FADE: Fractional Anomalous Dynamics Extrapolation for Training-Free Diffusion Transformers Acceleration

Jinlong Yang
Sichuan University
Chengdu, China
2023141461083@stu.scu.edu.cn

Lizilin
Sichuan University
Chengdu, China
2023141461067@stu.scu.edu.cn

Jinke Wu
Sichuan University
Chengdu, China
w_rmsl@stu.scu.edu.cn

Yao Zhou*
School of Artificial Intelligence
Sichuan University
Chengdu, China
yaozhou@scu.edu.cn

Abstract

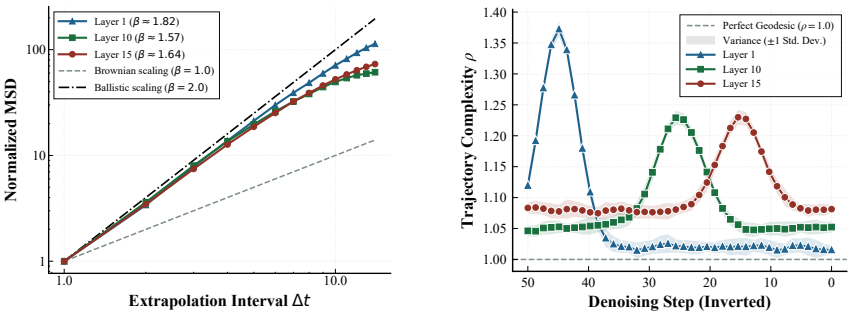
Diffusion Transformers (DiTs) have established themselves as the preeminent paradigm for high-fidelity generative modeling across various modalities, yet their practical deployment is hindered by the severe computational overhead of iterative sampling. While recent feature-caching methods mitigate this latency through feature reuse or forecasting based on integer-order operators, they remain limited in adapting to the non-stationary and highly curved dynamics of DiT latent trajectories. In this work, we reveal that DiT feature evolution exhibits anomalous diffusion characteristics and a certain degree of spatiotemporal heterogeneity, rendering static, integer-order forecasting operators inadequate and prone to error accumulation. To address this, we propose Fractional Anomalous Dynamics Extrapolation (FADE), a unified framework for training-free diffusion transformers acceleration. FADE models DiT feature evolution via anomalous diffusion and employs a Mittag–Leffler fractional operator to better adapt to high-curvature and non-linear latent trajectories of DiTs. Leveraging the empirical observation that these complex dynamics maintain cross-sample stability, we further construct a Look-Up Table with multiple dimensions through offline profiling. This enables negligible LUT-retrieval overhead, spatiotemporally adaptive parameter retrieval during online inference. Extensive evaluations across diverse architectures and modalities demonstrate that FADE achieves state-of-the-art acceleration, preserving exceptional generation fidelity under aggressive skip intervals with negligible latency overhead. Our project page is available at https://youngkinlon.github.io/fade-project-page/.

1 Introduction

Diffusion Models Ho et al. [2020] have become a dominant paradigm for generative modeling across images Chen et al. [2023], Cai et al. [2025], Labs [2024], videos Kong et al. [2024], Wan et al. [2025], and audio Kong et al. [2021], Liu et al. [2023], owing to their strong high-fidelity synthesis capability. Recent advances in Diffusion Transformers (DiTs) Peebles and Xie [2023] have further improved this capability, but also introduce substantial computational overhead. Their iterative sampling process requires sequential forward passes through large-scale models, resulting in high latency that hinders practical deployment and motivates training-free acceleration without modifying model weights.

*Corresponding author.

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



(a) Anomalous diffusion scaling (b) Spatiotemporal trajectory consistency

Figure 1: Empirical evidence of anomalous and structured dynamics in DiT. (a) The normalized mean squared displacement across layers follows an anomalous diffusion scaling law. (b) Cross-sample trajectory consistency across different layers and denoising steps.

To alleviate this bottleneck, feature caching has emerged as a training-free acceleration strategy. Traditional caching methods Ma et al. [2024b], Selvaraju et al. [2024] reuse features from previous timesteps by exploiting temporal consistency in intermediate representations. However, this cache-and-reuse paradigm degrades generation quality under large skip intervals, as feature similarity diminishes over longer temporal distances Selvaraju et al. [2024], Ma et al. [2024b]. Methods such as TaylorSeer Liu et al. [2025b] and HiCache Feng et al. [2026] formulate feature forecasting based on integer-order forecasting operators. Yet such formulations implicitly assume locally smooth dynamics with a fixed coefficient law, providing only limited approximations to DiT latent trajectories with complex geometry. As a result, they struggle to capture the intrinsic curvature, stage-dependent variations, and abrupt trajectory changes observed across denoising stages.

This limitation reflects a deeper geometric mismatch between existing forecasting schemes and DiT latent dynamics. Prior studies have shown that flat Euclidean assumptions are insufficient to capture the topology of deep generative latent spaces Kalatzis et al. [2020], while recent analyses further suggest the existence of fractal phase-transition structures in diffusion latents Lobashev et al. [2025]. Since anomalous diffusion commonly arises on fractal structures Oliveira et al. [2019], this connection motivates us to investigate whether DiT feature evolution exhibits anomalous diffusion behavior. In anomalous diffusion studies, the power-law scaling exponent of the mean squared displacement (MSD) serves as a standard empirical diagnostic for deviations from Brownian diffusion Metzler and Klafter [2000]. Using this diagnostic, we observe that the normalized MSD of DiT features deviates from Brownian linear scaling, as shown in Fig. 1(a), thereby supporting a fractional perspective for modeling such dynamics. To further characterize the structure of feature evolution, we examine its variation across denoising stages, network layers, and input samples from an anomalous-diffusion perspective. As shown in Fig. 1(b), feature dynamics vary substantially across layers and timesteps, while exhibiting relatively smaller variation across input samples. This observation suggests structured stage- and layer-dependent dynamics with a degree of cross-sample consistency, providing empirical support for calibrating forecasting parameters offline by denoising stage and network location and reusing them across samples.

Building on these observations, our key insight is to treat DiT feature evolution as an anomalous diffusion process over complex latent geometry, while exploiting its empirical cross-sample regularity for offline calibration. This motivates a forecasting framework that combines a fractional dynamical model with offline-calibrated stage-aware adaptation. Accordingly, we propose FADE (Fractional Anomalous Dynamics Extrapolation), a training-free framework that models DiT feature evolution via anomalous diffusion for diffusion transformers acceleration. A Mittag–Leffler fractional operator is introduced to generalize the classical integer-order shift operator, enabling flexible prediction of non-Brownian feature evolution. To operationalize this adaptive formulation, FADE profiles the optimal fractional parameters offline and stores them in a multi-dimensional LUT, which allows stage-aware feature forecasting through $\mathcal{O}(1)$ retrieval during inference. This design improves extrapolation robustness under large skip intervals while preserving generation quality with negligible overhead.

In summary, the principal contributions of our work are as follows:

- **Fractional Anomalous Dynamics Extrapolation (FADE):** We propose a training-free forecasting framework that models DiT feature evolution as an anomalous diffusion process. By generalizing the classical exponential shift operator with a Mittag–Leffler fractional operator, FADE better adapts to the complex, high-curvature dynamics of latent trajectories.

- **Zero-Overhead Adaptive Forecasting:** To translate our fractional framework into practical acceleration, we construct a Look-Up Table indexed by multiple dimensions through offline profiling. This design enables spatiotemporally adaptive parameter retrieval in constant time, ensuring robust feature forecasting without introducing any additional inference latency.

- **Empirical Characterization of DiT Feature Dynamics:** We reveal that DiT feature evolution exhibits anomalous diffusion behavior and structured spatiotemporal heterogeneity across denoising stages and network locations. Meanwhile, these dynamics remain relatively stable across input samples, providing empirical support for offline calibration.

- **State-of-the-Art Acceleration:** Extensive evaluations across diverse architectures and modalities, including DiT-XL/2, FLUX.1, and HunyuanVideo, demonstrate that FADE consistently outperforms existing baselines, maintaining exceptional generation fidelity and negligible latency overhead even under long skip intervals.

2 Related Work

Computation Reduction Strategies. Previous efforts to accelerate diffusion models focus on either reducing the number of sampling steps or compressing the denoising network. Step-reduction methods accelerate the generative process through efficient ODE solvers like DDIM Song et al. [2022] and DPM-Solver Lu et al. [2022], Zhao et al. [2023], Lu et al. [2025], knowledge distillation Salimans and Ho [2022], Luo et al. [2023], or few-step paradigms such as Rectified Flow Liu et al. [2022] and Consistency Models Song et al. [2023]. However, these approaches often degrade generation quality or require expensive retraining Ma et al. [2024a]. Alternatively, network compression techniques such as pruning Fang et al. [2023], Yuan et al. [2024], quantization Li et al. [2023], Shang et al. [2023], Yang et al. [2025], and token merging Zhang et al. [2024b], Bolya and Hoffman [2023] reduce the FLOPs of each denoising step. However, their efficiency gains often come at the cost of reduced model expressivity, which may compromise generation fidelity under aggressive compression. In contrast, feature caching reduces redundant intermediate feature computation to accelerate inference.

Feature Caching. As a prominent training-free alternative, feature caching skips redundant intermediate computations. Existing methods generally fall into two paradigms: cache-then-reuse and cache-then-forecast. Reuse-based approaches reuse cached representations, including direct reuse methods such as DeepCache Ma et al. [2024b], PAB Zhao et al. [2024], and FORA Selvaraju et al. [2024], token-level methods such as ToCa Zou et al. [2025], Tokencache Lou et al. [2024], and DaTo Zhang et al. [2024a], and error-aware or adaptive caching variants such as TeaCache Liu et al. [2025a], AdaCache Kahatapitiya et al. [2025], and ERTACache Peng et al. [2025]. MEDA Zhou et al. [2026] uses multiobjective evolutionary search to optimize feature-reuse schedules, balancing computational cost and generation quality. However, larger skip intervals inevitably enlarge the temporal representation mismatch, degrading generation quality. However, larger skip intervals inevitably enlarge the temporal representation mismatch, degrading generation quality. Forecasting-based approaches, pioneered by TaylorSeer Liu et al. [2025b], actively predict feature evolution. Recent advancements seek to refine this paradigm: HiCache Feng et al. [2026] improves the standard Taylor expansion by introducing scaled Hermite coefficients, while FOCA Zheng et al. [2026] drives the forecasting process by utilizing both the features themselves and their first-order finite-difference derivatives. BRACE Yang et al. [2026] further explores barycentric rational forecasting with adapted Chebyshev weights, directly aggregating cached historical features to accommodate local trajectory irregularities. Despite these algorithmic refinements, these methods treat the non-equilibrium denoising trajectory as a globally uniform dynamical system, limiting their capacity to characterize the anomalous diffusion properties and complex temporal phase transitions of latent features. This ultimately results in trajectory deviation under large skip intervals.

3 Method

3.1 Preliminaries

Diffusion models generate data by iteratively reversing a predefined forward noising process via a neural network $\epsilon_\theta(\mathbf{x}_t, t)$ that estimates the denoising direction at each timestep. Starting from Gaussian noise, the latent variable $\mathbf{x}_t$ is progressively refined over T timesteps toward a clean sample:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right) + \sigma_t \boldsymbol{\epsilon}. \tag{1}$$

Since diffusion sampling requires T sequential model evaluations, inference is costly, especially for large-scale DiTs Peebles and Xie [2023]. To reduce full forward passes at selected timesteps, *Predictive feature caching* reduces this cost by forecasting intermediate features $\mathbf{F}^l$ at layer l. With a recursive discrete difference operator Δ, the m-th order Taylor predictor over distance δ is:

$$\mathbf{F}_{\text{pred}}(\mathbf{x}^l_{t-\delta}) = \mathbf{F}(\mathbf{x}^l_t) + \sum_{i=1}^{m} \frac{\Delta^i \mathbf{F}(\mathbf{x}^l_t)}{i!} (-\delta)^i. \tag{2}$$

Based on operator calculus, the above polynomial extrapolation over a prediction distance δ can be interpreted as applying the exponential shift operator e^z to the current feature $\mathbf{F}_t$, where $z = -\delta \nabla_t$ and $\nabla_t = \partial / \partial t$ denotes the temporal derivative operator:

$$\mathbf{F}_{t-\delta} \approx e^z \mathbf{F}_t. \tag{3}$$

From this perspective, conventional integer-order extrapolation methods typically rely on finite-order local approximations of the shift operator, which may struggle to capture the complex and non-stationary dynamics of DiT's latent trajectories.

3.2 Overall Framework of FADE

Motivated by the anomalous diffusion of DiT latent feature evolution, we propose Fractional Anomalous Dynamics Extrapolation (FADE). As shown in Figure 2, FADE consists of an offline *FADE Profiling Framework* and an online *FADE Inference Architecture*. In the profiling stage, calibration samples are used for *Optimal Order Search* with *Directional Alignment*, where the optimal fractional order $\gamma^*_{t,s}$ is obtained by minimizing the directional loss $\mathcal{L}_{\text{dir}}$ between the predicted feature $\mathbf{F}_{\text{FADE}}$ and the ground-truth feature $\mathbf{F}_{\text{GT}}$. The calibrated orders are then stored in a static look-up table (LUT), where s indexes the network location(e.g., layer, module, stream, etc). In the inference stage, denoising follows a cache interval N and alternates between *Full Compute Steps* and *Prediction Steps*: Full Compute Steps perform forward passes and cache the finite-difference set $\{\Delta^i \mathbf{F}_t\}_{i=1}^m$, while Prediction Steps retrieve $\hat{\gamma}_{t,s}$ from the LUT and apply the *Fractional Operator* $E_\gamma(\xi)$ to generate the predicted feature $\mathbf{F}_{\text{pred}}$. The geometric view in Figure 2 further illustrates that FADE better adapts to the complex latent manifold of DiTs.

3.3 Modeling DiT Feature Evolution via Anomalous Diffusion

Building upon the observed latent dynamics and prior studies on DiT latent geometry, we model DiT feature evolution from an anomalous-diffusion perspective. Anomalous transport on fractal structures provides a geometric motivation for this view Oliveira et al. [2019], while our observed deviations from Brownian MSD scaling provide empirical support. To formulate a forecasting rule for such dynamics, we draw on classical fractional dynamics, where Mittag–Leffler functions commonly arise in propagation and relaxation solutions Metzler and Klafter [2000]. Since the Mittag–Leffler function reduces to the exponential function at unit order, we adopt it as a fractional operator prior to generalize the classical exponential shift operator. This provides a flexible coefficient law for controlling how higher-order local variations contribute to feature prediction:

$$E_\gamma(\xi) = \sum_{k=0}^{\infty} \frac{\xi^k}{\Gamma(\gamma k + 1)}. \tag{4}$$

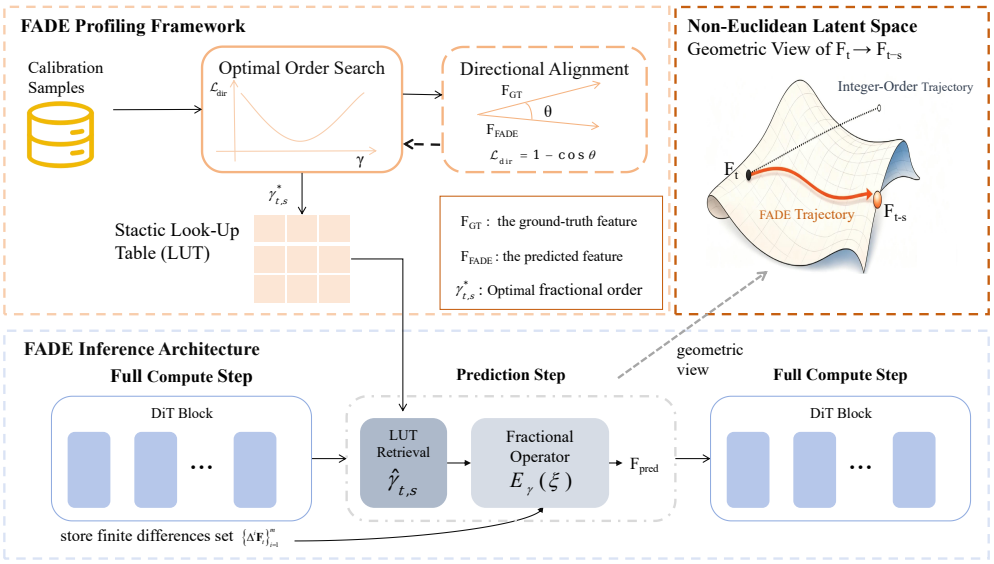


Figure 2: **Overall framework of FADE.** FADE consists of an offline profiling stage and an online inference stage. The offline stage calibrates and stores the optimal fractional orders in a static LUT, while the online stage alternates between full-compute and prediction steps using LUT-retrieved orders for feature forecasting. The upper-right panel illustrates the geometric contrast between the FADE trajectory and the integer-order trajectory in the latent space.

Here, $\Gamma(\cdot)$ denotes the Gamma function used to parameterize the fractional coefficients. Following this fractional coefficient law, we construct a finite-order feature predictor as

$$\mathbf{F}_{\text{pred}}(\mathbf{x}_{t-\delta}^{l}) = \mathbf{F}(\mathbf{x}_{t}^{l}) + \sum_{i=1}^{m} \frac{\Delta^{i}\mathbf{F}(\mathbf{x}_{t}^{l})}{\Gamma(\gamma i + 1)}(-\delta)^{i}. \tag{5}$$

Under this formulation, the fractional order γ controls the coefficient law used for feature extrapolation. Since DiT denoising is highly non-stationary, feature dynamics vary across timesteps and latent states, often shifting between sub- and super-diffusive-like behaviors. This makes a fixed integer-order rule insufficient for modeling local evolution. FADE therefore assigns state-specific fractional orders to adapt the extrapolation law to heterogeneous anomalous dynamics. To obtain these orders without runtime search, we introduce a Stage-Aware Calibration Strategy.

3.4 Stage-Aware Calibration Strategy

As established in our empirical analysis, existing static caching frameworks overlook a key structural property of Diffusion Transformers: feature dynamics vary substantially across latent states and timesteps, yet exhibit a certain degree of cross-sample regularity. Leveraging this cross-sample consistency, we propose a Stage-Aware Calibration Strategy that offline pre-computes the optimal fractional order $\gamma_{t,s}^{*}$ for each state s and timestep t, thereby avoiding online search overhead with a short one-time calibration stage and negligible LUT storage.

Directional Monitoring Objective. Calibration is formulated as a local trajectory-matching problem. We use cosine distance to emphasize directional agreement between predicted and reference features. A purely directional objective is therefore adopted, and $\mathcal{L}_{\text{dir}}$ is defined as the cosine distance between the predicted feature $\mathbf{F}_{\text{FADE}}(\gamma)$ and the ground-truth feature $\mathbf{F}_{\text{GT}}$:

$$\mathcal{L}_{\text{dir}}(\gamma) = 1 - \frac{\mathbf{F}_{\text{FADE}}(\gamma) \cdot \mathbf{F}_{\text{GT}}}{\|\mathbf{F}_{\text{FADE}}(\gamma)\|_2 \|\mathbf{F}_{\text{GT}}\|_2}. \tag{6}$$

Optimal Order Search and Global Refinement. Using a small calibration dataset $\mathcal{D}_{\text{calib}}$, the optimal fractional order is obtained by minimizing the expected directional error over a discrete

search grid $\mathcal{G}$, which spans both sub- and super-diffusion-like regimes:

$$\gamma^*_{t,s} = \arg\min_{\gamma \in \mathcal{G}} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_{\text{calib}}} \left[\mathcal{L}_{\text{dir}} \left(\mathbf{F}_{\text{FADE}}(\gamma); \mathbf{x}, t, s \right) \right]. \tag{7}$$

The offline-calibrated fractional orders $\gamma^*_{t,s}$ may still exhibit a global scale bias in practical inference. To account for this potential bias, we introduce a global refinement factor λ and use $\hat{\gamma}_{t,s} = \lambda \cdot \gamma^*_{t,s}$, providing a lightweight global adjustment to the calibrated orders. The refined orders $\hat{\gamma}_{t,s}$ are then stored in the LUT described in Sec. 3.2 for inference-time retrieval. Detailed calibration settings are provided in Appendix B.

3.5 Error Analysis from a Fractional Perspective

We provide a local and conditional error analysis under a fractional model of local latent evolution. Assume that the local ground-truth evolution admits the following representation:

$$\mathbf{F}^{\text{GT}}_{t-\delta,s} = \mathbf{F}_{t,s} + \sum_{i=1}^{m} \frac{\Delta^i \mathbf{F}_{t,s}}{\Gamma(\gamma^\dagger_{t,s} i + 1)} (-\delta_{\mathcal{M},t,s})^i + \mathbf{R}^{(t,s)}_{m+1}, \qquad \|\mathbf{R}^{(t,s)}_{m+1}\| \le C_{t,s}\, \delta^{m+1}_{\mathcal{M},t,s}, \tag{8}$$

where $\delta_{\mathcal{M},t,s}$ denotes the intrinsic propagation scale along the local feature trajectory, which reflects the underlying distance traveled by features on the latent manifold, while δ denotes the prediction distance used in practice. This distinction captures a potential propagation-scale mismatch. The profiled order $\gamma^*_{t,s}$ is used as an estimate of the intrinsic order $\gamma^\dagger_{t,s}$, and FADE applies the globally refined order $\hat{\gamma}_{t,s} = \lambda \gamma^*_{t,s}$ at inference time. Substituting Eq. (5) into Eq. (8), we obtain

$$\|\mathbf{F}^{\text{GT}}_{t-\delta,s} - \widehat{\mathbf{F}}^{\text{FADE}}_{t-\delta,s}\| \le C_{t,s}\, \delta^{m+1}_{\mathcal{M},t,s} + \sum_{i=1}^{m} \left\|\Delta^i \mathbf{F}_{t,s}\right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^\dagger_{t,s} i + 1)} - \frac{(-\delta)^i}{\Gamma(\lambda \gamma^*_{t,s} i + 1)} \right|. \tag{9}$$

The second term combines the intrinsic–extrinsic scale mismatch and the coefficient mismatch. For Taylor extrapolation in Eq. (2), the corresponding mismatch is

$$\sum_{i=1}^{m} \left\|\Delta^i \mathbf{F}_{t,s}\right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^\dagger_{t,s} i + 1)} - \frac{(-\delta)^i}{i!} \right|. \tag{10}$$

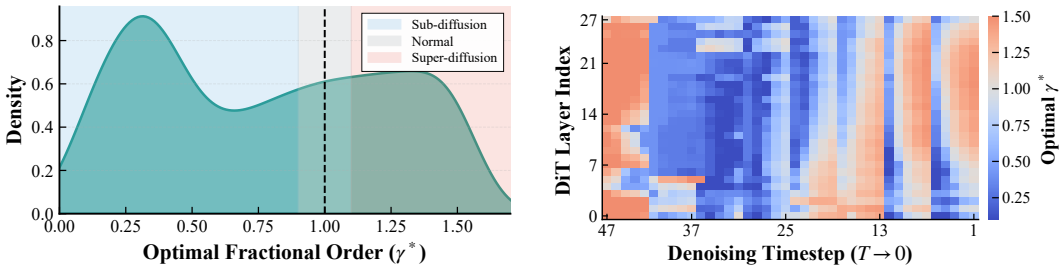
Since Taylor extrapolation corresponds to the $\gamma = 1$ case of the fractional coefficient family, it can be viewed as a special case of FADE. In contrast, FADE calibrates a stage-adaptive order $\gamma^*_{t,s}$ to accommodate the heterogeneous dynamics across DiT timesteps and network states, and applies a refined order $\hat{\gamma}_{t,s} = \lambda \gamma^*_{t,s}$ at inference time. The global factor λ can be viewed as a lightweight correction that helps accommodate residual propagation-scale mismatch. Together, these designs provide a more flexible local approximation of the intrinsic dynamics, consistent with the improved robustness observed under large prediction distances.

4 Experiments

4.1 Experimental Setup

FADE is evaluated across three representative generative tasks: (i) class-conditional image generation on ImageNet-1K Deng et al. [2009] using DiT-XL/2 Peebles and Xie [2023]; (ii) text-to-image generation on DrawBench Saharia et al. [2022] using the rectified flow-based FLUX.1-dev Labs et al. [2025], Labs [2024], Liu et al. [2022]; and (iii) text-to-video generation using HunyuanVideo Kong et al. [2024] on the HunyuanLarge architecture. For DiT-XL/2, we generate images at 256×256 resolution using the DDIM sampler with 50 denoising steps. For FLUX.1-dev, we use the Flow Match Euler scheduler with 50 sampling steps at 1024×1024 resolution. HunyuanVideo follows its official inference pipeline with 50 flow-matching sampling steps and default video generation settings.

We compare FADE with representative training-free caching methods, including FORA Selvaraju et al. [2024], TaylorSeer Liu et al. [2025b], and HiCache Feng et al. [2026]. Within each comparison, caching methods use the same pretrained checkpoint, sampler, number of sampling steps, resolution,



(a) Global distribution of γ^*.

(b) Spatiotemporal heatmap of γ^*.

Figure 3: **Empirical validation of fractional dynamics on DiT-XL/2.** The optimal fractional orders deviate markedly from the static integer-order assumption and exhibit a pronounced non-stationary temporal pattern across denoising timesteps.

guidance settings, and evaluation inputs. Qualitative comparisons additionally use matched random seeds and acceleration schedules to facilitate direct visual assessment.

Quantitative performance is assessed using FID-50K Heusel et al. [2018], sFID, and Inception Score for DiT-XL/2. For FLUX.1-dev, we report ImageReward Xu et al. [2023], CLIP Score Hessel et al. [2022], and CycleReward Bahng et al. [2025], together with PSNR, SSIM Horé and Ziou [2010], and LPIPS Zhang et al. [2018] to assess consistency with the full-compute baseline. HunyuanVideo is evaluated using the 16-dimensional VBench suite Huang et al. [2024]. Computational efficiency is assessed using FLOPs and measured generation latency. FLOPs reduction and wall-clock speedup are computed separately as the ratios of baseline FLOPs and latency to those of the accelerated method, respectively. Offline calibration cost is reported separately from inference latency.

4.2 Empirical Validation of Fractional Dynamics

To validate the fractional prior underlying FADE, we profile the optimal fractional orders γ^* across representative architectures and modalities. If DiT feature evolution were well described by conventional integer-order dynamics, the optimal states would concentrate around $\gamma = 1$, which is not observed in practice. On DiT-XL/2, Fig. 3 shows that the global distribution of γ^* is clearly shifted away from $\gamma = 1$, indicating that standard integer-order extrapolation captures only a limited portion of the effective generative states. The spatiotemporal heatmap further reveals a U-shaped temporal pattern: the middle denoising phase is more sub-diffusion-like, whereas the early and late phases are more super-diffusion-like in a phenomenological sense.

This heterogeneity also extends across architectures. FLUX.1 shows an inverted temporal trend and clear DoubleStream–SingleStream decoupling, while HunyuanVideo exhibits a broader super-diffusion-like tendency. Together, these results suggest that optimal feature dynamics are non-integer, architecture-dependent, and stage-dependent, motivating the stage-aware calibration design of FADE. More details are provided in Appendix D. Meanwhile, the cross-sample variance of γ^* is consistently small on DiT-XL/2 and remains within an acceptable range on FLUX.1 and HunyuanVideo.

4.3 Class-Conditional Image Generation

As demonstrated in Table 1, FADE consistently dominates both generation quality and efficiency metrics across all acceleration ratios on DiT-XL/2. When operating at a caching interval of $N = 4$, FADE achieves an outstanding FID of 2.41 and an sFID of 4.79. This not only crushes the 15-step DDIM (FID 5.09) with fewer FLOPs, but also outperforms TaylorSeer's Liu et al. [2025b] FID of 2.59 and HiCache's Feng et al. [2026] 2.51 under strictly identical computational costs. This superiority becomes even more pronounced under aggressive acceleration regimes. When scaling to $N = 6$, FADE maintains a highly competitive FID of 2.87, completely mitigating the severe structural collapse observed in FORA Selvaraju et al. [2024] and standard Taylor caching Liu et al. [2025b], which suffer from severely degraded FIDs of 9.22 and 3.17 respectively.

Table 1: Quantitative comparison of class-conditional image generation with DiT-XL/2

Method	Lat. (s)↓	FLOPs (T)↓	Spd.↑	FID↓	sFID↓	IS↑
DDIM-50 steps Song et al. [2022]	1.18	23.74	1.00×	2.29	4.29	237.31
DDIM-25 steps Song et al. [2022]	0.62	11.87	2.00×	2.91	4.57	232.22
DDIM-15 steps Song et al. [2022]	0.42	7.12	3.33×	5.09	6.11	205.54
FORA(N=4) Selvaraju et al. [2024]	0.45	6.66	3.56×	4.73	8.45	214.20
Taylor (N=4, m=4) Liu et al. [2025b]	0.61	6.66	3.56×	2.59	5.18	232.90
HiCache (N=4, m=4) Feng et al. [2026]	0.57	6.66	3.56×	2.51	5.20	232.72
FADE(N=4, m=4)	0.55	**6.66**	**3.56×**	**2.41**	**4.79**	**236.76**
DDIM-12 steps Song et al. [2022]	0.33	5.70	4.17×	7.94	8.07	183.88
FORA(N=5) Selvaraju et al. [2024]	0.37	5.24	4.53×	5.81	9.61	199.96
Taylor (N=5, m=4) Liu et al. [2025b]	0.55	5.24	4.53×	2.73	5.34	228.75
HiCache (N=5, m=4) Feng et al. [2026]	0.50	5.24	4.53×	2.67	5.48	229.44
FADE(N=5, m=4)	0.49	**5.24**	**4.53×**	**2.55**	**5.15**	**230.89**
DDIM-10 steps Song et al. [2022]	0.30	4.75	5.00×	12.17	11.24	159.20
FORA(N=6) Selvaraju et al. [2024]	0.34	4.76	4.98×	9.22	14.86	169.02
Taylor (N=6, m=4) Liu et al. [2025b]	0.50	4.76	4.98×	3.17	6.36	222.00
HiCache (N=6, m=4) Feng et al. [2026]	0.46	4.76	4.98×	3.09	6.30	220.55
FADE(N=6, m=4)	0.45	**4.76**	**4.98×**	**2.87**	**5.73**	**225.78**

Table 2: Quantitative comparison of text-to-image generation on FLUX

Method	Latency(s)↓	FLOPs(T)↓	Speed↑	Img. Reward↑	CLIP Score↑	Cycle Reward↑	PSNR↑	SSIM↑	LPIPS↓
FLUX.1 [dev] - 50 steps†	17.18	3719.50	1.00×	0.9898	27.4611	0.8800	∞	1.0000	0.0000
FLUX.1 [dev] - 25 steps	8.80	1859.75	2.00×	0.9375	27.3388	0.8466	19.2599	0.7749	0.3583
FLUX.1 [dev] - 20 steps	7.12	1487.80	2.62×	0.9387	27.2505	0.8466	18.1371	0.7484	0.4006
FORA ($\mathcal{N}$=5) Selvaraju et al. [2024]	5.01	893.54	4.16×	0.8253	27.2377	0.8353	15.9899	0.6677	0.5206
TaylorSeer ($\mathcal{N}$=5, m=1) Liu et al. [2025b]	5.14	893.54	4.16×	0.9919	27.5143	0.8540	18.4497	0.7424	0.4146
HiCache ($\mathcal{N}$=5, m=1) Feng et al. [2026]	5.12	893.54	4.16×	0.9350	27.4034	0.8346	19.2595	0.7503	0.3911
FADE‡ ($\mathcal{N}$=5, m=1)	5.04	893.54	4.16×	**1.0071**	**27.5724**	**0.8600**	**19.6683**	**0.7593**	**0.3815**
FORA ($\mathcal{N}$=6) Selvaraju et al. [2024]	4.40	744.81	4.99×	0.7836	27.1541	0.8087	15.5290	0.6663	0.5253
TaylorSeer ($\mathcal{N}$=6, m=1) Liu et al. [2025b]	4.57	744.81	4.99×	0.9850	27.5199	0.8480	17.5396	0.7078	0.4657
HiCache ($\mathcal{N}$=6, m=1) Feng et al. [2026]	4.55	744.81	4.99×	0.9427	27.6898	0.8539	18.5159	0.7155	0.4503
HiCache ($\mathcal{N}$=6, m=2) Feng et al. [2026]	5.03	744.81	4.99×	0.9892	27.6586	0.8428	18.5044	0.7268	0.4316
FADE‡ ($\mathcal{N}$=6, m=1)	4.43	744.81	4.99×	**1.0019**	**27.7060**	**0.8706**	**18.7114**	**0.7320**	**0.4207**
FORA ($\mathcal{N}$=7) Selvaraju et al. [2024]	4.10	670.44	5.55×	0.7289	27.0135	0.7962	15.5290	0.6548	0.5437
TaylorSeer ($\mathcal{N}$=7, m=1) Liu et al. [2025b]	4.31	670.44	5.55×	0.9420	27.4039	0.7970	16.8563	0.6844	0.5062
HiCache ($\mathcal{N}$=7, m=1) Feng et al. [2026]	4.20	670.44	5.55×	0.9191	27.6949	0.8494	**18.2778**	0.7032	0.4708
HiCache ($\mathcal{N}$=7, m=2) Feng et al. [2026]	4.72	670.44	5.55×	0.9881	27.5651	0.8384	17.9911	0.7098	0.4592
FADE‡ ($\mathcal{N}$=7, m=1)	4.12	670.44	5.55×	0.9861	**27.8089**	**0.8621**	18.1925	**0.7137**	**0.4491**
FADE‡ ($\mathcal{N}$=7, m=2)	4.25	670.44	5.55×	**0.9914**	27.7372	0.8592	17.7636	0.7106	0.4561

4.4 Text-to-Image Generation

Quantitative Analysis Table 2 reports the quantitative results on FLUX. FADE achieves a strong quality–efficiency trade-off across acceleration settings, consistently outperforming existing baselines on semantic alignment and structural fidelity at standard configurations ($\mathcal{N} \in \{5, 6\}$). Under a more aggressive skip interval, FADE still maintains competitive generation quality with negligible latency overhead, demonstrating the robustness of fractional forecasting.

Qualitative comparison. As illustrated in Figure 4, FADE maintains close visual consistency with the original FLUX.1 baseline in text rendering, texture, and scene composition. It preserves the "Text to Image" storefront and the "Diffusion" sign, whereas FORA Selvaraju et al. [2024] alters the sign geometry and produces unclear lettering. For the brick cube, FADE retains the reddish material and mortar details, while TaylorSeer Liu et al. [2025b] produces a pale stone-like appearance, FORA generates smoother surfaces, and HiCache Feng et al. [2026] changes the brick arrangement. FADE also preserves the cat press-conference composition and animal yearbook scene, while TaylorSeer introduces sunglasses and more human-like bodies in the latter. In the magnifying-glass example, FADE retains the train imagery within the lens, whereas FORA replaces it with text-like patterns.

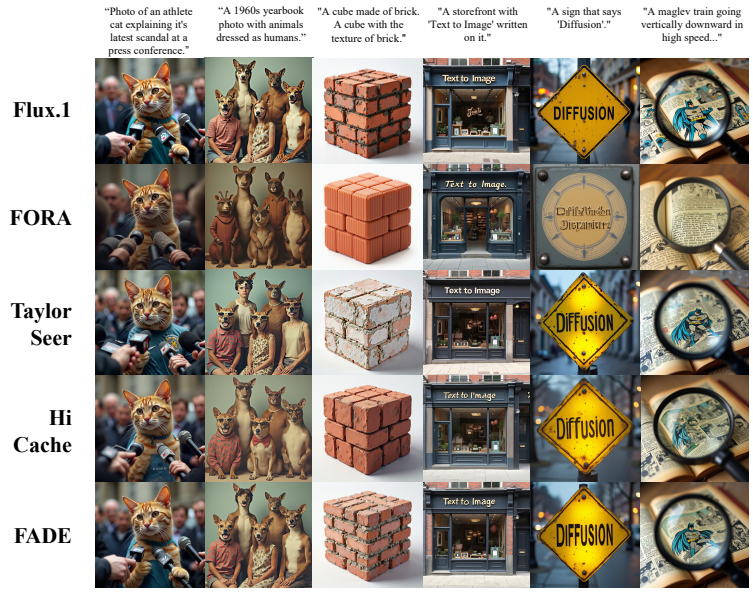


Figure 4: Visualization examples for different acceleration methods on Flux.

Table 3: Quantitative comparison of text-to-video generation on HunyuanVideo.

Method	**Lat. (s)↓**	**FLOPs (T)↓**	**Spd.↑**	**VBench (%)↑**
Original (50 steps)	190.2	29773	1.00×	80.68
DDIM (22% steps) Song et al. [2022]	42.7	6550	4.55×	78.73
FORA (N=6) Selvaraju et al. [2024]	37.1	5359	5.56×	78.86
TaylorSeer (N=6, m=1) Liu et al. [2025b]	37.8	5359	5.56×	79.87
HiCache (N=6, m=1) Feng et al. [2026]	37.8	5359	5.56×	79.93
FADE (N=6, m=1)	37.9	5359	5.56×	**80.12**

4.5 Text-to-Video Generation

Quantitative Analysis. Table 3 reports the quantitative results on HunyuanVideo. Under the same FLOPs budget and acceleration ratio, FADE achieves competitive latency and the best VBench score, indicating that fractional forecasting remains effective for text-to-video generation.

Qualitative Comparison. As shown in Fig. 5, FADE preserves clearer text and structure in the static case, correctly rendering the word "STOP", while better maintaining motion semantics and scene context in dynamic prompts. In particular, FADE better captures the intended temporal behavior and preserves prompt-specific motion cues, rather than merely maintaining frame-level appearance. In contrast, the baselines either miss the intended intense shaking effect in the astronaut case or introduce incorrect motion directions and off-road backgrounds in the motorcycle case.

4.6 Ablation Study

On DiT-XL/2, we compare several strategies for determining the fractional order, including fixed global tuning, polynomial fitting over denoising progress, and the proposed stage-aware LUT retrieval. We also compare MSE-based and directional calibration objectives. As shown in Table 4, the full FADE design achieves the best performance, suggesting that the optimal fractional dynamics cannot be well captured by either a static setting or a simple smooth fit, and that directional calibration provides a more effective signal for stable offline calibration.

On FLUX.1, we study two practical aspects of offline calibration: the calibration set size and the global refinement factor λ. Table 5 shows that FADE remains effective with a small calibration set,



Figure 5: **Qualitative comparison on text-to-video generation.**

Table 4: Ablation on DiT-XL/2.

Method	Speed↑	FID↓	sFID↓
DDIM-50	1.00×	2.29	4.29
DDIM-10	5.00×	12.17	11.24
HiCache	4.98×	3.09	6.30
FADE ($\gamma = 1$)	4.98×	3.17	6.36
FADE (fixed $\gamma = 0.85$)	4.98×	3.06	6.05
FADE (poly-fit $\gamma(t)$)	4.98×	3.00	6.19
FADE (LUT + MSE)	4.98×	2.97	5.91
FADE (LUT + Dir.)	**4.98×**	**2.87**	**5.73**

Table 5: Ablation on FLUX.1.

Method	Speed↑	IR↑	CLIP↑
FADE ($\lvert\mathcal{D}_{\text{calib}}\rvert = 1$)	4.99×	0.9963	27.6781
FADE ($\lvert\mathcal{D}_{\text{calib}}\rvert = 5$)	4.99×	1.0015	27.7454
FADE ($\lvert\mathcal{D}_{\text{calib}}\rvert = 10$)	4.99×	**1.0019**	27.7060
FADE ($\lvert\mathcal{D}_{\text{calib}}\rvert = 20$)	4.99×	1.0008	27.7256
FADE ($\lambda = 1.0$)	4.16×	0.9974	27.4618
FADE ($\lambda = 1.2$)	4.16×	0.9994	**27.6198**
FADE ($\lambda = 1.3$)	**4.16×**	**1.0071**	27.5724
FADE ($\lambda = 1.4$)	4.16×	0.9858	27.5050

saturating around $|\mathcal{D}_{\text{calib}}| = 10$. A moderate refinement factor gives the best overall performance, with $\lambda = 1.3$ achieving the best ImageReward and $\lambda = 1.2$ the best CLIPScore. This indicates that λ is a useful lightweight correction, though its optimal value depends on the prediction interval and sampling setting. Additional ablations are provided in Appendix E.

We further report the practical cost of offline calibration and LUT retrieval. Using 10 calibration samples, end-to-end offline calibration takes approximately 50 s, 550 s, and 2,000 s for DiT-XL/2, FLUX.1, and HunyuanVideo, respectively, on their respective evaluation hardware. The LUT has a small storage footprint, occupying only 44.8 kB for DiT-XL/2. The accumulated LUT-retrieval time per generation is 0.14 ms, 0.57 ms, and 0.61 ms, respectively.

5 Conclusion and Future Work

This work revisits training-free DiT acceleration from the perspective of fractional dynamics. Motivated by the limitations of cache-based methods under complex stage-dependent latent dynamics, we view DiT feature evolution through the lens of anomalous diffusion and formulate FADE using a Mittag–Leffler-based fractional operator. With an offline-profiled LUT, FADE enables adaptive forecasting at inference time with $\mathcal{O}(1)$ retrieval and no extra runtime overhead. Experiments across diverse architectures and modalities show that FADE achieves a stronger quality–efficiency trade-off than existing baselines while remaining robust under large prediction distances.

FADE relies on manually designed calibration, and its Mittag–Leffler operator is chosen primarily for simplicity. Future work will explore alternative fractional operators and order parameterizations to improve forecasting accuracy, stability, and the quality–efficiency trade-off across denoising stages, network locations, and skip intervals.

Acknowledgments and Disclosure of Funding

This work was supported by National Natural Science Foundation of China under Grant No.62376172.

References

Hyojin Bahng, Caroline Chan, Fredo Durand, and Phillip Isola. Cycle consistency as reward: Learning image-text alignment without human preferences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22934–22946, October 2025.

Daniel Bolya and Judy Hoffman. Token merging for fast stable diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4599–4603, 2023.

Qi Cai, Jingwen Chen, Yang Chen, Yehao Li, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Yiheng Zhang, Fengbin Gao, Peihan Xu, Yimeng Wang, Kai Yu, Wenxuan Chen, Ziwei Feng, Zijian Gong, Jianzhuang Pan, Yi Peng, Rui Tian, Siyu Wang, Bo Zhao, Ting Yao, and Tao Mei. Hidream-i1: A high-efficient image generative foundation model with sparse diffusion transformer, 2025. URL https://arxiv.org/abs/2505.22705.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. URL https://arxiv.org/abs/2310.00426.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. Structural pruning for diffusion models, 2023. URL https://arxiv.org/abs/2305.10924.

Liang Feng, Shikang Zheng, Jiacheng Liu, Yuqi Lin, Qinming Zhou, Peiliang Cai, Xinyu Wang, Junjie Chen, Chang Zou, Yue Ma, and Linfeng Zhang. Hicache: A plug-in scaled-hermite upgrade for taylor-style cache-then-forecast diffusion acceleration. In *International Conference on Learning Representations (ICLR)*, 2026.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022. URL https://arxiv.org/abs/2104.08718.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL https://arxiv.org/abs/1706.08500.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Alain Horé and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th International Conference on Pattern Recognition*, pages 2366–2369, 2010. doi: 10.1109/ICPR.2010.579.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

Kumara Kahatapitiya, Haozhe Liu, Sen He, Ding Liu, Menglin Jia, Chenyang Zhang, Michael S Ryoo, and Tian Xie. Adaptive caching for faster video generation with diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15240–15252, 2025.

Dimitris Kalatzis, David Eklund, Georgios Arvanitidis, and Søren Hauberg. Variational autoencoders with riemannian brownian motion priors. *arXiv preprint arXiv:2002.05227*, 2020.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis, 2021. URL https://arxiv.org/abs/2009.09761.

Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17535–17545, 2023.

Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7353–7363, 2025a.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

Jiacheng Liu, Chang Zou, Yuanhuiyi Lyu, Junjie Chen, and Linfeng Zhang. From reusing to forecasting: Accelerating diffusion models with taylorseers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15853–15863, 2025b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL https://arxiv.org/abs/2209.03003.

Alexander Lobashev, Dmitry Guskov, Maria Larchenko, and Mikhail Tamm. Hessian geometry of latent space in generative models. *arXiv preprint arXiv:2506.10632*, 2025.

Jinming Lou, Wenyang Luo, Yufan Liu, Bing Li, Xinmiao Ding, Weiming Hu, Yuming Li, and Chenguang Ma. Token caching for diffusion transformer acceleration. *arXiv preprint arXiv:2409.18523*, 2024.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in neural information processing systems*, 35:5775–5787, 2022.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *Machine Intelligence Research*, 22(4):730–751, June 2025. ISSN 2731-5398. doi: 10.1007/s11633-025-1562-4. URL http://dx.doi.org/10.1007/s11633-025-1562-4.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2023. URL https://arxiv.org/abs/2310.04378.

Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. Sit: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision*, pages 23–40. Springer, 2024a.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15762–15772, 2024b.

Ralf Metzler and Joseph Klafter. The random walk's guide to anomalous diffusion: a fractional dynamics approach. *Physics Reports*, 339(1):1–77, 2000. ISSN 0370-1573. doi: https://doi.org/10.1016/S0370-1573(00)00070-3. URL https://www.sciencedirect.com/science/article/pii/S0370157300000703.

Fernando A Oliveira, Rogelma MS Ferreira, Luciano C Lapas, and Mendeli H Vainstein. Anomalous diffusion: A basic mechanism for the evolution of inhomogeneous systems. *Frontiers in Physics*, 7:18, 2019.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

Xurui Peng, Chenqian Yan, Hong Liu, Rui Ma, Fangmin Chen, Xing Wang, Zhihua Wu, Songwei Liu, and Mingbao Lin. Ertacache: Error rectification and timesteps adjustment for efficient diffusion. *arXiv preprint arXiv:2508.21091*, 2025.

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, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. URL https://arxiv.org/abs/2205.11487.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models, 2022. URL https://arxiv.org/abs/2202.00512.

Pratheba Selvaraju, Tianyu Ding, Tianyi Chen, Ilya Zharkov, and Luming Liang. Fora: Fast-forward caching in diffusion transformer acceleration, 2024. URL https://arxiv.org/abs/2407.01425.

Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *CVPR*, 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models, 2023. URL https://arxiv.org/abs/2303.01469.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: learning and evaluating human preferences for text-to-image generation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 15903–15935, 2023.

Jinlong Yang, Jinke Wu, Lizilin, and Yao Zhou. Brace: Taming sharp irregularities via barycentric rational forecasting for fast diffusion transformers inference, 2026. URL https://arxiv.org/abs/2608.07572.

Lianwei Yang, Haokun Lin, Tianchen Zhao, Yichen Wu, Hongyu Zhu, Ruiqi Xie, Zhenan Sun, Yu Wang, and Qingyi Gu. Lrq-dit: Log-rotation post-training quantization of diffusion transformers for image and video generation. *arXiv preprint arXiv:2508.03485*, 2025.

Zhihang Yuan, Hanling Zhang, Lu Pu, Xuefei Ning, Linfeng Zhang, Tianchen Zhao, Shengen Yan, Guohao Dai, and Yu Wang. DiTFastattn: Attention compression for diffusion transformer models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=51HQpkQy3t.

Evelyn Zhang, Bang Xiao, Jiayi Tang, Qianli Ma, Chang Zou, Xuefei Ning, Xuming Hu, and Linfeng Zhang. Token pruning for caching better: 9 times acceleration on stable diffusion for free. *arXiv preprint arXiv:2501.00375*, 2024a.

Evelyn Zhang, Bang Xiao, Jiayi Tang, Qianli Ma, Chang Zou, Xuefei Ning, Xuming Hu, and Linfeng Zhang. Token pruning for caching better: 9 times acceleration on stable diffusion for free, 2024b. URL https://arxiv.org/abs/2501.00375.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. *arXiv preprint arXiv:2408.12588*, 2024.

Shikang Zheng, Liang Feng, Xinyu Wang, Qinming Zhou, Peiliang Cai, Chang Zou, Jiacheng Liu, Yuqi Lin, Junjie Chen, Yue Ma, and Linfeng Zhang. Forecast then calibrate: Feature caching as ODE for efficient diffusion transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. URL https://arxiv.org/abs/2508.16211.

Yao Zhou, Zhang Yi, and Gary G. Yen. Multiobjective evolutionary diffusion transformer acceleration. *IEEE Transactions on Cybernetics*, 56(11):6085–6096, 2026. doi: 10.1109/TCYB.2026.3731615.

Chang Zou, Xuyang Liu, Ting Liu, Siteng Huang, and Linfeng Zhang. Accelerating diffusion transformers with token-wise feature caching, 2025. URL https://arxiv.org/abs/2410.05317.

Technical appendices

A Detailed Derivation of the FADE Predictive Formulation

In this section, we provide the rigorous step-by-step mathematical derivation detailing the transition from the classical integer-order extrapolation to our generalized Fractional Anomalous Dynamics Extrapolation (FADE) formulation.

A.1 From Polynomial Extrapolation to the Exponential Shift Operator

Based on operator calculus, the polynomial extrapolation in Eq. (2) can be interpreted as applying an exponential shift operator to the current feature representation. Let $\mathbf{F}_t = \mathbf{F}(\mathbf{x}_t^l)$ denote the feature at timestep t, and let δ denote the prediction distance from t to $t-\delta$. We define the temporal derivative operator as

$$\nabla_t = \frac{\partial}{\partial t}. \tag{11}$$

For a sufficiently smooth continuous trajectory, the Taylor expansion of $\mathbf{F}_{t-\delta}$ around t can be written as

$$\mathbf{F}_{t-\delta} = \mathbf{F}_t + \sum_{i=1}^{\infty} \frac{\nabla_t^i \mathbf{F}_t}{i!}(-\delta)^i. \tag{12}$$

Equivalently, by collecting the derivative operator terms, we obtain

$$\mathbf{F}_{t-\delta} = \left(\sum_{i=0}^{\infty} \frac{(-\delta \nabla_t)^i}{i!} \right) \mathbf{F}_t. \tag{13}$$

Since the infinite series corresponds to the Maclaurin expansion of the exponential function, this transformation can be compactly expressed as

$$\mathbf{F}_{t-\delta} = e^{-\delta\nabla_t}\mathbf{F}_t. \tag{14}$$

By defining

$$z = -\delta\nabla_t, \tag{15}$$

the above relation becomes

$$\mathbf{F}_{t-\delta} \approx e^{z}\mathbf{F}_t, \tag{16}$$

where the approximation emphasizes the practical setting in which the continuous operator is implemented through finite-order truncation and discrete feature differences.

In practical predictive caching, the infinite-dimensional operator is truncated to the m-th order:

$$e^{-\delta\nabla_t}\mathbf{F}_t \approx \left(\sum_{i=0}^{m}\frac{(-\delta\nabla_t)^i}{i!}\right)\mathbf{F}_t = \mathbf{F}_t + \sum_{i=1}^{m}\frac{\nabla_t^i\mathbf{F}_t}{i!}(-\delta)^i. \tag{17}$$

By approximating the continuous derivatives with discrete feature differences, i.e.,

$$\Delta^i\mathbf{F}(\mathbf{x}_t^l) \approx \nabla_t^i\mathbf{F}_t, \tag{18}$$

we recover the polynomial forecasting form:

$$\mathbf{F}_{\text{pred}}(\mathbf{x}_{t-\delta}^l) = \mathbf{F}(\mathbf{x}_t^l) + \sum_{i=1}^{m}\frac{\Delta^i\mathbf{F}(\mathbf{x}_t^l)}{i!}(-\delta)^i. \tag{19}$$

Therefore, the commonly used m-th order polynomial extrapolation can be viewed as a truncated and discretized approximation of the exponential shift operator $e^{-\delta\nabla_t}$.

A.2 Detailed Derivation of the FADE Predictive Formulation

In this section, we provide a step-by-step derivation of the FADE predictive formulation, showing how the classical integer-order extrapolation can be generalized through a fractional Mittag–Leffler operator.

Operator Equivalence. We first revisit the continuous integer-order baseline. Let $\mathbf{F}_t = \mathbf{F}(\mathbf{x}_t^l)$ denote the latent feature at timestep t, and let δ denote the prediction distance from t to $t-\delta$. For a sufficiently smooth feature trajectory, the Taylor expansion of $\mathbf{F}_{t-\delta}$ around t is given by

$$\mathbf{F}_{t-\delta} = \sum_{i=0}^{\infty}\frac{(-\delta)^i}{i!}\frac{\partial^i\mathbf{F}_t}{\partial t^i}. \tag{20}$$

By defining the temporal derivative operator as

$$\nabla_t = \frac{\partial}{\partial t}, \tag{21}$$

the above expansion can be written in operator form as

$$\mathbf{F}_{t-\delta} = \left(\sum_{i=0}^{\infty}\frac{(-\delta\nabla_t)^i}{i!}\right)\mathbf{F}_t = e^{-\delta\nabla_t}\mathbf{F}_t. \tag{22}$$

Equivalently, by defining

$$z = -\delta\nabla_t, \tag{23}$$

the integer-order extrapolation can be interpreted as applying the exponential shift operator:

$$\mathbf{F}_{t-\delta} \approx e^{z}\mathbf{F}_t, \tag{24}$$

where the approximation emphasizes the practical setting in which the continuous operator is implemented through finite-order truncation and discrete feature differences.

Continuous Fractional Generalization. To model the anomalous and stage-dependent evolution of DiT features, FADE replaces the classical exponential operator with a Mittag–Leffler (ML) operator. The ML function is defined as

$$E_\gamma(z) = \sum_{i=0}^{\infty} \frac{z^i}{\Gamma(\gamma i + 1)}, \tag{25}$$

where γ controls the fractional propagation order. Substituting the exponential shift variable $z = -\delta\nabla_t$ into the ML operator gives

$$E_\gamma(-\delta\nabla_t)\mathbf{F}_t = \sum_{i=0}^{\infty} \frac{(-\delta\nabla_t)^i}{\Gamma(\gamma i + 1)}\mathbf{F}_t. \tag{26}$$

By isolating the zero-order term and using $\Gamma(1) = 1$, we obtain

$$E_\gamma(-\delta\nabla_t)\mathbf{F}_t = \mathbf{F}_t + \sum_{i=1}^{\infty} \frac{(-\delta)^i}{\Gamma(\gamma i + 1)}\nabla_t^i\mathbf{F}_t. \tag{27}$$

Truncation and Numerical Discretization. For tractable inference, the infinite fractional series is truncated to a finite order m. Meanwhile, since diffusion sampling proceeds over discrete timesteps, the continuous derivative $\nabla_t^i\mathbf{F}_t$ is approximated by the i-th order discrete feature difference, denoted by $\Delta^i\mathbf{F}_t$. The backward difference operator is recursively defined as

$$\Delta^1\mathbf{F}_t = \mathbf{F}_t - \mathbf{F}_{t-1}, \tag{28}$$

$$\Delta^i\mathbf{F}_t = \Delta^{i-1}\mathbf{F}_t - \Delta^{i-1}\mathbf{F}_{t-1}, \quad i \geq 2. \tag{29}$$

Substituting the continuous derivatives with discrete feature differences yields the FADE predictive formulation:

$$\mathbf{F}_{\text{pred}}(\mathbf{x}_{t-\delta}^l) = \mathbf{F}(\mathbf{x}_t^l) + \sum_{i=1}^{m} \frac{\Delta^i\mathbf{F}(\mathbf{x}_t^l)}{\Gamma(\gamma i + 1)}(-\delta)^i. \tag{30}$$

Here, δ denotes the effective prediction distance, instantiated in practice as the skip interval between the latest computed step and the target prediction step.

Degeneration to the Integer-Order Baseline. The classical integer-order extrapolation is recovered as a special case when $\gamma = 1$. Using the Gamma function identity

$$\Gamma(i + 1) = i!, \tag{31}$$

Eq. (30) becomes

$$\mathbf{F}_{\text{pred}}(\mathbf{x}_{t-\delta}^l)\big|_{\gamma=1} = \mathbf{F}(\mathbf{x}_t^l) + \sum_{i=1}^{m} \frac{\Delta^i\mathbf{F}(\mathbf{x}_t^l)}{i!}(-\delta)^i. \tag{32}$$

Therefore, the conventional integer-order polynomial forecasting rule can be viewed as the $\gamma = 1$ instance of the more general FADE formulation, while $\gamma \neq 1$ provides a flexible coefficient law for modeling anomalous, stage-dependent feature dynamics.

B Implementation Details of Stage-Aware Calibration

B.1 Model-Specific Calibration and Extreme Data Efficiency

A significant advantage of FADE is its extreme data efficiency during the offline calibration phase. We construct the calibration dataset $\mathcal{D}_{\text{calib}}$ using only **10 randomly sampled conditions** paired with standard Gaussian noise initializations. To ensure the calibration perfectly aligns with the native inference pipelines of different generation paradigms, the conditioning inputs are specifically configured as follows:

- **Class-Conditional Models (e.g., DiT-XL/2):** $\mathcal{D}_{\text{calib}}$ is constructed by randomly sampling 10 class labels from the ImageNet training label space, coupled with random initial noise seeds. The calibration seeds are kept disjoint from those used for final evaluation.

- **Text-to-Image Models (e.g., FLUX.1):** The calibration conditions are formed by randomly selecting 10 diverse text prompts from the MS-COCO training split, paired with random initial seeds. These calibration prompts and seeds are not reused in the final DrawBench evaluation.
- **Text-to-Video Models (e.g., HunyuanVideo):** For video generation, $\mathcal{D}_{\text{calib}}$ is constructed using 10 randomly sampled prompts from T2V-CompBench, together with random initial noise seeds. These calibration prompts and seeds are kept disjoint from those used in the final VBench evaluation.

Despite the limited sample size ($N = 10$) and the inherent stochasticity of diverse semantic conditions, the derived Look-Up Table (LUT) for $\hat{\gamma}_{t,s}$ captures a stable profile of latent dynamics that generalizes effectively to unseen prompts during online inference.

B.2 Stage-Aware Calibration Procedure

Calibration follows the online compute–predict schedule using a closed-loop procedure, as summarized in Algorithm 1. We approximate the calibration objective in Eq. (7) through sample-wise directional grid searches followed by aggregation. At each prediction location (t, s), candidate orders in $\mathcal{G}$ are evaluated using Eqs. (5) and (6). The prediction associated with the minimum directional loss is injected into the residual stream:

$$\mathbf{H}_{t,s} \leftarrow \mathbf{H}_{t,s} + \mathrm{gate}_{t,s} \odot \hat{\mathbf{F}}^{*}_{t,s}, \tag{33}$$

where $\mathbf{H}_{t,s}$ is the current hidden residual state and $\mathrm{gate}_{t,s}$ is the corresponding model gate. Subsequent calibration therefore accounts for recursively propagated prediction errors.

For each timestep–location pair (t, s), the selected orders are collected across calibration samples and aggregated to obtain $\gamma^{*}_{t,s}$. The refined order $\hat{\gamma}_{t,s} = \lambda\gamma^{*}_{t,s}$ is then stored in the LUT. The feature cache contains the zeroth-order feature $\Delta^0\mathbf{F}_{t_c,s} = \mathbf{F}_{t_c,s}$ and its finite differences, where t_c denotes the latest full-compute timestep.

B.3 Dissecting Trajectory Dynamics: Implicit Regularization vs. Dynamic Rectification

To rigorously validate the efficacy of the optimal fractional orders found during the aforementioned offline calibration, we shift our focus to the long-horizon error propagation during online inference. By injecting the extrapolated features back into the recursive forward pass, we compare our FADE framework against the Ground-Truth full computation (GT-FI) and Standard Taylor extrapolation (ST-FI).

B.3.1 The Illusion of Implicit Regularization in Taylor Expansion

A critical observation in our closed-loop experiments is the highly non-linear nature of the diffusion trajectory. Due to the inherent architectural complexity of Diffusion Transformers (e.g., the global receptive field of self-attention mechanisms), the true latent manifold (GT-FI) exhibits dense, high-frequency structural transitions and abrupt numerical spikes across layers.

However, when employing the standard Taylor expansion (ST-FI) for caching, we observe a phenomenon we term **Implicit Regularization**. The strict integer-order assumption ($\gamma = 1.0$) acts as a rigid mathematical constraint, aggressively smoothing out these necessary trajectory spikes. It forces the feature evolution into a sub-optimal, overly-regularized path. While this may locally suppress extreme outliers, it fundamentally destroys the high-frequency semantics required for complex generation, inevitably leading to catastrophic directional divergence in later stages.

B.3.2 FADE: Dual-Stage Rectification under Recursive Error

In contrast, FADE avoids this rigid regularization by treating the fractional order as a dynamic, stage-aware degree of freedom. By explicitly simulating the recursive error accumulation during the offline calibration, FADE introduces a **Dual-Stage Rectification** mechanism to faithfully reconstruct the GT trajectory:

- **Directional Rectification via Fractional Order:** Instead of suppressing the structural spikes, the optimized Mittag-Leffler parameters (γ^*) flexibly adapt to the local curvature of

Algorithm 1 Stage-Aware Closed-Loop Offline Calibration for FADE

Input: calibration dataset $\mathcal{D}_{\text{calib}}$, candidate grid $\mathcal{G}$, compute-step schedule $\mathcal{C}$ including warm-up, maximum order m, refinement factor λ
Output: refined fractional-order table $\text{LUT}[t, s]$
Initialize an empty collection $\mathcal{A}_{t,s}$ for each target pair (t, s)
for each calibration sample $\mathbf{q} \in \mathcal{D}_{\text{calib}}$ **do**
  Initialize the diffusion latent and conditioning from $\mathbf{q}$
  for each denoising timestep t **do**
    if $t \in \mathcal{C}$ **then**
      Run the standard full forward pass to obtain model output $\mathbf{y}_t$
      Update feature histories and caches $\mathcal{H}_s \leftarrow \{\Delta^i \mathbf{F}_{t,s}\}_{i=0}^{m}$ for all target locations s
      $t_c \leftarrow t$
    else
      Compute the prediction distance δ from t_c to t
      for each target location s in forward execution order **do**
        Advance normal forward computations to location s
        Obtain the reference feature $\mathbf{F}_{\text{GT}}(t, s)$ by full computation
        Evaluate all $\gamma \in \mathcal{G}$ using cache $\mathcal{H}_s$, distance δ, and Eq. (5)
        Select $\gamma^*_{t,s}(\mathbf{q})$ minimizing Eq. (6) and retain its prediction $\hat{\mathbf{F}}^*_{t,s}$
        Append $\gamma^*_{t,s}(\mathbf{q})$ to $\mathcal{A}_{t,s}$
        $\mathbf{H}_{t,s} \leftarrow \mathbf{H}_{t,s} + \text{gate}_{t,s} \odot \hat{\mathbf{F}}^*_{t,s}$
      end for
      Complete the remaining forward computations to obtain model output $\mathbf{y}_t$
    end if
    Advance the diffusion latent using $\mathbf{y}_t$ and the native sampler
  end for
end for
for each pair (t, s) with $\mathcal{A}_{t,s} \neq \emptyset$ **do**
  $\gamma^*_{t,s} \leftarrow \text{Aggregate}(\mathcal{A}_{t,s})$
  $\text{LUT}[t, s] \leftarrow \lambda \gamma^*_{t,s}$
end for
return LUT

the manifold. By strictly monitoring the cosine similarity (Algorithm **??**), FADE actively corrects the *directional drift* that plagues standard caching methods.

- **Intrinsic–Extrinsic Mismatch Correction:** The stage-aware LUT provides state-specific fractional orders, but practical inference still uses a proxy distance instead of the intrinsic propagation scale. To account for this residual propagation mismatch, FADE applies a global refinement factor λ to obtain $\hat{\gamma}_{t,s} = \lambda \gamma^*_{t,s}$, while preserving the state-specific structure encoded in the LUT.

Through this decoupled rectification strategy, FADE is able to faithfully trace the abrupt fluctuations of the ground-truth manifold, completely bypassing the implicit regularization trap of Euclidean-based forecasting methods.

B.4 Empirical Validation of High Sample Consistency on DiT

To rigorously justify the use of a minuscule calibration set, we conducted an extensive empirical study focusing on the consistency of the optimal fractional order specifically within the **DiT-XL/2** architecture. We computed the optimal $\gamma^*_{t,s}$ across 10 independent random noise initializations on ImageNet-512.

As summarized in Table 6, the optimal fractional order at the stable denoising phase ($t = 40$) exhibits a clear layer-dependent pattern. While the early and middle layers naturally display moderate variance to adapt to diverse initial noise inputs, the deep layers (Layers 19–27) maintain strong cross-sample consistency. Most importantly, empirical generation results confirm that the moderate standard deviation observed in the shallow stages has **negligible impact** on the final perceptual

quality. The stable deeper layers effectively absorb these initial fluctuations, ensuring that a single offline calibration remains highly robust across any unseen prompts.

Table 6: Cross-sample consistency of the optimal fractional order γ^* in DiT attention layers. We highlight the statistics at the most stable denoising phase (Timestep $t = 40$). The consistent behavior in deep layers and its robustness to early-stage variance formally validate our offline calibration strategy.

Network Phase	**Layer Indices**	**Attention Module (Attn)**		
		Mean γ^*	**Std (σ)**	**MAD**
Early Layers	Layers 0–9	0.64	0.144	0.128
Middle Layers	Layers 10–18	0.96	0.129	0.119
Deep Layers	Layers 19–27	0.46	0.128	0.109
Global Summary	**All Layers**	**0.68**	0.120	0.123

C Error Propagation and Stability Analysis

Scope of the analysis. We provide a local and conditional error analysis for the regime where the local latent evolution is better described by an intrinsic fractional propagation law. The goal is not to prove a universal advantage of FADE over all possible predictors, but to explain why a state-adaptive fractional coefficient law can reduce the mismatch between the intrinsic feature trajectory and its practical approximation based on an extrinsic prediction scale.

Notation. Let t denote the denoising timestep and s denote the latent state index. We write

$$\mathbf{F}_{t,s} := \mathbf{F}(\mathbf{x}_t; s),$$

and let $\Delta^i \mathbf{F}_{t,s}$ denote the i-th order backward finite difference computed from cached historical features at state s. For compactness, define the fractional coefficient

$$b_i(\gamma) := \frac{1}{\Gamma(\gamma i + 1)}. \tag{34}$$

Here, δ denotes the extrinsic prediction scale used in practical inference, while $\delta_{\mathcal{M},t,s}$ denotes the intrinsic propagation scale along the local feature trajectory. The difference between these scales gives rise to a potential intrinsic–extrinsic propagation mismatch.

Assumption 1: intrinsic fractional local evolution. For each timestep–state pair (t, s), assume that there exists a local intrinsic fractional order $\gamma^\dagger_{t,s} > 0$ such that the ground-truth local evolution admits the following truncated intrinsic expansion:

$$\mathbf{F}^{\mathrm{GT}}_{t-\delta,s} = \mathbf{F}_{t,s} + \sum_{i=1}^{m} b_i(\gamma^\dagger_{t,s}) \Delta^i \mathbf{F}_{t,s} (-\delta_{\mathcal{M},t,s})^i + \mathbf{R}^{(t,s)}_{m+1}, \tag{35}$$

where the truncation remainder satisfies

$$\left\| \mathbf{R}^{(t,s)}_{m+1} \right\| \le C_{t,s} \delta^{m+1}_{\mathcal{M},t,s}. \tag{36}$$

This assumption is local and conditional: it only states that the nearby feature evolution can be approximated by a truncated intrinsic ML-type expansion.

FADE predictor. FADE uses the extrinsic prediction scale δ and the globally refined fractional order

$$\hat{\gamma}_{t,s} = \lambda \gamma^*_{t,s}, \tag{37}$$

where $\gamma^*_{t,s}$ is obtained from offline stage-aware calibration. The corresponding predictor is

$$\widehat{\mathbf{F}}^{\mathrm{FADE}}_{t-\delta,s} = \mathbf{F}_{t,s} + \sum_{i=1}^{m} b_i(\lambda \gamma^*_{t,s}) \Delta^i \mathbf{F}_{t,s} (-\delta)^i. \tag{38}$$

Theorem 1: residual-aware FADE error bound. Under Assumption 1, the prediction error of FADE satisfies

$$\left\|\mathbf{F}^{\mathrm{GT}}_{t-\delta,s} - \widehat{\mathbf{F}}^{\mathrm{FADE}}_{t-\delta,s}\right\| \le C_{t,s}\delta^{m+1}_{\mathcal{M},t,s} + \sum_{i=1}^{m} \left\|\Delta^i \mathbf{F}_{t,s}\right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^{\dagger}_{t,s} i + 1)} - \frac{(-\delta)^i}{\Gamma(\lambda\gamma^{*}_{t,s} i + 1)} \right|. \tag{39}$$

Proof. Subtracting Eq. (38) from Eq. (35) gives

$$\mathbf{F}^{\mathrm{GT}}_{t-\delta,s} - \widehat{\mathbf{F}}^{\mathrm{FADE}}_{t-\delta,s} = \mathbf{R}^{(t,s)}_{m+1} + \sum_{i=1}^{m} \left[b_i(\gamma^{\dagger}_{t,s})(-\delta_{\mathcal{M},t,s})^i - b_i(\lambda\gamma^{*}_{t,s})(-\delta)^i \right] \Delta^i \mathbf{F}_{t,s}. \tag{40}$$

Taking norms and applying the triangle inequality yields

$$\left\|\mathbf{F}^{\mathrm{GT}}_{t-\delta,s} - \widehat{\mathbf{F}}^{\mathrm{FADE}}_{t-\delta,s}\right\| \le \left\|\mathbf{R}^{(t,s)}_{m+1}\right\| + \sum_{i=1}^{m} \left\|\Delta^i \mathbf{F}_{t,s}\right\| \left| b_i(\gamma^{\dagger}_{t,s})(-\delta_{\mathcal{M},t,s})^i - b_i(\lambda\gamma^{*}_{t,s})(-\delta)^i \right|. \tag{41}$$

Using Eq. (36) and the definition of $b_i(\cdot)$ gives Eq. (39). □

Decomposing the mismatch term. The second term in Eq. (39) contains both intrinsic–extrinsic scale mismatch and coefficient mismatch. Indeed, for each order i, we have

$$\begin{aligned} &\left| b_i(\gamma^{\dagger}_{t,s})(-\delta_{\mathcal{M},t,s})^i - b_i(\lambda\gamma^{*}_{t,s})(-\delta)^i \right| \\ &\le \underbrace{\left| b_i(\gamma^{\dagger}_{t,s}) \right| \left| \delta^i_{\mathcal{M},t,s} - \delta^i \right|}_{\text{intrinsic–extrinsic scale mismatch}} + \underbrace{\delta^i \left| b_i(\gamma^{\dagger}_{t,s}) - b_i(\lambda\gamma^{*}_{t,s}) \right|}_{\text{fractional coefficient mismatch}}. \end{aligned} \tag{42}$$

This shows explicitly that FADE can reduce the coefficient mismatch by calibrating $\gamma^{*}_{t,s}$ and refining it with λ, while the remaining scale discrepancy arises from the difference between the extrinsic prediction scale δ and the intrinsic propagation scale $\delta_{\mathcal{M},t,s}$.

Smoothness of the coefficient law. Assume that the relevant fractional orders lie in a compact interval

$$I = [\gamma_{\min}, \gamma_{\max}] \subset (0, \infty).$$

Since $b_i(\gamma) = 1/\Gamma(\gamma i + 1)$ is smooth on I, there exists a finite constant

$$L_i := \sup_{\gamma \in I} \left| \frac{d}{d\gamma} b_i(\gamma) \right| < \infty, \tag{43}$$

such that

$$|b_i(\gamma_1) - b_i(\gamma_2)| \le L_i |\gamma_1 - \gamma_2|, \qquad \forall \gamma_1, \gamma_2 \in I. \tag{44}$$

Therefore, Eq. (42) implies

$$\begin{aligned} &\left| b_i(\gamma^{\dagger}_{t,s})(-\delta_{\mathcal{M},t,s})^i - b_i(\lambda\gamma^{*}_{t,s})(-\delta)^i \right| \\ &\le \left| b_i(\gamma^{\dagger}_{t,s}) \right| \left| \delta^i_{\mathcal{M},t,s} - \delta^i \right| + L_i \left| \gamma^{\dagger}_{t,s} - \lambda\gamma^{*}_{t,s} \right| \delta^i. \end{aligned} \tag{45}$$

This bound separates the error into an intrinsic–extrinsic scale mismatch and an order-induced coefficient mismatch.

Taylor predictor. The vanilla Taylor predictor corresponds to the special case $\gamma = 1$, since

$$b_i(1) = \frac{1}{\Gamma(i+1)} = \frac{1}{i!}.$$

Its prediction form is

$$\widehat{\mathbf{F}}^{\mathrm{Taylor}}_{t-\delta,s} = \mathbf{F}_{t,s} + \sum_{i=1}^{m} \frac{1}{i!} \Delta^i \mathbf{F}_{t,s} (-\delta)^i. \tag{46}$$

Following the same derivation as above, we obtain

$$\begin{aligned} \left\| \mathbf{F}^{\mathrm{GT}}_{t-\delta,s} - \widehat{\mathbf{F}}^{\mathrm{Taylor}}_{t-\delta,s} \right\| \leq\ & C_{t,s} \delta^{m+1}_{\mathcal{M},t,s} \\ & + \sum_{i=1}^{m} \left\| \Delta^i \mathbf{F}_{t,s} \right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^{\dagger}_{t,s} i + 1)} - \frac{(-\delta)^i}{i!} \right|. \end{aligned} \tag{47}$$

Conditional advantage over Taylor. The truncation term in Eq. (39) and Eq. (47) is shared. Therefore, FADE has a tighter derived upper bound than Taylor whenever

$$\sum_{i=1}^{m} \left\| \Delta^i \mathbf{F}_{t,s} \right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^{\dagger}_{t,s} i + 1)} - \frac{(-\delta)^i}{\Gamma(\lambda \gamma^{*}_{t,s} i + 1)} \right| < \sum_{i=1}^{m} \left\| \Delta^i \mathbf{F}_{t,s} \right\| \left| \frac{(-\delta_{\mathcal{M},t,s})^i}{\Gamma(\gamma^{\dagger}_{t,s} i + 1)} - \frac{(-\delta)^i}{i!} \right|. \tag{48}$$

This condition states that the refined fractional coefficient law must approximate the intrinsic local evolution more closely than the rigid Taylor coefficient law, accounting for the difference between intrinsic and extrinsic propagation scales. It is a local and conditional statement, rather than a universal guarantee over arbitrary latent processes.

Role of the global refinement factor λ. The offline-calibrated fractional orders $\gamma^{*}_{t,s}$ may still exhibit a global scale bias in practical inference. To account for this potential bias, we introduce a global refinement factor λ and use

$$\hat{\gamma}_{t,s} = \lambda \gamma^{*}_{t,s}.$$

This provides a lightweight global adjustment to the calibrated orders while preserving the state-specific structure captured by the offline LUT.

Summary of the local error analysis. Under the intrinsic fractional local evolution assumption, the prediction error is bounded by a truncation term and a mismatch term involving propagation scales and fractional coefficients. Taylor extrapolation corresponds to the $\gamma = 1$ case of the fractional coefficient family. FADE calibrates stage-adaptive orders $\gamma^{*}_{t,s}$ and applies the refined orders $\hat{\gamma}_{t,s} = \lambda \gamma^{*}_{t,s}$ during inference. When its mismatch term is smaller than that of Taylor, FADE has a tighter derived upper bound. These designs provide a more flexible local approximation of the intrinsic dynamics, consistent with the improved robustness observed under large prediction distances.

D Comprehensive Analysis of Latent Trajectory Dynamics

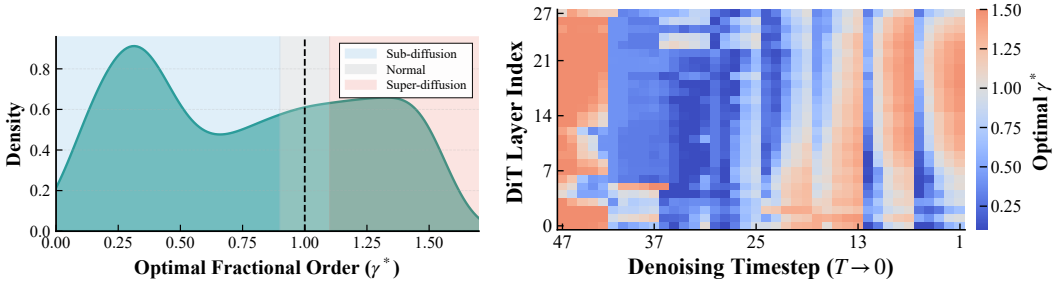
D.1 Empirical Analysis of Fractional Dynamics on DiT-XL/2

We analyze the optimal fractional orders γ^{*} across the 28 layers of DiT-XL/2 during a standard 50-step generation process. Statistics are averaged over 50 randomly sampled generation trajectories. Figure 6 shows the global distribution and spatiotemporal variation of the calibrated orders.

Global Distribution: Fractional Regimes Dominate. As shown in Figure 6(a), orders below 0.9 occur more frequently than those above 1.1, while the near-integer regime ($\gamma \approx 1$) accounts for approximately 13% of the analyzed states. Offline calibration therefore frequently selects orders that differ from the unit order used in standard Taylor extrapolation.

Temporal Heterogeneity: A U-Shaped Phase Pattern. Figure 6(b) reveals a U-shaped temporal pattern: lower orders are concentrated in the middle phase, while higher orders appear more frequently in the early and late phases.

Spatial Consistency across Network Depth. At a given timestep, the calibrated orders remain relatively aligned across network layers. Together with the temporal variation, this motivates stage-aware selection of the forecasting order.



(a) Global Density Distribution of γ^* (b) Spatiotemporal Heatmap of γ^*

Figure 6: **Spatiotemporal visualization of the optimal fractional orders (γ^*) on DiT-XL/2.** (a) Global KDE distribution of γ^*. (b) Variation across denoising timesteps and transformer layers, showing a broadly shared U-shaped temporal pattern.

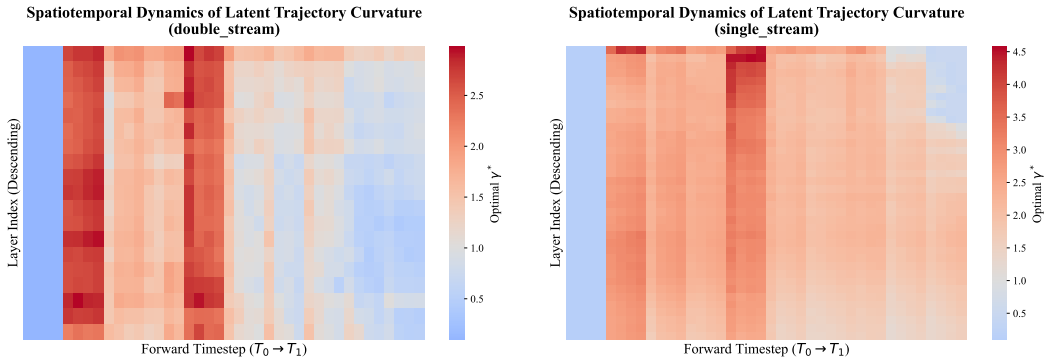
D.2 Empirical Analysis of Fractional Dynamics on FLUX.1

We extend the analysis to FLUX.1, which uses rectified flow matching and contains distinct *DoubleStream* and *SingleStream* blocks. For this analysis, we trace γ^* over 50 generation trajectories using text prompts randomly sampled from DrawBench. Figure 7 shows the calibrated orders across both block types.

Architectural Decoupling: A Shift toward Higher-Order Fractional Regimes. In the *DoubleStream* blocks (Figure 7(a)), the higher-order region ($\gamma > 1.1$) accounts for 56.8% of the states, while orders below 0.9 account for 32.4%. In the *SingleStream* blocks (Figure 7(b)), the higher-order proportion increases to **87.0%**, while the near-integer regime ($\gamma \approx 1$) accounts for only approximately 0.5%.

Temporal Heterogeneity: An Inverted Phase Pattern. Along the timeline $T_0 \to T_1$, FLUX exhibits a temporal pattern opposite to the U-shaped trend observed in DiT. Relatively lower orders appear in the early and late phases, while higher orders are concentrated in the middle phase, as shown in Figure 7.

These observations motivate adapting the forecasting order across both architectural blocks and temporal stages.



(a) DoubleStream Blocks (b) SingleStream Blocks

Figure 7: **Spatiotemporal decoupling of latent dynamics across FLUX.1 architectures.** Calibrated orders in (a) DoubleStream and (b) SingleStream blocks. SingleStream blocks exhibit a greater proportion of higher-order states. Across both streams, higher orders are concentrated in the middle stages of $T_0 \to T_1$.

Table 7: Additional ablation on fractional-order scheduling strategies for FLUX.1.

Method	Speed↑	IR↑	CLIP↑
FADE (fixed $\gamma = 1.2$)	4.99×	0.9883	27.5061
FADE (fixed $\gamma = 1.6$)	4.99×	**1.0016**	27.6207
FADE (fixed $\gamma = 1.8$)	4.99×	0.9978	27.6239
FADE (fixed $\gamma = 2.0$)	4.99×	0.9936	27.6617
FADE (dynamic fit $\gamma(t)$)	4.99×	0.9948	**27.7077**

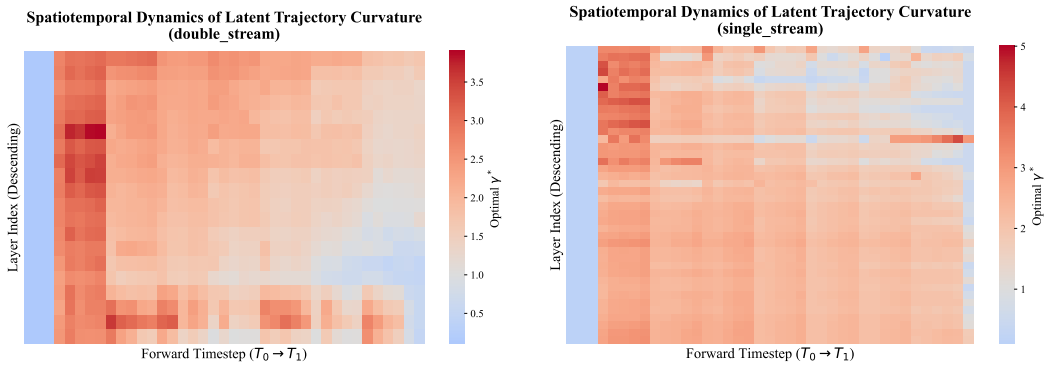
D.3 Empirical Analysis of Fractional Dynamics on HunyuanVideo

We extend the analysis to HunyuanVideo by tracing the optimal fractional orders γ^* across its transformer blocks using diverse video text prompts. Figure 8 presents the distributions across its DoubleStream and SingleStream blocks.

A Broad Shift toward Super-Diffusion-Like Regimes. Figure 8 shows a broad shift toward higher-order regimes ($\gamma > 1.1$) across both block types. This trend resembles the behavior observed in the *SingleStream* blocks of FLUX.1 (Section D.2).

Limitations of a Uniform Integer-Order Assumption. The near-integer regime ($\gamma \approx 1$) accounts for only a small fraction of the calibrated states, while orders below 0.9 are also less prominent than in the image DiT analyzed above. These distributions show that calibration frequently selects orders beyond the fixed $\gamma = 1$ baseline.

Together with the results on DiT and FLUX, these observations motivate adapting the forecasting order across denoising stages and model architectures.



(a) DoubleStream Blocks (b) SingleStream Blocks

Figure 8: **Spatiotemporal decoupling of latent dynamics across HunyuanVideo architectures.** Both (a) DoubleStream and (b) SingleStream blocks show a broad shift toward higher-order regimes ($\gamma > 1.1$), while near-integer states ($\gamma \approx 1$) are comparatively uncommon.

E Additional Ablation Study

E.1 Ablation on Fractional-Order Scheduling

We further study different fractional-order scheduling strategies on FLUX.1 under the $N = 6$ setting. This ablation is designed to examine whether a manually specified fixed order or a simple time-dependent fitted schedule can reliably replace stage-aware calibration.

As shown in Table 7, different fractional-order schedules favor different evaluation metrics. A fixed order of $\gamma = 1.6$ achieves the best ImageReward, while the dynamic fitted schedule obtains the highest CLIP score. This suggests that time-dependent order variation can better align with text-image semantic consistency, but does not uniformly dominate fixed-order schedules across all metrics. These

Table 8: Additional ablation on fractional-order scheduling strategies for DiT-XL/2. "–" indicates that the metric was not recorded.

Method	**FID↓**	**sFID↓**
DDIM-50	2.29	4.29
HiCache	2.51	5.20
Taylor	2.590	5.18
FADE (fixed $\gamma = 0.85$)	2.49	4.81
FADE (Bernoulli-fit $\gamma(t)$)	2.51	5.14

results indicate that fractional-order scheduling is sensitive to both the metric and denoising regime, further motivating stage-aware calibration instead of relying on a single manually specified order.

We also evaluate simple fractional-order scheduling strategies on DiT-XL/2. This study compares the original integer-order Taylor baseline with a manually tuned fixed fractional order and a fitted time-dependent schedule.

As shown in Table 8, both the fixed fractional order and the fitted time-dependent schedule improve over the integer-order Taylor baseline in terms of FID. However, the fitted schedule does not consistently outperform the manually tuned fixed order, suggesting that simple parametric schedules can be sensitive to the chosen functional form. Together with the FLUX.1 results, this supports the use of stage-aware calibration, which avoids imposing a single fixed or manually fitted γ schedule.

F Detailed Experimental Protocols

As mentioned in the main text, experiments are conducted on three representative diffusion transformer models across different generative tasks: FLUX.1-dev Labs [2024] for text-to-image generation, HunyuanVideo Kong et al. [2024] for text-to-video generation, and DiT-XL/2 Peebles and Xie [2023] for class-conditional image generation. This section provides detailed experimental configurations, including hardware infrastructure, sampling protocols, FADE-specific calibration settings, and baseline hyperparameters.

F.1 Hardware and Software Infrastructure

Hardware platforms are selected according to the memory and computation requirements of each model. Specifically, DiT-XL/2 is evaluated on 32GB NVIDIA vGPUs, FLUX.1-dev on NVIDIA H800 GPUs, and HunyuanVideo on NVIDIA A800 GPUs. All models follow their official inference pipelines and default precision settings. Classifier-free guidance (CFG) scales, schedulers, input resolutions, and prompt settings are kept consistent with the standard configurations of the corresponding models.

F.2 Model Configurations and Sampling Parameters

Across all evaluated models, official pretrained checkpoints and inference pipelines are strictly followed. Unless otherwise specified, all methods are evaluated under the same number of inference steps, sampling schedules, random seeds, and acceleration ratios for fair comparison.

- **DiT-XL/2 (ImageNet-1K):** The pretrained DiT-XL/2 model is used for class-conditional image generation at a spatial resolution of 256×256. The inference process is driven by the DDIM sampler with a total of 50 denoising steps.
- **FLUX.1-dev (DrawBench):** The rectified-flow-based FLUX.1-dev model is evaluated for high-fidelity text-to-image generation. Sampling is performed using the default Flow Match Euler scheduler at a resolution of 1024×1024, with 50 inference steps.
- **HunyuanVideo (VBench):** Text-to-video generation experiments are conducted using the official HunyuanVideo inference pipeline. The default flow-matching sampler is adopted

with 50 denoising steps, and all video generation settings follow the official configuration unless otherwise stated.

F.3 FADE and Baseline Configurations

A detailed configuration scheme for the proposed FADE method and the reproduced baselines is provided below.

- **FADE (Ours):** FADE is implemented as a training-free feature forecasting framework. For each model, we first run an offline calibration stage to profile the optimal fractional order γ^* for different denoising stages and network locations. The calibration objective is defined as a directional trajectory matching loss based on cosine distance between the FADE-predicted feature and the corresponding full-computation feature. This choice avoids over-penalizing magnitude variations caused by normalization layers and focuses on the angular consistency of latent feature trajectories.

  The calibrated fractional orders are stored in a static lookup table (LUT), which is indexed by model-specific structural states. For DiT-XL/2, the LUT is indexed by timestep, layer, and module. For FLUX.1-dev, the LUT additionally considers the stream and block structure of the model. For HunyuanVideo, we adopt a coarser LUT indexing strategy due to the substantially higher computational cost and inherent spatiotemporal redundancy of video generation. The LUT is therefore constructed based on timestep and coarse video-transformer block indices. During online inference, FADE directly retrieves the corresponding γ from the LUT and performs fractional-order feature extrapolation without any additional optimization or adaptive search.
- **Baselines:** We compare FADE with representative training-free diffusion acceleration methods, including FORA Selvaraju et al. [2024], HiCache Feng et al. [2026], and TaylorSeer Liu et al. [2025b]. All reproduced baselines strictly follow the implementation protocols and default hyperparameter settings reported in their original papers and open-source repositories. For a fair comparison, all baselines are evaluated using the same base model, sampler, number of inference steps, resolution, CFG scale, prompt set, and acceleration schedule as FADE.

Offline calibration protocol. For each calibration trajectory, a full-computation reference pass collects the oracle features $\mathbf{F}_{t,s}^{\mathrm{GT}}$ at prediction locations. Given the cached feature basis $\{\mathbf{D}_{t,s}^{k}\}_{k=0}^{K-1}$ and prediction distance x, each candidate order γ is evaluated by

$$\widehat{\mathbf{F}}_{t,s}(\gamma) = \sum_{k=0}^{K-1} \frac{1}{\Gamma(\gamma k + 1)} \mathbf{D}_{t,s}^{k} x^{k}, \tag{49}$$

where K is the number of retained basis terms and the zeroth-order coefficient is 1. The candidate minimizing the directional error is selected:

$$\gamma_{t,s}^{*} = \arg\min_{\gamma \in \mathcal{G}} \left[1 - \cos\left(\widehat{\mathbf{F}}_{t,s}(\gamma), \mathbf{F}_{t,s}^{\mathrm{GT}} \right) \right]. \tag{50}$$

The selected predictions are propagated using the closed-loop procedure in Algorithm 1.

For DiT-XL/2, s indexes the layer and module, and we use $\mathcal{G} = \{0.1, 0.2, \ldots, 3.0\}$ with all available cached orders. For FLUX.1-dev, s additionally specifies the stream and block, and we use $\mathcal{G} = \{0.1, 0.2, \ldots, 5.0\}$ with a two-term cached basis for numerical stability. HunyuanVideo follows the FLUX.1-dev protocol with a coarser search and LUT indexed by timestep and video-transformer block. The calibrated orders are stored in a static LUT for inference.

Online inference settings. For each model and cache interval N, we tune λ offline and compute the refined orders:

$$\hat{\gamma}_{t,s} = \lambda \gamma_{t,s}^{*}. \tag{51}$$

The refined orders are stored in the LUT and retrieved directly during inference. We keep λ fixed across all evaluation prompts. The settings are summarized in Table 9.

Table 9: Online inference settings of FADE. The refinement factor λ is fixed after offline calibration and introduces no online search overhead.

Model	Cache interval N	Refinement factor λ
DiT-XL/2	4	0.6
DiT-XL/2	5	0.6
DiT-XL/2	6	0.7
FLUX.1-dev	5	1.3
FLUX.1-dev	6	1.0
FLUX.1-dev	7	1.1
HunyuanVideo	default	1.0

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main contributions of FADE, including feature modeling based on anomalous diffusion, Mittag–Leffler fractional forecasting, offline-profiled LUT retrieval, and training-free DiT acceleration. The stated claims are supported by the theoretical analysis and empirical evaluations across multiple architectures and modalities.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The paper discusses limitations related to the local and conditional nature of the theoretical analysis, the reliance on a one-time offline calibration stage, and the possible dependence of optimal fractional parameters on backbone architectures, sampling schedules, and prediction intervals.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.

- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: The paper states the assumptions underlying the local conditional error analysis, including the intrinsic fractional evolution model and the bounded remainder term. The corresponding derivations are provided in the main text and further detailed in the appendix.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The paper provides the model configurations, sampling settings, acceleration schedules, calibration details, baseline settings, hardware information, and evaluation protocols needed to reproduce the main experiments. An anonymous code repository is also provided to facilitate reproducibility.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same

dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: We provide an anonymous code repository containing the implementation of FADE and scripts for reproducing the main experimental results. The paper and supplemental material describe the experimental settings, model checkpoints, calibration procedure, and evaluation protocols needed for reproduction.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes].

Justification: The paper reports the model configurations, datasets, sampling parameters, acceleration schedules, calibration settings, baseline hyperparameters, hardware platforms, and evaluation protocols in the experimental section and appendix.

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

7. **Experiment statistical significance**

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

   Answer: [No].

   Justification: The paper does not report formal error bars or statistical significance tests. However, the evaluations are conducted under fixed sampling settings and standard benchmark protocols, and the reported inference metrics are empirically stable across the evaluated prompt sets; additional ablations are provided to support the main conclusions.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The paper reports the GPU platforms and memory settings used for the main experiments, including 32GB NVIDIA vGPUs for DiT-XL/2, NVIDIA H800 GPUs for FLUX.1-dev, and NVIDIA A800 GPUs for HunyuanVideo. Runtime and hardware details are provided in the experimental protocol to support reproducibility.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research uses publicly available models, benchmarks, and evaluation protocols, and does not involve human subjects, private data, or sensitive personal information. The anonymous code release preserves author anonymity during submission.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes].

    Justification: The paper discusses that FADE can reduce the computational cost and energy consumption of diffusion model inference, improving the accessibility of high-quality generative models. It also acknowledges that more efficient generation may lower the cost of misuse, such as producing misleading synthetic content, and should be accompanied by responsible deployment and existing safeguards for generative models.

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

11. **Safeguards**

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

    Answer: [N/A]

Justification: The paper does not release new pretrained generative models, scraped datasets, or high-risk data assets. The released code implements a training-free inference acceleration method for existing diffusion models, and responsible use should follow the safeguards and usage policies of the underlying base models.

Guidelines:

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

12. **Licenses for existing assets**

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

Answer:[Yes]

Justification: The paper cites the original works for the pretrained models, benchmarks, and baseline methods used in the experiments, including DiT-XL/2, FLUX.1, HunyuanVideo, DrawBench, VBench, and related acceleration baselines. We use these assets according to their released terms and licenses, and our anonymous code release includes attribution to the corresponding sources.

Guidelines:

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

13. **New assets**

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

Answer: [Yes]

Justification: The paper introduces the FADE implementation as a new code asset. It is released through an anonymized repository with documentation and experimental instructions to support reproduction of the main results.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.

- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments, human-subject studies, or newly collected human annotations. All evaluations are conducted using existing public benchmarks and automated metrics.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments, human-subject studies, or newly collected human annotations. Therefore, IRB approval or equivalent review is not applicable.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: The core method development does not involve LLMs as an important, original, or non-standard component. Any use of language tools, if applicable, is limited to writing, editing, or formatting and does not affect the methodology, experiments, scientific rigor, or originality of the work.

Guidelines:

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