LEAP: Accelerating ODE-based Diffusion Sampling with Optimal Linear Leaps

Haowei Zheng[*†]
zhwei7107@zju.edu.cn
Zhejiang University
Hangzhou, China

Zhenyu Zhou[*†]
zhyzhou@zju.edu.cn
Zhejiang University
Hangzhou, China

Jiawei Chen[*†]
sleepyhunt@zju.edu.cn
Zhejiang University
Hangzhou, China

Can Wang[*†]
wcan@zju.edu.cn
Zhejiang University
Hangzhou, China

Defang Chen[‡]
defchern@berkeley.edu
University of California, Berkeley
California, United States

arXiv:2610.25918v1 [cs.CV] 19 Oct 2026

Abstract

Diffusion models primarily rely on general-purpose ODE-based numerical methods for training-free accelerated sampling, but performance degrades with coarse time discretization. In this paper, we propose *optimal linear leaps* (LEAP), a specialized method grounded in theoretical and empirical investigations of sampling trajectory geometry. Built upon low-dimensionality and cross-trajectory consistency, LEAP formulates a supervised least-squares optimization problem within the local trajectory subspace, achieving a precise approximation of future states without model evaluations. Extensive experiments demonstrate that LEAP outperforms previous methods across diverse modalities, including text-to-image (Stable Diffusion 3.5, FLUX.1-dev) and image-to-3D (TRELLIS.2) generation. Furthermore, LEAP can be seamlessly integrated into existing solver-based methods, offering a plug-and-play enhancement for modern diffusion models. Code is available at https://github.com/zju-pi/LEAP.

CCS Concepts

• **Computing methodologies** → **Computer vision**.

Keywords

Diffusion Models, Training-free Acceleration, Solver-based

ACM Reference Format:
Haowei Zheng, Zhenyu Zhou, Jiawei Chen, Can Wang, and Defang Chen. 2026. LEAP: Accelerating ODE-based Diffusion Sampling with Optimal Linear Leaps. In *Proceedings of the 34th ACM International Conference on Multimedia (MM '26), November 10–14, 2026, Rio de Janeiro, Brazil*. ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3767308.3836106

1 Introduction

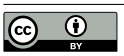
Diffusion-based generative models [9, 29, 32, 33] have achieved unprecedented success in multimedia content synthesis, ranging from image [24, 25, 27], video [3, 10], audio [6, 13], and 3D [11, 41]. Despite their remarkable generation quality, the slow sampling process imposes a critical bottleneck for latency-sensitive applications.

To address this issue, numerous acceleration techniques have been proposed. Training-based approaches distill a pre-trained multi-step model into an extremely few-step student model [12, 20, 28, 31]. However, distilling modern diffusion models with billions of parameters (e.g., 8B Stable Diffusion 3.5 [8] and 12B FLUX [2]) is challenging and inflexible to accommodate community-tuned variants and LoRA modules. In contrast, training-free methods require minimal parameter updates, offering superior plug-and-play flexibility. Recent progress includes **(i)** solver-based method for calculating ordinary differential equations (ODEs) [16, 18, 30, 46, 49], **(ii)** optimized timestep scheduling [4, 26, 43], and **(iii)** feature caching [14, 21, 39, 44]. Existing high-order multi-step solvers [16, 18, 46] estimate future states via a linear combination of historical states where the coefficients are derived from numerical analysis, assuming sufficiently small and near-uniform step sizes. However, practical acceleration typically employs large and non-uniform step sizes, making their assumptions and associated convergence guarantees less reliable. Concurrently, emerging research on the geometric regularity of diffusion trajectories [4, 5, 36, 38] has uncovered intriguing empirical patterns, such as cross-trajectory consistency and low-dimensional concentration. While previous studies have qualitatively observed these regularities, a rigorous mathematical foundation remains largely absent. To move beyond heuristic designs toward a principled approach, a deeper and more rigorous investigation into the underlying mechanics is essential.

To bridge this gap, we conduct a comprehensive theoretical analysis of the geometric properties inherent in diffusion trajectories, yielding the following two theoretical insights. Firstly, we establish a theoretical bound on the alignment error between trajectories after orthogonal rotational alignment, revealing the cross-trajectory consistency of diffusion models. Secondly, we rigorously characterize the intrinsic low-dimensionality of diffusion sampling trajectories, proving that individual paths are confined to a low-dimensional subspace. Our analysis reveals that despite different noise initializations, these generative paths share consistent low-dimensional dynamics.

Leveraging these two mathematically grounded priors, we propose *optimal linear leaps* (LEAP), a training-free, solver-based acceleration framework. In contrast to traditional numerical solvers

[*]State Key Laboratory of Blockchain and Data Security, Zhejiang University.
[†]Hangzhou High-Tech Zone (Binjiang) Institute of Blockchain and Data Security.
[‡]Corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License.
MM '26, Rio de Janeiro, Brazil
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2213-4/2026/11
https://doi.org/10.1145/3767308.3836106

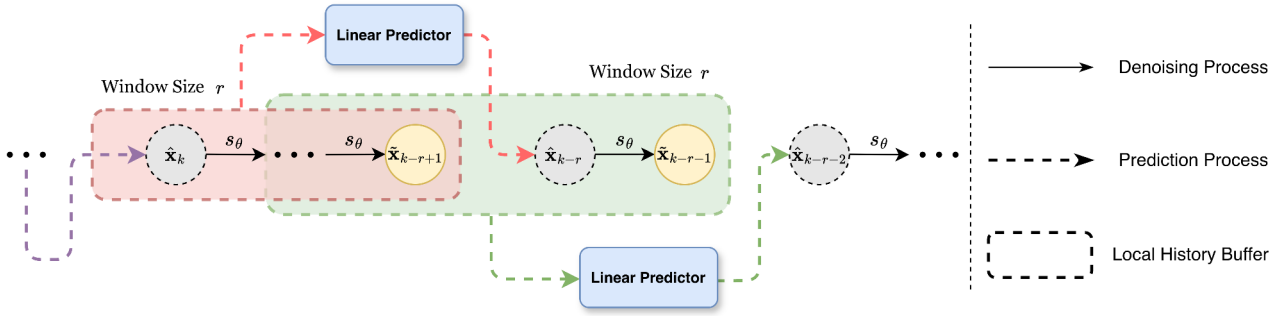


Figure 1: Overview of the proposed LEAP. A linear predictor (blue boxes) extrapolates the subsequent score from a sliding history buffer of size r (dashed boxes), yielding an approximated state $\hat{\mathbf{x}}$ (grey circles) without model evaluations. To prevent error drift, a model evaluation subsequently refines this approximation into a corrected state $\tilde{\mathbf{x}}$ (yellow circles).

that rely on rigid Taylor coefficients, our method introduces a data-driven multi-step strategy powered by least-squares optimization. The intrinsic low-dimensionality of diffusion trajectories suggests that local path segments can be accurately characterized by a linear subspace. Leveraging this geometric regularity, we formulate the prediction as a least-squares optimization problem, which identifies the optimal linear coefficients by minimizing the discrepancy between linearly extrapolated historical states and multi-step ground-truth trajectory samples. Motivated by the cross-trajectory consistency, we demonstrate that the optimal linear extrapolation coefficients are sample-agnostic. This property guarantees the transferability of coefficients derived from a limited set of reference trajectories before inference, bypassing the overhead for per-sample optimization. LEAP significantly enhances the sampling efficiency while preserving high generation quality. Crucially, as it operates entirely at the macroscopic trajectory level, our approach functions as a model-agnostic and plug-and-play module. This allows for seamless integration into diverse diffusion models without any architectural modifications. Extensive experiments across diverse modalities, including text-to-image generation and text-to-3D generation demonstrate the superior performance and versatility of our method.

In summary, our main contributions are as follows:

- We prove the intrinsic low-dimensionality and geometric isomorphism of generative trajectories across diverse noise initializations.
- Motivated by these theoretical insights, we propose LEAP, a novel multi-step ODE solver that leverages least-squares alignment to learn sample-agnostic extrapolation coefficients, enabling highly accurate linear leaps with minimal computational overhead.
- Extensive experiments across diverse generative tasks that spanning both text-to-tmage synthesis and complex image-to-3D generation demonstrate the effectiveness of our method compared to state-of-the-art training-free baselines.

2 Preliminary

Diffusion Models. Continuous-time diffusion models progressively corrupt data $\mathbf{x}_0 \sim p_{data}(\mathbf{x})$ into a tractable noise distribution $\mathbf{x}_T \sim p_{prior}(\mathbf{x})$ via a forward Stochastic Differential Equation (SDE) [33]:

$$\mathrm{d}\mathbf{x}_t = f(t)\mathbf{x}_t\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}_t, \qquad t \in [0, T], \tag{1}$$

where $\mathbf{w}_t$ is the standard Wiener process [22], while $f(t)$ and $g(t)$ denote the drift and diffusion coefficients. Crucially, the generative process that reverses the forward SDE to recover the data distribution is governed by the reverse-time SDE $\mathrm{d}\mathbf{x}_t = [f(t)\mathbf{x}_t - g(t)^2\nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t)]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}}_t$, where $\bar{\mathbf{w}}_t$ denotes a standard Wiener process in the reverse-time direction. To implement the generative process, diffusion models estimate the intractable *score function* $\nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t)$ using a neural network $s_\theta(\mathbf{x}_t, t)$ trained by score matching objectives. For any diffusion SDE, there exists an Ordinary Differential Equation (ODE) [33] that shares the identical marginal distributions $p_t(\mathbf{x})$ for all $t \in [0, T]$, which is known as the Probability Flow ODE (PF-ODE):

$$\mathrm{d}\mathbf{x}_t = \left[f(t)\mathbf{x}_t - \frac{1}{2}g(t)^2\nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t)\right]\mathrm{d}t. \tag{2}$$

By casting the generative process as a numerical initial value problem, PF-ODE enables deterministic sampling and underpins various training-free solver-based acceleration methods [18, 46].

Flow-based Models. Instead of explicitly defining a diffusion process through an SDE, flow-based generative models formulate a continuous-time Normalizing Flow [1, 15, 17]. These models learn a time-dependent vector field $v(\mathbf{x}_t, t)$ that transports noise to the data distribution via a deterministic ODE:

$$\mathrm{d}\mathbf{x}_t = v(\mathbf{x}_t, t)\mathrm{d}t. \tag{3}$$

To circumvent the direct estimation of intractable marginal vector fields, Flow Matching introduces a simulation-free objective which regresses a neural vector field $v_\theta(\mathbf{x}_t, t)$ towards a conditional vector field $\mathbf{u}_t(\mathbf{x}_t|\mathbf{x}_0)$ that defines the target probability path $p_t(\mathbf{x})$:

$$\mathbb{E}_{t, p_0(\mathbf{x}_0), p_t(\mathbf{x}_t|\mathbf{x}_0)}\left[\|v_\theta(\mathbf{x}_t, t) - \mathbf{u}_t(\mathbf{x}_t \mid \mathbf{x}_0)\|_2^2\right]. \tag{4}$$

Specifically, by setting $T = 1$, analytically solving Eq. 1 yields a reparameterization of $\mathbf{x}_t$ as

$$\mathbf{x}_t = \alpha_t\mathbf{x}_0 + \sigma_t\mathbf{x}_1, \qquad \mathbf{x}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{5}$$

where

$$\alpha_t = \exp\left(\int_0^t f(u)\mathrm{d}u\right), \quad \sigma_t = \left(\int_0^t g(u)^2\exp\left(-2\int_0^u f(s)\mathrm{d}s\right)\mathrm{d}u\right)^{1/2}. \tag{6}$$

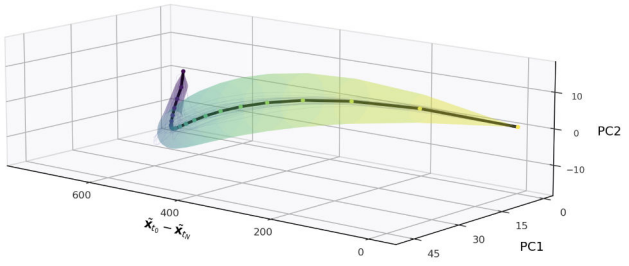


Figure 2: Visualizing the trajectory regularity of diffusion models. Following the visualization protocol in [5], we project 100 sampling trajectories generated with distinct noise and prompt initializations using SD3.5-Medium [8].

The conditional velocity field is thus given by

$$\mathbf{u}_t(\mathbf{x}_t \mid \mathbf{x}_0) = \frac{\dot{\sigma}_t}{\sigma_t}\mathbf{x}_t + \left(\dot{\alpha}_t - \alpha_t \frac{\dot{\sigma}_t}{\sigma_t}\right)\mathbf{x}_0. \tag{7}$$

Contemporary cutting-edge text-to-image models increasingly favor the Flow Matching framework due to its conceptual simplicity and flexible path design [2, 8]. Since they inherit the architecture and training conventions of diffusion models, we still refer to the flow-based formulation as "diffusion" when no ambiguity arises.

Trajectory Regularity. While diffusion models operate in an immensely high-dimensional ambient space, recent empirical observations reveal that their sampling trajectories exhibit remarkable geometric regularity [4, 5, 37]. Principal Component Analysis (PCA) on individual sampling trajectory exhibits that the generative dynamics are inherently *low-dimensional* where the explained variance is dominated by the first few principal components. Furthermore, as illustrated in Fig. 2, when projecting multiple trajectories with distinct noise and prompt initializations into a shared 3D space, they bundle into a smooth 'tube'. This visual alignment highlights a striking *cross-trajectory consistency*, indicating that different diffusion paths share nearly isomorphic macroscopic dynamics. Together, this inter-trajectory low-dimensionality and inter-trajectory consistency constitute the trajectory regularity, serving as the foundational intuition for our subsequent theoretical analysis and the design of our acceleration algorithm.

3 Methodology

3.1 The Intrinsic Geometry of Generative Trajectories

While previous works [4, 5] have observed strong geometric regularity of ODE sampling trajectories to justify existing accelerated sampling methods, a theoretical understanding remains unclear. In this section, we prove that diffusion sampling trajectories can be effectively aligned through orthogonal transformation and are embedded in low-dimensional subspaces. To this end, we perform an analysis within the Gaussian mixture model (GMM) framework with shared spherical covariance to analytically characterize diffusion dynamics. Since GMMs are universal approximators capable of modeling highly complex data manifolds, this assumption not only facilitates closed-form mathematical derivations but also ensures the broad applicability of our theoretical findings. Under this mild assumption, the data distribution is given by

$$p_{\text{data}}(\mathbf{x}) = \sum_{s=1}^{S} \pi_s \mathcal{N}(\boldsymbol{\mu}_s, \sigma_0^2 \mathbf{I}_d), \quad \pi_s > 0, \quad \sum_{r=1}^{S} \pi_s = 1, \tag{8}$$

where S is the number of components, π_s are the mixing weights and $\boldsymbol{\mu}_s$ are the cluster means. Let

$$U := \operatorname{span}\{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_S\}, \quad r := \dim(U), \quad R := \max_{1 \le s \le S} \|\boldsymbol{\mu}_s\|.$$

We consider the exact PF-ODE associated with the variance-preserving forward diffusion. Let $t \in [0, T]$ denote the standard forward diffusion time. Denote $\mathbf{x}_t^{(m)}$ the m-th sampling trajectory evolving backward in time which is initialized from a pure noise state $\mathbf{x}_T^{(m)} \sim p_T$. To analyze the trajectory at discrete intervals, we consider a backward time grid $T = t_N > t_{N-1} > \cdots > t_0 = 0$ and define the trajectory states as

$$\mathbf{D}_n^{(m)} := \mathbf{x}_{t_n}^{(m)} - \mathbf{x}_T^{(m)}, \tag{9}$$

which represents the displacement from the initial noise. Let

$$\alpha_T := \exp\Big(-\frac{1}{2}\int_0^T \beta(s)\, ds\Big), \qquad v_T^2 := \alpha_T^2 \sigma_0^2 + (1 - \alpha_T^2). \tag{10}$$

For brevity, we present a simplified statement with a proof sketch below. The complete theorem including all constants and rigorous proof is detailed in the Appendix A.

Theorem 3.1 (Cross-trajectory consistency). *Suppose that $\mathbf{x}_T^{(1)}, \ldots, \mathbf{x}_T^{(M)}$ are i.i.d. samples from the exact noised marginal p_T. Then for every $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\max_{1 \le m_1, m_2 \le M} \inf_{\mathbf{H} \in O(d)} \max_{0 \le n \le N} \left\| \mathbf{D}_n^{(m_1)} - \mathbf{H}\mathbf{D}_n^{(m_2)} \right\| \le C_{T,1}\, v_T \left(\sqrt{r} + \sqrt{\log \frac{M}{\delta}}\right) + C_{T,2}\, R, \tag{11}$$

where $C_{T,1}, C_{T,2} > 0$ depend only on the diffusion schedule and not on d, n, or the mixture parameters.

Corollary 3.2 (Asymptotic consistency). *If $r + \log M = o(d)$ and $R = o(v_T\sqrt{d})$, then for every fixed $\delta \in (0, 1)$,*

$$\frac{1}{v_T\sqrt{d}} \max_{1 \le m_1, m_2 \le M} \inf_{\mathbf{H} \in O(d)} \max_{0 \le n \le N} \left\| \mathbf{D}_n^{(m_1)} - \mathbf{H}\mathbf{D}_n^{(m_2)} \right\| \xrightarrow[d \to \infty]{\mathbb{P}} 0. \tag{12}$$

Proof sketch. Let P and $Q = \mathbf{I} - P$ denote the orthogonal projections onto U and $U^\perp$, respectively. In the Appendix A, we show that the reverse-time PF-ODE can be written as

$$\mathrm{d}\mathbf{x}_t = \big[a_t \mathbf{x}_t + b_t \mathbf{m}_t(P\mathbf{x}_t)\big]\mathrm{d}t, \tag{13}$$

where $\mathbf{m}_t(\cdot) \in U$ is the posterior mean of the component centers at time t. Consequently, separating the dynamics yields

$$Q\,\mathrm{d}\mathbf{x}_t = a_t Q\mathbf{x}_t\,\mathrm{d}t, \qquad P\,\mathrm{d}\mathbf{x}_t = \big[a_t P\mathbf{x}_t + b_t \mathbf{m}_t(P\mathbf{x}_t)\big]\mathrm{d}t. \tag{14}$$

Therefore, the $U^\perp$-component evolves by a common scalar rescaling, while all mixture-specific non-linear effects are confined to the r-dimensional subspace U.

Geometrically, this implies that after applying the orthogonal alignment, the maximum discrepancy between any pair of trajectories is determined by two factors: the norm difference of the initial noise within the orthogonal complement (i.e., $\big|\|Q\mathbf{x}_T^{(m_1)}\| -$

$\|Q\mathbf{x}_T^{(m_2)}\|\big|$), and the absolute magnitude of the signal component within the data subspace (i.e., $\|P\mathbf{x}_T^{(m)}\|$). Since $Q\mathbf{x}_T^{(m)}$ is a Gaussian vector in dimension $d-r$, its norm concentrates uniformly over m, such that the norm difference is bounded by $O\big(v_T\sqrt{\log(M/\delta)}\big)$. On the other hand, $P\mathbf{x}_T^{(m)}$ lives in an r-dimensional subspace and its norm satisfies

$$\|P\mathbf{x}_T^{(m)}\| = O\left(\alpha_T R + v_T\left(\sqrt{r} + \sqrt{\log(M/\delta)}\right)\right) \tag{15}$$

uniformly over m with high probability. Combining these bounds yields Thm. 3.1, and Cor. 3.2 follows by normalization with $v_T\sqrt{d}$. □

Remark. Cor. 3.2 indicates that in high-dimensional settings ($d \to \infty$) which matches setting of diffusion models, the error bound in Thm. 3.1 becomes practically effectively vanishes. This provides a strong theoretical guarantee that our cross-trajectory alignment is highly accurate in realistic generation scenarios.

Corollary 3.3 (Intrinsic low-dimensionality of individual trajectories). *Under the assumptions of Thm. 3.1, for each sampling trajectory $\mathbf{x}_t^{(m)}$, there exists a linear subspace*

$$\mathcal{S}_m := U \oplus \mathrm{span}\{Q\mathbf{x}_T^{(m)}\} \tag{16}$$

with $\dim(\mathcal{S}_m) \le r+1$, such that

$$\mathbf{x}_{t_n}^{(m)} \in \mathcal{S}_m, \qquad 0 \le n \le N. \tag{17}$$

Equivalently, if $\mathbf{X}^{(m)} \in \mathbb{R}^{(N+1)\times d}$ denotes the trajectory matrix whose n-th row is $(\mathbf{x}_{t_n}^{(m)})^\top$, then

$$\mathrm{rank}(\mathbf{X}^{(m)}) \le r+1. \tag{18}$$

Remark. Cor. 3.3 reveals that all trajectory variance is captured by at most $r+1$ principal components. However, this $(r+1)$-dimensional subspace $\mathcal{S}_m$ is sample-dependent, as it is spanned by the initial noise vector $Q\mathbf{x}_T^{(m)}$. The following corollary establishes a remarkable geometric invariance: by applying a suitable orthogonal transformation to each trajectory, all trajectories can be perfectly aligned into a **single, shared** low-dimensional subspace.

Corollary 3.4 (Universal low-dimensional alignment of diffusion trajectories). *For any fixed unit vector $\mathbf{e} \in U^\perp$, there exist orthogonal matrices $\mathbf{H}_1, \dots, \mathbf{H}_M \in O(d)$ satisfying $\mathbf{H}_m|_U = \mathbf{I}$ and*

$$\mathbf{H}_m Q\mathbf{x}_T^{(m)} = \|Q\mathbf{x}_T^{(m)}\|\,\mathbf{e}, \qquad \forall m \in \{1,\dots,M\}. \tag{19}$$

Under such transformations, the aligned trajectories defined by

$$\bar{\mathbf{x}}_{t_n}^{(m)} := \mathbf{H}_m \mathbf{x}_{t_n}^{(m)} \tag{20}$$

all reside in the same $(r+1)$-dimensional subspace

$$\bar{\mathcal{S}} := U \oplus \mathrm{span}\{\mathbf{e}\}. \tag{21}$$

PF-ODE exhibits two complementary structural properties: cross-trajectory consistency and intrinsic individual low-dimensionality. Thm. 3.1 establishes that, different sampling trajectories can be highly aligned through simple orthogonal transformation. Cor. 3.3 further demonstrates that each trajectory is strictly confined to an $(r+1)$-dimensional subspace. Notably, after a simple per-trajectory alignment introduced by Cor. 3.4, the entire collection of trajectories resides within a universal $(r+1)$-dimensional subspace. Therefore, the high-dimensional sampling dynamics are effectively governed by only a few latent coordinates: one radial coordinate in $U^\perp$ and at most r coordinates within the mean subspace U.

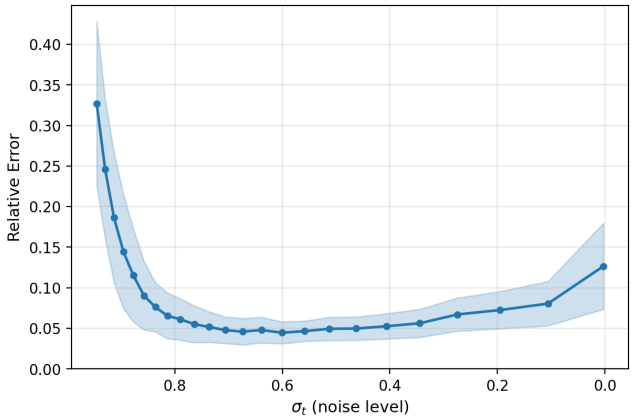


Figure 3: Relative prediction error vs. noise level σ_t. We quantify the projection error of each score onto the subspace spanned by its corresponding historical buffer. The relative error is strictly bounded, using a small window size of $r = 3$.

More importantly, the intrinsic low-dimensionality established in Cor. 3.3 implies that, within any local window of a given trajectory, both the historical scores and the target output reside in a subspace. While this geometric regularity is rigorously established under mild assumption, its validity in real-world generative scenarios warrants empirical verification. To bridge this gap, we investigate the local score dynamics using sampling trajectories from Stable Diffusion 3.5. Specifically, we quantify the orthogonal projection error of the current score $\mathbf{s}_t$ onto the local subspace spanned by a short historical buffer, denoted as $\mathrm{span}(\mathbf{s}_{t+r}, \mathbf{s}_{t+r-1}, \dots, \mathbf{s}_{t+1})$. As illustrated in Fig. 3, the relative projection error is strictly bounded, dropping to a mere 4%–5% even with a small window size $r = 3$. This empirical finding confirms that even for highly complex data distributions, the immediate future score remains strictly confined within the linear span of its recent history. This empirical evident for low-dimensionality seamlessly transforms our theoretical insights into a practical algorithm, leading to our linear extrapolation framework in the subsequent section.

3.2 Optimal Linear Leaps with Aligned Trajectories

In this section, we present our method grounded in the the local linear dependence established above. This geometric regularity underpins the use of a linear extrapolator to predict next score function. The overall pipeline of our approach is illustrated in Fig. 1. To formalize this process, suppose we collect an ensemble of M fine-grained reference sampling trajectories. For the m-th trajectory ($m = 1, \dots, M$), denote $\tilde{\mathbf{s}}_n^{(m)}$ the model output (e.g., estimated score function) at timestep t_n, and let $\tilde{\mathbf{x}}_{n-1}^{(m)}$ be the next state updated using $\tilde{\mathbf{s}}_n^{(m)}$.

Given a window size r, our objective at timestep t_n is to predict the next model output $\tilde{\mathbf{s}}_{n-1}$ by utilizing r stored previous model outputs along the sampling trajectory. Denote the local history

Algorithm 1 Coefficient Bank Construction

Require: Reference score trajectories $\{\mathcal{T}^{(m)}\}_{m=1}^{M}$, where $\mathcal{T}^{(m)} = \{\tilde{\mathbf{s}}_N^{(m)}, \tilde{\mathbf{s}}_{N-1}^{(m)}, \dots, \tilde{\mathbf{s}}_1^{(m)}\}$; window size r; ridge coefficient λ
Ensure: Coefficient bank $C = \{\mathbf{c}_{N-r+1}, \mathbf{c}_{N-r}, \dots, \mathbf{c}_1\}$
for $n = N-r+1, N-r, \dots, 1$ **do**
  for $m = 1, 2, \dots, M$ **do**
    for $j = 1, \dots, r-1$ **do**
      $\Delta_{n,j}^{(m)} \leftarrow \tilde{\mathbf{s}}_{n+j}^{(m)} - \tilde{\mathbf{s}}_{n+j-1}^{(m)}$
    end for
    $\mathbf{B}_n^{(m)} \leftarrow \left[\Delta_{n,r-1}^{(m)}, \dots, \Delta_{n,1}^{(m)}, \tilde{\mathbf{s}}_n^{(m)}\right]$
  end for
  $\mathbf{c}_n^{\star} \leftarrow \arg\min_{\mathbf{c}\in\mathbb{R}^r} \sum_{m=1}^{M} \left\|\tilde{\mathbf{s}}_{n-1}^{(m)} - \mathbf{B}_n^{(m)}\mathbf{c}\right\|_2^2 + \lambda\|\mathbf{c}\|_2^2$
end for
return $C = \{\mathbf{c}_{N-r+1}^{\star}, \mathbf{c}_{N-r}^{\star}, \dots, \mathbf{c}_1^{\star}\}$

Algorithm 2 LEAP Sampling (Predict-Correct)

Require: Initial state $\tilde{\mathbf{x}}_N$; coefficient bank C; window size r; score network s_θ
Ensure: Final sample $\tilde{\mathbf{x}}_0$
Warm-up: run base sampler for r steps
$\mathcal{B} \leftarrow \{\tilde{\mathbf{s}}_N, \tilde{\mathbf{s}}_{N-1}, \dots, \tilde{\mathbf{s}}_{N-r+1}\}$
Update current state to $\tilde{\mathbf{x}}_{N-r}$
for $n = N-r+1$ **down to** 3 **step** 2 **do**
  Predict:
    Compute differences $\Delta_{n,j}$ from $\mathcal{B}$ for $j = 1, \dots, r-1$
    $\hat{\mathbf{s}}_{n-1} \leftarrow \mathbf{c}_{n,r-1}\tilde{\mathbf{s}}_n + \sum_{j=0}^{r-2} \mathbf{c}_{n,j}\,\Delta_{n,r-j-1}$
    Evolve to $\hat{\mathbf{x}}_{n-2}$ via numerical solver using $\hat{\mathbf{s}}_{n-1}$
    Append $\hat{\mathbf{s}}_{n-1}$ to $\mathcal{B}$ and remove the oldest score
  Correct:
    Evaluate exact score $\tilde{\mathbf{s}}_{n-2} \leftarrow s_\theta(\hat{\mathbf{x}}_{n-2})$
    Evolve to $\tilde{\mathbf{x}}_{n-3}$ via numerical solver using $\tilde{\mathbf{s}}_{n-2}$
    Append $\tilde{\mathbf{s}}_{n-2}$ to $\mathcal{B}$ and remove the oldest score
end for
return $\tilde{\mathbf{x}}_0$

matrix of scores and the corresponding target by

$$\mathbf{H}_n^{(m)} = \left[\tilde{\mathbf{s}}_{n+r-1}^{(m)}, \tilde{\mathbf{s}}_{n+r-2}^{(m)}, \dots, \tilde{\mathbf{s}}_n^{(m)}\right] \in \mathbb{R}^{d\times r}. \tag{22}$$

We seek an optimal coefficient vector $\mathbf{w}_n \in \mathbb{R}^r$ that satisfies:

$$\tilde{\mathbf{s}}_{n-1}^{(m)} \approx \mathbf{H}_n^{(m)}\mathbf{w}_n. \tag{23}$$

The cross-trajectory consistency presented in Thm. 3.1 demonstrates that, after proper alignment, different trajectories follow nearly isomorphic low-dimensional dynamics. Consequently, a *single* coefficient vector $\mathbf{w}_n$ can effectively characterize the score evolution of different reference trajectories at a given sampling timestep, enabling the extraction of a universal local extrapolation law. Formally, we derive the step-wise optimal predictor within the family of linear extrapolators by solving the following multi-trajectory least-squares problem:

$$\mathbf{w}_n^{\star} = \arg\min_{\mathbf{w}\in\mathbb{R}^r} \sum_{m=1}^{M} \left\|\tilde{\mathbf{s}}_{n-1}^{(m)} - \mathbf{H}_n^{(m)}\mathbf{w}\right\|_2^2. \tag{24}$$

While the formulation above is conceptually straightforward, directly regressing on $[\tilde{\mathbf{s}}_{n+r-1}, \tilde{\mathbf{s}}_{n+r-2}, \dots, \tilde{\mathbf{s}}_n]$ is often numerically ill-posed. Since consecutive model outputs along smooth segments of the diffusion path are highly correlated, the columns of the history matrix tend to be nearly collinear. As a result, the associated least-squares system becomes poorly conditioned, making the fitted coefficients sensitive to finite-sample error, imperfect cross-trajectory alignment, and numerical instabilities. To mitigate this issue, we reparameterize the local subspace using a *difference basis* rather than the *raw basis*. Specifically, for each trajectory window, we define the first-order differences of the scores as:

$$\Delta_{n,j}^{(m)} := \tilde{\mathbf{s}}_{n+j}^{(m)} - \tilde{\mathbf{s}}_{n+j-1}^{(m)}, \qquad j = 1, \dots, r-1, \tag{25}$$

and define the modified history matrix as:

$$\mathbf{B}_n^{(m)} = \left[\Delta_{n,r-1}^{(m)}, \dots, \Delta_{n,1}^{(m)}, \tilde{\mathbf{s}}_n^{(m)}\right] \in \mathbb{R}^{d\times r}. \tag{26}$$

Rather than directly estimating $\mathbf{w}_n$, we fit an auxiliary coefficient vector $\mathbf{c}_n \in \mathbb{R}^r$ such that

$$\tilde{\mathbf{s}}_{n-1}^{(m)} \approx \mathbf{B}_n^{(m)}\mathbf{c}_n. \tag{27}$$

In practice, we aggregate samples from all reference trajectories and solve a ridge regression problem within the difference basis for each timestep $n = N-1, N-2, \dots, r$:

$$\mathbf{c}_n^{\star} = \arg\min_{\mathbf{c}\in\mathbb{R}^r} \sum_{m=1}^{M} \left\|\tilde{\mathbf{s}}_{n-1}^{(m)} - \mathbf{B}_n^{(m)}\mathbf{c}\right\|_2^2 + \lambda\|\mathbf{c}\|_2^2. \tag{28}$$

Once the step-wise coefficient bank $C = \{\mathbf{c}_{N-r+1}, \mathbf{c}_{N-r}, \dots, \mathbf{c}_1\}$ has been solved, inference proceeds in two phases. First, we execute a warm-up phase by running the initial r denoising steps in a standard way. This initializes the history buffer with the initial r model outputs and their corresponding states:

$$\{\tilde{\mathbf{s}}_N, \tilde{\mathbf{s}}_{N-2}, \dots, \tilde{\mathbf{s}}_{N-r+1}\}, \quad \{\tilde{\mathbf{x}}_N, \tilde{\mathbf{x}}_{N-2}, \dots, \tilde{\mathbf{x}}_{N-r}\}. \tag{29}$$

This warm-up phase provides the minimum context required by the extrapolator. Moving forward, we employ an interleaved sampling strategy that alternates between a network-free prediction step and a network-based correction step. Specifically, suppose the current history buffer contains r most recent scores:

$$\{\tilde{\mathbf{s}}_{n+r-1}, \tilde{\mathbf{s}}_{n+r-2}, \dots, \tilde{\mathbf{s}}_n\}. \tag{30}$$

We first predict the next model output using the learned step-wise coefficients:

$$\hat{\mathbf{s}}_{n-1} := \mathbf{c}_{n,r-1}\tilde{\mathbf{s}}_n + \sum_{j=0}^{r-2} \mathbf{c}_{n,j}\,\Delta_{n,r-j-1}. \tag{31}$$

The extrapolated score $\hat{\mathbf{s}}_{n-1}$ is then utilized to evolve the trajectory to $\hat{\mathbf{x}}_{n-2}$ via numerical solvers, eliminating the need for an explicit network evaluation. The extrapolated score $\hat{\mathbf{s}}_{n-1}$ then serves as a proxy for $\tilde{\mathbf{s}}_{n-1}$ and is pushed into the history buffer, displacing the oldest entry to maintain a fixed window size of r.

Extrapolation alone inevitably accumulates numerical drift over time. To ensure trajectory stability, we implement an interleaved Predict-Correct scheme. Following the short warm-up stage, the solver alternates between network-free extrapolation and exact network evaluation, balancing computational efficiency with numerical robustness, providing a practical acceleration framework for diffusion sampling. Given the step-wise optimality of our linear

extrapolators and their capability to bypass redundant network evaluations, we name our method as *Optimal Linear Leaps* (LEAP). The full algorithms of coefficient bank construction and LEAP sampling are provided in Algorithm 1 and 2.

4 Experiments

We evaluate the proposed LEAP on three cutting-edge diffusion models: SD3.5-Medium [8] and FLUX.1-Dev [2] for text-image task and TRELLIS.2 [41] for image-3D task. The quality of generated images is quantitatively evaluated using three metrics, including ImageReward [42], CLIP Score [23], and HPSv2 [40]. The prompts are drawn from the DrawBench [27] dataset. For image-3D task, we evaluate 3D object on a test set of 100 randomly sampled images from the Google Scanned Objects (GSO) dataset [7] by PSNR and CLIP Score. By default, we fix a window size of $r = 5$ and set the ridge regularization coefficient to $\lambda = 0.0001$ across all experiments. Unless otherwise specified, LEAP is compared with existing state-of-the-art solvers under identical inference settings.

4.1 Main Results

Text-to-image Generation. We evaluate LEAP on text-to-image generation, benchmarking it against representative solvers including Euler method [30], DPM++ [19], UniPC [46], iPNDM [35], LTC-Accel [50] and STORK-4 [34]. As shown in Tab. 1, LEAP consistently achieves top-tier performance across diverse sampling budgets, effectively preserving both text-image alignment and perceptual quality for accelerated sampling. Moreover, LEAP consistently enhances the performance of both first-order and higher-order solvers, demonstrating its versatile plug-and-play capability. By leveraging a pre-computed, step-wise coefficient bank, LEAP seamlessly adapts any solver without any solver-specific hyperparameter tuning or model fine-tuning, making it a solver-agnostic enhancement module. Fig. 4 provides a qualitative comparison of LEAP against state-of-the-art solvers on SD3.5-Medium. Our LEAP exhibits superior performance across four challenging scenarios including (1) text rendering: LEAP accurately synthesizes legible text, whereas baselines suffer from typographic errors; (2) physical lighting: LEAP captures realistic transparency and intricate light refraction; (3) attribute binding: LEAP precisely binds the localized attributes to their respective partitions without semantic leakage.

Image-to-3D Generation. We further validate LEAP on image-to-3D generation using the TRELLIS.2 [41] architecture. As presented in Tab. 2, our method achieves systematic improvements in both PSNR and CLIP Score over the baseline on GSO dataset. The qualitative results shown in Fig. 5 demonstrate LEAP's superior capability in (1) topological correctness, maintaining precise connectivity in complex jointed structures; (2) surface fidelity; (3) geometric completeness, effectively generating intricate and high-frequency geometries, yielding sharper textural details and superior color accuracy.

4.2 Ablation Study

To systematically validate the contribution of each key design choice in the proposed method, we conduct comprehensive ablation studies on SD3.5-Medium. To ensure a rigorous comparison, we isolate the impact of each component by maintaining a fixed

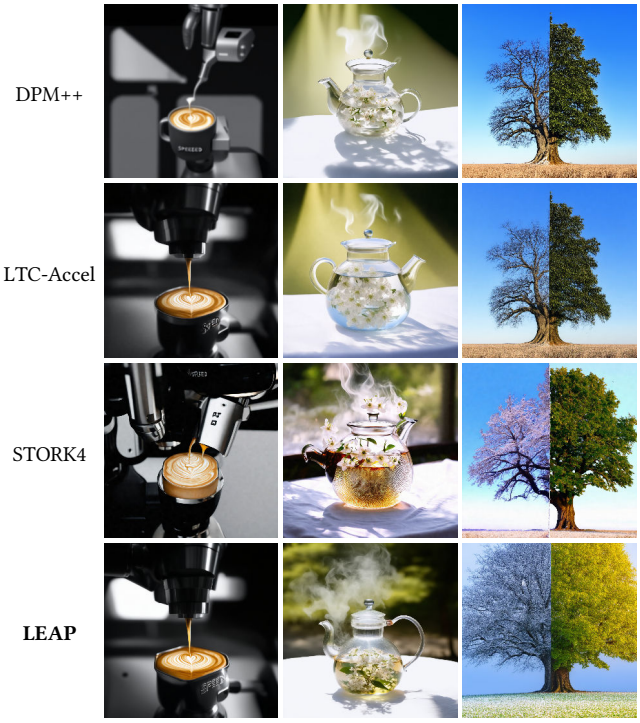


Figure 4: Text-to-image generation on SD3.5-Medium using 10 inference steps across diverse prompts.

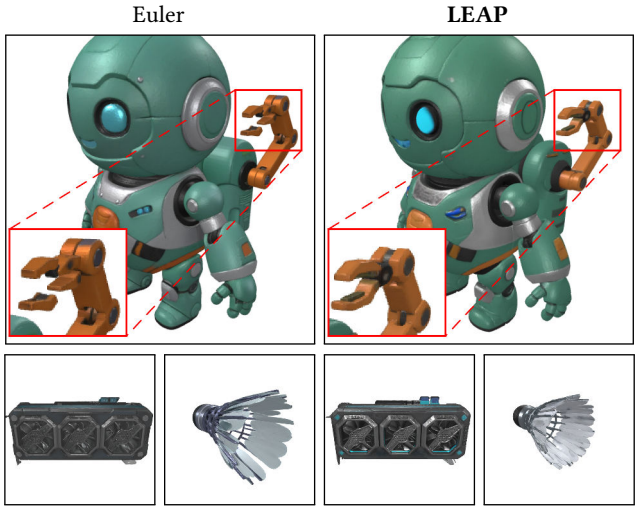


Figure 5: Image-to-3D generation on TRELLIS.2 using 10 inference steps across different image prompts.

N-step sampling schedule, identical random seeds and a consistent ensemble of reference trajectory (default $M = 32$). (a) **Window Size** r. As shown in Fig 6a, we adopt $r = 5$ as the default window size, as it delivers peak or near-peak performance in most scenarios. (b) **Penalty Strength** λ. Fig. 6b demonstrates that LEAP is robust to the penalty coefficient λ across a wide range of values. We retain this penalty term across all experiments to ensure theoretical

Table 1: Quantitative results on SD3.5-Medium and FLUX.1-dev. The best and second-best results are marked in bold and underlined, respectively. Our proposed method (LEAP) is highlighted with a gray background.

Steps	Method	SD3.5-Medium			FLUX.1-dev		
		HPSv2 ↑	ImageReward ↑	CLIPScore ↑	HPSv2 ↑	ImageReward ↑	CLIPScore ↑
10	Euler	25.8866	0.6653	32.7920	28.3538	0.8110	30.6939
	DPM++	26.4565	0.7178	32.9027	28.6365	0.8331	30.7257
	iPNDM	26.4041	0.7063	32.7579	28.8272	0.8956	30.9178
	UniPC	26.4450	0.7122	32.6428	28.8193	0.8839	31.0634
	LTC-Accel	26.2546	0.7324	33.1578	28.9543	0.9193	31.1217
	STORK-4	26.0567	0.6619	32.8704	29.2846	0.9729	31.2895
	LEAP (Euler)	26.8419	0.7692	**33.2982**	29.0536	0.9204	30.9412
	LEAP (DPM++)	**27.0237**	**0.8177**	33.2850	**29.4004**	**0.9816**	**31.4320**
15	Euler	27.4725	0.8416	33.4644	29.2580	0.9470	31.0488
	DPM++	27.8642	0.8438	33.4704	29.5030	0.9588	31.2782
	iPNDM	27.9220	0.8770	33.4416	29.5056	0.9491	31.3836
	UniPC	27.8680	0.8788	33.4703	29.4668	0.9414	31.2339
	LTC-Accel	27.8174	0.9196	**33.7184**	29.5874	0.9576	31.1906
	STORK-4	27.4224	0.8253	33.5179	**29.9693**	**1.0308**	31.3041
	LEAP (Euler)	28.1826	0.9522	33.6817	29.6785	0.9719	31.1046
	LEAP (DPM++)	**28.2152**	**0.9806**	33.5617	29.9024	1.0101	**31.4981**
20	Euler	28.2691	0.9201	33.6945	29.4556	0.9398	30.9759
	DPM++	28.4878	0.9560	33.7391	29.6523	0.9460	31.0277
	iPNDM	28.4252	0.9317	**33.9325**	29.6963	0.9886	31.2203
	UniPC	28.4756	0.9364	33.7757	29.6773	0.9840	31.1858
	LTC-Accel	28.3403	0.9547	33.7901	29.7326	0.9884	31.1142
	STORK-4	28.1835	0.9035	33.8360	29.7927	1.0066	31.2650
	LEAP (Euler)	28.5727	0.9701	33.7192	29.9370	1.0320	31.3433
	LEAP (DPM++)	**28.6259**	**0.9915**	33.5356	**30.1514**	**1.0514**	**31.6484**

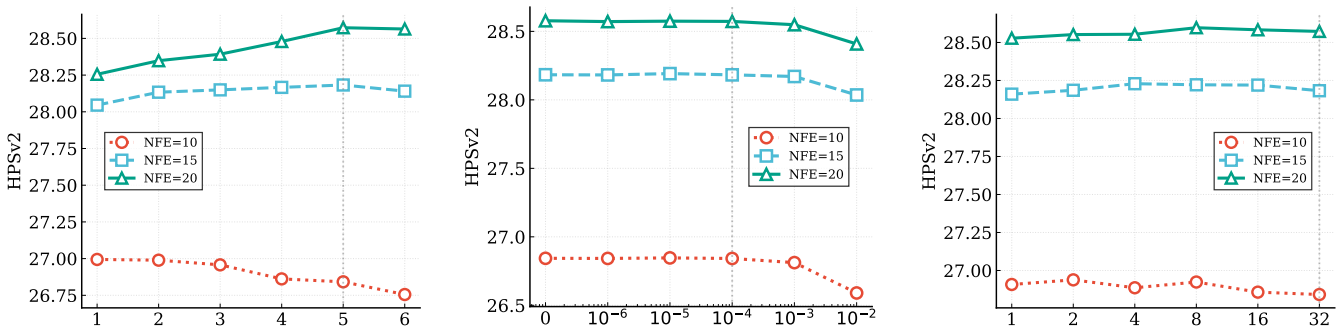
Table 2: Quantitative comparison of image-to-3D generation on the GSO dataset via TRELLIS.2.

Method	PSNR ↑					CLIP Score ↑				
	10	15	20	25	30	10	15	20	25	30
Euler	7.250	7.396	7.235	7.361	7.405	0.8638	0.8654	0.8682	0.8650	0.8658
LEAP (Euler)	**7.414**	**7.537**	**7.337**	**7.410**	**7.408**	**0.8677**	**0.8674**	**0.8693**	**0.8678**	**0.8709**

consistency and numerical stability. (c) **Number of Reference Trajectories** M. As shown in Fig 6c, LEAP exhibits remarkable data efficiency, maintaining high performance even with minimal reference data ($M \leq 2$). This observation perfectly corroborates our theoretical analysis regarding cross-trajectory consistency. We default to $M = 32$ to ensure statistically robust coverage across diverse prompts. (d) **Basis Construction.** Fig. 7 underscores the critical role of our difference-basis reparameterization. The difference basis effectively rectifies the ill-conditioning, ensuring numerical stability and enabling consistent quality improvements across all NFE settings.

5 Related Works

Fast numerical solvers. A predominant branch of training-free acceleration methods focuses on improving the numerical solver employed in reverse-time sampling. Early work such as DDIM [30] established the ODE-based paradigm for fast deterministic generation. Subsequent research has developed increasingly effective high-order and multi-step solvers. PNDM [16] introduced multi-step pseudo numerical methods, showing that leveraging historical samples can significantly improve the quality-speed trade-off over first-order samplers. Its improved implementation, iPNDM [35], is frequently utilized in modern diffusion sampling. DEIS [45] further exploited the semi-linear structure of diffusion dynamics through



(a) Window Size (r) (b) Penalty Strength (λ) (c) Number of Reference Trajectories (M)

Figure 6: Ablation studies on key hyperparameters in Stable Diffusion 3.5-Medium. We evaluate the impact of (a) Window Size, (b) Penalty Strength, and (c) Number of Tokens on the HPSv2 score across different NFEs. The default configuration is indicated by the dashed vertical line.

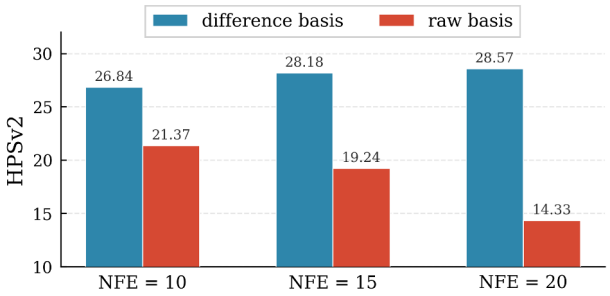


Figure 7: Ablation study on basis construction in Stable Diffusion 3.5-Medium. Our proposed different basis effectively addresses the issue of ill-conditioning.

an exponential-integrator formulation. Similarly, DPM-Solver [18] and DPM++ [19] derived dedicated high-order solvers by analytically isolating the linear components, significantly reducing discretization errors, and have emerged as a robust practical standard, particularly for guided sampling scenarios. UniPC [46] unified the predictor-corrector paradigm. Subsequent refinements such as DPM-Solver-v3 [48] and DC-Solver [47], further improved solver parameterization and correction stability. LTC-Accel [50] leverages the local transition coherence of generative trajectories to dynamically approximate adjacent network outputs, achieving training-free acceleration. STORK [34] addressed numerical stiffness and removed the reliance on semi-linear assumptions and proposed a stabilized solver. Despite their differences, these methods still construct update rules primarily from numerical analysis or step-wise trend heuristics. In contrast, our method revisits multi-step solving from a trajectory-geometric perspective and seeks to exploit the intrinsic low-dimensional and cross-trajectory consistent structure of generative trajectories more directly.

Diffusion sampling dynamics. Prior research on diffusion sampling dynamics has mainly established trajectory regularity through empirical observations, supported by preliminary theoretical insights under idealized settings. Notably, Wang et al. [37] observed that individual reverse-time sampling trajectories are effectively low-dimensional, often resembling two-dimensional rotations. Chen et al. [5] studied deterministic PF-ODE trajectories, finding that each trajectory closely follows the displacement between its endpoints, with residuals neatly captured by two or three orthogonal directions, yielding a characteristic boomerang-like shape. Their subsequent work [4] strengthened the cross-trajectory consistency claim by aligning projected trajectories with orthogonal Procrustes, revealing a nearly canonical trajectory shape after sample-specific rotations and reflections. While these works have empirically uncovered the trajectory regularity of diffusion sampling dynamics, their theoretical grounding relies largely on KDE-based surrogates of the empirical data distribution, analytically tractable Gaussian models, and supportive Gaussian-mixture calculations. To the best of our knowledge, prior literature lacks a unified theorem within a Mixture-of-Gaussian framework that jointly formalizes both trajectory low-dimensionality and cross-trajectory isomorphism. In contrast, our work provides a principled derivation that simultaneously establishes these two properties.

6 Conclusion

In this paper, we introduced LEAP, a novel, training-free multi-step ODE solver that effectively addresses the performance degradation of diffusion models in low-NFE regimes. Our approach is built upon a rigorous investigation of the geometric properties of diffusion sampling trajectories. By proving the intrinsic low-dimensionality and cross-trajectory consistency of generative paths, we formulated a supervised least-squares optimization within the local trajectory subspace. This allows LEAP to derive optimal, sample-agnostic extrapolation coefficients, achieving accurate future state approximations with minimal computational overhead. Extensive evaluations demonstrate that LEAP serves as a versatile, plug-and-play enhancement for modern large-scale diffusion models. LEAP consistently outperforms state-of-the-art baselines across diverse modalities, including text-to-image generation using Stable Diffusion 3.5 and FLUX.1-Dev and image-to-3D generation using TRELLIS.2. Future work will delve deeper into the geometric properties of diffusion sampling trajectories under practical settings to further align our theoretical framework with realistic, finite-step sampling scenarios.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62476244), the Starry Night Science Fund of Zhejiang University Shanghai Institute for Advanced Study, China (Grant No: SN-ZJU-SIAS-001) and the advanced computing resources provided by the Supercomputing Center of Hangzhou City University.

References

[1] Michael S Albergo and Eric Vanden-Eijnden. 2022. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571* (2022).
[2] Black Forest Labs. 2024. FLUX.1 [dev]. https://github.com/black-forest-labs/flux.
[3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. 2023. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127* (2023).
[4] Defang Chen, Zhenyu Zhou, Can Wang, and Siwei Lyu. 2025. Geometric regularity in deterministic sampling dynamics of diffusion-based generative models. *Journal of Statistical Mechanics: Theory and Experiment* 2025, 12 (2025), 124002.
[5] Defang Chen, Zhenyu Zhou, Can Wang, Chunhua Shen, and Siwei Lyu. 2024. On the Trajectory Regularity of ODE-based Diffusion Sampling. In *International Conference on Machine Learning*. 7905–7934.
[6] Nanxin Chen, Yu Zhang, Heiga Zen, Ron J Weiss, Mohammad Norouzi, and William Chan. 2020. Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713* (2020).
[7] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. 2022. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*. Ieee, 2553–2560.
[8] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*.
[9] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.
[10] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. 2022. Video diffusion models. *Advances in neural information processing systems* 35 (2022), 8633–8646.
[11] Team Hunyuan3D, Shuhui Yang, Mingxin Yang, Yifei Feng, Xin Huang, Sheng Zhang, Zebin He, Di Luo, Haolin Liu, Yunfei Zhao, et al. 2025. Hunyuan3d 2.1: From images to high-fidelity 3d assets with production-ready pbr material. *arXiv preprint arXiv:2506.15442* (2025).
[12] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. 2023. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. *arXiv preprint arXiv:2310.02279* (2023).
[13] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761* (2020).
[14] Senmao Li, Taihang Hu, Joost van de Weijer, Fahad S Khan, Tao Liu, Linxuan Li, Shiqi Yang, Yaxing Wang, Ming-Ming Cheng, and Jian Yang. 2024. Faster diffusion: Rethinking the role of the encoder for diffusion model inference. *Advances in Neural Information Processing Systems* 37 (2024), 85203–85240.
[15] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747* (2022).
[16] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. 2022. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778* (2022).
[17] Xingchao Liu, Chengyue Gong, and Qiang Liu. 2022. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003* (2022).
[18] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. 2022. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in neural information processing systems* 35 (2022), 5775–5787.
[19] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. 2025. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *Machine Intelligence Research* 22, 4 (2025), 730–751.
[20] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. 2023. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378* (2023).
[21] Xinyin Ma, Gongfan Fang, and Xinchao Wang. 2024. Deepcache: Accelerating diffusion models for free. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15762–15772.
[22] Bernt Oksendal. 2013. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media.
[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PmLR, 8748–8763.
[24] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* 1, 2 (2022), 3.
[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 10684–10695.
[26] Amirmojtaba Sabour, Sanja Fidler, and Karsten Kreis. 2024. Align your steps: Optimizing sampling schedules in diffusion models. *arXiv preprint arXiv:2404.14507* (2024).
[27] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems* 35 (2022), 36479–36494.
[28] Tim Salimans and Jonathan Ho. 2022. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512* (2022).
[29] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. pmlr, 2256–2265.
[30] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502* (2020).
[31] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. 2023. Consistency models. (2023).
[32] Yang Song and Stefano Ermon. 2020. Improved techniques for training score-based generative models. *Advances in neural information processing systems* 33 (2020), 12438–12448.
[33] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456* (2020).
[34] Zheng Tan, Weizhen Wang, Andrea L Bertozzi, and Ernest K Ryu. 2025. STORK: Faster Diffusion And Flow Matching Sampling By Resolving Both Stiffness And Structure-Dependence. *arXiv preprint arXiv:2505.24210* (2025).
[35] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. 2022. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers.
[36] Zhengchao Wan, Qingsong Wang, Gal Mishne, and Yusu Wang. 2025. Elucidating flow matching ODE dynamics via data geometry and denoisers. In *Forty-second International Conference on Machine Learning*.
[37] Binxu Wang and John J Vastola. 2023. Diffusion models generate images like painters: an analytical theory of outline first, details later. *arXiv preprint arXiv:2303.02490* (2023).
[38] Guangyi Wang, Wei Peng, Lijiang Li, Wenyu Chen, Yuren Cai, and Songzhi Su. 2024. Diffusion Sampling Correction via Approximately 10 Parameters. *arXiv preprint arXiv:2411.06503* (2024).
[39] Felix Wimbauer, Bichen Wu, Edgar Schoenfeld, Xiaoliang Dai, Ji Hou, Zijian He, Artsiom Sanakoyeu, Peizhao Zhang, Sam Tsai, Jonas Kohler, et al. 2024. Cache me if you can: Accelerating diffusion models through block caching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6211–6220.
[40] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. 2023. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341* (2023).
[41] Jianfeng Xiang, Xiaoxue Chen, Sicheng Xu, Ruicheng Wang, Zelong Lv, Yu Deng, Hongyuan Zhu, Yue Dong, Hao Zhao, Nicholas Jing Yuan, et al. 2025. Native and compact structured latents for 3d generation. *arXiv preprint arXiv:2512.14692* (2025).
[42] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. 2023. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems* 36 (2023), 15903–15935.
[43] Shuchen Xue, Zhaoqiang Liu, Fei Chen, Shifeng Zhang, Tianyang Hu, Enze Xie, and Zhenguo Li. 2024. Accelerating diffusion sampling with optimized time steps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8292–8301.
[44] Zichao Yu, Zhen Zou, Guojiang Shao, Chenwei Zhang, Shengze Xu, Jie Huang, Feng Zhao, Xiaodong Cun, and Wenyi Zhang. 2025. Ab-cache: Training-free acceleration of diffusion models via adams-bashforth cached feature reuse. In *Proceedings of the 33rd ACM International Conference on Multimedia*. 10408–10417.
[45] Qinsheng Zhang and Yongxin Chen. 2022. Fast sampling of diffusion models with exponential integrator. *arXiv preprint arXiv:2204.13902* (2022).

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

[47] Wenliang Zhao, Haolin Wang, Jie Zhou, and Jiwen Lu. 2024. Dc-solver: Improving predictor-corrector diffusion sampler via dynamic compensation. In *European Conference on Computer Vision*. Springer, 450–466.

[48] Kaiwen Zheng, Cheng Lu, Jianfei Chen, and Jun Zhu. 2023. Dpm-solver-v3: Improved diffusion ode solver with empirical model statistics. *Advances in Neural Information Processing Systems* 36 (2023), 55502–55542.

[49] Zhenyu Zhou, Defang Chen, Can Wang, and Chun Chen. 2024. Fast ode-based sampling for diffusion models in around 5 steps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7777–7786.

[50] Shangwen Zhu, Han Zhang, Zhantao Yang, Qianyu Peng, Zhao Pu, Huangji Wang, and Fan Cheng. 2025. Accelerating Diffusion Sampling via Exploiting Local Transition Coherence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 18284–18293.

A Proofs for Section 3.1

This appendix section proves Theorems and Corollaries from Section 3.1. First, we provide an exact decomposition of the PF-ODE under a mild assumption. Next, we prove the deterministic pairwise alignment bound. Finally, we apply the Gaussian concentration theorem to get a high-probability bound.

A.1 Notations

Consider the variance-preserving forward diffusion situation:

$$d\mathbf{X}_t = -\frac{1}{2}\beta(t)\mathbf{X}_t\,dt + \sqrt{\beta(t)}\,d\mathbf{W}_t, \qquad t \in [0,T], \tag{32}$$

where $\beta : [0,T] \to (0,\infty)$ is integrable. We define

$$\alpha_t := \exp\Big(-\frac{1}{2}\int_0^t \beta(s)\,ds\Big), \qquad \nu_t^2 := \alpha_t^2\sigma_0^2 + (1-\alpha_t^2). \tag{33}$$

Suppose data distribution satisfies GMM assumption:

$$p_0(\mathbf{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\boldsymbol{\mu}_m, \sigma_0^2\mathbf{I}_d), \qquad \pi_m > 0, \qquad \sum_{m=1}^{M}\pi_m = 1. \tag{34}$$

To describe the layout of the mixed distribution center, we define the space formed by their means as

$$U := \operatorname{span}\{\boldsymbol{\mu}_1,\dots,\boldsymbol{\mu}_M\}, \qquad r := \dim(U), \qquad R := \max_{1\le m\le M}\|\boldsymbol{\mu}_m\|. \tag{35}$$

We also let P to be the orthogonal projection onto U, with $Q := \mathbf{I} - P$.

A.2 Structural lemmas

We now begin to derive the closed-form expressions of PF-ODE dynamics. We start our proofs by formulating the marginal probability density of the forward diffusion.

Lemma A.1 (Closed Form of Marginal Probability Function). *For every $t \in [0,T]$, we have*

$$p_t(\mathbf{x}) = \sum_{s=1}^{S} \pi_s \mathcal{N}(\alpha_t\boldsymbol{\mu}_s, \nu_t^2\mathbf{I}_d). \tag{36}$$

Proof. Suppose $C \in \{1,\dots,S\}$ satisfy condition $\mathbb{P}(C=s) = \pi_s$, and suppose the conditional probability density of the initial state is

$$p(\mathbf{x}_0 \mid C=s) = \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_s, \sigma_0^2\mathbf{I}_d). \tag{37}$$

Because forward diffusion process defines the transition probability density:

$$p(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \alpha_t\mathbf{x}_0, (1-\alpha_t^2)\mathbf{I}_d). \tag{38}$$

Hence, conditional on $C=s$, the probability density of $\mathbf{x}_t$ is obtained by the following equation:

$$\begin{aligned} p(\mathbf{x}_t \mid C=s) &= \int p(\mathbf{x}_t \mid \mathbf{x}_0)p(\mathbf{x}_0 \mid C=s)\,\mathrm{d}\mathbf{x}_0 \\ &= \mathcal{N}\big(\mathbf{x}_t; \alpha_t\boldsymbol{\mu}_s, (\alpha_t^2\sigma_0^2 + 1 - \alpha_t^2)\mathbf{I}_d\big) \\ &= \mathcal{N}(\mathbf{x}_t; \alpha_t\boldsymbol{\mu}_s, \nu_t^2\mathbf{I}_d). \end{aligned} \tag{39}$$

Averaging over s gives the claim. □

Lemma A.2 (Closed-form of score function). *Define*

$$w_s(\mathbf{x},t) := \frac{\pi_s \exp\Big(\frac{\alpha_t}{\nu_t^2}\langle \mathbf{x}, \boldsymbol{\mu}_s\rangle - \frac{\alpha_t^2}{2\nu_t^2}\|\boldsymbol{\mu}_s\|^2\Big)}{\sum_{j=1}^{S}\pi_j \exp\Big(\frac{\alpha_t}{\nu_t^2}\langle \mathbf{x}, \boldsymbol{\mu}_j\rangle - \frac{\alpha_t^2}{2\nu_t^2}\|\boldsymbol{\mu}_j\|^2\Big)}, \tag{40}$$

and

$$\mathbf{m}_t(\mathbf{x}) := \sum_{s=1}^{S} w_s(\mathbf{x},t)\boldsymbol{\mu}_s. \tag{41}$$

Then

$$\nabla \log p_t(\mathbf{x}) = -\frac{1}{\nu_t^2}\big(\mathbf{x} - \alpha_t\mathbf{m}_t(\mathbf{x})\big). \tag{42}$$

Proof. By Lemma A.1,

$$p_t(\mathbf{x}) = \sum_{s=1}^{S} \pi_s \phi_{\nu_t^2\mathbf{I}_d}(\mathbf{x} - \alpha_t\boldsymbol{\mu}_s), \tag{43}$$

where ϕ_Σ denotes the Gaussian density with covariance Σ. Therefore,

$$\nabla p_t(\mathbf{x}) = -\frac{1}{\nu_t^2}\sum_{s=1}^{S}\pi_s(\mathbf{x} - \alpha_t\boldsymbol{\mu}_s)\phi_{\nu_t^2\mathbf{I}_d}(\mathbf{x} - \alpha_t\boldsymbol{\mu}_s). \tag{44}$$

Therefore, the closed-form of score function can be derived by dividing Eq. 44 by $p_t(\mathbf{x})$:

$$\nabla \log p_t(\mathbf{x}) = -\frac{1}{\nu_t^2}\Big(\mathbf{x} - \alpha_t\sum_{s=1}^{S} w_s(\mathbf{x},t)\boldsymbol{\mu}_s\Big) = -\frac{1}{\nu_t^2}\big(\mathbf{x} - \alpha_t\mathbf{m}_t(\mathbf{x})\big). \tag{45}$$

□

Lemma A.3 (Sufficiency of Mean Subspace). *For every $\mathbf{x} \in \mathbb{R}^d$ and $t \in [0,T]$,*

$$w_s(\mathbf{x},t) = w_s(P\mathbf{x},t), \qquad \mathbf{m}_t(\mathbf{x}) = \mathbf{m}_t(P\mathbf{x}) \in U. \tag{46}$$

Furthermore, let $\mathbf{x}_{t\in[0,T]}$ be a PF-ODE trajectory. Its derivative with respect to t is given by

$$\frac{d\mathbf{x}_t}{dt} = a_t\mathbf{x}_t + b_t\mathbf{m}_t(P\mathbf{x}_t), \tag{47}$$

where

$$a_t := \frac{1}{2}\beta(t)\Big(1 - \nu_t^{-2}\Big), \qquad b_t := \frac{1}{2}\beta(t)\alpha_t\nu_t^{-2}. \tag{48}$$

Proof. Note that the expression for $w_s(\mathbf{x},t)$ only depends on $\mathbf{x}$ through the inner products $\langle \mathbf{x}, \boldsymbol{\mu}_s\rangle$. Since for $\boldsymbol{\mu}_s \in U$, we have

$$\langle \mathbf{x}, \boldsymbol{\mu}_s\rangle = \langle P\mathbf{x}, \boldsymbol{\mu}_s\rangle, \tag{49}$$

which leads to $w_s(\mathbf{x},t) = w_s(P\mathbf{x},t)$. Furthermore, it is obvious that $\mathbf{m}_t(x) \in U$ because it is simply a sum of vectors $\mu_s \in U$:

$$\mathbf{m}_t(\mathbf{x}) = \sum_{s=1}^{S} w_s(\mathbf{x},t)\boldsymbol{\mu}_s. \tag{50}$$

In forward process, the PF-ODE is

$$\begin{aligned} \frac{d\mathbf{x}_t}{dt} &= -\frac{1}{2}\beta(t)\mathbf{x}_t - \frac{1}{2}\beta(t)\nabla\log p_t(\mathbf{x}_t) \\ &= \frac{1}{2}\beta(t)\Big[\mathbf{x}_t - \nu_t^{-2}\mathbf{x}_t + \alpha_t\nu_t^{-2}\mathbf{m}_t(\mathbf{x}_t)\Big]. \end{aligned} \tag{51}$$

Using Eq. 46 immediately yields Eq.47. □

Lemma A.4 (Exact $U \oplus U^\perp$ decomposition). *Let*

$$\Phi(t_2,t_1) := \exp\Big(\int_{t_1}^{t_2} a_v\,dv\Big), \qquad \rho_t := \Phi(t,T). \tag{52}$$

Then we obtain the following equation by integrating from T to t:

$$Q\mathbf{x}_t = \rho_t Q\mathbf{x}_T, \tag{53}$$

and

$$P\mathbf{x}_t = \rho_t P\mathbf{x}_T - \int_t^T \Phi(t,\tau) b_\tau \, \mathbf{m}_\tau(P\mathbf{x}_\tau)\, d\tau. \tag{54}$$

As a result,

$$Q\mathbf{D}_t = (\rho_t - 1) Q\mathbf{x}_T. \tag{55}$$

Proof. Apply Q to Eq. 47. Since $\mathbf{m}_t(P\mathbf{x}_t) \in U$, we obtain

$$Q\frac{d\mathbf{x}_t}{dt} = a_t Q\mathbf{x}_t, \tag{56}$$

whose exact solution is

$$Q\mathbf{x}_t = \exp\Big(\int_T^t a_v\, dv\Big) Q\mathbf{x}_T = \rho_t Q\mathbf{x}_T. \tag{57}$$

Similarly, for the projection P, we can obtain that

$$P\frac{d\mathbf{x}_t}{dt} = a_t P\mathbf{x}_t + b_t \mathbf{m}_t(P\mathbf{x}_t), \tag{58}$$

and the variation-of-constants formula (integrating from T to t) leads to

$$P\mathbf{x}_t = \rho_t P\mathbf{x}_T + \int_T^t \Phi(t,\tau) b_\tau \, \mathbf{m}_\tau(P\mathbf{x}_\tau)\, d\tau. \tag{59}$$

Subtracting $Q\mathbf{x}_T$ immediately gives the Eq. 55. □

Lemma A.5 (Mean-Subspace Displacement Bound). *Define*

$$A_T := \sup_{0\le t\le T} |\rho_t - 1|, \qquad B_T := \sup_{0\le t\le T} \int_t^T \Phi(t,\tau)|b_\tau|\, d\tau. \tag{60}$$

Then every trajectory satisfies

$$\sup_{0\le t\le T} \|P\mathbf{D}_t\| \le A_T \|P\mathbf{x}_T\| + B_T R. \tag{61}$$

Proof. By Lemma A.4,

$$P\mathbf{D}_t = (\rho_t - 1) P\mathbf{x}_T - \int_t^T \Phi(t,\tau) b_\tau \, \mathbf{m}_\tau(P\mathbf{x}_\tau)\, d\tau. \tag{62}$$

Since $\mathbf{m}_\tau(P\mathbf{x}_\tau)$ is a convex combination of $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$, it's obvious that

$$\|\mathbf{m}_\tau(P\mathbf{x}_\tau)\| \le R. \tag{63}$$

Hence by using the triangle inequality, we have

$$\|P\mathbf{D}_t\| \le |\rho_t - 1| \|P\mathbf{x}_T\| + \int_t^T \Phi(t,\tau)|b_\tau|\, R\, d\tau. \tag{64}$$

Taking the supremum over $t \in [0,T]$ proves the lemma. □

Lemma A.6 (Minimum orthogonal alignment error). *For any* $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$,

$$\min_{\mathbf{H}\in O(m)} \|\mathbf{u} - \mathbf{H}\mathbf{v}\| = \big|\|\mathbf{u}\| - \|\mathbf{v}\|\big|. \tag{65}$$

Proof. For any $\mathbf{H} \in O(m)$, the inequality

$$\|\mathbf{u} - \mathbf{H}\mathbf{v}\| \ge \big|\|\mathbf{u}\| - \|\mathbf{H}\mathbf{v}\|\big| = \big|\|\mathbf{u}\| - \|\mathbf{v}\|\big| \tag{66}$$

always holds. Conversely, if $\mathbf{u}, \mathbf{v} \ne 0$, choose $\mathbf{H} \in O(m)$ such that

$$\mathbf{H}\frac{\mathbf{v}}{\|\mathbf{v}\|} = \frac{\mathbf{u}}{\|\mathbf{u}\|}. \tag{67}$$

Then $\mathbf{H}\mathbf{v} = (\|\mathbf{v}\|/\|\mathbf{u}\|)\mathbf{u}$, so

$$\|\mathbf{u} - \mathbf{H}\mathbf{v}\| = \Big\|\mathbf{u} - \frac{\|\mathbf{v}\|}{\|\mathbf{u}\|}\mathbf{u}\Big\| = \big|\|\mathbf{u}\| - \|\mathbf{v}\|\big|. \tag{68}$$

Finally, if $\mathbf{u} = 0$ or $\mathbf{v} = 0$, the result holds trivially. □

A.3 A uniform high-probability bound

Building upon the deterministic trajectory decomposition provided in the previous subsection, this part gives a uniform high-probability bound for pairwise trajectory alignment proof.

Proposition A.7 (Deterministic pairwise alignment). *Suppose* $\mathbf{x}^{(1)}_{t\in[0,T]}$ *and* $\mathbf{x}^{(2)}_{t\in[0,T]}$ *are two exact PF-ODE trajectories with starting noise* $\mathbf{x}^{(1)}_T$ *and* $\mathbf{x}^{(2)}_T$. *Then there exists* $\mathbf{H} \in O(d)$ *such that*

$$\sup_{0\le t\le T} \|\mathbf{D}^{(1)}_t - \mathbf{H}\mathbf{D}^{(2)}_t\| \le A_T \big|\|Q\mathbf{x}^{(1)}_T\| - \|Q\mathbf{x}^{(2)}_T\|\big| + A_T\big(\|P\mathbf{x}^{(1)}_T\| + \|P\mathbf{x}^{(2)}_T\|\big) + 2B_T R. \tag{69}$$

Consequently,

$$\max_{0\le n\le N} \|\mathbf{D}^{(1)}_n - \mathbf{H}\mathbf{D}^{(2)}_n\| \le A_T \big|\|Q\mathbf{x}^{(1)}_T\| - \|Q\mathbf{x}^{(2)}_T\|\big| + A_T\big(\|P\mathbf{x}^{(1)}_T\| + \|P\mathbf{x}^{(2)}_T\|\big) + 2B_T R. \tag{70}$$

Proof. Decompose $\mathbb{R}^d = U \oplus U^\perp$. By Lemma A.6, there exists $\mathbf{H}_Q \in O(U^\perp)$ such that

$$\|Q\mathbf{x}^{(1)}_T - \mathbf{H}_Q Q\mathbf{x}^{(2)}_T\| = \big|\|Q\mathbf{x}^{(1)}_T\| - \|Q\mathbf{x}^{(2)}_T\|\big|. \tag{71}$$

Choose any $\mathbf{H}_P \in O(U)$ and define

$$\mathbf{H} := \mathbf{H}_P \oplus \mathbf{H}_Q \in O(d). \tag{72}$$

By Lemma A.4,

$$Q\mathbf{D}^{(k)}_t = (\rho_t - 1) Q\mathbf{x}^{(k)}_T. \tag{73}$$

Hence

$$Q\big(\mathbf{D}^{(1)}_t - \mathbf{H}\mathbf{D}^{(2)}_t\big) = (\rho_t - 1)\big(Q\mathbf{x}^{(1)}_T - \mathbf{H}_Q Q\mathbf{x}^{(2)}_T\big), \tag{74}$$

so

$$\sup_{0\le t\le T} \|Q(\mathbf{D}^{(1)}_t - \mathbf{H}\mathbf{D}^{(2)}_t)\| \le A_T \big|\|Q\mathbf{x}^{(1)}_T\| - \|Q\mathbf{x}^{(2)}_T\|\big|. \tag{75}$$

For the U-component, Lemma A.5 gives

$$\|P(\mathbf{D}^{(1)}_t - \mathbf{H}\mathbf{D}^{(2)}_t)\| \le \|P\mathbf{D}^{(1)}_t\| + \|P\mathbf{D}^{(2)}_t\| \le A_T\|P\mathbf{x}^{(1)}_T\| + B_T R + A_T\|P\mathbf{x}^{(2)}_T\| + B_T R. \tag{76}$$

Combining the U and $U^\perp$ parts via the triangle inequality proves the continuous-time bound. The discrete-time bound follows from

$$\max_{0\le n\le N} \|\mathbf{D}^{(1)}_n - \mathbf{H}\mathbf{D}^{(2)}_n\| \le \sup_{0\le t\le T} \|\mathbf{D}^{(1)}_t - \mathbf{H}\mathbf{D}^{(2)}_t\|. \tag{77}$$

□

Lemma A.8 (Gaussian norm concentration). *If* $\mathbf{g} \sim \mathcal{N}(0, \mathbf{I}_m)$, *then for every* $t > 0$,

$$\mathbb{P}\big(|\|\mathbf{g}\| - \mathbb{E}\|\mathbf{g}\|| \ge t\big) \le 2e^{-t^2/2}, \qquad \mathbb{E}\|\mathbf{g}\| \le \sqrt{m}. \tag{78}$$

Proof. The map $\mathbf{g} \mapsto \|\mathbf{g}\|$ is 1-Lipschitz, so the first inequality is the standard Gaussian concentration inequality. The second follows from Jensen's inequality:

$$\mathbb{E}\|\mathbf{g}\| \le \sqrt{\mathbb{E}\|\mathbf{g}\|^2} = \sqrt{m}. \tag{79}$$

□

Theorem A.9 (Explicit version of Theorem 3.1). *Suppose that $\mathbf{x}_T^{(1)}, \dots, \mathbf{x}_T^{(M)}$ are i.i.d. samples from the exact noised marginal p_T. Then for every $\delta \in (0,1)$, with probability at least $1-\delta$,*

$$\max_{1\le a,b\le M} \inf_{\mathbf{H}\in O(d)} \max_{0\le n\le N} \|\mathbf{D}_n^{(a)} - \mathbf{H}\mathbf{D}_n^{(b)}\| \le 2\sqrt{2}\, A_T \nu_T \sqrt{\log\frac{4M}{\delta}} + 2A_T\Big(\alpha_T R + \nu_T\Big[\sqrt{r} + \sqrt{2\log\frac{2M}{\delta}}\Big]\Big) + 2B_T R. \tag{80}$$

Proof. By Lemma A.1, the noised marginal at time T is

$$p_T(\mathbf{x}) = \sum_{m=1}^{M} \pi_m \mathcal{N}(\alpha_T \boldsymbol{\mu}_m, \nu_T^2 \mathbf{I}_d). \tag{81}$$

Hence each sample can be represented as

$$\mathbf{x}_T^{(k)} = \alpha_T \boldsymbol{\mu}_{C_k} + \nu_T \mathbf{g}^{(k)}, \tag{82}$$

where $C_k \in \{1,\dots,M\}$, $\mathbb{P}(C_k = m) = \pi_m$, and $\mathbf{g}^{(k)} \sim \mathcal{N}(0, \mathbf{I}_d)$. Since $\boldsymbol{\mu}_{C_k} \in U$,

$$Q\mathbf{x}_T^{(k)} = \nu_T Q\mathbf{g}^{(k)} \sim \mathcal{N}(0, \nu_T^2 \mathbf{I}_{d-r}), \tag{83}$$

and

$$P\mathbf{x}_T^{(k)} = \alpha_T \boldsymbol{\mu}_{C_k} + \nu_T P\mathbf{g}^{(k)}. \tag{84}$$

We first control the Q-component. By Lemma A.8 and a union bound, with probability at least $1-\delta/2$,

$$\max_{1\le k\le M} \Big| \|Q\mathbf{x}_T^{(k)}\| - \mathbb{E}\|Q\mathbf{x}_T^{(k)}\| \Big| \le \nu_T \sqrt{2\log\frac{4M}{\delta}}. \tag{85}$$

Therefore, on this event,

$$\max_{1\le a,b\le M} \Big| \|Q\mathbf{x}_T^{(a)}\| - \|Q\mathbf{x}_T^{(b)}\| \Big| \le 2\nu_T \sqrt{2\log\frac{4M}{\delta}}. \tag{86}$$

Next we control the P-component. Since $P\mathbf{g}^{(k)} \sim \mathcal{N}(0, \mathbf{I}_r)$ on U, Lemma A.8 and $\|\boldsymbol{\mu}_{C_k}\| \le R$ imply that, with probability at least $1-\delta/2$,

$$\max_{1\le k\le M} \|P\mathbf{x}_T^{(k)}\| \le \alpha_T R + \nu_T\Big(\sqrt{r} + \sqrt{2\log\frac{2M}{\delta}}\Big). \tag{87}$$

On the intersection of these two events, whose probability is at least $1-\delta$, Proposition A.7 gives, for every pair (a,b), some orthogonal matrix $\mathbf{H}_{a,b} \in O(d)$ such that

$$\max_{0\le n\le N} \|\mathbf{D}_n^{(a)} - \mathbf{H}_{a,b}\mathbf{D}_n^{(b)}\| \le A_T \cdot 2\nu_T \sqrt{2\log\frac{4M}{\delta}} + A_T \cdot 2\Big(\alpha_T R + \nu_T\Big[\sqrt{r} + \sqrt{2\log\frac{2M}{\delta}}\Big]\Big) + 2B_T R. \tag{88}$$

Taking the maximum over (a,b) and then the infimum over $\mathbf{H} \in O(d)$ yields the stated inequality. □

Proof of Corollary 3.2. Divide the bound in Theorem A.9 by $\nu_T\sqrt{d}$. This gives

$$\frac{1}{\nu_T\sqrt{d}} \max_{1\le a,b\le n} \inf_{\mathbf{H}\in O(d)} \max_{0\le \ell\le L} \|\mathbf{D}_\ell^{(a)} - \mathbf{H}\mathbf{D}_\ell^{(b)}\| \le 2\sqrt{2}\, A_T \sqrt{\frac{\log(4M/\delta)}{d}} + 2A_T\sqrt{\frac{r}{d}} + 2A_T\sqrt{\frac{2\log(2M/\delta)}{d}} + 2A_T\frac{\alpha_T R}{\nu_T\sqrt{d}} + 2B_T\frac{R}{\nu_T\sqrt{d}}. \tag{89}$$

Since $r + \log n = o(d)$ and $R = o(\nu_T\sqrt{d})$, the right-hand side converges to 0 for every fixed $\delta \in (0,1)$. This proves the claim. □

A.4 Proofs of the low-dimensionality Corollary

We now show that under mild assumptions, each PF-ODE trajectory is confined to a low-dimensional subspace. Moreover, a simple alignment brings all trajectories into a single shared subspace.

Corollary A.10 (Explicit version of Corollary 3.3). *Suppose $\mathbf{x}_{t\in[0,T]}^{(k)}$ is an exact PF-ODE trajectory. Then*

$$\mathbf{x}_t^{(k)} \in \mathcal{S}_k := U \oplus \operatorname{span}\{Q\mathbf{x}_T^{(k)}\}, \qquad \forall t \in [0,T]. \tag{90}$$

In particular,

$$\dim(\mathcal{S}_k) \le r+1. \tag{91}$$

Hence, on any discrete reverse-time grid $T = t_0 > t_1 > \cdots > t_L = 0$, the trajectory matrix

$$\mathbf{X}^{(k)} \in \mathbb{R}^{(L+1)\times d}, \qquad row_\ell(\mathbf{X}^{(k)}) = (\mathbf{x}_\ell^{(k)})^\top, \tag{92}$$

satisfies

$$\operatorname{rank}(\mathbf{X}^{(k)}) \le r+1. \tag{93}$$

Equivalently, the empirical trajectory covariance

$$\Sigma_{\text{traj}}^{(k)} := \frac{1}{M+1}\sum_{m=0}^{M}\big(\mathbf{x}_\ell^{(k)} - \bar{\mathbf{x}}^{(k)}\big)\big(\mathbf{x}_\ell^{(k)} - \bar{\mathbf{x}}^{(k)}\big)^\top \tag{94}$$

has rank at most $r+1$, and therefore

$$\lambda_j\Big(\Sigma_{\text{traj}}^{(k)}\Big) = 0, \qquad j \ge r+2. \tag{95}$$

Proof. By Lemma A.4,

$$Q\mathbf{x}_t^{(k)} = \rho_t Q\mathbf{x}_T^{(k)}, \tag{96}$$

which lies in the one-dimensional subspace $\operatorname{span}\{Q\mathbf{x}_T^{(k)}\}$. On the other hand,

$$P\mathbf{x}_t^{(k)} \in U \tag{97}$$

for all $t \in [0,T]$. Therefore,

$$\mathbf{x}_t^{(k)} = P\mathbf{x}_t^{(k)} + Q\mathbf{x}_t^{(k)} \in U \oplus \operatorname{span}\{Q\mathbf{x}_T^{(k)}\}. \tag{98}$$

This proves that the whole trajectory lies in the linear subspace $\mathcal{S}_k$ with $\dim(\mathcal{S}_k) \le r+1$. Since every row of $\mathbf{X}^{(k)}$ belongs to $\mathcal{S}_k$, we have

$$\operatorname{rank}(\mathbf{X}^{(k)}) \le \dim(\mathcal{S}_k) \le r+1. \tag{99}$$

Since $\mathcal{S}_k$ is a linear subspace, it contains the centered trajectory vectors. Thus, the image of the covariance matrix $\Sigma_{\text{traj}}^{(k)}$ lies within $\mathcal{S}_k$, implying

$$\operatorname{rank}\Big(\Sigma_{\text{traj}}^{(k)}\Big) \le r+1. \tag{100}$$

This directly leads to the conclusion about the eigenvalues. □

Corollary A.11 (Explicit version of Corollary 3.4). *Fix any unit vector* $\mathbf{e} \in U^{\perp}$. *For each trajectory* k, *choose an orthogonal matrix* $\mathbf{H}_k \in O(d)$ *such that*

$$\mathbf{H}_k|_U = \mathbf{I}, \qquad \mathbf{H}_k Q\mathbf{x}_T^{(k)} = \|Q\mathbf{x}_T^{(k)}\|\,\mathbf{e}. \tag{101}$$

Define the aligned trajectory

$$\tilde{\mathbf{x}}_t^{(k)} := \mathbf{H}_k \mathbf{x}_t^{(k)}. \tag{102}$$

Then

$$\tilde{\mathbf{x}}_t^{(k)} \in \widetilde{\mathcal{S}} := U \oplus \mathrm{span}\{\mathbf{e}\}, \qquad \forall k,\ \forall t \in [0,T]. \tag{103}$$

In particular,

$$\dim(\widetilde{\mathcal{S}}) \le r+1. \tag{104}$$

Hence, if $\widetilde{\mathbf{X}} \in \mathbb{R}^{n(L+1)\times d}$ *denotes the pooled aligned trajectory matrix obtained by stacking all aligned trajectory states, then*

$$\mathrm{rank}(\widetilde{\mathbf{X}}) \le r+1. \tag{105}$$

Equivalently, the pooled covariance of the aligned trajectories has rank at most $r+1$.

Proof. By construction, $\mathbf{H}_k$ acts as the identity on U. Therefore, since $P\mathbf{x}_t^{(k)} \in U$,

$$\mathbf{H}_k P\mathbf{x}_t^{(k)} = P\mathbf{x}_t^{(k)} \in U. \tag{106}$$

Moreover, using Lemma A.4,

$$Q\mathbf{x}_t^{(k)} = \rho_t Q\mathbf{x}_T^{(k)}, \tag{107}$$

and thus

$$\mathbf{H}_k Q\mathbf{x}_t^{(k)} = \rho_t \mathbf{H}_k Q\mathbf{x}_T^{(k)} = \rho_t \|Q\mathbf{x}_T^{(k)}\|\,\mathbf{e} \in \mathrm{span}\{\mathbf{e}\}. \tag{108}$$

Combining the U-part and the $U^{\perp}$-part gives

$$\tilde{\mathbf{x}}_t^{(k)} = \mathbf{H}_k \mathbf{x}_t^{(k)} \in U \oplus \mathrm{span}\{\mathbf{e}\} = \widetilde{\mathcal{S}}. \tag{109}$$

Therefore all aligned trajectories share a single subspace $\widetilde{\mathcal{S}}$ of dimension at most $r+1$, which immediately yields the rank results. □

Remark 1 (Connection to least-squares reconstruction). The above corollaries provide a direct explanation for the least-squares residual observed in practice. Specifically, every state $\mathbf{x}_n^{(k)}$ on a given trajectory k belongs to the same linear subspace $\mathcal{S}_k$ with

$$\dim(\mathcal{S}_k) \le r+1. \tag{110}$$

Therefore, if a history buffer

$$\{\mathbf{x}_{n-B}^{(k)}, \mathbf{x}_{n-B+1}^{(k)}, \ldots, \mathbf{x}_{n-1}^{(k)}\} \tag{111}$$

spans $\mathcal{S}_k$, then the least-squares reconstruction error of $\mathbf{x}_n^{(k)}$ from this buffer is exactly zero:

$$\min_{\mathbf{c}\in\mathbb{R}^B} \left\| \mathbf{x}_n^{(k)} - \sum_{j=1}^{B} c_j \mathbf{x}_{n-j}^{(k)} \right\| = 0. \tag{112}$$

Thus, in the exact spherical GMM setting, a sufficiently rich short buffer already determines the future point by linear regression. In realistic diffusion models we do not expect this identity to hold exactly, but small least-squares residuals strongly suggest that the denoising dynamics remain approximately low-dimensional.

Table 3: LEAP combined with different base solvers in SD3.5-Medium. All results use the same 15-step schedule and $r = 5$ in LEAP's setting.

Solver + Method	HPSv2↑	ImageReward↑	CLIPScore↑
Euler	27.4725	0.8416	33.4644
LEAP (Euler)	**28.1826**	**0.9522**	**33.6817**
DPM++	27.8642	0.8438	33.4704
LEAP (DPM++)	**28.2152**	**0.9806**	**33.5617**
iPNDM	27.9220	0.8770	33.4416
LEAP (iPNDM)	**28.2204**	**0.9642**	**33.6601**
UniPC	27.8680	0.8788	33.4703
LEAP (UniPC)	**28.2309**	**0.9800**	**33.6077**

Table 4: Ablation on the prediction-to-correction (P:C) schedule on SD3.5-Medium.

NFE	P:C	HPSv2↑	ImageReward↑	CLIPScore↑
10	1:1	**26.8419**	**0.7692**	**33.2982**
	1:2	26.3182	0.7224	33.1515
15	1:1	**28.1826**	**0.9522**	33.6817
	1:2	27.9594	0.9004	**33.7109**
20	1:1	**28.5727**	**0.9701**	33.7192
	1:2	28.3575	0.9499	**33.8125**

B Experimental Results

B.1 Ablation Study

B.1.1 Compatibility with Different Base Solvers. To further evaluate the robustness and generalization ability of LEAP, we conduct an additional ablation experiment by integrating LEAP with various widely used ODE solvers, including Euler, DPM++, iPNDM, and UniPC. This experiment is conducted on SD3.5 with NFE=15 and $r = 5$. As shown in Table 3, LEAP improves human preference and text-image alignment metrics (HPSv2, ImageReward, and CLIPScore) across all tested baseline solvers.

B.1.2 Predict-correct schedule. We further test the ratio between prediction and correction steps. Consecutive linear predictions are theoretically degradate to our current implementation as the same linear subspace is used. We thus add results regarding 2-step consecutive corrections below, which are inferior to our default 1:1 setting, as shown in Table 4.

B.2 Robustness, Transferability, and Generalization

B.2.1 Prompt-distribution robustness. The coefficient bank is constructed using reference prompts sampled from DrawBench, while evaluation is conducted on four prompt categories from the HPSv2 benchmark. As shown in Table 5, LEAP improves Euler across every category for HPSv2, ImageReward, and CLIPScore. These results indicate that LEAP remains effective under a prompt distribution shift.

Table 5: Ablation study on prompt distributions (NFE= 15). Reference prompts are sampled from DrawBench, while evaluation prompts are sampled from the HPSv2 benchmark.

Metric	Method	Anime	ConceptArt	Paintings	Photo
HPSv2	Euler	30.14	28.63	28.99	26.56
	LEAP (Euler)	**31.22**	**30.15**	**30.14**	**27.37**
ImageReward	Euler	1.10	1.02	1.06	0.80
	LEAP (Euler)	**1.18**	**1.15**	**1.16**	**0.96**
CLIPScore	Euler	35.18	33.61	33.41	32.28
	LEAP (Euler)	**35.19**	**33.96**	**33.69**	**32.79**

B.2.2 Coefficient transferability. We evaluate whether learned coefficients can be reused beyond the solver, model, or sampling schedule from which it was constructed. Specifically, we study cross-model, cross-solver, and cross-step transfer. As reported in Table 6, every transferred configuration improves its corresponding baseline on all three metrics.

Table 6: Transferability of the coefficient bank. Cross-model and cross-solver results use 10 NFE; cross-step results use 15 NFE.

Transfer setting	Method	HPSv2↑	ImageReward↑	CLIPScore↑
Cross-model	Euler	25.89	0.67	32.79
	LEAP (FLUX coeffs.)	**26.97**	**0.75**	**33.26**
Cross-solver	DPM++	26.46	0.72	32.90
	LEAP (Euler coeffs.)	**27.01**	**0.80**	**33.48**
Cross-step	Euler	27.47	0.84	33.46
	LEAP (10-NFE coeffs.)	**28.17**	**0.91**	**33.69**

B.3 Additional Qalitative Results

We first provide additional qualitative results for both text-image task and image-3D task under the same setting in Sec.4.1. For text-image task, the qualitative results are shown in Fig. 9. For image-3D task, the qualitative results are shown in Fig. 10.

B.4 Experimental Details

B.4.1 Hyperparameters. In this subsection, we discuss our experiment setup in greater detail. In Sec. 4, we evaluate the proposed LEAP on three diffusion models: SD3.5-Medium [8] and FLUX.1-Dev [2] for text-image task and TRELLIS.2 [41] for image-3D task.

We test LEAP against previous solvers in text-to-image tasks, including DPM++ [19], STORK [34], and LTC-Accel [50]. In our evaluation, DPM++ is configured to its second-order version, STORK utilizes $k = 4$, $n = 1$, and $s = 9$, while LTC-Accel adopts $r = 2$ and $\tau = 0.1$ for both quantitative and qualitative assessments. We use a fixed window size of $r = 5$ and set the ridge regularization coefficient to $\lambda = 0.0001$ as the default experiments settings of LEAP.

In the context of image-to-3D tasks, we benchmark our approach against the Euler baseline. For these experiments, we maintain the exact same configuration as mentioned in the text-to-image tasks described above.

B.4.2 Prompts. In the text-to-image qualitative experiments, we adopt the following text prompts to generate images:

- "*A close-up of a futuristic barista robot pouring latte art in the shape of a perfect hexagon. The robot's metallic arm has the word 'SPEED' etched into it. High contrast, cinematic lighting, photorealistic.*"
- "*A transparent glass teapot filled with blooming jasmine flowers. Steam is rising, and sunlight is refracting through the water and glass, casting complex caustics onto a white tablecloth.*",
- "*A swarm of tiny, glowing robotic drones weaving an intricate, glowinglaser web between two giant, ancient redwood trees in a foggy forest. The web perfectly forms the exact shape of a rigid, symmetrical pentagram. Volumetric fog.* "

In the image-to-3D qualitative experiments, we use Nano Banana2 to generate reference images, as shown in Fig. 8.



Figure 8: Reference images adopted in image-3D qualitative experiments.

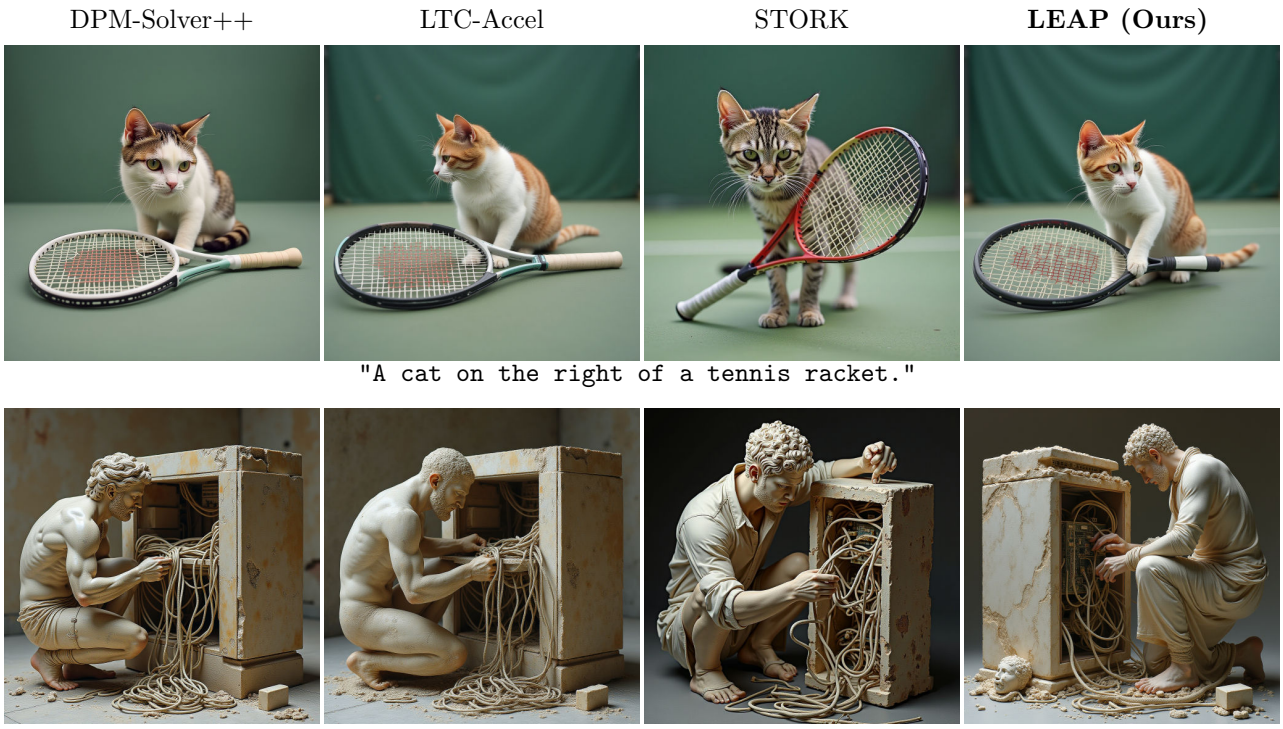


Figure 9: Additional qualitative comparison on the FLUX model (Part 1).

DPM-Solver++ LTC-Accel STORK4 **LEAP (Ours)**



"A red colored car."



"A sphere made of kitchen tile. A sphere with the texture of kitchen tile."



"A small vessel propelled on water by oars, sails, or an engine."



"A mechanical or electrical device for measuring time."

Figure 9: Additional qualitative comparison on the FLUX model (Continued).

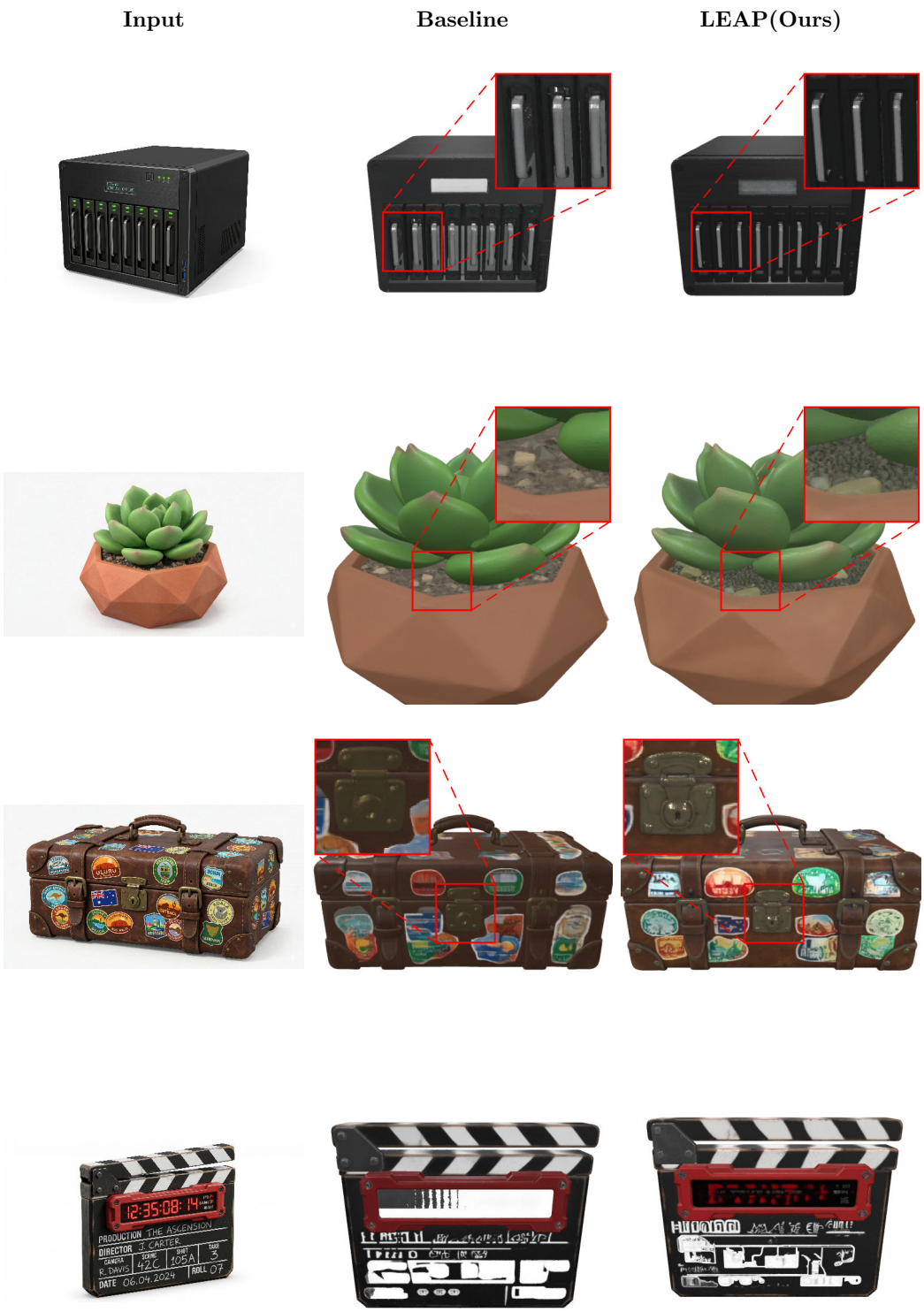


Figure 10: Additional image-3D qualitative results. Columns from left to right show the input conditions, generation results from the baseline solver (NFE=10), and results from our proposed LEAP method (NFE=10).