. Zezario, S.-S. Wang, J. Sherman, Y.-Y. Hsieh, X. Lu, H.-M. Wang, and Y. Tsao, "Speech enhancement based on denoising autoencoder with multi-branched encoders," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2538–2549, 2020.

[41] S.-W. Fu, T.-W. Wang, Y. Tsao, X. Lu, and H. Kawai, "End-to-end waveform utterance enhancement for direct evaluation metrics optimization by fully convolutional neural networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 9, pp. 1570–1584, 2018.

[42] N. Kandpal, O. Nieto, and Z. Jin, "Music enhancement via image translation and vocoding," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Singapore, 2022, pp. 946–950.

[43] H. Manor and T. Michaeli, "Zero-shot unsupervised and text-based audio editing using DDPM inversion," in *Proc. Int. Conf. Mach. Learn. (ICML)*, ser. Proc. Mach. Learn. Res., vol. 235. PMLR, 2024, pp. 34 603–34 629.

8. APPENDIX: STRING-TO-CLARINET TIMBRE TRANSFER

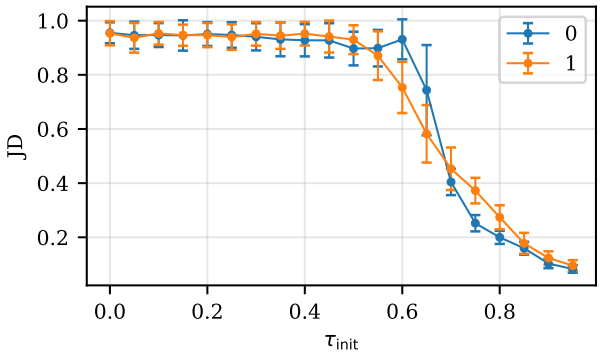

Figure 6: JD as a function of τ_{init} for string-to-clarinet timbre transfer for $\lambda = 0$ and $\lambda = 1$. $\tau_{\text{init}} = 0$ denotes that the model performs all reverse steps, while $\tau_{\text{init}} = 1$ denotes that no reverse steps are performed.

Similar to the oboe-to piano experiment, we perform timbre transfer from string-to-clarinet using Algorithm 1. The diffusion model is run for T = 100 inference steps with a guidance scale of $\omega = 30$. To evaluate the effect of the initialization time t_{init}, we compute the Jaccard Distance (JD) between the guide signal and 50 samples generated via warm initialization for each value of t_{init}, and report the mean and standard deviation. In addition, we compute FAD between a reference set of 100 unconditionally generated clarinet samples, produced using the default guidance scale $\omega = 7$, and 100 samples generated via warm initialization. The corresponding results are shown in Figures 6 and 7.

Informal listening suggests a perceptual sweet spot around $\tau_{\text{init}} = 0.8$. This is consistent with the guideline observed in the oboe-to-piano experiment, where a sweet spot tends to appear once the Jaccard distance is below 0.6 and FAD is below 0.7. In this case, FAD acts as the limiting factor. Audio examples are available on the companion website.

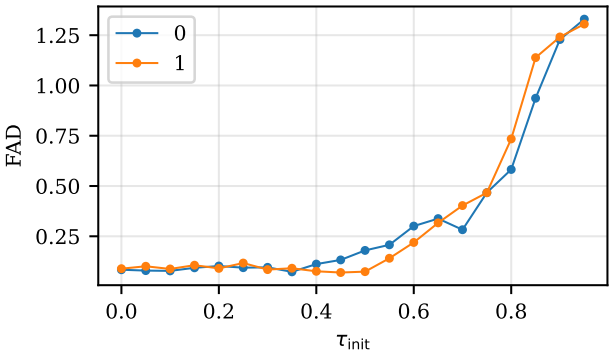

Figure 7: FAD as a function of τ_{init} for string-to-clarinet timbre transfer for $\lambda = 0$ and $\lambda = 1$. $\tau_{\text{init}} = 0$ denotes that the model performs all reverse steps, while $\tau_{\text{init}} = 1$ denotes that no reverse steps are performed.