

FlashAudio: Rectified Flows for Fast and High-fidelity Text-to-Audio Generation

Huadai Liu^{1,2*}, Jialei Wang^{3*}, Rongjie Huang⁴, Yang Liu³
Heng Lu², Zhou Zhao^{3†}, Wei Xue^{1†}

¹Hong Kong University of Science and Technology

²Tongyi Lab, Alibaba Group

³Zhejiang University

⁴Fundamental AI Research (FAIR), Meta

Abstract

Recent advancements in latent diffusion models (LDMs) have markedly enhanced text-to-audio generation, yet their iterative sampling processes impose substantial computational demands, limiting practical deployment. While recent methods utilizing consistency-based distillation aim to achieve few-step or single-step inference, their one-step performance is constrained by curved trajectories, preventing them from surpassing traditional diffusion models. In this work, we introduce FlashAudio with rectified flows to learn straight flow for fast simulation. To alleviate the inefficient timesteps allocation and suboptimal distribution of noise, FlashAudio optimizes the time distribution of rectified flow with Bifocal Samplers and proposes immiscible flow to minimize the total distance of data-noise pairs in a batch via assignment. Furthermore, to address the amplified accumulation error caused by the classifier-free guidance (CFG), we propose Anchored Optimization, which refines the guidance scale by anchoring it to a reference trajectory. Experimental results on text-to-audio generation demonstrate that FlashAudio’s one-step generation performance surpasses the diffusion-based models with hundreds of sampling steps on audio quality and enables a sampling speed of 400x faster than real-time on a single NVIDIA 4090Ti GPU. Code will be available at <https://github.com/liuhuadai/FlashAudio>.¹

1 Introduction

Diffusion models (Ho et al., 2020a,b; Song et al., 2021) have demonstrated significant capabilities in modeling various modalities, including images (Rombach et al., 2022; Saharia et al., 2022), audio (Liu et al., 2023b; Huang et al., 2023b),

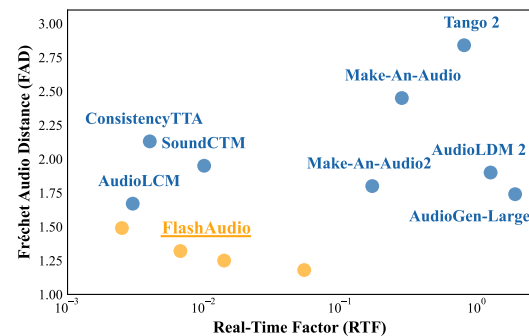


Figure 1: FlashAudio is a fast and high-fidelity text-to-audio generation model with rectified flow. It has a remarkable speed with 400x faster than real-time, on a single NVIDIA 4090Ti GPU.

and video (Ho, Chan, Saharia, Whang, Gao, Gritsenko, Kingma, Poole, Norouzi, Fleet, and Salimans, 2022; Gupta, Yu, Sohn, Gu, Hahn, Fei-Fei, Essa, Jiang, and Lezama, 2023). They have become the de-facto approach for generating high-fidelity audio from natural language inputs due to their impressive generalization capabilities. However, their iterative nature and the associated computational costs, along with prolonged sampling times during inference, limit their application in real-time scenarios (Song et al., 2023). Recent advances (Salimans and Ho, 2022; Kim et al., 2024) have thus concentrated on distillation models that estimate the integral along the Probability Flow (PF) ODE sample trajectory, effectively reducing the computational load associated with numerical solvers. For instance, AudioLCM (Liu et al., 2024a) utilizes guided consistency distillation with multi-step ODE solvers, achieving two-step generation performance on par with diffusion models. In contrast, some studies adopt one-step generation (Saito et al., 2024; Bai et al., 2024) but struggle to align with latent diffusion models.

While mapping any point in the trajectory into initial point leads to efficient training, it introduces the challenge of estimating data points along

*Equal Contribution.

†Corresponding Author.

¹Audio Samples are available at <https://FlashAudio-TTA.github.io/>.

curved trajectories. This choice can significantly impact sampling, as demonstrated by Kim et al. (2024). For instance, a forward process that fails to accurately map from noise to data may lead to a discrepancy between the training and test distributions, resulting in artifacts such as noisy audio samples. Although curved paths require many integration steps to simulate the process, a straight path can be simulated in a single step, reducing the risk of error accumulation (Esser et al., 2024). Since each step corresponds to an evaluation of the neural network, this directly influences the sampling speed.

Rectified Flow (RF)(Liu et al., 2022) is a novel flow-based generative model that linearly transfers the source distribution to the target distribution. This property of linear transfer facilitates simulation-free training and faster sampling at inference time. Recently, RF-based methods have garnered significant attention in image synthesis(Liu et al., 2023e; Esser et al., 2024) and text-to-speech applications (Guan et al., 2024). For instance, InstaFlow (Liu et al., 2023e) generates high-quality images in just one step. While this model class offers superior theoretical properties and has proven effective in both image and speech domains, it has not yet been explored for text-to-audio generation.

In this paper, we introduce FlashAudio, a text-to-audio generation model using rectified flows to learn straight flows for fast and high-fidelity output. Traditional RFs uniformly distribute time steps, often resulting in suboptimal performance, as easier tasks consume resources that could be better allocated to more challenging ones. To overcome this, we propose the Bifocal Samplers, which reallocates time steps to focus on more difficult aspects of the task, improving both model efficiency and stability. Additionally, we introduce immiscible flow to minimize the total distance of data-noise pairs in a batch via assignment. Finally, to mitigate the amplified accumulation error caused by classifier-free guidance (CFG), we propose Anchored Optimization, which refines the guidance scale by anchoring it to a reference trajectory at $\omega = 1$, enhancing both the audio quality and text-audio alignment with larger guidance scales.

Experimental results on text-to-audio generation demonstrate that FlashAudio achieves state-of-the-art performance in both objective and subjective metrics with significantly reduced inference time. FlashAudio also outperforms strong

diffusion-based baselines and consistency-based models in both few-step and one-step generation settings. Our extensive preliminary analysis and ablation studies show that each design in FlashAudio is effective. The main contributions of this study are summarized below:

- We propose FlashAudio with rectified flows for fast and high-fidelity text-to-audio generation, which is the first work in TTA to learn straight flows for fast simulation.
- FlashAudio includes bifocal samplers and immiscible flow techniques for efficient and stable training of rectified flow models.
- FlashAudio introduces anchored optimization to refine the guidance scale by anchoring it to a reference trajectory.
- Experimental results demonstrate that FlashAudio achieves state-of-the-art performance with much less inference time in both multi-step and one-step generation. This makes generative models practically applicable for text-to-audio generation deployment.

2 Preliminaries

In this section, we briefly introduce the theory of flow matching and rectified flow.

2.1 Flow Matching

Flow matching (FM) (Chen et al., 2018) is a generative model for training objective to regress onto a target vector field that generates a desired probability path. Let \mathbb{R}^d denote the data space with data point x_0 , flow matching aims to learn vector field $v_\theta(t, x, c) : [0, 1] \times \mathbb{R}^d \mapsto \mathbb{R}^d$, such that the solution of the ordinary differential equation (ODE) can transfer standard Gaussian distribution $z_0 \sim \pi_0$ to latent audio distribution $z_1 \sim \pi_1$:

$$\frac{d\phi_t(z)}{dt} = v_\theta(\phi_t(z), t) \quad (1)$$

where we model the vector field v_t with a neural network $v_\theta(z, t)$. Given the ground truth vector field $u(z, t)$ that generates probability path π_t under the two marginal constraints that $\pi_{t=0} = \pi_0$ and $\pi_{t=1} = \pi_1$, the vector field is learned by minimizing a simple mean square objective (MSE):

$$\min_v \mathbb{E}_{(z_0, z_1) \sim \gamma} \|v_\theta(z_t, t) - u(z_t, t)\|^2 \quad (2)$$

where the γ is any coupling of (π_0, π_1) .

However, it is computationally intractable to find such u and π_t . Instead of directly optimizing Eq. 2, Conditional Flow Matching (CFM) (Lipman et al., 2022) regress $v_\theta(z, t, c)$ on the conditional vector field $u(t, z_t|z_1)$ and probability path $\pi_t(z_t|z_1)$:

$$\min_v \mathbb{E}_{(z_0, z_1) \sim \gamma} \|v_\theta(z_t, t) - u(z_t|z_1, t)\|^2 \quad (3)$$

2.2 Rectified Flow

The rectified flow (Liu et al., 2022) is an ODE model that transport distribution π_0 to π_1 by following straight line paths as much as possible. In rectified flow, the drift force v is set to drive the flow to follow the direction $(z_1 - z_0)$ of the linear path pointing from z_0 to z_1 as much as possible:

$$\min_v \mathbb{E}_{(z_0, z_1) \sim \gamma} \|(z_1 - z_0) - v_\theta(z_t, t)\|^2 \quad (4)$$

where z_t is the linear interpolation of z_0 and z_1 and $z_t = (1 - t)z_0 + tz_1$.

Reflow Reflow is an iterative procedure to straighten the trajectories without modifying the marginal distributions, hence allowing fast simulation at inference time. In text-to-audio generation, the reflow objective is as follows:

$$v_{k+1} = \arg \min_v \mathbb{E}_{z_0 \sim \pi_0} \|(z_1 - z_0) - v_\theta(z_t, t, c)\|^2 \quad (5)$$

where c is the text embeddings and $z_1 = \text{ODE}[v_k](z_0|c)$ and v_{k+1} is optimized using the objective equation 4, but with (Z_0, Z_1) pairs constructed from the previous $\text{ODE}[v_k]$.

3 FlashAudio

This section introduces FlashAudio, a novel rectified flow framework designed for fast and high-fidelity text-to-audio generation. We begin by discussing the motivation behind each design choice in FlashAudio. Subsequently, we detail the selection process for a powerful pre-trained CFM model to initialize the 1-rectified flow model. Next, we elaborate on our advanced training techniques, including bifocal samplers and the immiscible flow method. This is followed by an introduction to reflow and anchored optimization for better few-step performance. Finally, we employ the distillation for one-step generation and display the training, inference procedures and algorithm employed in FlashAudio

3.1 Motivation

Diffusion models (Song and Ermon, 2019; Song et al., 2020) have made notable advancements in domains such as image and audio generation. However, the iterative nature of current latent diffusion text-to-audio models requires extensive computational resources, leading to slow inference times and limited real-time applicability (Luo et al., 2023). This constraint impedes the practical deployment of these models in real-world scenarios. Recent methods (Liu et al., 2024a) have employed consistency distillation to enhance inference speed by mapping points on curved trajectories with initial points. Nonetheless, curved trajectories are prone to greater error accumulation compared to straight paths, which restricts their performance relative to diffusion models in few-step settings.

In contrast, rectified models (Liu et al., 2022) offer a novel generative approach by mapping data and noise along straight paths, facilitating efficient few-step and one-step generation. Despite their promising theoretical benefits, rectified models have been under-explored in the context of text-to-audio generation. To bridge this gap, we introduce advanced training techniques for rectified flow models, such as immiscible flow and bifocal samplers, to ensure efficient and stable training. Furthermore, to address the error amplification caused by Classifier-Free Guidance (CFG) (Ho and Salimans, 2021) during reflow and couplings generation, we propose anchored optimization to enhance sample accuracy for subsequent reflow procedures.

3.2 Initialization from Pre-trained Flow Matching Models

As a blossoming class of generative models, flow matching models (Lipman et al., 2022) have demonstrated their powerful generation abilities across image (Gat et al., 2024) and audio (Vyas et al., 2023), with less sampling steps and more high-quality generated samples. Compared to training rectified flow models from scratch, initializing the neural network with flow matching model benefits in inheriting powerful capabilities and speeding up convergence. The experimental results in Section 4.3 demonstrate the advantage of adopting the pre-trained models.

3.3 Advanced Techniques for RF Models

3.3.1 Bifocal Samplers for Timesteps

In the training of rectified flow models, the common approach is to sample timesteps t uniformly across the interval $[0, 1]$, i.e., $t \sim \text{Uniform}(0, 1)$. However, the choice of sampling distribution significantly influences the effectiveness of the training process. Ideally, more computational resources should be allocated to the timesteps that are more challenging for the model, rather than distributing effort uniformly.

During 1-rectified flow training, the model learns to approximate the dataset’s average when $t = 1$ and the noise average when $t = 0$. This makes the intermediate timesteps within $[0, 1]$ particularly challenging. A practical solution to this issue is to modify the time distribution from the standard uniform distribution to one that prioritizes intermediate timesteps. Inspired by Esser et al. (2024), we adopt the logit-normal distribution (Atchison and Shen, 1980) $p(t)$ for this purpose:

$$p_1(t; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \frac{1}{t(1-t)} \times \exp\left(-\frac{1}{2\sigma^2} [\text{logit}(t) - \mu]^2\right), \quad (6)$$

where the logit function is defined as:

$$\text{logit}(y) = \log\left(\frac{y}{1-y}\right). \quad (7)$$

In this formulation, μ and σ^2 denote the mean and variance in the logit space, respectively. Figure 6 provides a visual representation of the logit-normal distribution.

For the reflow process, which is applied only once in FlashAudio, the 2-rectified flow model learns to directly predict data from noise at $t = 1$ and noise from data at $t = 0$. So it is non-trivial to assign greater emphasis to the timesteps near the noise and data. To tackle this, we employ a mixture of exponential distributions (Mix-Exp) defined as:

$$p_2(t) \propto \exp(at) + \exp(-at), \quad (8)$$

where a serves as the scale parameter. The distribution of Mix-Exp is depicted in Figure 6.

3.3.2 Immiscible Flow Mechanism

The phenomenon of particles becoming tightly jumbled together during diffusion, making them difficult to separate individually, is akin to the challenges observed in rectified flow models. When

particles are rendered immiscible, they maintain a similar overall distribution but remain distinctly identifiable. This concept can be translated into rectified flow models where each audio sample can map to any point in the noise space, and vice versa, making it challenging for the model to differentiate during the reverse process.

To simulate this immiscible phenomenon, we propose a method where each noise point is assigned to a limited set of audio samples, reducing confusion for the rectified flow model. Despite this, the noise space must remain strictly Gaussian to ensure efficient sampling. Drawing inspiration from Li et al. (2024), we introduce the concept of Immiscible Flow. This approach involves assigning batches of noise samples to corresponding batches of audio samples based on their proximity in a shared space, minimizing the total distance between data-noise pairs during training.

The noise remains Gaussian after assignment, with each noise point being associated with closer audio samples, analogous to the immiscible phenomenon. This significantly alleviates the challenges associated with the simulation process. Practically, this assignment can be efficiently executed using the Hungarian algorithm, which can be implemented with Scipy (Virtanen et al., 2020) as follows:

$$\begin{aligned} \mathbf{M} &= \text{linear_sum_assignment}(\text{dist}(z_1, z_0)) \\ z'_0 &= z_0[\mathbf{M}] \\ z_t &= (1-t)z'_0 + tz_1 \end{aligned} \quad (9)$$

3.4 Reflow and Anchored Optimization

3.4.1 Straightening Flows via Reflow

To achieve optimal performance in few-step generative tasks, it is essential to ensure that the flow trajectories are as straight as possible. Training a rectified flow model just once is often insufficient for constructing adequately straight transport paths. Therefore, an additional reflow process is applied using the training objective defined in Eq. 5 to straighten the transport paths. As depicted in Figure 2, the 2-rectified flows achieve near-straight trajectories, effectively eliminating the need for further rectification (e.g., 3-rectified flow).

3.4.2 Anchored Optimization

Classifier-Free Guidance (CFG) (Ho and Salimans, 2021) plays a critical role in generating audios that align with text prompts. During sampling, Liu et al. (2023e) introduce a guided velocity field in the

context of text-conditioned rectified flows using the formula:

$$v_{cfg}(z_t, t, c) = \omega v(z_t, t, c) + (1 - \omega)v(z_t, t, \emptyset) \quad (10)$$

where ω is the guidance scale.

In practice, the multiple regression of the velocity field v_θ and the application of sampling function in generating \hat{z}_1 inevitably incorporate a slight error in every step. These errors are further amplified when a larger guidance scale $\omega > 1$ is used, which is likely to perturb the original marginal distribution. To alleviate these errors, InstalFlow employs a much smaller guidance scale ($\omega = 1.5$) in the final stage while their initial scale is 5. While using $\omega = 1$ can roughly preserve the original marginal distribution, the lack of CFG makes it struggle to balance the audio quality and alignment with text prompts.

In \hat{z}_1 generation stage, inspired by Mokady et al. (2023), we use the initial simulation with $\omega = 1$ as a pivot trajectory and optimize around it with a guidance scale $\omega > 1$ during all paired generation stages. Direct optimization of textual embeddings often leads to non-interpretable representations, as the optimized tokens may not correspond to actual words. Instead, we leverage CFG’s core characteristic: the outcome is strongly influenced by the unconditional prediction. Therefore, we focus on optimizing only the unconditional embedding \emptyset while keeping the model parameters and textual embeddings fixed.

In practice, simulation with $\omega = 1$ generates a sequence $z_0, z_{1/T}^*, \dots, z_1^*$. We initialize with the same Gaussian noise z_0 and perform the following optimization with guidance scale $\omega > 1$ for timesteps $t = 1/T, 2/T, \dots, 1$, iterating T times at each step:

$$\min_{\emptyset_t} \|z_{t-1}^* - z_{t-1}\|^2 \quad (11)$$

Here, z_t represents the intermediate result of the guided sampling. Early stopping is employed to minimize computational time.

In reflow training, we use v_{cfg} as the target of regression, the reflow objective can be written as:

$$\min_v \mathbb{E}_{(z_0, z_1) \sim \gamma} \|(z_1 - z_0) - v_{cfg}(z_t, t)\|^2 \quad (12)$$

3.5 Distillation for One-step Generation

As shown in Figure 2, the trajectories of the 2-rectified flow model become nearly straight after a single reflow step. With these approximately

straight ODEs, a promising approach to enhance the performance of one-step generation is through one-step distillation:

$$\hat{v}_k = \arg \min_v \mathbb{E}_{z_0 \sim \pi_0} \|\text{ODE}[v_k] - (z_0 + v_{cfg}(z_0, c))\|^2 \quad (13)$$

In this process, we learn a single Euler step $z + v(z|\mathbb{T})$ to compress the mapping from z_0 to $\text{ODE}[v_k](z_0|\mathcal{T})$, effectively simplifying and accelerating the generation process.

3.6 Training and Inference Procedures

The training algorithm for FlashAudio, incorporating rectified flows, is detailed in Algorithm 1.

Algorithm 1 Training FlashAudio with Rectified Flows

- 1: **Input:** Initialize the velocity field network with pre-trained flow matching; a dataset of text prompts $\mathcal{D}_\mathcal{T}$; encoding training data into latent space \mathcal{D}_z .
 - 2: Sample $z_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $z_1 \sim \mathcal{D}_z$.
 - 3: Re-weight timestep t using the **logit-normal distribution** and assign z_0 via **immiscible flow**.
 - 4: Train the 1-rectified flow v_1 by minimizing the objective in Equation 4.
 - 5: Generate the couplings $[z'_0, z'_1]$ for reflow with **anchored optimization**.
 - 6: Re-weight timestep t using the **Mix-Exp distribution** and assign z_0 via **immiscible flow**.
 - 7: Train the 2-rectified flow v_2 by minimizing the objective in Equation 12.
 - 8: Generate the couplings $[z''_0, z''_1]$ with **anchored optimization** for one-step distillation.
 - 9: Train the final model \hat{v}_2 by minimizing the objective in Equation 13.
-

3.6.1 Training Procedure

The training process begins by training the 1-rectified flow models using a pre-trained flow matching model, followed by the generation of reflow couplings. Our experiments indicate that the 1-rectified model performs exceptionally well in the CFG setting, whereas the 2-rectified model with $\omega > 1$ fails to achieve comparable performance with $\omega = 1$. For detailed experimental analysis, please refer to Section 4.2.3. Therefore, anchored optimization is conducted when generating couplings $[z'_0, z'_1]$ and $[z''_0, z''_1]$. All rectified

models utilize Mean Squared Error (MSE) loss as the distance function.

3.6.2 Inference Procedure

For inference, we adopt the Euler Solver for fast simulation, which is governed by the equation:

$$z_{t+\Delta t} = z_t + \Delta t v(z_t, t | \mathcal{T}) \quad (14)$$

Here, the simulation is performed with a step size of $\Delta t = 1/T$, completing the process in T steps. In the few-step setting, we utilize the v_2 model for sampling. In the one-step setting, we employ the \hat{v}_2 model to generate audio latents. These latents are then transformed into mel-spectrograms, which are subsequently used to generate waveforms using a pre-trained vocoder.

4 Experiments

4.1 Experimental setup

4.1.1 Dataset

For text-to-sound generation, we use the training split of Audiocaps dataset (Kim et al., 2019) to train all our models. For text-to-music generation, we exclusively employ the LP-Musicaps (Doh et al., 2023) dataset for FlashAudio training endeavors.

4.1.2 Model configurations

Initialized model is originally trained with objective equation 3, we utilize its pre-trained VAE, a continuous 1D-convolution-based network. This VAE is used to compress the mel-spectrogram into a 20-channel latent representation with a temporal axis downsampling rate of 2. Training 1-rectified flow involves 40,000 iterations on 4 NVIDIA 4090Ti GPU, with a batch size of 16 per GPU. We use the AdamW optimizer with a learning rate of $4.8e-5$. Our choice for the vocoder is BigVGAN-v2 (Lee et al., 2023), which is known for its universal applicability to different scenarios. We train the vocoder on the AudioSet dataset to ensure robust performance. More details on the model configuration can be found in Appendix B.

4.1.3 Evaluation Metrics

Our models conduct a comprehensive evaluation (Cui et al., 2021) using both objective and subjective metrics to measure audio quality, text-audio alignment fidelity, and inference speed. Objective assessment includes Kullback-Leibler (KL) divergence, Frechet audio distance (FAD), and CLAP

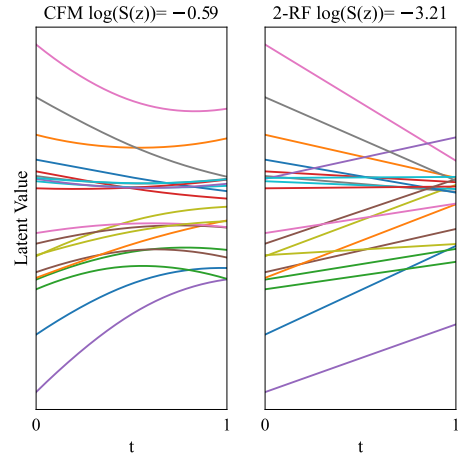


Figure 2: The straightness and simulation figure of flow matching and 2-rectified flow (2-RF) models. The top of each subfigure has its corresponding value of $\log(S(z))$

score to quantify audio quality. The Real-time Factor (RTF) is also introduced to measure the system’s efficiency in generating audio for real-time applications. RTF is the ratio between the total time taken by the audio system to synthesize an audio sample and the duration of the audio. In terms of subjective evaluation, we conduct crowd-sourced human assessments employing the Mean Opinion Score (MOS) to evaluate both audio quality (MOS-Q) and text-audio alignment faithfulness (MOS-F). Detailed information regarding the evaluation procedure can be accessed in Appendix D.

4.2 Preliminary Analyses

In this section, we analyze 2-rectified flows with a range of few-step settings (from 16 to 1) and investigate the influence of CFG on the training of rectified flows. Additionally, we explore the effects of reflow on straightening flows and clarify why reflow is applied only once.

4.2.1 Straighen Effects of Reflow

We evaluate the straightness of the 2-rectified flow model and analyze the effectiveness of the reflow procedure in achieving straighter flows. Following the methodology of InstalFlow, we quantify the straightness by measuring the deviation of the velocity along the trajectory. Specifically, we use the metric:

$$S(z) = \int_0^1 \mathbb{E} [\|(z_1 - z_0) - v(z_t, t)\|^2] dt, \quad (15)$$

where $S(z)$ represents the mean squared deviation of the velocity from the desired trajectory. We

compare the trajectories of flow matching and 2-rectified flow models through simulation. As illustrated in Figure 2, reflow significantly reduces the estimated $S(z)$, demonstrating its effectiveness in straightening the flow trajectories. Furthermore, the simulation results show that the 2-rectified model achieves nearly straight trajectories.

4.2.2 Straight Flows Yield Fast Generation

In both diffusion and flow matching models, selecting the sampling step T involves a trade-off between computational cost and accuracy: a larger T provides a better approximation of the ODE but increases computational expense, while a smaller T may struggle to maintain accuracy. For efficient simulation, it is essential to learn ODEs that can be accurately and rapidly simulated with a small T . To address this, we compared our nearly straight 2-RF models with curved baselines such as flow matching and consistency models in a few-step setting. The results, presented in Figure 3, lead to several key observations: (1) Compared to curved flow matching models, our proposed 2-RF models consistently deliver superior performance across all sampling steps, from 16 to 1. This finding underscores the importance of straightening the ODE trajectories to enhance performance in fast generation tasks. (2) Our 2-RF models also outperform AudioLCM, particularly in the one-step setting, with a lower FAD (2.24 vs. 3.86) and a higher CLAP score (0.63 vs. 0.583). This demonstrates the superiority of our rectified flow approach in few-step generation scenarios.

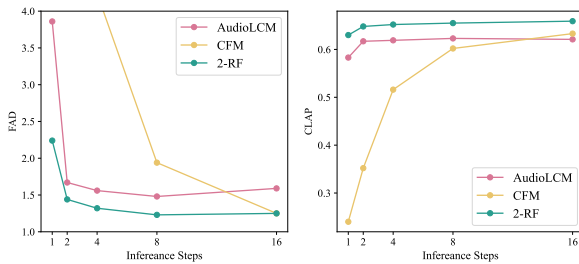


Figure 3: Comparisons with AudioLCM and conditional flow matching in few-step setting, measured by fad and CLAP score.

4.2.3 Impact of CFG on Audio Quality

Rectified Flows utilize reflow to straighten trajectories while maintaining the marginal distribution. However, the deployment of CFG is likely to perturb the distribution. We conduct experiments on rectified flow to explore the impact of CFG in RF and the effect of anchored optimization. The

key observations are as follows: (1) As the guidance scale increases, there is a notable decline in audio quality for original RF models. This supports our assumption in Section 3.4.2. (2) After adding anchored optimization, it leads to a significant improvement in audio quality and text alignment. Specifically, the FAD score decreases from 1.43 to 1.26 and the CLAP score increases from 0.639 to 0.652. This demonstrates the effectiveness of our proposed anchored optimization.

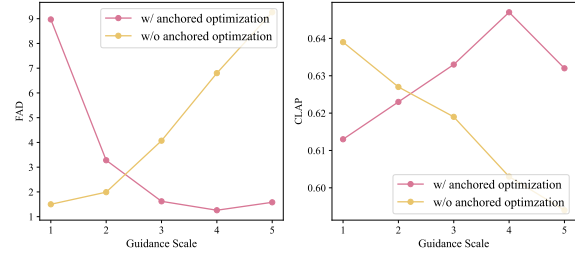


Figure 4: Comparison between with and without Anchored Optimization varying different guidance scale.

4.3 Performance on Text-to-Sound Generation

We conduct a comparative analysis of the quality of generated audio samples and inference latency across various systems, including GT (i.e., ground-truth audio), AudioGen (Kreuk et al., 2023a), Make-An-Audio (Huang et al., 2023c), AudioLDM 2 (Liu et al., 2023b), TANGO 2 (Majumder et al., 2024), Make-An-Audio 2 (Huang et al., 2023a), ConsistencyTTA (Bai et al., 2024), SoundCTM (Saito et al., 2024), AudioLCM (Liu et al., 2024a), and our constructed CFM. utilizing the published models as per the respective paper. The evaluations are conducted using the AudioCaps test set and then calculate the objective and subjective metrics. The results are compiled and presented in Table 1. From these findings, we draw the following conclusion:

Audio Quality In terms of audio quality, we first compare FlashAudio’s 2-RF models with our constructed CFM and other baselines. With the same number of steps as CFM, FlashAudio not only outperforms CFM but also significantly surpasses other baselines, achieving a notably lower FAD score of 1.18. In a few-step setting with four steps, FlashAudio remains competitive with CFM and markedly outperforms diffusion-based models and AudioLCM. Further evaluation of FlashAudio’s one-step generation performance, enhanced by additional distillation, shows substantial im-

Model	NFE	Objective Metrics				Subjective Metrics	
		FAD (\downarrow)	KL (\downarrow)	CLAP (\uparrow)	RTF (\downarrow)	MOS-Q(\uparrow)	MOS-F(\uparrow)
GT	/	/	/	0.670	/	87.90	85.48
AudioGen-Large	/	1.74	1.43	0.601	1.890	/	/
Make-An-Audio	100	2.45	1.59	0.616	0.280	69.79	66.19
AudioLDM 2	100	1.90	1.48	0.622	1.250	73.38	71.22
Make-An-Audio 2	100	1.80	1.32	0.645	0.170	75.56	73.14
Tango 2	100	2.84	1.20	0.680	0.800	73.46	72.08
CFM	24	1.22	1.34	0.640	0.054	76.32	74.75
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \times$)	24	1.18	1.28	0.658	0.054	78.86	76.98
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \times$)	4	1.26	1.30	0.652	0.014	78.23	76.14
AudioLCM	2	1.67	1.37	0.617	0.003	76.48	73.92
ConsistencyTTA	1	2.13	1.33	0.655	0.004	73.19	70.08
SoundCTM	1	1.95	1.36	0.656	0.01	73.87	71.16
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \times$)	1	2.24	1.40	0.630	0.0025	72.67	70.84
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \checkmark$)	1	1.49	1.32	0.648	0.0025	77.56	75.43

Table 1: The audio quality and sampling speed comparisons. The evaluation is conducted on a server with 1 NVIDIA 4090Ti GPU and batch size 1. NFE (number of function evaluations) measures the computational cost, which refers to the total number of times the denoiser function is evaluated during the generation process. \mathcal{R} denotes reflow and \mathcal{D} denotes one-step distillation.

provements in both objective and subjective metrics. FlashAudio excels over consistency-based models in nearly all metrics, aside from a slightly lower CLAP score, and also achieves superior audio quality compared to diffusion-based models—a notable advancement that consistency-based models have yet to achieve. These observations suggest that FlashAudio possesses the capability to generate high-fidelity audio with fast simulation.

Sampling Speed FlashAudio surpasses diffusion models and consistency-based models with an exceptional RTF of 0.0025, demonstrating superior speed while maintaining competitive audio quality. This translates to a remarkable speed, approximately 400x faster than real-time, on a single NVIDIA 4090Ti GPU. This impressive reduction in inference time establishes FlashAudio as a leading solution in efficient TTA generation.

4.4 Ablation Studies

In this section, we conduct ablation studies to validate the effectiveness of initialization from CFM, our proposed bifocal samplers, and immiscible flow. We attach more exploratory experimental results to Appendix C.

The results are presented in Table 2, and we have the following observations: (1) The significant improvement of all metrics from ID 1 to 0 demonstrates the effectiveness of initialization from pre-trained condition flow matching. (2) Although the adoption of Logit-Norm in the training of 1-rectified flow slightly causes the decrease of FAD score, the improvement of KL and CLAP validates the positive impact of re-weighting timesteps

in middle timesteps, with the KL from 1.27 to 1.25 and CLAP score from 0.649 to 0.659. And ID 4 outperforms ID 5 across all objective metrics, which shows the necessity to concentrate on the boundary of timesteps between $[0, 1]$. These two sets of results demonstrate the effectiveness of our proposed bifocal samplers for improving the training of rf. (3) The degradation from ID 0 to ID 3 and from ID 4 to ID 6 proves that the immiscible flow mechanism plays an important role in RFs training.

ID	Model	FAD (\downarrow)	KL (\downarrow)	CLAP (\uparrow)
0	1-rf w/ LN	1.12	1.25	0.659
2	1-rf w/o LN	1.08	1.27	0.649
1	1-rf w/o Init	1.49	1.55	0.602
3	1-rf w/o Immi	1.19	1.35	0.648
4	2-rf w/ Mix-Exp	1.18	1.28	0.658
5	2-rf w/o Mix-Exp	1.23	1.28	0.637
6	2-rf w/o Immi	1.26	1.29	0.641

Table 2: Ablation studies results about FlashAudio. LN: Logit Norm Distribution, Immi: Immiscible Flow, Init: Initialization from pre-trained flow matching. The "w/o LN" and "w/o Mix-Exp" mean the adoption of uniform distribution.

5 Conclusion

In this work, we presented FlashAudio, a novel rectified flow model designed to overcome the limitations of iterative sampling in text-to-audio generation. By introducing Bifocal Samplers and Immiscible Flow, FlashAudio optimized time step allocation and noise distribution, addressing key inefficiencies in the rectified flow process. The proposed Anchored Optimization method further mitigated

the impact of classifier-free guidance errors by anchoring the guidance scale to a reference trajectory, thus stabilizing the generation process. Experimental evaluations demonstrated that FlashAudio achieved superior performance in both speed and quality, outperforming existing diffusion-based and consistency-based models, even with one-step generation. Comprehensive analysis and ablation studies confirmed the effectiveness of FlashAudio’s components, highlighting its potential for high-fidelity text-to-audio applications. We envisage that our work could serve as a basis for future text-to-audio studies.

6 Limitation And Potential Risks

Despite FlashAudio’s notable performance in both few-step and one-step generation settings, it has two primary limitations: (1) GPU resource constraints have limited our ability to explore the scaling potential of our transformer backbone. Future work will focus on expanding the model size of FlashAudio and applying it to a broader range of real-world scenarios. (2) The current model design is restricted to generating fixed-length audio, lacking the capability to produce audio of varying lengths. Addressing these limitations will be a focus of future research efforts.

This paper focuses on achieving efficient and high-fidelity text-to-audio generation, making generative models more feasible for real-world text-to-audio deployment. Meanwhile it could lead to bias and discrimination. If the training data contains biases, the generative model may not only inherit these biases but also amplify them. This can result in the generated audio content exhibiting discriminatory tendencies related to race, gender, or other social categories. Such outcomes can be damaging to affected groups and lead to unfair decisions or behaviors, making it crucial to identify and address biases in the training data during model development.

Acknowledgements

The research was supported by NSFC (No.62206234) from Mainland China, Early Career Scheme (ECS-HKUST22201322) from Hong Kong RGC, and National Natural Science Foundation of China under Grant No.62222211 and No.U24A20326.

References

- Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. 2023. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*.
- Jhon Atchison and Sheng M Shen. 1980. Logistic-normal distributions: Some properties and uses. *Biometrika*, 67(2):261–272.
- Yatong Bai, Trung Dang, Dung Tran, Kazuhito Koishida, and Somayeh Sojoudi. 2024. [Consistencytta: Accelerating diffusion-based text-to-audio generation with consistency distillation](#). *Preprint*, arXiv:2309.10740.
- Ke Chen, Yusong Wu, Haohe Liu, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2024. Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1206–1210. IEEE.
- Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. 2018. Neural ordinary differential equations. *Advances in neural information processing systems*, 31.
- Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. [Simple and controllable music generation](#). *Preprint*, arXiv:2306.05284.
- Chenye Cui, Yi Ren, Jinglin Liu, Feiyang Chen, Rongjie Huang, Ming Lei, and Zhou Zhao. 2021. Emovie: A mandarin emotion speech dataset with a simple emotional text-to-speech model. *arXiv preprint arXiv:2106.09317*.
- SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. 2023. [Lp-musiccaps: Llm-based pseudo music captioning](#). *Preprint*, arXiv:2307.16372.
- Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*.
- Seth Forsgren and Hayk Martiros. 2022. Riffusion-stable diffusion for real-time music generation. *URL <https://riffusion.com>*.
- Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky T. Q. Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. 2024. [Discrete flow matching](#). *Preprint*, arXiv:2407.15595.
- Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. 2023. [Text-to-audio generation using instruction-tuned llm and latent diffusion model](#). *Preprint*, arXiv:2304.13731.

- Wenhao Guan, Qi Su, Haodong Zhou, Shiyu Miao, Xingjia Xie, Lin Li, and Qingyang Hong. 2024. [Reflow-tts: A rectified flow model for high-fidelity text-to-speech](#). *Preprint*, arXiv:2309.17056.
- Yiwei Guo, Chenpeng Du, Ziyang Ma, Xie Chen, and Kai Yu. 2024. Voiceflow: Efficient text-to-speech with rectified flow matching. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11121–11125. IEEE.
- Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Li Fei-Fei, Irfan Essa, Lu Jiang, and José Lezama. 2023. [Photorealistic video generation with diffusion models](#). *Preprint*, arXiv:2312.06662.
- Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: a reference-free evaluation metric for image captioning. In *EMNLP*.
- Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. 2022. [Imagen video: High definition video generation with diffusion models](#). *Preprint*, arXiv:2210.02303.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020a. Denoising diffusion probabilistic models. In *Proc. of NeurIPS*.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020b. [De-noising diffusion probabilistic models](#). *Preprint*, arXiv:2006.11239.
- Jonathan Ho and Tim Salimans. 2021. Classifier-free diffusion guidance. In *NeurIPS 2021 workshop on deep generative models and downstream applications*.
- Jiawei Huang, Yi Ren, Rongjie Huang, Dongchao Yang, Zhenhui Ye, Chen Zhang, Jinglin Liu, Xiang Yin, Zejun Ma, and Zhou Zhao. 2023a. [Make-an-audio 2: Temporal-enhanced text-to-audio generation](#). *Preprint*, arXiv:2305.18474.
- Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. 2023b. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. *arXiv preprint arXiv:2301.12661*.
- Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. 2023c. [Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models](#). *Preprint*, arXiv:2301.12661.
- Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. 2018. Fr²echet audio distance: A metric for evaluating music enhancement algorithms. *arXiv preprint arXiv:1812.08466*.
- Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. 2019. AudioCaps: Generating captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 119–132.
- Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. 2024. [Consistency trajectory models: Learning probability flow ode trajectory of diffusion](#). *Preprint*, arXiv:2310.02279.
- Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. 2023a. [AudioGen: Textually guided audio generation](#). *Preprint*, arXiv:2209.15352.
- Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. 2023b. AudioGen: Textually Guided Audio Generation.
- Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, et al. 2024. Voicebox: Text-guided multilingual universal speech generation at scale. *Advances in neural information processing systems*, 36.
- Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. 2023. BigVGAN: A Universal Neural Vocoder with Large-Scale Training.
- Yiheng Li, Heyang Jiang, Akio Kodaira, Masayoshi Tomizuka, Kurt Keutzer, and Chenfeng Xu. 2024. [Immiscible diffusion: Accelerating diffusion training with noise assignment](#). *Preprint*, arXiv:2406.12303.
- Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.
- Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D. Plumbley. 2023a. AudioLDM: Text-to-Audio Generation with Latent Diffusion Models.
- Haohe Liu, Qiao Tian, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. 2023b. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *arXiv preprint arXiv:2308.05734*.
- Huadai Liu, Rongjie Huang, Jinzheng He, Gang Sun, Ran Shen, Xize Cheng, and Zhou Zhao. 2023c. Wav2sql: Direct generalizable speech-to-sql parsing. *arXiv preprint arXiv:2305.12552*.

- Huadai Liu, Rongjie Huang, Xuan Lin, Wenqiang Xu, Maozong Zheng, Hong Chen, Jinzheng He, and Zhou Zhao. 2023d. Vit-tts: visual text-to-speech with scalable diffusion transformer. *arXiv preprint arXiv:2305.12708*.
- Huadai Liu, Rongjie Huang, Yang Liu, Hengyuan Cao, Jialei Wang, Xize Cheng, Siqi Zheng, and Zhou Zhao. 2024a. [Audiolcm: Text-to-audio generation with latent consistency models](#). *Preprint*, arXiv:2406.00356.
- Huadai Liu, Jialei Wang, Rongjie Huang, Yang Liu, Jiayang Xu, and Zhou Zhao. 2024b. Medic: Zero-shot music editing with disentangled inversion control. *arXiv preprint arXiv:2407.13220*.
- Qiang Liu. 2022. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*.
- Xingchao Liu, Chengyue Gong, and Qiang Liu. 2022. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*.
- Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. 2023e. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*.
- Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. 2023. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*.
- Navonil Majumder, Chia-Yu Hung, Deepanway Ghosal, Wei-Ning Hsu, Rada Mihalcea, and Soujanya Poria. 2024. [Tango 2: Aligning diffusion-based text-to-audio generations through direct preference optimization](#). *Preprint*, arXiv:2404.09956.
- Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2023. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047.
- Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. [Learning transferable visual models from natural language supervision](#). *Preprint*, arXiv:2103.00020.
- Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.
- Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding.
- Koichi Saito, Dongjun Kim, Takashi Shibuya, Chieh-Hsin Lai, Zhi Zhong, Yuhta Takida, and Yuki Mitsufuji. 2024. Soundctm: Uniting score-based and consistency models for text-to-sound generation. *arXiv preprint arXiv:2405.18503*.
- Tim Salimans and Jonathan Ho. 2022. [Progressive distillation for fast sampling of diffusion models](#). *Preprint*, arXiv:2202.00512.
- Flavio Schneider, Zhijing Jin, and Bernhard Schölkopf. 2023. [Moûsai: Text-to-music generation with long-context latent diffusion](#). *Preprint*, arXiv:2301.11757.
- Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. In *Proc. of ICLR*.
- Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. 2023. Consistency models. *arXiv preprint arXiv:2303.01469*.
- Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.
- Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2021. [Score-based generative modeling through stochastic differential equations](#). *Preprint*, arXiv:2011.13456.
- Aaron Van Den Oord, Oriol Vinyals, et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. [Attention is all you need](#). *Preprint*, arXiv:1706.03762.
- Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. 2020. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272.
- Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, Jeff Wang, Ivan Cruz, Bapi Akula, Akinniyi Akinyemi, Brian Ellis, Rashel Moritz, Yael Yungster, Alice Rakotoarison, Liang Tan, Chris Summers, Carleigh Wood, Joshua Lane, Mary Williamson, and Wei-Ning Hsu. 2023. [Audiobox: Unified audio generation with natural language prompts](#). *Preprint*, arXiv:2312.15821.

Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. 2023. Diff-sound: Discrete Diffusion Model for Text-to-sound Generation.

Zhen Ye, Zeqian Ju, Haohe Liu, Xu Tan, Jianyi Chen, Yiwen Lu, Peiwen Sun, Jiahao Pan, Weizhen Bian, Shulin He, et al. 2024. Flashspeech: Efficient zero-shot speech synthesis. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6998–7007.

A Related Works

A.1 Text-to-audio Generation

Text-to-Audio Generation is an emerging task that has witnessed notable advancements in recent years. For instance, DiffSound (Yang et al., 2023) leverages a pre-trained VQ-VAE (Van Den Oord et al., 2017) on mel-spectrograms to encode audio into discrete codes, subsequently utilized by a diffusion model for audio synthesis. AudioGen (Kreuk et al., 2023b) frames text-to-audio generation as a conditional language modeling task, while Make-An-Audio (Huang et al., 2023c), AudioLDM 2 (Liu et al., 2023a), and TANGO (Ghosal et al., 2023) are all founded on the Latent Diffusion Model (LDM), which significantly enhances sample quality. However, a notable drawback of diffusion models (Liu et al., 2023d, 2024b) lies in their iterative sampling process, leading to slow inference and restricting real-world applications. Recent work has thus focused on consistency distillation to achieve few-step and one-step generation. For instance, AudioLCM leverages consistency distillation with a multi-step ODE solver, achieving two-step generation performance comparable to diffusion models. Other works adopt one-step generation but struggle to align with LDMs. In this work, we introduce a novel rectified flow model for fast and high-fidelity text-to-audio generation and achieve superior performance in both few-step and one-step generation.

A.2 Flow-based Models

Flow matching (Lipman et al., 2022) models the vector field of the transport probability path from noise to data samples. Compared to score-based models like DDPM (Ho et al., 2020b), flow matching offers more stable and robust training, along with superior performance. Specifically, rectified flow matching (Liu et al., 2022) learns the transport ODE to follow straight paths between noise and data points as closely as possible, reducing transport costs and enabling fewer sampling steps through the reflow technique. This approach has shown exceptional performance in accelerating image generation (Liu et al., 2023e; Liu, 2022; Esser et al., 2024). In the realm of audio generation, Voicebox (Le et al., 2024) employs flow matching to develop a large-scale multi-task speech generation model. Its successor, Audiobox (Vyas et al., 2023; Ye et al., 2024), expands this approach into a unified audio generation model guided by natural language prompts. VoiceFlow (Guo et al., 2024) in-

troduces rectified flow matching into text-to-speech (TTS), achieving speech generation with fewer inference steps. However, in the task of text-to-audio (TTA) generation, rectified flow models have not yet been explored to enhance sample efficiency and high-quality audio generation.

B Architecture

We list the hyper-parameters of FlashAudio in Table 3 and the model architecture of FlashAudio is displayed in Figure 5.

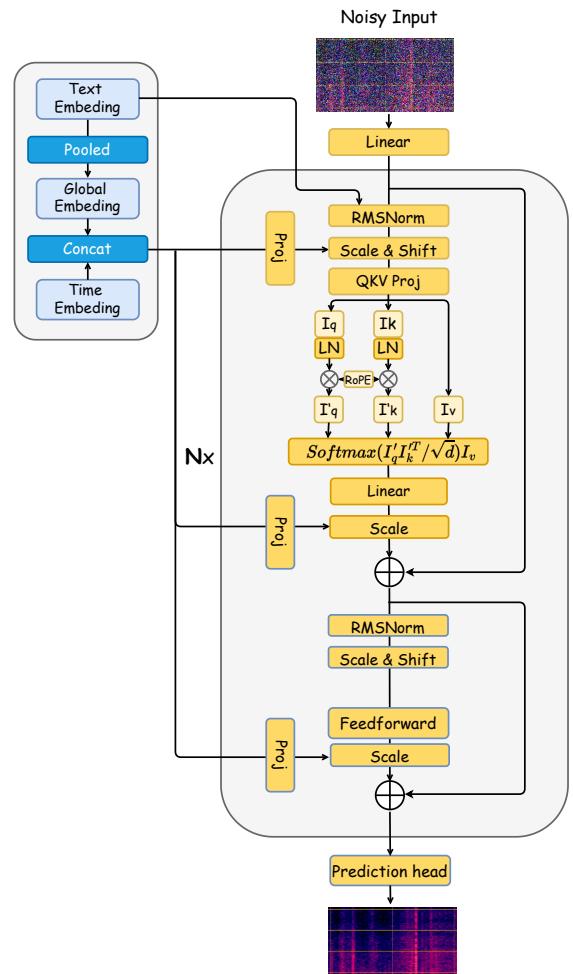


Figure 5: The architecture of FlashAudio.

C Additional Quantitative Comparison

Due to the page limit, we provide additional quantitative results with performance on text-to-music generation, scalable model size of transformer backbone (Vaswani et al., 2023; Liu et al., 2023c), and more ablation results on bifocal samplers.

Hyperparameter		FlashAudio
Spectrogram Autoencoders	Input/Output Channels	80
	Hidden Channels	20
	Residual Blocks	2
	Spectrogram Size	80×624
	Channel Mult	[1, 2, 4]
Transformer Backbone	Input shape	(20, T)
	Condition Embed Dim	1024
	Feed-forward Hidden Size	768
	Transformer Heads	32
	Transformer Blocks	16
CLAP Text Encoder	Transformer Embed Channels	768
	Output Project Channels	1024
	Token Length	77

Table 3: Hyperparameters of FlashAudio models.

Model	NFE	FAD (\downarrow)	Objective Metrics			Subjective Metrics	
			KL (\downarrow)	CLAP (\uparrow)	RTF (\downarrow)	MOS-Q(\uparrow)	MOS-F(\uparrow)
GroundTruth	/	/	/	0.46	/	89.32	91.04
Riffusion	/	13.31	2.10	0.19	0.40	75.23	76.12
Mousai	/	7.50	/	/	/	/	/
Melody	/	5.41	/	/	/	/	/
MusicLM	/	4.00	/	/	/	/	/
MusicGen	/	4.50	1.41	0.42	1.28	81.15	84.21
MusicLDM	200	5.20	1.47	0.40	1.40	79.34	82.10
AudioLDM 2	200	3.81	1.22	0.43	2.20	82.42	85.64
AudioLCM	2	3.92	1.24	0.40	0.003	82.56	85.71
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \times$)	4	3.25	1.20	0.44	0.014	83.96	85.83
FlashAudio ($\mathcal{R} \checkmark \mathcal{D} \checkmark$)	1	3.48	1.22	0.042	0.0025	83.14	85.06

Table 4: The comparison between FlashAudio and baseline models on the MusicCaps Evaluation set. We borrow the results of Mousai, Melody, and MusicLM from the MusicGen (Copet et al., 2023).

C.1 Performance on Text-to-Music Generation

We perform a comparative analysis of audio samples generated by FlashAudio against several established music generation systems. These include: 1) GT, the ground-truth audio; 2) MusicGen (Copet et al., 2023); 3) MusicLM (Agostinelli et al., 2023); 4) Mousai (Schneider et al., 2023); 5) Riffusion (Forsgren and Martiros, 2022); 6) MusicLDM (Chen et al., 2024); 7) AudioLDM 2 (Liu et al., 2023b); 8) AudioLCM (Liu et al., 2024a). The results are presented in Table 4, and we have the following observations: (1) In terms of audio quality, FlashAudio outperforms all diffusion-based methods and language models across a spectrum of both objective and subjective metrics with a much less inference time. This highlights FlashAudio’s effectiveness in producing high-quality music samples and establishes it as a leading model in audio synthesis. (2) In terms of sampling speed, FlashAudio stands out for its exceptional efficiency.

The optimal sampling speed of FlashAudio requires only RTF of 0.0025 while maintaining high-quality output. This illustrates its potent capability to strike an optimal balance between the quality of the samples and the time required for inference.

C.2 Analyses about Scalable Transformer

We investigate the performance of a novel transformer backbone designed to scale up the trainable parameters, as showcased in Table 5. We observe that when the model parameters are reduced to 74M, performance declines across all metrics. However, when the parameter number increases to 429M, there is a performance degradation across multiple metrics. We speculate that this anomalous phenomenon may be due to the convergence difficulty for the larger model under similar training steps, or the redundant model capacity tends to cause overfitting on a relatively small dataset like Audiotaps, deteriorating the model’s generalization performance. Due to the limit of GPU

resources, we cannot expand the batch size to a larger value which also has a negative effect on the larger model.

Model	Parameters	FAD	KL	CLAP
Small	74M	1.40	1.43	0.639
Base	197M	1.18	1.28	0.658
Large	429M	1.20	1.30	0.657

Table 5: The presented figures only account for trainable parameters, i.e., those within the transformer architecture, evaluated on AudioCaps.

We evaluate the performance of various parameters for the logit-normal and Mix-Exp distributions. The results, summarized in Table 6, reveal the following insights: (1) As illustrated in Figure 6, Logit(-0.5,1.0) and Logit(0.5,1.0) emphasize the left and right regions of the timesteps, respectively, while Logit(0.0,1.0) focuses on the center of the timesteps. The Logit(0.0,1.0) distribution achieves the best performance, which supports our view that 1-RF should prioritize the middle timesteps. (2) Among the Mix-Exp distributions, $a = 4$ produces the best results. Therefore, we adopt $a = 4$ for training 2-RF.

Model	FAD	KL	CLAP
Logit(-0.5,1.0)	1.13	1.30	0.641
Logit(0.5,1.0)	1.20	1.26	0.647
Logit(0.0,1.0)	1.12	1.25	0.659
Mix-Exp(1)	1.24	1.31	0.636
Mix-Exp(2)	1.23	1.33	0.638
Mix-Exp(3)	1.20	1.30	0.649
Mix-Exp(4)	1.18	1.28	0.658

Table 6: Performance of different parameters for the logit-normal and Mix-Exp distributions.

D Evaluation

D.1 Subjective evaluation

To directly reflect the quality of the audio generated, we carry out MOS (Mean Opinion Score) tests. These tests involve scoring two aspects: MOS-Q, which assesses the quality of the audio itself, and MOS-F, which measures the faithfulness of the alignment between the text and the audio.

For assessing audio quality, the evaluators were specifically directed to “concentrate on quality and naturalness of the audio.” They were provided with audio samples and asked to give their subjective rating (MOS-Q) on a 20-100 Likert scale.

To assess text-audio faithfulness, human evaluators were presented with both the audio and its corresponding caption. They were then asked to

answer the question, “Does the natural language description align with the audio?” The raters had to select one of the options: “completely,” “mostly,” or “somewhat,” using a 20-100 Likert scale for their response.

Our crowd-sourced subjective evaluation tests were conducted via Amazon Mechanical Turk where participants were paid \$8 hourly.

D.2 Objective evaluation

The Fréchet Audio Distance (FAD) (Kilgour et al., 2018), adapted from the Fréchet Inception Distance (FID) for the audio domain, is a reference-free perceptual metric designed to measure the difference between the distributions of generated audio and ground truth audio. FAD is commonly used to assess the quality of generated audio content.

KL divergence is calculated at the level of paired samples between the generated audio and the ground truth audio. This metric is determined by the label distribution and is then averaged to produce the final result.

CLAP score, adapted from the CLIP score (Hessel et al., 2021; Radford et al., 2021) for the audio domain, is a reference-free evaluation metric to measure audio-text alignment for this work that closely correlates with human perception.

E Bifocal Samplers for Rectified Flow Models

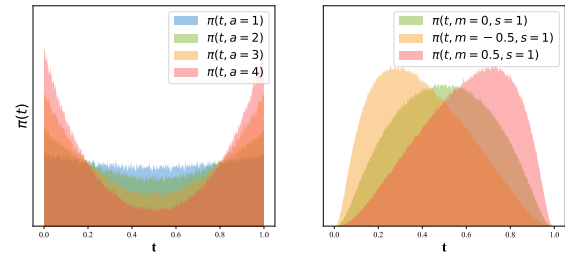


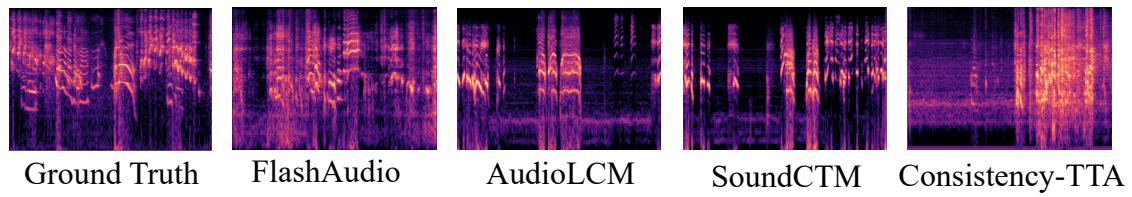
Figure 6: The Mix-Exp distribution (left) and logit-normal distributions (right), which are used in biasing the sampling of training timesteps.

As discussed in Section 3.3.1 we introduce a novel sampler named Bifocal sampler which combines a mixture of exponential distribution (Mix-Exp) and logit-normal distribution in different processing stages respectively, which leads to obvious improvement. The distribution of both Mix-Exp sampler and logit-normal sampler are been visualized in Figure 6. The peaks of these two distributions are located at the middle of t and the boundary of t , respectively. The use of two distinct peaks at

different stages is why we refer to this sampling strategy as the Bifocal Samplers.

F QUALITATIVE RESULTS

We visualize FlashAudio with mel-spectrograms and compare it with baselines in both multi-step generation and one-step generation setting in Figure 7. FlashAudio is prone to generate the most similar mel-spectrum with the ground-truth while other baselines like Consistency-TTA produces noisy output and SoundCTM deviate from the GT. Compared to diffusion-based models, FlashAudio shows the capabilities to generate clear and high-fidelity mel-spectrum.

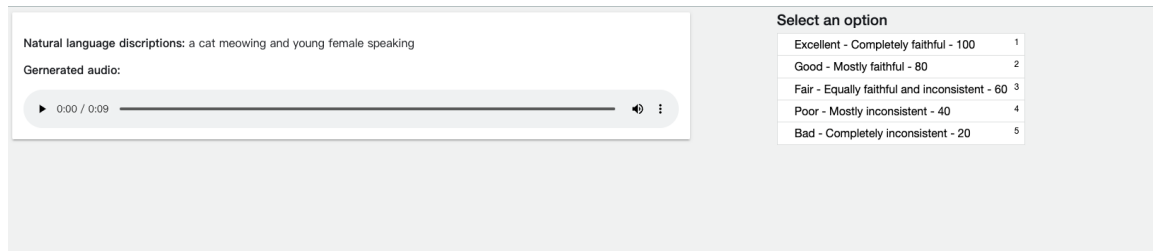


(a) One-step Generation Comparisons

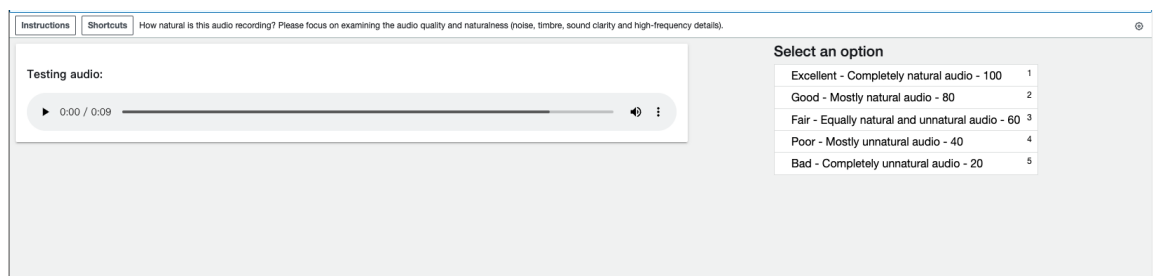


(b) Multi-step Generation Comparisons

Figure 7: Mel-Spectrums for caption: A man and woman laughing followed by a man shouting then a woman laughing as a child laughs.



(a) Screenshot of MOS-F testing.



(b) Screenshot of MOS-Q testing.

Figure 8: Screenshots of subjective evaluations.