WHAT SHOULD FEW-STEP DIFFUSION SCHEDULES OPTIMIZE? FEATURE-ALIGNED TIMESTEP REFINEMENT BEYOND NUMERICAL ERROR

Anonymous authors
Paper under double-blind review

ABSTRACT

Few-step diffusion sampling is highly sensitive to timestep placement, yet existing scheduling objectives are typically derived from numerical or trajectory-level accuracy, while generation quality is evaluated in a nonlinear feature space. We show that this mismatch produces markedly different optimization geometry in the extreme few-step regime: schedules with similar state-space error can have very different Fréchet Inception Distance (FID), whereas the feature-mean discrepancy to a high-accuracy reference closely follows the local FID landscape. We therefore use this reference feature-mean discrepancy, the mean component of the FID to the reference, as a local schedule-refinement objective. Rather than assuming global equivalence to data FID, we characterize when its gradient follows the data-mean component and when decreasing that component also decreases FID. We further show that the naive finite-sample plug-in estimator reintroduces a paired feature-dispersion term and derive an unbiased U-statistic estimator. The resulting method requires neither model retraining nor real-data statistics during schedule search. Across eight pixel-space settings spanning three datasets, four solvers, and 4–16 function evaluations (NFE), our schedules lie 0.47–4.13% above best-found direct-FID solutions (median 1.78%), versus median gaps of 16.48% for paired feature error and 59.8% for state-space error. The same objective also transfers to LD3's richer two-sequence parameterization, improving its released schedule from FID 9.632 to 8.093 at 4 NFE and from 2.369 to 2.201 at 10 NFE.

1 INTRODUCTION

Diffusion models have become a leading paradigm for high-quality image generation (Ho et al., 2020; Song et al., 2021; Dhariwal & Nichol, 2021), but their strong generative performance often comes at substantial inference cost, as sampling requires repeated evaluations of a large neural network along a discretized denoising trajectory (Ho et al., 2020; Song et al., 2021). Existing approaches to reducing this cost broadly fall into two categories. Training-based approaches modify or retrain the generative process for fast sampling, including progressive distillation (Salimans & Ho, 2022), consistency-based generation (Song et al., 2023), and flow-based formulations such as Rectified Flow (Liu et al., 2023) and MeanFlow (Geng et al., 2025). Training-free approaches instead retain the pretrained model and improve the sampling procedure itself, for example through higher-order numerical solvers (Lu et al., 2022; 2025; Karras et al., 2022). Among them, timestep schedule optimization is particularly attractive in the few-step regime, where the placement of only a handful of model evaluations can substantially affect generation quality without modifying the underlying model (Xue et al., 2024; Sabour et al., 2024; Tong et al., 2025; Zhu et al., 2026).

A fundamental question, however, remains: *what should a few-step sampling schedule actually optimize?* Existing methods derive their scheduling criteria from different aspects of the sampling process, including numerical discretization error, stochastic-process-derived criteria, and learned or optimized surrogate objectives (Xue et al., 2024; Sabour et al., 2024; Tong et al., 2025; Zhu et al., 2026). Although these approaches differ in formulation, their objectives are primarily constructed from the sampling dynamics rather than from the geometry of the downstream evaluation metric. This distinction matters because generation quality is typically assessed after a nonlinear feature transformation, such as the Inception representation used by Fréchet Inception Distance

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

(FID) (Heusel et al., 2017). A schedule that more faithfully approximates the sampling trajectory may therefore not be the one that optimizes generation quality.

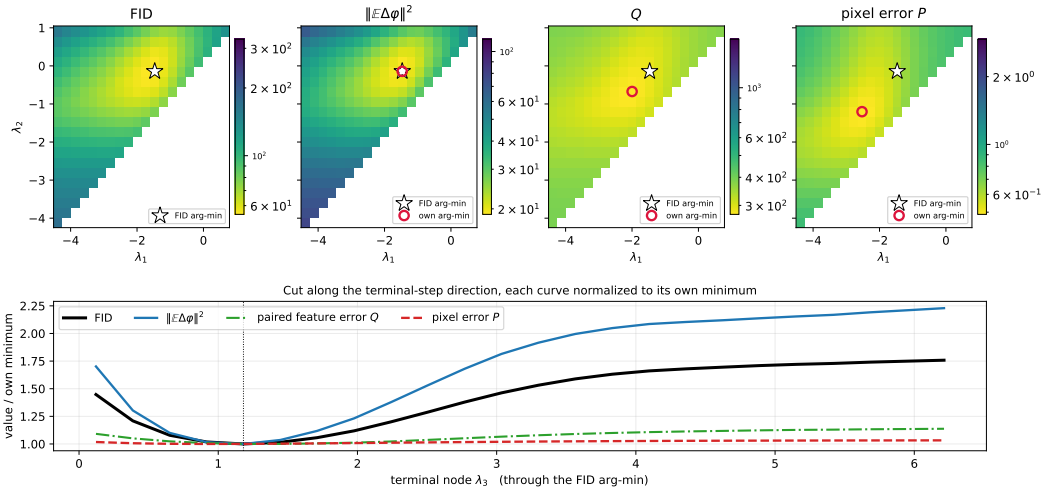

Figure 1: **Numerical accuracy does not necessarily imply metric-aligned accuracy.** CIFAR-10, four-step Euler. Top: the four objectives over λ_1 and λ_2, with the terminal node λ_3 fixed at its FID-optimal value; the star marks the FID arg-min and the circle each objective's own. On this plane the arg-min of $\|\mathbb{E}\Delta\varphi\|^2$ coincides with the FID optimum, while those of the paired feature error Q and the pixel error P are 6.4% and 30.6% above it (FID from 2000 samples). Bottom: along the terminal node, pixel endpoint error largely misses the sensitivity exhibited by FID, whereas $\|\mathbb{E}\Delta\varphi\|^2$ closely follows it. Optimizing this surrogate from the standard schedule reaches within 0.7% of direct FID optimization, while pixel and paired-feature objectives remain substantially worse.

We find that this mismatch becomes especially consequential in the extreme few-step regime. As illustrated in Fig. 1, under four-step sampling, FID is highly sensitive to the placement of the terminal timestep, whereas pixel endpoint error is nearly insensitive along the same direction. In contrast, the discrepancy between feature means closely reproduces the local anisotropy of the FID landscape. These observations suggest that the key is not merely to approximate the sampling trajectory accurately, but to identify which component of the feature-space discrepancy preserves the optimization geometry relevant to generation quality.

To this end, we compare a candidate sampler S with a high-accuracy reference sampler R using the same latent z, and define $\Delta\varphi(z) = \varphi(S(z)) - \varphi(R(z))$, with φ the Inception feature map. The paired feature error decomposes as

$$\mathbb{E}_z\left[\|\Delta\varphi(z)\|_2^2\right] = \|\mathbb{E}_z[\Delta\varphi(z)]\|_2^2 + \mathrm{tr}(\mathrm{Cov}(\Delta\varphi))\,. \tag{1}$$

The first term is also exactly the mean component of the FID between the candidate and reference samplers (Heusel et al., 2017). This motivates our feature-mean objective

$$\mathcal{L}_{\mathrm{FM}} = \|\mathbb{E}_z[\Delta\varphi(z)]\|_2^2 = \|\mathbb{E}_z[\varphi(S(z))] - \mathbb{E}_z[\varphi(R(z))]\|_2^2\,. \tag{2}$$

Rather than assuming that $\mathcal{L}_{\mathrm{FM}}$ is globally equivalent to FID, we establish a local justification: decreasing the mean component of FID is guaranteed to decrease FID whenever its gradient is sufficiently aligned with that of the covariance component. Empirically, around standard sampling schedules, the reference-based mean term in Eq. 2 closely follows this descent direction. This makes $\mathcal{L}_{\mathrm{FM}}$ a practical local surrogate for schedule refinement, while avoiding the mismatched geometry of pixel error and the additional paired-dispersion term in Eq. 1.

Few-step samplers are deployed with standard schedules, so we cast schedule optimization as refining them, exactly the regime covered by the local analysis. We initialize from a standard sampling schedule and optimize the timestep locations using $\mathcal{L}_{\mathrm{FM}}$, while keeping the pretrained model and solver fixed. Across three datasets, four solvers, and sampling budgets from four to sixteen evaluations, the resulting schedules lie 0.47–4.13% above the best-found solutions obtained by direct

FID optimization, with a median gap of 1.78%. In contrast, optimizing paired feature error or pixel-space error leaves gaps of 9.60–41.6% and 6.71–108%, respectively. The objective also transfers to richer schedule parameterizations: when plugged into the decoupled two-sequence search space of LD3 (Tong et al., 2025), it improves the released LD3 schedule from FID 9.632 to 8.093 at four evaluations and from 2.369 to 2.201 at ten evaluations under the same evaluation pipeline.

Our contributions are summarized as follows:

- We identify a *geometry mismatch* in few-step scheduling: state-space accuracy can be poorly aligned with the FID landscape, especially along the terminal timestep direction.
- We show that seemingly more complete feature-space discrepancies make schedule optimization worse: the paired feature error is dominated by a paired-dispersion term that masks the variation of the feature mean, and the naive finite-sample estimate of the feature-mean term reintroduces this term with weight $1/K$; an unbiased U-statistic removes it.
- We show when the model's own high-accuracy sampler can replace real-data statistics: the reference-mean and data-mean gradients differ only by a projected offset that stays small around practical schedules, and decreasing the data-mean term decreases FID under a descent condition that holds at 99.5% of our diagnostic grid.
- Across datasets, solvers, and sampling budgets, the resulting training-free objective acts as an effective local refinement of standard schedules: it recovers schedules close to best-found direct-FID solutions and improves LD3's released two-sequence schedules.

2 Related Work

Fast diffusion sampling. A major line of work accelerates diffusion sampling by improving the numerical integration of the reverse-time dynamics. DPM-Solver and DPM-Solver++ exploit the semi-linear structure of diffusion ODEs to obtain accurate samples with substantially fewer function evaluations (Lu et al., 2022; 2025), while the Elucidated Diffusion Model (EDM) framework of Karras et al. (2022) jointly studies model parameterization, numerical solvers, and noise-level discretization. These methods primarily improve how a trajectory is integrated under a fixed sampling budget. Our work is complementary: we keep the pretrained model and numerical solver fixed, and study how the available evaluations should be placed along the trajectory.

Timestep schedule optimization. Orthogonal to solver design, timestep scheduling determines where a fixed number of model evaluations are allocated. Xue et al. (2024) optimize non-uniform timesteps by minimizing the discrepancy between numerical and high-accuracy ODE solutions. Align Your Steps (AYS) derives solver-specific schedules using stochastic calculus (Sabour et al., 2024), while GITS exploits the regularity of diffusion ODE trajectories and uses dynamic programming to adapt the discretization to their geometry (Chen et al., 2024). LD3 directly learns discretization parameters against a high-accuracy teacher and further decouples the solver timestep from the timestep supplied to the denoiser (Tong et al., 2025). HSO combines global initialization search with local schedule refinement using a midpoint-error proxy and spacing regularization (Zhu et al., 2026). More recently, INDIS extends global schedules to instance-dependent discretizations conditioned on the initial sample and generation condition (Yuan et al., 2026). Two approaches instead compare with real data: DDSS learns sampler coefficients and timesteps by backpropagating through the sampler to minimize a linear-kernel KID (Watson et al., 2022), and OYS treats the target metric as a black box and optimizes it by Bayesian optimization (Zhang et al., 2026). These methods differ substantially in search strategy and schedule parameterization, but apart from these two, their optimization criteria are primarily derived from the sampling process itself. In contrast, we ask which feature-space discrepancy to match, against which target, and how to estimate it: the feature mean relative to the model's own high-accuracy sampler preserves the local FID geometry and needs no real data (Section 5.2). To our knowledge, no prior schedule or solver optimization compares feature means with the model's own sampler. This question is complementary to schedule parameterization: we show that the same feature-aligned objective remains effective when inserted into the richer two-sequence search space of LD3. Among the global schedule methods above, only LD3 and GITS provide schedules for our checkpoints, solvers, and budgets; we compare with both in Table 2 and discuss the others in Appendix E.3.

3 PRELIMINARIES

3.1 DIFFUSION ODE SAMPLING AND TIMESTEP SCHEDULES

Diffusion sampling can be formulated as numerical integration of a probability-flow ODE (Song et al., 2021). For the EDM models used in our main analysis, we adopt the variance-exploding parameterization (Karras et al., 2022),

$$\frac{dx}{d\sigma} = \frac{x - D_\theta(x;\sigma)}{\sigma}, \tag{3}$$

where $D_\theta(x;\sigma)$ is a pretrained denoiser and σ decreases from a high-noise state toward a clean sample. A sampling schedule specifies an ordered sequence of noise levels $\sigma_0 > \sigma_1 > \cdots > \sigma_N$ through which a numerical solver advances. For a fixed model, solver, and number of function evaluations (NFE), schedule optimization changes the intermediate noise levels while leaving the underlying generative model unchanged. The schedule therefore determines how a limited computational budget is distributed along the sampling trajectory. Our formulation is not specific to EDM; for Stable Diffusion we express its discrete schedule through the EDM-form noise levels $\sigma_t = \sqrt{(1-\bar{\alpha}_t)/\bar{\alpha}_t}$, where $\bar{\alpha}_t$ is its cumulative noise schedule, query the U-Net at the correspondingly interpolated continuous timestep, and optimize these noise levels with the same objective.

3.2 DISTRIBUTIONAL EVALUATION IN FEATURE SPACE

Let $\varphi(x) \in \mathbb{R}^{d_\varphi}$ denote a feature representation, and let $(\mu_{p_1}, \Sigma_{p_1})$ and $(\mu_{p_2}, \Sigma_{p_2})$ be the feature means and covariances of two distributions p_1 and p_2. The FID (Heusel et al., 2017) is

$$\mathrm{FID}(p_1, p_2) = \|\mu_{p_1} - \mu_{p_2}\|_2^2 + \mathrm{tr}\left(\Sigma_{p_1} + \Sigma_{p_2} - 2\left(\Sigma_{p_1}^{1/2}\Sigma_{p_2}\Sigma_{p_1}^{1/2}\right)^{1/2}\right). \tag{4}$$

We denote the first term by

$$M(p_1, p_2) = \|\mu_{p_1} - \mu_{p_2}\|_2^2, \tag{5}$$

and the second, Bures covariance term by $C(p_1, p_2)$, such that

$$\mathrm{FID}(p_1, p_2) = M(p_1, p_2) + C(p_1, p_2). \tag{6}$$

This decomposition is central to our analysis. The diffusion ODE is integrated in its state space, whereas FID evaluates the resulting samples after the nonlinear transformation φ. Thus, reducing numerical error in the ODE state need not reduce either component of FID. In Section 4, we use a high-accuracy sampler as a reference and derive our objective from the mean component M.

4 METHODOLOGY

We formulate few-step schedule optimization as reference-based distribution matching. Section 4.1 defines the high-accuracy reference sampler and the feature-space discrepancy. Section 4.2 motivates the feature-mean objective and derives its unbiased finite-sample estimator. Finally, Section 4.3 relates this reference-based objective to the data FID locally, and Section 4.4 describes the schedule parameterization and optimization procedure.

4.1 REFERENCE-BASED FORMULATION

Let $S(z)$ and $R(z)$ respectively denote the sample produced from latent z by the schedule under evaluation and the sample obtained by accurately integrating the same pretrained diffusion model from the same latent. Thus, R represents the high-budget behavior of the model, and the difference between S and R measures the error introduced by few-step sampling.

Let $\varphi : \mathbb{R}^d \to \mathbb{R}^{2048}$ denote the Inception feature representation used for FID, and define $\Delta\varphi(z) = \varphi(S(z)) - \varphi(R(z))$. At the population level, let μ_S and μ_R denote the feature means $\mathbb{E}_z[\varphi(S(z))]$ and $\mathbb{E}_z[\varphi(R(z))]$. Our reference-based mean discrepancy, $\mathcal{L}_{\mathrm{FM}}$ in Eq. 2 and $\|\mathbb{E}\Delta\varphi\|^2$ in tables and figures, is

$$M_R = \|\mu_S - \mu_R\|_2^2 = \|\mathbb{E}_z[\Delta\varphi(z)]\|_2^2. \tag{7}$$

Unlike direct $\mathrm{FID}(S, \mathcal{D})$, M_R requires no statistics from the real-data distribution $\mathcal{D}$ since it only uses samples from a high-accuracy reference sampler of the same model. This makes the objective well-defined even when condition-specific real-data statistics are unavailable. The reference and real-data distributions are generally different even under accurate sampling. Therefore, M_R is not globally equivalent to data FID, and its usefulness as a scheduling objective must instead be justified through their local optimization geometry, which we analyze in Section 4.3.

4.2 Feature-Mean Objective and Finite-Sample Estimation

To understand why we isolate the mean discrepancy, consider the paired feature-space error

$$Q = \mathbb{E}_z[\|\Delta\varphi(z)\|_2^2]. \tag{8}$$

Expanding $\Delta\varphi$ around its population mean yields

$$Q = \underbrace{\|\mathbb{E}_z[\Delta\varphi(z)]\|_2^2}_{M_R} + \operatorname{tr}\operatorname{Cov}(\Delta\varphi). \tag{9}$$

Hence M_R captures the distribution-level mean feature displacement, whereas the second term measures sample-wise dispersion around that mean. The same M_R is also exactly the mean component of $\mathrm{FID}(S, R)$. This decomposition organizes the quantities we analyze along two axes: the target distribution and whether the FID covariance term is included.

Target	Mean term only	Mean + Bures term
Reference R	M_R (ours)	$\mathrm{FID}(S, R) = M_R + C_R$
Data $\mathcal{D}$	$M_{\mathcal{D}}$	$\mathrm{FID}(S, \mathcal{D}) = M_{\mathcal{D}} + C_{\mathcal{D}}$

Here, C_R and $C_{\mathcal{D}}$ denote the Bures covariance terms in FID, which are different from the paired-dispersion term $\operatorname{tr}\operatorname{Cov}(\Delta\varphi)$ appearing in Q. Moving horizontally adds a marginal covariance term, while moving vertically changes the target distribution. The paired objective Q lies outside this grid because its additional term is the covariance of the *paired discrepancy*, $\operatorname{Cov}(\Delta\varphi)$, rather than the marginal covariance term used by FID. We additionally consider

$$P = \left(\mathbb{E}_z\|S(z) - R(z)\|_2^2\right)^{1/2} / \mathbb{E}_z\|R(z)\|_2, \tag{10}$$

as a state-space control. For the pixel-space models this is evaluated in pixel space, while for Stable Diffusion it is evaluated in the latent state space.

Finite-sample estimation. The objective M_R in Eq. 7 is a population quantity, *i.e.*, $M_R = \|\mu\|_2^2$ with $\mu = \mathbb{E}_z[\Delta\varphi(z)]$. In practice, we estimate it from K i.i.d. latent samples $z_1, \ldots, z_K$. Let

$$\hat{\mu} = \frac{1}{K}\sum_{i=1}^{K} \Delta\varphi_i, \qquad \hat{Q} = \frac{1}{K}\sum_{i=1}^{K} \|\Delta\varphi_i\|_2^2, \tag{11}$$

where $\Delta\varphi_i = \Delta\varphi(z_i)$. A natural estimator of M_R is the plug-in estimator $\|\hat{\mu}\|_2^2$. For an estimator $\widehat{T}$ of a population quantity T, we define its bias as $\mathrm{Bias}(\widehat{T}) = \mathbb{E}[\widehat{T}] - T$, where the expectation is over the random draw of the finite latent set. The following proposition shows that the plug-in estimator of M_R has a systematic positive bias. More importantly, this bias is exactly proportional to the paired-dispersion term in Eq. 9, so finite-sample estimation would otherwise reintroduce the component that M_R is intended to exclude.

Proposition 1. *Let $\Sigma = \operatorname{Cov}(\Delta\varphi)$, and assume $\Delta\varphi$ has a finite second moment. Then*

$$\mathrm{Bias}\left(\|\hat{\mu}\|_2^2\right) = \frac{\operatorname{tr}\Sigma}{K}, \tag{12}$$

and equivalently

$$\mathbb{E}\|\hat{\mu}\|_2^2 = M_R + \frac{\operatorname{tr}\Sigma}{K} = \left(1 - \frac{1}{K}\right) M_R + \frac{1}{K} Q. \tag{13}$$

Moreover,

$$\widehat{M_R} = \frac{K\|\hat{\mu}\|_2^2 - \hat{Q}}{K-1} = \frac{1}{K(K-1)} \sum_{i \neq j} \langle \Delta\varphi_i, \Delta\varphi_j \rangle \tag{14}$$

is an unbiased estimator of M_R, the one-sample U-statistic of the squared linear-kernel MMD (Gretton et al., 2012). A detailed derivation is provided in Appendix B.

Proposition 1 reveals a finite-sample version of the same distinction motivating Eq. 9. Although the population objective contains only the mean feature displacement, its naive plug-in estimator implicitly mixes that objective with the paired feature error Q. We therefore optimize the unbiased estimator in Eq. 14 in all main-table optimizations; a few diagnostic experiments in the appendix predate the correction and use the plug-in estimator.

Once the latent set is drawn and fixed, Eq. 14 is deterministic as a function of the sampling schedule. The correction removes the systematic bias but not the variance from the finite latent set.

4.3 Local Relation to Data FID

The proposed objective is defined relative to a high-accuracy model reference, whereas final performance is measured against real data. We therefore analyze the two local links $M_R \to M_{\mathcal{D}} \to \mathrm{FID}$: 1) when our objective follows the data-mean term and 2) when decreasing that term decreases FID.

From the reference mean to the data mean. Let $\mu_{\mathcal{D}}$ be the real-data feature mean and define $b = \mu_R - \mu_{\mathcal{D}}$. Since $\mu_S - \mu_{\mathcal{D}} = \mathbb{E}[\Delta\varphi] + b$, the mean component of data FID satisfies

$$M_{\mathcal{D}} = \|\mathbb{E}[\Delta\varphi] + b\|_2^2 = M_R + 2\,\langle \mathbb{E}[\Delta\varphi], b\rangle + \|b\|_2^2. \tag{15}$$

Let h denote the schedule step lengths and

$$J = \frac{\partial\, \mathbb{E}[\Delta\varphi]}{\partial h}. \tag{16}$$

Within the tangent space of feasible schedules,

$$\nabla M_{\mathcal{D}} = \nabla M_R + 2J^\top b. \tag{17}$$

Thus, the reference-mean and data-mean objectives have similar local directions when $2J^\top b$, the fixed offset b seen through the schedule Jacobian, is small relative to ∇M_R.

From the data mean to FID. Using the decomposition from Section 3.2,

$$\mathrm{FID}(S, \mathcal{D}) = M_{\mathcal{D}} + C_{\mathcal{D}}. \tag{18}$$

The directional derivative of FID along $-\nabla M_{\mathcal{D}}$ is

$$\begin{aligned} D_{-\nabla M_{\mathcal{D}}}\mathrm{FID} &= -\|\nabla M_{\mathcal{D}}\|_2^2 - \langle \nabla C_{\mathcal{D}}, \nabla M_{\mathcal{D}}\rangle \\ &= -\|\nabla M_{\mathcal{D}}\|_2^2\,(1 + r\cos\theta)\,, \end{aligned} \tag{19}$$

where

$$r = \frac{\|\nabla C_{\mathcal{D}}\|_2}{\|\nabla M_{\mathcal{D}}\|_2}, \qquad \theta = \angle(\nabla C_{\mathcal{D}}, \nabla M_{\mathcal{D}}). \tag{20}$$

Therefore, $-\nabla M_{\mathcal{D}}$ is a strict FID descent direction exactly when

$$1 + r\cos\theta > 0. \tag{21}$$

Together, Eqs. 17 and 21 provide a local rather than global justification for M_R. The surrogate is useful where (i) the reference-mean direction remains close to the data-mean direction and (ii) decreasing the data-mean term also decreases FID. Combining the two, $-\nabla M_R = -\nabla M_{\mathcal{D}} + 2J^\top b$ is itself a strict FID descent direction whenever $\|2J^\top b\|_2\,\|\nabla \mathrm{FID}\|_2 < \|\nabla M_{\mathcal{D}}\|_2^2\,(1 + r\cos\theta)$, that is, whenever the offset is small relative to the descent margin. Section 5.2 tests both conditions directly. The full gradient and landscape analyses are given in the appendix.

4.4 Schedule Optimization

We optimize timestep locations with the pretrained model and solver fixed. For EDM models, let

$$\lambda_i = -\log\sigma_i, \qquad h_i = \lambda_{i+1} - \lambda_i, \tag{22}$$

with both schedule endpoints fixed. Writing

$$L = \lambda_N - \lambda_0 = \sum_{i=0}^{N-1} h_i, \tag{23}$$

we parameterize the positive step lengths as

$$h = L \cdot \mathrm{softmax}(\vartheta), \qquad \lambda_i = \lambda_0 + \sum_{j<i} h_j, \qquad \sigma_i = e^{-\lambda_i}. \tag{24}$$

The shift invariance of the softmax leaves $N - 1$ effective degrees of freedom. For Stable Diffusion the same parameterization is applied to the EDM-form noise levels of its discrete schedule (Section 3.1).

We initialize from the standard discretization associated with each model and locally refine the schedule by minimizing $\widehat{M}_R$ in Eq. 14. The same latent samples are reused throughout an optimization run, making the finite-sample objective deterministic under a fixed execution configuration. We use Nelder–Mead, which requires neither a differentiable sampler nor backpropagation through the solver. This refinement is intentionally local. Our main results start from the standard schedule associated with each model; alternative and random initializations are used only to characterize sensitivity to initialization. We do not assume that M_R is globally equivalent to FID or that arbitrary initializations recover the same solution. Exact optimization budgets, restart protocols, and initialization diagnostics are reported in Appendices A and D.2.

5 Experiments

We evaluate whether M_R preserves the local FID geometry, how closely its schedules approach best-found direct-FID solutions, and whether it transfers to richer schedule parameterizations.

5.1 Experimental protocol

We evaluate eight pixel-space settings spanning CIFAR-10, FFHQ, and AFHQv2, with DPM-Solver++ (2M), Euler, Heun, and iPNDM (Zhang & Chen, 2023) (3M), and additionally Stable Diffusion 1.5 with DPM-Solver++. Unless otherwise stated, schedule optimization uses $K = 2000$ fixed latent samples and the unbiased estimator in Eq. 14. Final quality is measured using FID with $50\,000$ generated samples for the pixel-space benchmarks and $30\,000$ samples against COCO statistics for Stable Diffusion. Appendix A reports compute costs.

Within each experimental cell, all objectives use the same schedule parameterization, initialization, and optimization budget. When optimizing FID itself, we use its finite-sample estimate with the same latents and budget and call the result the *best-found direct-FID solution*, not a global optimum. Appendix A details checkpoints, references, NFE conventions, budgets, restarts, and statistics.

5.2 Why the feature mean?

We first test the two local links underlying the objective. For the reference-to-data link, Eq. 17 gives $\nabla M_{\mathcal{D}} = \nabla M_R + 2J^\top b$. Although J depends on the schedule, the projected offset $2J^\top b$ is empirically stable in the local region we measure: its norm varies by only 7.0% across four base schedules, while $\|\nabla M_R\|$ varies by a factor of seven. At the three practical base schedules, ∇M_R and $\nabla M_{\mathcal{D}}$ differ by only $0.41°$, $0.42°$, and $0.52°$. As this small offset predicts, optimizing $M_{\mathcal{D}}$ itself, the objective of DDSS (Watson et al., 2022), on CIFAR-10 with four-step DPM-Solver++ ends at nearly the same schedule and FID as M_R (16.950 versus 16.927; Appendix C.1). For the data-mean-to-FID link, the descent condition $1 + r\cos\theta > 0$ holds at 99.5% of the feasible nodes on our CIFAR-10 diagnostic grid. Full gradient measurements and finite-difference checks are reported in Appendices C.1–C.3.

The resulting ordering is consistent with these local directions. On the 2300-schedule grid, the FID minimizer ranks **2nd** under M_R, compared with **51st** under paired feature error Q and **315th** under state-space error P. Among thirty-four promising schedules remeasured at $K = 10^4$, the minimizer of M_R coincides with the FID minimizer, whereas the schedule selected by Q has 15.75% higher FID. At the optima selected by the two controls, the gradient of M_R remains within 5–6° of the FID gradient, while the corresponding control gradients differ by 37–56°. Appendix D.1 reports the complete landscape and precision analysis.

Adding more discrepancy terms does not improve this alignment. For Q, the paired-dispersion term accounts for 96.3–97.3% of the objective near the FID optimum, largely masking the variation

Table 1: Controlled comparison of optimization objectives under the same search space and budget: each entry is the final FID of the schedule obtained by optimizing that column's objective from the initial schedule, Karras $\rho = 7$ for pixel-space models (FID@50k) and time-uniform for Stable Diffusion (COCO FID@30k). Direct FID is the best-found finite-budget FID optimization. Appendix D.1 reports other initializations; gaps in the text use unrounded FIDs.

Dataset	Solver	NFE	Initial schedule	Direct FID	M_R	Q	P
CIFAR-10	DPM-Solver++	4	36.055	16.594	16.927	20.113	20.030
FFHQ	DPM-Solver++	4	29.509	17.446	18.167	19.358	20.042
AFHQv2	DPM-Solver++	4	15.654	12.922	12.983	15.777	13.789
CIFAR-10	Euler	4	66.758	41.414	41.706	45.391	83.445
CIFAR-10	Heun	8	38.065	29.274	30.457	34.044	60.888
CIFAR-10	DPM-Solver++	16	2.580	2.073	2.106	2.276	2.420
CIFAR-10	iPNDM	4	29.489	14.071	14.354	16.416	27.996
CIFAR-10	iPNDM	10	3.691	2.656	2.672	3.760	5.397
SD 1.5	DPM-Solver++	4	23.093	16.399	16.117	17.260	15.913
SD 1.5	DPM-Solver++	5	15.438	12.214	12.471	13.797	12.694
SD 1.5	DPM-Solver++	6	13.055	11.363	11.793	12.838	12.318
SD 1.5	DPM-Solver++	7	12.157	10.782	11.128	12.859	12.442

carried by the feature mean. Adding the Bures covariance term instead, by directly optimizing $\mathrm{FID}(S, R)$, also makes optimization less reliable under the same finite-sample budget: across four tested cells, its resulting data FID is 1.62%, 0.67%, 43.7%, and 1.87% higher than that obtained from M_R. The large finite-sample distortion of the covariance estimate accompanies this harder search surface, although we do not attribute the failure to estimator bias alone (Appendices B and E.4).

5.3 Main results

Table 1 compares the schedules obtained by optimizing each objective from the standard initialization. Across the eight pixel-space settings, minimizing M_R produces schedules only 0.47–4.13% above the best-found direct-FID solutions, with a median gap of 1.78%; for reference, evaluating a fixed schedule on disjoint seeds moves its FID by 0.27–1.13% (Table 12). The corresponding median gaps are 16.48% for Q and 59.8% for P. Thus, the separation is not driven by a single dataset or solver: even the worst gap obtained with M_R remains below the best gap obtained with either Q or P.

Direct FID optimization is a search reference rather than an oracle. All methods use the same parameterization, initialization, and optimization budget, but they induce different search surfaces. Consequently, a surrogate can occasionally return a lower final FID than the direct-FID run without beating the true FID optimum. This occurs in the richer parameterization of Section 5.4, where warm-start experiments show that the original direct-FID run had not reached its best solution.

Stable Diffusion. The last four rows extend the comparison to Stable Diffusion 1.5. Across four to seven NFE, M_R stays within 1.72–3.78% of the best-found direct-FID result, compared with 2.96–15.39% for P and 5.25–19.26% for Q. Unlike the pixel-space models, P is measured in the learned latent state and performs best at four NFE, but this advantage does not persist as the budget increases. Complete protocol details and schedule comparisons are given in Appendix E.1.

5.4 Transfer to a richer parameterization

We next ask whether the objective remains useful when the schedule space is enlarged. LD3 (Tong et al., 2025) uses two time sequences, one for the denoiser query and one for the solver update. We evaluate its released schedules in our pipeline and then optimize M_R directly in the same two-sequence parameterization. Appendix E.3 reports the pipeline reproduction checks.

Within the single-sequence space, M_R reaches FID 14.354 at four NFE and 2.672 at ten NFE, 2.0% and 0.6% above the corresponding best-found direct-FID schedules and below the released GITS schedules (15.623 and 2.776). The ordering is unchanged under GITS's own fourth-order iPNDM

Table 2: CIFAR-10 with third-order iPNDM. All FIDs use $50\,000$ samples in our pipeline. The upper block uses a single time sequence, while the lower block uses LD3's richer two-sequence parameterization. In the lower block, every search starts from its own objective's single-sequence optimum; the restarted row instead starts the direct-FID search at the schedule found by minimizing $\|\mathbb{E}\Delta\varphi\|^2$. Bold marks the lowest FID per block among rows not optimizing FID directly.

	4 NFE	10 NFE
Single sequence — this paper's parameterization, run in our pipeline		
Uniform in λ (LD3's baseline)	35.153	3.643
Karras $\rho = 7$	29.489	3.691
LD3 released, first sequence only	15.025	3.351
GITS released schedule (Chen et al., 2024)	15.623	2.776
Minimizing $\|\mathbb{E}\Delta\varphi\|^2$	**14.354**	**2.672**
Best found, direct FID	14.071	2.656
Two sequences — LD3's parameterization, run in our pipeline		
LD3 released pair	9.632	2.369
Minimizing $\|\mathbb{E}\Delta\varphi\|^2$, query sequence only	8.140	2.275
Minimizing $\|\mathbb{E}\Delta\varphi\|^2$, both sequences	**8.093**	**2.201**
Direct FID, query sequence only	9.446	2.297
Direct FID, both sequences, same protocol	13.767	2.167
Direct FID, both sequences, restarted at the $\|\mathbb{E}\Delta\varphi\|^2$ schedule	7.859	—
Minimizing $\mathrm{FID}(S, R)$, both sequences	11.633	2.242

(Appendix E.3). Moving the same objective to LD3's two-sequence space reduces these values to 8.093 and 2.201, improving the released LD3 schedules (9.632 and 2.369) by 16.0% and 7.09%, respectively. M_R uses no dataset statistics and no backpropagation through the sampler.

Most of the four-NFE gain can already be obtained by optimizing only the denoiser query sequence, which reaches 8.140. At ten NFE the restriction costs more, 2.275 against 2.201, and the gap holds on two further disjoint sets of $50\,000$ seeds. Under the same protocol, direct FID optimization in the full space reaches 2.167 at ten NFE but stalls at 13.767 at four NFE; restarted at the schedule found by M_R, it reaches 7.859. Against these best-found values, M_R lies 2.98% and 1.59% above direct FID at four and ten NFE, within the range of Table 1. Appendix E.4 reports the search details.

5.5 Scope

The objective is most reliable as a local refinement criterion: its Spearman correlation with FID is $+0.997$ over the full grid but $+0.479$ among schedules within 10% of the best FID, so surrogate differences between near-optimal schedules are not equally precise differences in quality.

Refinement of standard schedules is the intended use, and it does not require a tuned start: on CIFAR-10, Karras and two clamp-based starts end within 0.2% of one another in FID (Appendix D.1). The standard schedules used for all headline results lie in the practical basin, where the surrogate follows the local FID geometry; Appendix D.2 maps the boundary of this basin. The derivation holds for any fixed feature map, so the same objective applies to other feature-based metrics; we evaluate the Inception features used by FID.

6 Conclusion

We studied what a few-step diffusion schedule should optimize when generation quality is measured by FID. The mean feature displacement from a high-accuracy reference provides a simple objective whose local directions align with data FID better than paired feature or state-space errors. We derived an unbiased estimator that prevents the usual finite-sample estimate from reintroducing paired dispersion. Across eight pixel-space settings and Stable Diffusion, optimizing this objective produces schedules close to those found by direct FID search. It also improves LD3's released schedules in the two-sequence parameterization. Together, these results show that choosing the scheduling objective matters even when the model and solver are fixed.

AI USE STATEMENT

In this work, we used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, and interpret results. Generating synthetic data sets, cleaning or reformatting datasets, and qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to suggest experimental parameters, create or edit software code, create artifacts, draft parts of the paper, summarize and analyze existing literature, discover research topics and identify gaps, brainstorm, search for information, edit the paper to improve readability, identify relevant literature, and suggest a structure for the paper. We have reviewed all AI-assisted work. The authors checked the derivations step by step, and all authors read and approved the final text. The AI-written code was verified with tests against independent reference implementations and analytic solutions, our evaluation pipeline reproduces the published EDM FID, and the reported results are recomputed from the recorded experiment logs by a script that compares them with the paper; the tests and the script are included in the supplementary material. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We train no models; all experiments use publicly released pretrained checkpoints, listed in Section 5.1 and Appendix A. The objective and its unbiased estimator are defined in Section 4.2, with the derivation in Appendix B, and the schedule parameterization and optimizer in Section 4.4. Appendix A gives the implementation details, including reference construction, numerical precision, and the NFE convention; the search hyperparameters and budgets (Table 3); the image processing used for FID; and the compute cost (Table 4). Appendix E.1 describes the Stable Diffusion protocol, and Appendices B–E contain the additional experiments and ablations. The supplementary material contains the code for reference generation, schedule optimization, and FID evaluation, including the COCO prompt and reference-statistics preparation for Stable Diffusion; the final schedules for all entries of Tables 1 and 2; the logs from which the reported numbers are computed, with a script that recomputes these numbers; and tests for the samplers and estimators.

REFERENCES

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018.

Defang Chen, Zhenyu Zhou, Can Wang, Chunhua Shen, and Siwei Lyu. On the trajectory regularity of ODE-based diffusion sampling. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 2024.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J. Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25), 2012.

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. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with Rectified Flow. In *International Conference on Learning Representations*, 2023.

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. In *Advances in Neural Information Processing Systems*, volume 35, 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), 2025.

Amirmojtaba Sabour, Sanja Fidler, and Karsten Kreis. Align your steps: Optimizing sampling schedules in diffusion models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, 2024.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 2023.

Vinh Tong, Trung-Dung Hoang, Anji Liu, Guy Van den Broeck, and Mathias Niepert. Learning to discretize denoising diffusion ODEs. In *International Conference on Learning Representations*, 2025.

Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022.

Shuchen Xue, Zhaoqiang Liu, Fei Chen, Shifeng Zhang, Tianyang Hu, Enze Xie, and Zhenguo Li. Accelerating diffusion sampling with optimized time steps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Liangyu Yuan, Ruoyu Wang, Tong Zhao, Dingwen Fu, Mingkun Lei, Beier Zhu, and Chi Zhang. Few-step diffusion sampling through instance-aware discretizations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Qinsheng Zhang and Yongxin Chen. Fast sampling of diffusion models with exponential integrator. In *International Conference on Learning Representations*, 2023.

Travis Zhang, Christian Belardi, Justin Lovelace, Jin Peng Zhou, Saebyeol Shin, Carla P. Gomes, and Kilian Q. Weinberger. Optimize your sampling: Tuned diffusion sampling with Bayesian optimization. arXiv preprint arXiv:2608.18040, 2026.

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

Aihua Zhu, Rui Su, Qinglin Zhao, Li Feng, Meng Shen, and Shibo He. Hierarchical schedule optimization for fast and robust diffusion model sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, 2026.

A Experimental and Optimization Details

This appendix provides the implementation details needed to reproduce the reference construction, schedule search, diagnostic measurements, and evaluation protocol used throughout the paper.

Reference construction and numerical precision. For the unconditional pixel-space models, the reference sample $R(z)$ is generated from the same latent z as the candidate sample by integrating the same pretrained probability-flow ODE with Heun on a Karras $\rho = 7$ schedule. We use 64 Heun steps, corresponding to 128 network evaluations. For Stable Diffusion, where each network evaluation is substantially more expensive, the reference uses 32 steps of DPM-Solver++ (2M) on the same Karras schedule, corresponding to 32 network evaluations.

For the pixel-space EDM models, ODE states are maintained in `float64`, while denoiser evaluations use `float32`. Stable Diffusion follows its native lower-precision inference pipeline; the consequences of its sensitivity to execution configuration are quantified below. For FID evaluation, generated images are mapped to $\{0, \ldots, 255\}$ and converted to `uint8` before feature extraction with the standard Inception network. For Stable Diffusion, the Inception network received inputs in $[0, 1]$ rather than its expected $[-1, 1]$; since every Stable Diffusion FID and feature-based objective uses this same extractor, comparisons within Stable Diffusion remain consistent, but the absolute values are not directly comparable to standard COCO FID, including those reported by Tong et al. (2025).

The reference is a high-accuracy numerical approximation rather than an exact ODE solution. We therefore separately vary its integration budget. Appendix D shows that increasing the reference precision moves its feature mean slightly but leaves the relevant candidate ordering and selected arg-min unchanged.

Models and implementation checks. We use the public unconditional VP EDM checkpoints for CIFAR-10 (32×32), FFHQ and AFHQv2 (64×64). Our iPNDM implementation agrees bit for bit with the reference implementation of Zhou et al. (2024) on nonuniform nodes at the orders and step counts used here. Like that implementation, it uses the multistep coefficients derived for uniform steps, so its order counts the denoiser outputs combined in each step rather than giving its convergence order on nonuniform schedules. As an end-to-end check of our generation and FID pipeline, running EDM's 18-step Heun schedule (35 NFE) through it on the CIFAR-10 checkpoint gives FID@50k 1.959, compared with the 1.97 reported by Karras et al. (2022) as the minimum of three evaluations. The published LD3 discretizations and their reproduced FIDs are examined in Appendix E.3.

NFE and terminal-step convention. A schedule with $N + 1$ nodes executes N solver intervals. We count each denoiser evaluation toward NFE, execute every scheduled solver step, and terminate at the model's minimum noise level. We do not replace the final solver step with a separate terminal denoising call in the main comparisons. Appendix E.4 examines the alternative convention.

Optimization protocol and search budgets. For each optimization run, the K latent samples are drawn once and reused at every objective evaluation. The finite-sample objective is therefore deterministic once the execution configuration is fixed rather than being resampled at every optimizer step.

We optimize the schedule with Nelder–Mead. Within each experimental cell, all compared objectives use the same schedule parameterization, initialization, and function-evaluation allowance. For the pixel-space and decoupled experiments, a search consists of restarted Nelder–Mead rounds: after each round, a new simplex is initialized around the best point found by the previous round. Stable Diffusion is the exception and uses a single fixed-budget round. The final protocol is summarized below.

Stable Diffusion is thus reported as a fixed-budget search rather than as a search that has stabilized under restarts. For the remaining cells, the restarted protocol is used uniformly within each comparison. Nelder–Mead's own termination criterion is not interpreted as evidence of a global minimum, and throughout the paper "best-found" therefore means the best solution reached under the stated finite-budget search.

Restarts are consequential for our objective even in low-dimensional cells. For example, on CIFAR-10 with four-step DPM-Solver++, restarting from the first-round incumbent moves the final FID from 17.089 to 16.927. Applying the same additional budget to the other objectives changes direct FID, P, and Q by 0.000%, 0.000%, and 0.013%, respectively. We therefore apply the same restart protocol to every objective within a cell rather than allowing objective-dependent stopping rules.

Table 3: Nelder–Mead search budgets. "Budget × rounds" gives the maximum number of function evaluations per round and the number of restarted rounds. The last column reports the resulting nominal evaluations per effective free parameter.

dataset	solver	NFE	free params	budget × rounds	per param
CIFAR-10, FFHQ, AFHQv2	DPM-Solver++	4	3	180×3	180
CIFAR-10	Euler	4	3	180×3	180
CIFAR-10	Heun	8	3	180×3	180
CIFAR-10	iPNDM	4	3	300×3	300
CIFAR-10	iPNDM	10	9	400×3	133
CIFAR-10	DPM-Solver++	16	15	500×3	100
Stable Diffusion 1.5	DPM-Solver++	4–7	3–6	400×1	67–133
Decoupled iPNDM parameterization					
CIFAR-10	iPNDM	4	7	700×3	300
CIFAR-10	iPNDM	10	19	1600×3	253
CIFAR-10, query only	iPNDM	4	4	400×3	300
CIFAR-10, query only	iPNDM	10	10	400×3	120

The higher-dimensional searches remain more difficult. In the nineteen-parameter decoupled setting, an additional round can still substantially reduce the objective. Results in such cells should consequently be read as outcomes of a controlled finite-budget search rather than as proven minima of the corresponding objective.

Computational cost. One objective evaluation generates the $K = 2000$ samples, extracts their Inception features, and computes the objective from these features; the first two steps dominate. We measure this cost with the optimizer's own sampling and feature-extraction code on an otherwise idle NVIDIA RTX 3090, averaging ten evaluations after one warm-up, and multiply it by the number of evaluations each search actually used (Table 4). The Stable Diffusion searches ran on 16 NVIDIA GB200 GPUs on exclusively allocated nodes, and their times are taken from the job logs.

Table 4: Compute cost of the schedule search. Time per evaluation is the wall-clock time of one objective evaluation with $K = 2000$ on the stated hardware. Search cost is the number of evaluations used by the M_R search multiplied by its time per evaluation and by the number of GPUs, in GPU-hours of that hardware.

				time per evaluation (s)		search
dataset	solver	NFE	evaluations	M_R	direct FID	(GPU-h)
One RTX 3090						
CIFAR-10	DPM-Solver++	4	543	17.3	18.0	2.6
CIFAR-10	Euler	4	543	17.4	18.1	2.6
CIFAR-10	Heun	8	543	33.0	33.7	5.0
CIFAR-10	iPNDM	4	903	17.4	18.1	4.4
CIFAR-10	iPNDM	10	1203	40.8	41.4	13.6
CIFAR-10	DPM-Solver++	16	1503	64.1	64.8	26.8
FFHQ	DPM-Solver++	4	543	35.6	36.2	5.4
AFHQv2	DPM-Solver++	4	543	35.5	36.2	5.4
CIFAR-10, decoupled	iPNDM	4	2100	17.3	18.0	10.1
CIFAR-10, decoupled	iPNDM	10	4769	40.8	41.5	54.0
16 GB200, one round						
Stable Diffusion 1.5	DPM-Solver++	4	339	3.26	6.44	4.9
Stable Diffusion 1.5	DPM-Solver++	5	389	3.82	7.10	6.6
Stable Diffusion 1.5	DPM-Solver++	6	400	4.39	7.12	7.8
Stable Diffusion 1.5	DPM-Solver++	7	400	4.95	7.64	8.8

Generating the 2000 reference samples is a one-time cost per model: with 64 Heun steps, 8.4 minutes on CIFAR-10 and 18.1 on each of FFHQ and AFHQv2 on the same GPU, and with 32 DPM-Solver++ steps, 22 seconds for Stable Diffusion on 16 GB200 GPUs. The final FID@50k evaluation of a schedule, which belongs to the evaluation protocol rather than to the search, takes 8.8 minutes

at 32×32 and 16.9 minutes at 64×64; for Stable Diffusion, the final FID@30k evaluation runs on four NVIDIA H100 GPUs and takes 5 to 8 minutes per schedule.

In pixel space, direct FID search costs about the same per evaluation as M_R, because sampling dominates. In Stable Diffusion, where sampling on 16 GPUs is fast, the covariance square root of FID increases the time per evaluation by a factor of 1.5 to 2. Using a single round instead of three reduces the search cost by two thirds, at the FID cost reported above for CIFAR-10 (17.089 versus 16.927).

Diagnostic measurements. We use three diagnostics to separate properties of an objective from behavior specific to Nelder–Mead.

First, for the four-step CIFAR-10 analysis, we discretize the feasible log-noise interval into 25 levels and enumerate all

$$\binom{25}{3} = 2300 \tag{25}$$

strictly increasing interior-node triples. Every objective is evaluated on the complete grid. This allows us to compare objective landscapes and identify low-objective regions without relying on the trajectory of an optimizer. For basin analysis, the grid is treated as a graph under nearest-neighbor adjacency in index space, and connected components are counted within FID sublevel sets.

Second, because estimates at $K = 2000$ can misorder near-tied schedules, we remeasure leading candidates at $K = 10^4$. The remeasurement set is constructed from schedules selected using FID and the proposed objective. Consequently, correlations computed only within this set are subject to selection-induced range restriction. We therefore use these candidates primarily for arg-min and regret comparisons rather than as an unbiased estimate of global correlation.

Third, we estimate local gradient directions in the schedule step-length coordinates. At a base schedule h_0, we draw $n_d = 20$ directions d_k uniformly from the unit sphere of the tangent space

$$\left\{ d : \sum_i d_i = 0 \right\}, \tag{26}$$

so that the endpoint constraint is preserved. Directions producing non-positive step lengths are discarded. For an objective g, we evaluate

$$g(h_0 + \varepsilon L d_k) - g(h_0) \approx \varepsilon L \, d_k^\top \nabla g \tag{27}$$

and recover ∇g by least squares. We use two perturbation scales, $\varepsilon \in \{0.03, 0.08\}$, and report both rather than selecting the one that produces the more favorable alignment. For analyses involving the full schedule-reachable feature subspace, we construct the corresponding feature Jacobian column by column instead of estimating it from random directions.

Execution determinism and data parallelism. Fixing the latent set eliminates Monte Carlo resampling during search, but execution-level determinism also depends on how those latents and model evaluations are batched.

On the pixel-space pipeline, latent samples are generated in a fixed global order and whole batches are dispatched to the available devices. In a direct CIFAR-10 check with $K = 512$, executing the same samples on one and two devices produces feature matrices with maximum elementwise difference zero. The tested pixel-space pipeline is therefore bit-exact across those device partitions.

Stable Diffusion is more sensitive. Prompts are sharded across ranks, so changing the number of devices changes the batches in which samples are evaluated as well as the ordering of floating-point reductions. The U-Net, VAE, and text encoder also operate in lower precision. In one pair of searches from the same start and with the same objective and $K = 2000$, changing the run configuration gives

ranks	objective at start	objective at end	FID@30k
16	6.2311	3.0390	16.0753
8	6.2113	2.8085	15.6341
relative	0.32%	7.6%	2.7%

The two runs above also used different GPU hardware, so they do not isolate device count. A second comparison on H100 hardware with four versus eight devices changes the starting objective by 0.33%, the final objective by 0.48%, and FID by 0.29%. These comparisons show that the Stable Diffusion search outcome is not invariant to execution configuration.

We therefore keep hardware, device count, batch construction, and random number generation fixed within every Stable Diffusion comparison. The effect concerns the search trajectory rather than the final FID measurement, which is evaluated under a fixed configuration.

Statistical and comparison protocol. Several auxiliary comparisons involve only a small number of schedules. We therefore avoid asymptotic significance tests in these settings. For example, with $n = 4$ observations, the smallest attainable two-sided p-value for an exact permutation test of a rank statistic is 0.083. Whenever a small candidate family is used for a formal comparison, we report exact permutation results rather than relying on asymptotic library approximations.

When comparing two correlations measured on the same candidate set, we use a paired bootstrap confidence interval for their difference together with a label-swap permutation test, rather than comparing two independently obtained p-values. We also distinguish between complete-grid analyses and candidate sets selected by one or more objectives: correlations on selected sets are range-restricted by construction and are therefore not treated as unbiased estimates of global association.

Finally, Q and P are treated as matched controls rather than strawman baselines. Within a cell they use the same schedule parameterization, practical initialization, latent count, and search budget as the proposed objective. Accordingly, an explanation for the behavior of M_R should also account for why these closely related discrepancy measures can select different schedules.

B Finite-Sample Estimation of the Feature-Mean Objective

This appendix provides the derivation of Proposition 1 and examines how finite-sample estimation affects the resulting schedule optimization. The central issue is specific to the structure of our objective: although the population quantity M_R contains only the mean feature displacement, its naive finite-sample plug-in estimator reintroduces part of the paired-dispersion term that M_R is intended to exclude.

Derivation of Proposition 1. Let

$$X_i = \Delta\varphi_i, \qquad \mu = \mathbb{E}[X_i], \qquad \Sigma = \mathrm{Cov}(X_i), \tag{28}$$

where $X_1, \dots, X_K$ are i.i.d. samples induced by the latent draws. The sample mean is

$$\hat{\mu} = \frac{1}{K}\sum_{i=1}^{K} X_i. \tag{29}$$

Since

$$\mathbb{E}[\hat{\mu}] = \mu, \qquad \mathrm{Cov}(\hat{\mu}) = \frac{\Sigma}{K}, \tag{30}$$

the second-moment decomposition gives

$$\mathbb{E}\|\hat{\mu}\|_2^2 = \|\mathbb{E}\hat{\mu}\|_2^2 + \mathrm{tr}\,\mathrm{Cov}(\hat{\mu}) \tag{31}$$

$$= \|\mu\|_2^2 + \frac{\mathrm{tr}\,\Sigma}{K} \tag{32}$$

$$= M_R + \frac{\mathrm{tr}\,\Sigma}{K}. \tag{33}$$

Because

$$Q = \mathbb{E}\|X_i\|_2^2 = \|\mu\|_2^2 + \mathrm{tr}\,\Sigma = M_R + \mathrm{tr}\,\Sigma, \tag{34}$$

Eq. 33 can equivalently be written as

$$\mathbb{E}\|\hat{\mu}\|_2^2 = \left(1 - \frac{1}{K}\right) M_R + \frac{1}{K} Q. \tag{35}$$

Thus the bias of the naive plug-in estimator is

$$\mathrm{Bias}\left(\|\hat{\mu}\|_2^2\right) = \frac{\mathrm{tr}\,\Sigma}{K}. \tag{36}$$

Now define

$$\hat{Q} = \frac{1}{K}\sum_{i=1}^{K}\|X_i\|_2^2. \tag{37}$$

Since $\mathbb{E}[\hat{Q}] = Q$,

$$\mathbb{E}\left[\frac{K\|\hat{\mu}\|_2^2 - \hat{Q}}{K-1}\right] = \frac{K\left(M_R + \frac{\mathrm{tr}\,\Sigma}{K}\right) - (M_R + \mathrm{tr}\,\Sigma)}{K-1} \tag{38}$$

$$= M_R. \tag{39}$$

Hence

$$\widehat{M}_R = \frac{K\|\hat{\mu}\|_2^2 - \hat{Q}}{K-1} \tag{40}$$

is unbiased.

Its U-statistic form follows by expanding

$$K^2\|\hat{\mu}\|_2^2 = \sum_{i=1}^{K}\sum_{j=1}^{K}\langle X_i, X_j\rangle \tag{41}$$

and subtracting the diagonal terms

$$K\hat{Q} = \sum_{i=1}^{K}\|X_i\|_2^2. \tag{42}$$

Therefore,

$$\widehat{M}_R = \frac{1}{K(K-1)}\sum_{i\neq j}\langle X_i, X_j\rangle, \tag{43}$$

which is Eq. 14 in the main text. This is the one-sample U-statistic form of the unbiased MMD^2 estimator (Gretton et al., 2012) with a linear kernel, taking the pairs $(\varphi(S(z_i)), \varphi(R(z_i)))$ as observations. Gretton et al. (2012) state it for pairs whose two members are independent; it remains unbiased here, where both members come from the same latent, because each term $\langle X_i, X_j\rangle$ with $i \neq j$ involves two independent latents. The unbiased estimate can be negative at finite K, despite $M_R \geq 0$ at the population level; this does not affect its use as an optimization objective.

How much does finite-sample estimation matter? The derivation above identifies the bias analytically. We next ask whether finite-sample effects are large enough to alter the optimization landscape or the schedule returned by the optimizer.

We first compare the empirical mean term, the Bures covariance term, and their sum $\mathrm{FID}(S, R)$ over 16 schedules perturbed around the CIFAR-10 solution. We use $K = 10^4$ measurements as a high-sample reference and evaluate disjoint subsamples at smaller K.

Table 5: Finite-sample behavior of the reference-FID terms over 16 nearby schedules. "Bias" denotes relative deviation from the corresponding $K = 10^4$ measurement, which is used here as a high-sample reference rather than an exact population value.

	Spearman vs. $K = 10^4$			coeff. of variation (%)			bias (%)		
K	500	1000	2000	500	1000	2000	500	1000	2000
plug-in mean term	+0.92	+0.95	+0.97	4.4	6.3	3.8	+12.9	+4.8	+2.2
Bures covariance term	+0.97	+0.98	+1.00	1.9	2.3	1.3	+451	+264	+128
full $\mathrm{FID}(S, R)$	+0.87	+0.91	+0.95	1.8	2.6	1.7	+306	+178	+86

The Bures covariance term exhibits a much larger finite-sample shift in absolute value than the mean term. At $K = 2000$, the 2048-dimensional empirical covariance is also rank-deficient. Nevertheless, the shift is similar across nearby schedules, and its schedule ranking remains highly consistent with the high-sample remeasurement. Thus a large finite-sample distortion in value need not appear as high variance or unstable ranking; it can instead alter the surface on which the optimizer operates while preserving much of its ordering.

For the proposed objective, the more direct question is whether removing the known plug-in bias changes the optimized schedule. We therefore rerun each cell from the same initialization and with the same search budget, changing only the estimator of M_R from $\|\hat{\mu}\|_2^2$ to Eq. 14. The controls are not rerun in this ablation: $\hat{Q}$ is already unbiased for Q, while P and direct FID_K, the FID estimated from the K samples, do not receive this correction.

To quantify the magnitude of the contamination, we define the *dispersion share*

$$\eta = \frac{\operatorname{tr} \Sigma}{K M_R}. \tag{44}$$

Table 6: Controlled comparison between the naive plug-in estimator and the unbiased estimator of M_R. Each row keeps the initialization and search budget fixed and changes only the estimator. The reported values are final FIDs of the resulting schedules. This table is an estimator ablation; the main tables use the unbiased estimator together with the final search protocol and can therefore differ from the "unbiased" values shown here.

cell	dispersion share	plug-in	unbiased	FID change
CIFAR-10 Euler 4	0.7%	41.7028	41.7063	+0.008%
FFHQ DPM-Solver++ 4	1.0%	18.1621	18.1696	+0.041%
same cell, $\rho = 11$ start	1.0%	18.1652	18.1776	+0.068%
CIFAR-10 Heun 8	1.0%	30.5183	30.4734	−0.147%
AFHQv2 DPM-Solver++ 4	1.6%	12.9841	12.9817	−0.018%
same cell, $\rho = 11$ start	1.6%	13.0789	13.0768	−0.016%
CIFAR-10 DPM-Solver++ 4	1.9%	17.0931	17.0885	−0.027%
same cell, clamp start	1.9%	16.9522	16.9113	−0.241%
CIFAR-10 iPNDM 4	2.3%	14.3520	14.3508	−0.008%
CIFAR-10 iPNDM 10	12.9%	2.6800	2.6876	+0.282%
CIFAR-10 DPM-Solver++ 16	44.7%	2.1510	2.1153	−1.661%

Ten of the eleven controlled reruns change FID by less than one third of a percent, with a median change of -0.016%. The exception is the setting with the largest dispersion share, where debiasing improves FID by 1.66%.

The schedule itself can move substantially before this displacement becomes visible in FID. Taking the largest relative movement of any interior node, the change is at most 2.3% across the runs below a 5% dispersion share, 9.7% at a share of 12.9%, and 15.1% at 44.7%. At the intermediate share, the schedule has already moved appreciably while its FID change remains comparable to the evaluation variation of a fixed schedule. This is consistent with the flattening of FID among already-good schedules analyzed in Appendix D.

The dispersion share itself should not be interpreted as a universal threshold. For example, on Stable Diffusion the measured shares range from 3.6% to 43.6% over four to seven evaluations, yet replacing the plug-in estimator with the unbiased estimator changes final FID by only $+0.26\%$, -0.06%, 0.00%, and -0.17%, respectively. The effect of the correction therefore depends on the local FID landscape in addition to the magnitude of the estimator bias. We nevertheless use Eq. 14 uniformly in all final reported optimizations, rather than deciding post hoc whether the correction happens to change FID in a particular cell.

Pairing interpretation. At the population level,

$$M_R = \|\mathbb{E}\varphi(S) - \mathbb{E}\varphi(R)\|_2^2 \tag{45}$$

depends only on the marginal feature means of S and R and is therefore independent of the particular coupling used to pair samples from the two distributions. The same is true of $\mathrm{FID}(S, \mathcal{D})$. By contrast, Q and P are explicitly constructed from paired per-sample differences.

This distinction does not make pairing invariance the mechanism behind M_R. In particular, $\mathrm{FID}(S, R)$ is also a marginal, pairing-invariant quantity, yet its optimization behavior can differ substantially from that of M_R. Moreover, the finite-sample estimator in Eq. 14 is computed from the paired differences $\Delta\varphi_i$. For a fixed finite sample, changing that pairing changes $\hat{Q}$ and hence can change $\widehat{M}_R$, even though the population target itself remains pairing-invariant. We therefore use pairing invariance only to distinguish the population quantities conceptually, not as an explanation for the empirical success of the proposed objective.

C Local Geometry of the Objective

Section 4.3 justifies the proposed objective through two local links,

$$M_R \;\longrightarrow\; M_{\mathcal{D}} \;\longrightarrow\; \mathrm{FID}(S, \mathcal{D}). \tag{46}$$

This appendix examines both links in detail and then measures the resulting end-to-end geometry directly. All gradients are taken with respect to the schedule step lengths h in the feasible tangent space

$$\left\{ d : \sum_i d_i = 0 \right\}, \tag{47}$$

rather than with respect to the softmax coordinates ϑ. Gradient angles are not invariant to reparameterization, while the actual schedule search itself remains derivative-free.

C.1 From the Reference Mean to the Data Mean

Recall that

$$b = \mu_R - \mu_{\mathcal{D}}, \tag{48}$$

so that

$$M_{\mathcal{D}} = \|\mathbb{E}[\Delta\varphi] + b\|_2^2 . \tag{49}$$

Let

$$J = \frac{\partial \mathbb{E}[\Delta\varphi]}{\partial h}. \tag{50}$$

Differentiating gives the identity

$$\nabla M_{\mathcal{D}} - \nabla M_R = 2J^\top b. \tag{51}$$

The feature-space displacement b is fixed for a given reference and data distribution, but J depends on the schedule. Thus $2J^\top b$ is not in general a globally fixed gradient offset. The relevant question is instead whether it remains small relative to ∇M_R in the local region where schedule refinement is performed.

Let

$$a = \nabla M_R, \qquad \delta = 2J^\top b, \tag{52}$$

so that

$$\nabla M_{\mathcal{D}} = a + \delta. \tag{53}$$

Whenever $\|\delta\|_2 \le \|a\|_2$, the angular perturbation satisfies

$$\angle(a, a+\delta) \le \arcsin \frac{\|\delta\|_2}{\|a\|_2}. \tag{54}$$

The extremal configuration occurs when δ is orthogonal to the resulting vector $a + \delta$, equivalently

$$\delta^\top a = -\|\delta\|_2^2. \tag{55}$$

In particular, orthogonality between δ and a is not the equality condition.

We evaluate Eq. 51 at the same four base schedules used for the directional-gradient analysis. At $\varepsilon = 0.03$, the measured quantities are listed in Table 7.

At the three practical base schedules, ∇M_R and $\nabla M_{\mathcal{D}}$ are therefore nearly parallel. Across all four schedules, $\|2J^\top b\|_2$ varies by only 7.0% around its mean, whereas $\|\nabla M_R\|_2$ changes by a factor of

Table 7: Local discrepancy between the reference-mean and data-mean gradients. The final row is the worst-case angular bound from Eq. 54.

base schedule	clamp	min. of Q	min. of P	best-found FID solution
$\|\nabla M_R\|_2$	1.773	1.811	1.953	0.275
$\|2J^\top b\|_2$	0.243	0.235	0.237	0.202
$\|2J^\top b\|_2/\|\nabla M_R\|_2$	0.137	0.130	0.121	0.734
$\angle(\nabla M_\mathcal{D}, \nabla M_R)$	0.41°	0.42°	0.52°	29.6°
bound, Eq. 54	7.9°	7.5°	7.0°	47.3°

approximately seven. The deterioration near the best-found FID solution is consequently associated primarily with the reference-mean gradient becoming small rather than with a rapidly growing offset term.

This behavior should not be read as a dataset-independent quantitative guarantee. Equation 51 is general, but both J and $b = \mu_R - \mu_\mathcal{D}$ depend on the model, data distribution, and local schedule. The bound only states that whenever the projected offset is small relative to $\|\nabla M_R\|_2$, it cannot strongly rotate the data-mean gradient.

The schedule-reachable feature subspace. Equation 51 also shows that only the component of b visible through the schedule Jacobian can affect the gradient. A schedule with $N-1$ free directions can move the feature mean only within the column space induced by J. If a direction in the 2048-dimensional feature space were unrelated to a random $(N-1)$-dimensional subspace, its characteristic projection ratio at $N = 4$ would be approximately

$$\sqrt{\frac{N-1}{2048}} = 0.038. \tag{56}$$

Computing J column by column, we instead measure

$$\frac{\|P_J b\|_2}{\|b\|_2} = 0.45\text{–}0.53 \tag{57}$$

over the same four base schedules. Thus the reference-to-data displacement is not approximately orthogonal to the feature directions reachable by schedule perturbations in this experiment. The small angular discrepancy in Table 7 therefore cannot be explained simply by generic high-dimensional orthogonality.

Optimizing the data-mean term directly. Equation 51 guarantees agreement only locally: near the best-found FID solution the offset is no longer small relative to ∇M_R (Table 7), and at a minimizer of M_R the data-mean gradient reduces to $2J^\top b$, which need not vanish. Whether the two objectives lead to similar schedules is therefore an empirical question. We optimize $M_\mathcal{D}$ itself on CIFAR-10 with four-step DPM-Solver++, using the same initialization, latent set, search budget, and three-round protocol as for M_R, with the corresponding unbiased estimator and the mean of the FID reference statistics as $\mu_\mathcal{D}$. This quantity coincides with the objective optimized by DDSS (Watson et al., 2022), the linear-kernel special case of KID (Bińkowski et al., 2018). The resulting schedule, $[80,\ 5.18,\ 1.056,\ 0.278,\ 0.002]$, is close to that found by M_R, $[80,\ 5.17,\ 1.075,\ 0.272,\ 0.002]$. Its FID of 16.950 is within 0.13% of M_R (16.927), below the seed-to-seed variation of a fixed schedule in Table 12, and both lie 2.0–2.1% above the best-found direct-FID schedule (16.594). In this setting, the reference therefore recovers what the data statistics provide without requiring them.

C.2 From the Data Mean to FID

The second link concerns whether decreasing the data-mean term also decreases the full FID. Let

$$F(h) = \mathrm{FID}(S, \mathcal{D}) = M_\mathcal{D}(h) + C_\mathcal{D}(h). \tag{58}$$

For $\nabla M_\mathcal{D} \neq 0$, consider the direction

$$d = -\nabla M_\mathcal{D}. \tag{59}$$

The directional derivative is

$$D_d F = \langle \nabla F, d \rangle \tag{60}$$

$$= -\|\nabla M_{\mathcal{D}}\|_2^2 - \langle \nabla C_{\mathcal{D}}, \nabla M_{\mathcal{D}} \rangle \tag{61}$$

$$= -\|\nabla M_{\mathcal{D}}\|_2^2 \left(1 + r \cos\theta\right), \tag{62}$$

where

$$r = \frac{\|\nabla C_{\mathcal{D}}\|_2}{\|\nabla M_{\mathcal{D}}\|_2}, \qquad \theta = \angle(\nabla C_{\mathcal{D}}, \nabla M_{\mathcal{D}}). \tag{63}$$

Therefore, $-\nabla M_{\mathcal{D}}$ is a strict FID descent direction exactly when

$$1 + r \cos\theta > 0. \tag{64}$$

When $r < 1$, the condition holds regardless of the relative angle. For $r \geq 1$, alignment between the two components becomes important. We measure the condition on a 31×31 slice through the CIFAR-10 four-step solution at fixed λ_2, estimating FID and its two components from $K = 2000$ samples at each grid point. Of the 961 grid points, 234 have strictly ordered nodes, and the finite-difference gradient is defined at 204 of them.

Table 8: FID descent condition on a local CIFAR-10 four-step grid. The margin is $1 + r \cos\theta$.

FID band	points	median r	median $\cos\theta$	median margin	min margin	margin > 0
$\leq 1.1\times$	12	1.393	$+0.888$	$+1.441$	-0.046	91.7%
$\leq 1.5\times$	54	1.124	$+0.898$	$+1.870$	-0.046	98.1%
$\leq 2.0\times$	110	1.072	$+0.972$	$+1.920$	-0.046	99.1%
$\leq 4.0\times$	204	1.056	$+0.994$	$+1.992$	-0.046	99.5%

The condition holds at 203 of these 204 points. Among the 95 points with $r \leq 1$, every measured margin is positive; for points with $r < 1$, strict positivity follows directly from the bound $\cos\theta \geq -1$. The remaining 109 points have $r > 1$, so their validity depends on the relative gradient angle. Only one violates the condition, and it lies in the innermost FID band.

The margin becomes less uniform as the optimum is approached. Nine of the twelve points in the innermost band have $r > 1$, compared with approximately half of the complete grid, and the median margin decreases from 1.99 over the full measured region to 1.44 in the innermost band. Thus the covariance component becomes increasingly relevant precisely where the total FID landscape is approaching stationarity.

At an exact stationary point,

$$\nabla M_{\mathcal{D}} + \nabla C_{\mathcal{D}} = 0. \tag{65}$$

If the two component gradients are nonzero, they must have equal magnitudes and opposite directions, giving

$$r = 1, \qquad \cos\theta = -1, \tag{66}$$

and hence zero descent margin. If both component gradients vanish, θ is undefined, but the total FID gradient is already zero. The sufficient descent condition therefore naturally loses discriminative power near a stationary point.

C.3 Empirical Gradient and Landscape Analysis

The previous two subsections analyze the links $M_R \to M_{\mathcal{D}}$ and $M_{\mathcal{D}} \to$ FID separately. We now examine their end-to-end effect and compare the proposed objective with the paired feature error Q and the state-space control P.

Using the directional-gradient procedure described in Appendix A, we estimate gradients at four base schedules with $K = 10^4$ and perturbation scales $\varepsilon \in \{0.03, 0.08\}$.

At the three practical non-FID-optimal schedules, the gradients of M_R, $M_{\mathcal{D}}$, and $C_{\mathcal{D}}$ all lie within approximately 5–7° of the FID gradient at both perturbation scales. In contrast, the gradients of Q and P are displaced by tens of degrees.

Table 9: Angle to $\nabla\mathrm{FID}(S, \mathcal{D})$ in degrees for $\varepsilon = 0.03/0.08$. $M_{\mathcal{D}}$ and $C_{\mathcal{D}}$ denote the mean and Bures covariance components of data FID. $C_{\mathcal{D}}$ is distinct from $\mathrm{tr}\,\mathrm{Cov}(\Delta\varphi)$, the paired-dispersion term in Q.

base schedule	$\|\nabla\mathrm{FID}\|_2$	$M_{\mathcal{D}}$	M_R	$C_{\mathcal{D}}$	Q	P
clamp initialization	$3.77/11.08$	$6/6$	$6/6$	$7/7$	$83/43$	$88/50$
minimizer of Q	$3.92/11.54$	$7/6$	$6/5$	$7/6$	$56/40$	$53/37$
minimizer of P	$4.12/11.82$	$5/6$	$5/6$	$6/6$	$59/41$	$47/37$
best-found FID solution	$0.54/2.57$	$77/33$	$52/23$	$40/42$	$60/93$	$89/125$

The disagreement of Q cannot be explained simply by evaluating error in the wrong representation: Q uses the same Inception features as FID. The control angles should be read with care. They change by 10–40° between the two perturbation scales, and at the minimizers of Q and P the estimated control gradient grows by $6.1\times$ and $5.4\times$ when the perturbation scale grows by $2.7\times$, so second-order effects dominate these estimates. Each control is therefore nearly stationary at its own minimizer, whereas FID still has a substantial gradient there, which M_R follows within 5–6°.

The final row requires a different interpretation. At the best-found FID solution, $\|\nabla\mathrm{FID}\|_2$ is much smaller and the estimated angles become more sensitive to perturbation scale. This is expected from the analysis above: once the target landscape is nearly stationary, gradient angle is no longer a stable discriminator between objectives. We therefore use the first three rows to characterize local search directions and the final row primarily to illustrate the loss of resolution near the optimum.

Finite-displacement landscapes. Gradient alignment describes infinitesimal behavior. To test whether the same distinction is visible over finite schedule changes, we evaluate all four objectives on two-dimensional slices of the $N = 4$ schedule space with $K = 2000$. Figure 1 shows one slice for four-step Euler, while Fig. 2 reports all three two-dimensional slices for four-step DPM-Solver++, obtained by fixing each interior node in turn.

Across the three slices, FID varies by 3.7–8.2×, whereas the state-space error P varies by only 1.9–3.2×. The FID regret at the control arg-min ranges from $+14.4\%$ to $+67.6\%$ for P and $+4.5\%$ to $+11.5\%$ for Q, compared with $+1.4\%$ to $+12.0\%$ for M_R.

The ordering is not uniform across all slices: M_R is closer to the FID arg-min on two of the three, while Q is closer on the third. This is consistent with the scope of our claim. We do not require M_R to dominate every alternative at every point of the global feasible space; rather, the proposed objective is intended to preserve the local FID geometry around practical schedules well enough to guide schedule refinement.

Figure 3 repeats the three slices for four-step Euler; its bottom row is the slice shown in Fig. 1. Here FID varies by 3.0–3.7× across the three slices, whereas P varies by only 1.2–1.8×. On every slice the arg-min of M_R coincides with the FID arg-min, while the FID regret at the control arg-min is $+13.6\%$ to $+30.6\%$ for P and $+2.8\%$ to $+7.1\%$ for Q.

Appendix D examines where this local agreement ceases to hold.

D Locality, Robustness, and Initialization

The proposed objective is intended as a local surrogate around practical sampling schedules rather than as a globally faithful replacement for FID. This appendix examines both sides of that statement. We first test whether the conclusions are stable within the practical low-FID region, including sensitivity to measurement noise, reference precision, and reasonable initialization. We then deliberately leave that region using random feasible schedules to characterize the boundary of the method.

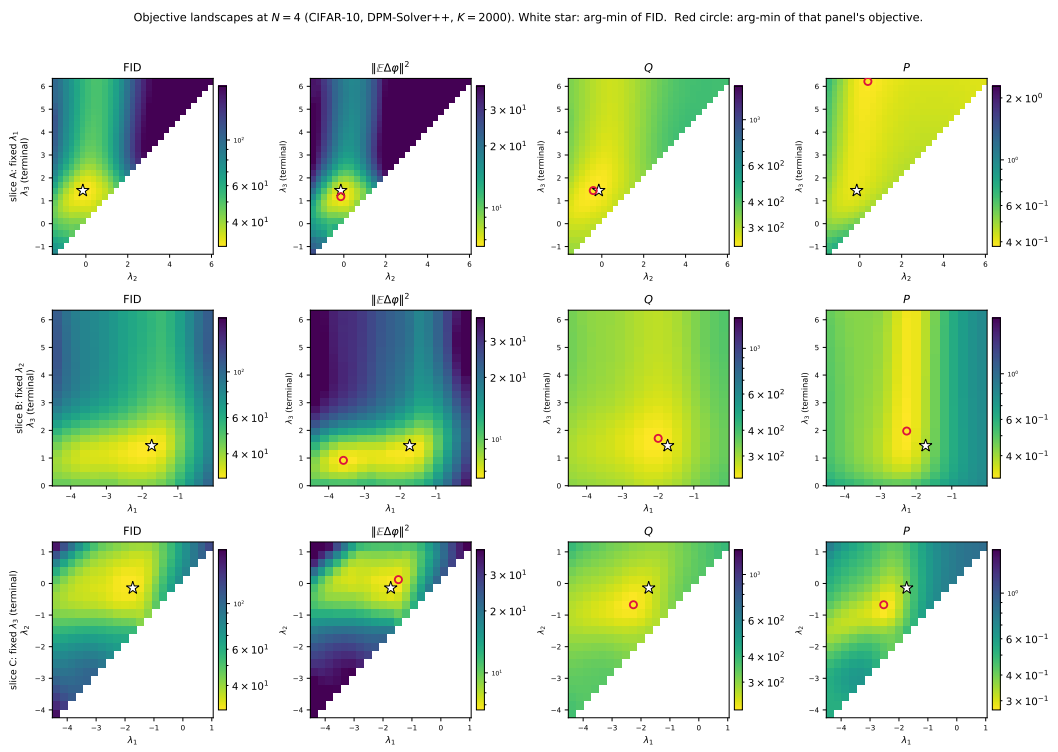

Figure 2: Three slices through the CIFAR-10 four-step DPM-Solver++ schedule space. Each row fixes one interior node and varies the remaining two. White star: FID arg-min. Red circle: arg-min of the objective shown in that panel.

D.1 Robustness Within the Practical Basin

We begin by examining the structure of the low-FID region on the complete 2300-schedule grid described in Appendix A. For each threshold t, we form the sublevel set

$$\left\{ h : \mathrm{FID}_K(h) \le (1+t) \min_{h'} \mathrm{FID}_K(h') \right\} \tag{67}$$

and count its connected components under nearest-neighbor adjacency in grid index space.

Table 10: Connected components of FID sublevel sets on the complete 2300-schedule CIFAR-10 four-step grid.

t	2%	5%	10%	20%	30%	40%	50%	60%	70%	100%
points	3	6	19	68	104	149	182	241	283	435
components	2	1	1	1	1	1	1	1	1	1

From the 5% sublevel onward, the low-FID region forms a single connected component. At 2%, only three grid points remain and two are not immediate neighbors; at this resolution, that observation is insufficient to establish a distinct second basin. Thus, within the accuracy of the grid, the practical low-FID region is better described as one broad basin than as several well-separated local minima.

High-sample remeasurement. Finite-K estimates can change the ordering of near-tied schedules. We therefore take the union of the top 25 candidates under FID_K and the top 25 under M_R, yielding 34 unique schedules, and remeasure them at $K = 10^4$.

On this set, the schedule selected by M_R is also the schedule with the lowest measured FID, giving zero arg-min regret. The same schedule remains the winner under a second sampling seed; the

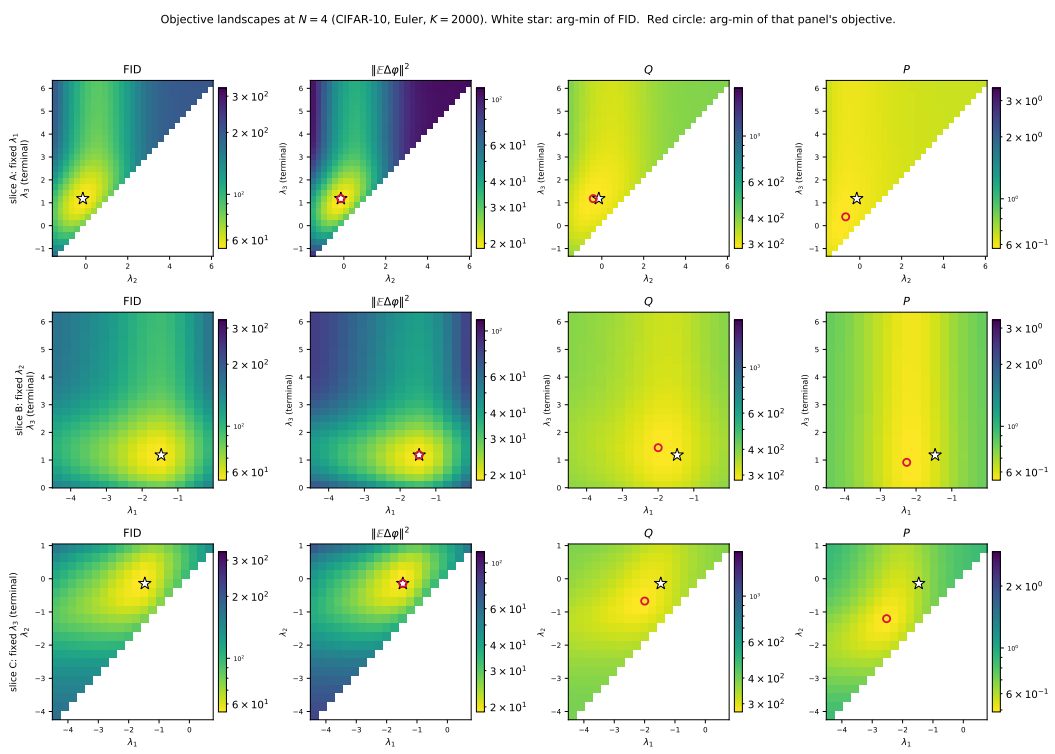

Figure 3: Three slices through the CIFAR-10 four-step Euler schedule space, in the same format as Fig. 2. Each row fixes one interior node and varies the remaining two. White star: FID arg-min. Red circle: arg-min of the objective shown in that panel.

margin to the runner-up is 3.26% under the first seed and 2.42% under the second. Across the two seeds, the schedule rankings have Spearman correlations $+0.992$ for FID and $+0.991$ for M_R.

The two control objectives behave differently. Both Q and P select the same schedule in this candidate set, whose FID is 15.75% above the best measured candidate. Because the candidate set itself is constructed using FID and M_R, this number is not an unbiased estimate of the controls' global regret; if anything, their best schedules outside this selected set are underrepresented. We therefore use this experiment only to test whether the proposed arg-min survives higher-sample remeasurement, not to estimate a population regret distribution.

Reference precision. The reference sampler is also finite-accuracy. We therefore rebuild R using 64, 128, and 256 Heun steps and repeat the candidate ranking. Increasing the integration budget moves the reference feature mean by 0.0154 from 64 to 128 steps and by a further 0.0050 from 128 to 256 steps, so the reference itself has not become numerically identical.

What matters for schedule selection, however, is highly stable: all three reference precisions select the same schedule with zero FID regret on the remeasured candidate set. Their rankings agree at Spearman $+0.9985$ between the 64-step reference and either refinement, and at $+1.0000$ between the 128- and 256-step references. The reference therefore need not be treated as an exact ODE solution; for the comparisons used here, its remaining numerical error is largely shared across candidate schedules and does not alter the observed selection.

Sensitivity to practical initialization. Our headline procedure starts from the standard schedule associated with each model. On CIFAR-10 we also constructed a clamp-based initialization whose $r_{\max}$ was selected from

$$\{1.1, 1.25, 1.5, 2, 3, 4, 5, 6\} \tag{68}$$

using FID. Because this particular initialization is therefore FID-informed, we test whether its selected hyperparameter is responsible for the eventual result.

Table 11: Sensitivity to practical initialization. The clamp setting $r_{\max} = 3$ was selected using its initial FID, while $r_{\max} = 2$ was not.

initialization	FID before refinement	FID after minimizing M_R
Karras $\rho = 7$	36.06	16.927
clamp $r_{\max} = 3$ (FID-selected)	20.17	16.954
clamp $r_{\max} = 2$	27.85	16.926

Table 12: Percentage gap above the best-found direct-FID solution across eight pixel-space settings. The final column counts settings whose gap falls within the 0.47–4.13% range of M_R. The last row reports the movement in FID when evaluating a fixed schedule on disjoint seeds.

Objective	Min.	Median	Max.	Within M_R range
M_R	0.47%	1.78%	4.13%	—
Q	9.60%	16.48%	41.6%	$0/8$
P	6.71%	59.8%	108%	$0/8$
Same schedule, disjoint seeds	0.27%	—	1.13%	—

The three initial schedules span a large range in FID, from 20.17 to 36.06, but refinement brings them to a much narrower range. In particular, the clamp setting that was *not* selected using FID ends 0.16% below the selected one. This difference is smaller than the 0.27–1.13% movement observed when a fixed schedule is evaluated on disjoint sampling seeds. The FID-informed choice of $r_{\max}$ therefore does not explain the final quality reached from this family of practical initializations.

For this reason, the main claims use the standard schedule rather than the FID-selected clamp initialization. The role of initialization is not to identify one specially tuned point, but to place the search inside the practical region in which the surrogate has the local geometry studied in Appendix C.

Final-protocol alternative starts. The final headline table uses the standard Karras initialization. The corresponding three-round FIDs for the alternative starts are 16.954 versus 16.927 on CIFAR-10, 18.175 versus 18.167 on FFHQ, and 13.082 versus 12.983 on AFHQv2 (alternative versus Karras in each case). The CIFAR-10 alternative is a clamp initialization whose $r_{\max} = 3$ was chosen using its initial FID; the FFHQ and AFHQv2 alternatives use Karras $\rho = 11$.

Distribution across pixel-space settings. Table 12 summarizes the gaps to the best-found direct-FID solutions for the eight pixel-space cells, using the Karras start for the proposed objective. The direct-FID solutions result from finite-sample, finite-budget optimization and are search references, not certified FID optima. The last row measures variation when a fixed schedule is evaluated using a disjoint set of $50\,000$ sampling seeds.

On independent seed sets, the single-round M_R and direct-FID schedules differ by $+0.56\%$ and $+0.79\%$ on CIFAR-10 and by $+1.21\%$ and $+0.34\%$ on AFHQv2. For the three-round schedules in the headline comparison, the corresponding gaps are $+2.17\%$ and $+1.86\%$ on CIFAR-10, and $+1.24\%$ and $+0.91\%$ on AFHQv2. The single-round pairs and the three-round AFHQv2 pair are comparable to the fixed-schedule seed variation, while the three-round CIFAR-10 gap exceeds it on both seed sets. These observations limit what can be inferred from small percentage differences in final FID.

Correlation loses resolution near the optimum. The complete grid also allows us to distinguish global association from fine-scale schedule selection. Table 13 reports Pearson and Spearman correlations between each surrogate and FID_K after progressively restricting the grid to better FID schedules.

Across the full grid, M_R tracks FID closely, with Pearson correlation $+0.994$ and Spearman correlation $+0.997$. The association weakens sharply as the range is restricted to already-good schedules. Among the 19 schedules within $1.1\times$ of the best measured FID, the correlations fall to $+0.533$ and $+0.479$.

Table 13: Correlation with FID_K on nested sublevel sets of the complete 2300-schedule CIFAR-10 grid. Entries are Pearson / Spearman.

range	points	max FID_K	M_R	Q	P
all	2300	425.1	$+0.994/+0.997$	$+0.932/+0.963$	$+0.464/+0.512$
$\mathrm{FID}_K \le 4.0 \times \min$	1252	121.5	$+0.981/+0.985$	$+0.819/+0.834$	$+0.385/+0.387$
$\mathrm{FID}_K \le 2.0 \times \min$	435	61.1	$+0.956/+0.954$	$+0.550/+0.527$	$+0.303/+0.253$
$\mathrm{FID}_K \le 1.5 \times \min$	182	45.7	$+0.877/+0.884$	$+0.456/+0.452$	$+0.430/+0.406$
$\mathrm{FID}_K \le 1.2 \times \min$	68	36.6	$+0.628/+0.615$	$+0.535/+0.553$	$+0.395/+0.418$
$\mathrm{FID}_K \le 1.1 \times \min$	19	33.4	$+0.533/+0.479$	$+0.262/+0.335$	$-0.107/-0.154$

Table 14: Changes to already reasonable schedules and their final FID. The final row gives the observed variation when remeasuring the same schedule with disjoint sets of 50 000 seeds.

Change	Size of change	Change in FID
Debiasing, nine runs at dispersion share $< 5\%$	schedule $\le 2.32\%$	-0.24 to $+0.07\%$
Debiasing, dispersion share 12.9%	schedule 9.69%	$+0.28\%$
Debiasing, dispersion share 44.7%	schedule 15.1%	-1.66%
Four condition-specific schedules	objective 16%	$+0.68\%$
Rounds two and three, nineteen parameters	objective -16.3%	-0.22%
Same schedule, disjoint seed set	—	0.27 to 1.13%

This loss of correlation is consistent with a flattening target landscape: once FID differences become small, fine-scale ordering becomes increasingly difficult even though the broader basin is well aligned. It also illustrates why correlation alone is not the quantity on which we base the method. On the 34 high-K candidates above, Q has the highest rank correlation with FID ($+0.723$), while M_R has a lower one ($+0.507$); nevertheless, their selected schedules have FID regrets of 15.75% and zero, respectively. A surrogate can therefore correlate better over a restricted candidate family while still selecting a worse minimum.

FID response to changes among good schedules. Table 14 brings together perturbations that move an already reasonable schedule or lower the achieved objective. The debiasing rows are the eleven controlled runs in Table 6; the per-condition row comes from Appendix E.2; and the nineteen-parameter row uses the decoupled CIFAR-10 iPNDM cell at ten NFE.

The nineteen-parameter row follows one search through its second and third rounds. They lower M_R by a further 16.3%, yet FID changes by only -0.22%, and by -0.39% and -0.23% on two further disjoint sets of 50 000 seeds, while the FID of each of the two schedules itself moves by 0.6–2.2% from the original seed set to either further set. The optimizer keeps finding deeper points of the objective after FID has stopped resolving them.

D.2 What Happens Outside the Practical Basin

Global grid minimizers. The 2300-schedule diagnostic grid also distinguishes local selection from the ordering across the enumerated feasible schedules. The grid minimizer of M_R has FID 13.2% above the grid's FID minimum, whereas the shared grid minimizer of Q and P is 8.7% above it. In the practical optimization cells, the median gaps to the best-found direct-FID solutions are instead 1.78%, 16.48%, and 59.8% for M_R, Q, and P. A better grid-wide minimizer is therefore not evidence of better schedule selection from the standard practical starts. These are minima on an enumerated grid, not proven global optima over the continuous feasible space.

The previous results establish robustness only within a practical initialization region. To test the local nature of the claim directly, we instead initialize from six randomly generated feasible schedules,

$$\vartheta \sim \mathcal{N}(0, I), \tag{69}$$

which induces a logistic-normal distribution over positive step lengths under the softmax parameterization. These starts are compared with the Karras and clamp schedules used above.

Table 15: Minimizing M_R from six random feasible initializations on CIFAR-10 with four-step sampling and $K = 2000$. Only two random starts recover the practical solution region.

initialization	resulting σ (objective)	outcome
clamp	$[80,\ 5.17,\ 1.09,\ 0.276]$ (6.265)	practical reference
Karras	$[80,\ 5.17,\ 1.08,\ 0.272]$ (6.264)	practical reference
seed 22	$[80,\ 5.16,\ 1.09,\ 0.275]$ (6.264)	recovers
seed 12	$[80,\ 6.85,\ 0.97,\ 0.314]$ (6.378)	recovers
seed 21	$[80,\ \mathbf{33.72},\ 1.03,\ 0.398]$ $(\mathbf{6.249})$	lower objective, wrong region
seed 13	$[80,\ 3.53,\ \mathbf{3.51},\ 0.349]$ (11.058)	interior collapse
seed 11	$[80,\ 3.76,\ 0.57,\ \mathbf{0.0029}]$ (14.594)	terminal collapse
seed 23	$[80,\ 3.83,\ 0.57,\ \mathbf{0.0023}]$ (14.576)	terminal collapse

Two of the six random starts return to the practical region. Three instead collapse one or more schedule intervals: two place the terminal node almost at $\sigma_{\min}$, while one nearly merges two interior nodes. These failures also have substantially larger surrogate values than the practical solutions.

The remaining run is more important. Seed 21 reaches an objective value of 6.249, slightly *lower* than either practical reference, yet places the first interior node at $\sigma_1 = 33.72$ rather than near 5. A $K = 10^4$ remeasurement places the corresponding region 25.3% above the best measured FID. Thus failure outside the practical basin cannot in general be detected simply by checking whether the optimized surrogate value is large.

The behavior also cannot be reduced to a finite-sample artifact. At $K = 10^4$, the global ordering changes enough that this region is no longer the measured minimum of M_R among the remeasured candidates. Nevertheless, the grid point nearest the random-start solution still ranks fourth under M_R while ranking thirtieth of 34 under FID. Increasing the sample count therefore removes the particular low-K arg-min without making the surrogate globally faithful to FID in that region.

This experiment defines the scope of the method. The proposed objective is not claimed to support unconstrained global search over all feasible schedules. A standard practical discretization is part of the procedure because it places the optimizer in the region where the local relations analyzed in Appendix C are informative. Within that region, the selection is stable to reasonable initialization, sampling seed, and reference precision; outside it, both degenerate schedules and genuinely low-surrogate but poor-FID schedules can occur.

E Additional Evaluations and Protocol Checks

This appendix collects evaluations that are important for interpreting the main comparisons but are not part of the core objective analysis. We first document the Stable Diffusion evaluation pipeline and its cross-pipeline discrepancy, then test whether condition-specific schedules are beneficial, verify the published LD3 and GITS baselines used in our comparison, and finally report two additional sampling-protocol ablations.

E.1 Stable Diffusion Protocol and Cross-Pipeline Check

The Stable Diffusion results in Table 1 are intended as an internal comparison: all objectives are optimized and evaluated using the same model, sampler implementation, prompt set, and FID pipeline. We nevertheless compare our absolute values with those reported in prior work and document the discrepancy here.

Optimized schedules. At four evaluations, the endpoints are fixed at $\sigma_0 = 14.6146$ and $\sigma_4 = 0.0292$, leaving three interior nodes. For each optimized schedule, we report the mean relative displacement of these nodes from the schedule obtained by direct FID optimization,

$$\frac{1}{3}\sum_{i=1}^{3}\frac{|\sigma_i - \sigma_i^{\mathrm{FID}}|}{\sigma_i^{\mathrm{FID}}}. \tag{70}$$

Table 16: Four-step Stable Diffusion schedules under the common evaluation pipeline. Distance is the mean relative interior-node displacement from the schedule returned by direct FID optimization.

schedule	σ_1	σ_2	σ_3	distance
time-uniform baseline	4.082	1.618	0.696	18.8%
direct FID	4.186	1.519	0.472	—
M_R	3.879	1.350	0.420	9.9%
Q	4.510	1.727	0.527	11.0%
P	3.744	1.301	0.409	12.8%

The two controls deviate from the direct-FID schedule in opposite directions: all three interior nodes returned by Q lie above their direct-FID counterparts, whereas those returned by P lie below them. The proposed objective also lies below the direct-FID schedule in this cell, but has the smallest mean node displacement of the three surrogate objectives.

Comparison with the published absolute FID. Tong et al. (2025) report Stable Diffusion 1.5 results at CFG 7.5 using COCO FID@30k, with values 17.76, 14.41, 13.86, and 13.76 at four through seven evaluations, obtained with second-order iPNDM and time-uniform steps; the iPNDM rows below use third order. Using our 512×512 evaluation pipeline, we obtain the values in Table 17. Because our feature extractor uses a different input range (Appendix A), these values are not on the same scale as the published ones.

Table 17: Stable Diffusion 1.5 absolute FID under combinations of solver and discretization in our pipeline, compared with the values reported by Tong et al. (2025).

	4 NFE	5 NFE	6 NFE	7 NFE
iPNDM + Karras $\rho = 7$	26.935	18.430	13.828	11.523
iPNDM + time-uniform	26.965	17.056	13.296	12.189
DPM-Solver++ + Karras $\rho = 7$	29.640	16.253	12.372	11.529
DPM-Solver++ + time-uniform	23.039	15.465	13.043	12.169
reported by Tong et al. (2025)	17.76	14.41	13.86	13.76

This sweep was performed separately from the optimization runs in Table 1. Its DPM-Solver++ time-uniform value at four evaluations is 23.039, compared with 23.093 for the corresponding baseline in the main experiment, a difference of 0.23%. As discussed in Appendix A, Stable Diffusion search and sampling are sensitive to execution configuration, so we report each experiment from its own internally consistent run rather than combining numbers across runs.

We tested several possible explanations for the larger discrepancy from the published few-step values. Changing the FID resolution from 512 to 256 moves the measured values by only 1–4%, which does not account for the largest gaps. A one-evaluation shift in the NFE convention also leaves a substantial residual. Extending our own curve to 10 and 25 evaluations gives FIDs of 11.400 and 12.781, indicating that the disagreement is concentrated mainly in the few-step portion rather than appearing as a constant offset over the entire curve. Changing solver and discretization narrows the disagreement in some cells but does not produce a consistent match across all four budgets; for example, iPNDM with Karras spacing is within 0.23% of the reported value at six evaluations despite being substantially different at lower NFE.

Apart from the known difference in the feature extractor's input range, these checks do not isolate a unique cause for the remaining cross-pipeline difference. We therefore treat the published absolute values only as an external protocol check. The conclusions in the main paper rely on comparisons among objectives optimized and evaluated within the same fixed pipeline.

E.2 Condition-Specific Scheduling

One practical distinction between the proposed reference-based objective and direct data FID is that M_R does not require condition-specific real-data statistics. We therefore test whether this permits

useful condition-dependent schedules on Stable Diffusion, the text-conditional model in our experiments.

COCO captions are embedded using the model's text encoder and partitioned into C clusters using spherical k-means. Each cluster is assigned its own schedule by minimizing Eq. 14 on prompts from that cluster. The $30\,000$ evaluation prompts are then assigned to the nearest cluster, sampled using the corresponding schedule, and pooled before computing a single final FID.

All comparison arms use the same set of $30\,000$ prompt–latent pairs. They differ only in whether generation uses the time-uniform baseline, one schedule optimized over all prompts, or C independently optimized cluster schedules. The compared searches use the same hardware and batch configuration; an earlier aggregate run under a different device configuration is excluded, as motivated by Appendix A.

As the number of clusters increases, reliable condition-specific FID optimization becomes increasingly sample-limited. For example, at $C = 64$ each cluster contains approximately 469 real images, far fewer than the 2048 Inception feature dimensions, yielding a severely rank-deficient empirical covariance. The reference-based objective remains well-defined without estimating these condition-specific real-data covariances.

We consider two resource allocations. In the equal-cost setting,

$$C \times \text{budget} \times \frac{K}{C} \tag{71}$$

is held fixed, so increasing C reduces the number of optimization samples per cluster. The equal-sample experiment instead uses 2000 latent samples for each of four clusters, removing this finite-sample disadvantage.

Table 18: Condition-specific schedule optimization on Stable Diffusion. The final column is positive when the per-condition schedules obtain lower pooled FID than the aggregate schedule.

setting	samples per cluster	aggregate	per-condition	change
equal cost, $C = 4$	500	15.576	15.583	-0.05%
equal sample, $C = 4$	2000	15.576	15.681	-0.68%
equal cost, $C = 16$	125	15.576	16.134	-3.59%
equal cost, $C = 64$	31	15.576	17.392	-11.66%

No tested partition improves over the aggregate schedule. The degradation at larger C under equal cost is consistent with the rapidly decreasing sample count per cluster. More importantly, the equal-sample $C = 4$ experiment removes this estimation disadvantage and still shows no benefit from condition-specific scheduling.

Earlier low-sample comparisons were also sensitive at approximately the one-percentage-point level to the evaluation sample set: repeating one earlier $C = 4$ comparison on a disjoint set of $30\,000$ prompt–latent pairs changed its paired difference from $+2.83\%$ to $+1.80\%$. The current equal-sample result is therefore interpreted as a null rather than evidence for a small condition-specific gain.

Cross-condition transfer. We additionally evaluate each learned cluster schedule on held-out latents from every cluster. If the optimal schedule depended strongly on caption condition, each cluster's own schedule should preferentially perform best on that cluster. Under equal-sample optimization, the diagonal schedule is best in only one of four rows; under equal-cost optimization, it is best in none. Moreover, in the equal-cost experiment each cluster's own schedule is 2.0–4.0% worse on its condition than the aggregate schedule.

Increasing the number of optimization samples from 500 to 2000 per cluster actually increases the separation among the resulting schedules: their three interior nodes differ by 4.9–7.7% in the lower-sample case and 9.3–11.8% in the higher-sample case. Thus the convergence toward different schedules cannot be explained simply by larger sampling noise at smaller K.

At the same time, the downstream metric remains comparatively insensitive: for one cluster, the objective separates the five candidate schedules by approximately 16%, while replacing the aggregate

schedule with all four cluster-specific schedules changes pooled FID by only 0.68%. The experiment therefore provides no evidence that the additional conditioning granularity is useful in this setting; instead, it is consistent with the near-optimal schedule region being sufficiently flat that different searches can return visibly different schedules with little change in final FID.

E.3 External Baselines and Pipeline Verification

The LD3 comparison in Table 2 requires identifying which discretizations correspond to the published baselines. The naming in the released results does not directly specify the σ-space formulas, so we sweep the standard choices using iPNDM order 3, four evaluations, and CIFAR-10 at $K = 10^4$.

Table 19: Candidate discretizations used to identify the published LD3 baselines.

discretization	$\mathrm{FID}_K(10^4)$	σ
Karras $\rho = 3$	55.230	[80, 34.75, 10.90, 1.609, 0.002]
Karras $\rho = 5$	29.788	[80, 23.10, 4.408, 0.364, 0.002]
Karras $\rho = 7$	31.325	[80, 17.53, 2.515, 0.170, 0.002]
Karras $\rho = 10$	33.929	[80, 13.44, 1.531, 0.095, 0.002]
uniform in λ	36.847	[80, 5.657, 0.400, 0.0283, 0.002]
uniform in σ	267.32	[80, 60.00, 40.00, 20.00, 0.002]
uniform in $\sqrt{\sigma}$	141.90	[80, 45.15, 20.20, 5.151, 0.002]

The identification is determined by the complete pattern rather than by a single matched number. The sweep reproduces the four published baseline families in the same order and across a wide numerical range: uniform in σ corresponds to the reported time-uniform baseline (267.32 versus 266.26), uniform in $\sqrt{\sigma}$ to time-quadratic (141.90 versus 139.72), Karras $\rho = 7$ to time-EDM (31.325 versus 29.78), and uniform in $\lambda = -\log\sigma$ to time-logSNR (36.847 versus 35.04).

Measured using the same $50\,000$-sample protocol as the published result, our uniform-in-λ baseline gives FID 35.153 versus 35.04, a difference of 0.32%. Two further checks use the released LD3 paired schedules directly: at ten evaluations our pipeline obtains 2.369 versus the reported 2.38 (0.46% difference), while at four evaluations it obtains 9.632 versus 9.31 (3.5% difference).

These checks are why Table 2 executes the released schedules in our own pipeline rather than mixing their reported absolute values with ours. They also matter for the baseline itself: using Karras $\rho = 7$ instead of the identified uniform-in-λ discretization would give 29.489 at 50k and would make the apparent cross-pipeline comparison substantially different. We therefore use the discretization identified from the complete baseline pattern rather than selecting the numerically closest individual entry.

GITS schedules. We run the CIFAR-10 schedules printed by Chen et al. (2024) in our pipeline. Both the GITS schedules and our single-sequence schedules are evaluated with the unmodified iPNDM code released with GITS, so the GITS schedules are run with their authors' own solver implementation. GITS reports its results with fourth-order iPNDM, whereas Table 2 follows LD3 in using third order. With fourth order, our pipeline reproduces the reported values to within 0.02: Karras $\rho = 7$ gives 24.812 and 2.767 (reported 24.82 and 2.77), and GITS gives 15.081 and 2.487 (reported 15.10 and 2.49), at four and ten NFE. Evaluated with fourth order without re-optimization, the schedules found by minimizing $\|\mathbb{E}\Delta\varphi\|^2$ reach 13.759 and 2.324, so they remain below GITS under both solver orders.

Other schedule methods. Several methods in Section 2 do not provide schedules for our settings, so a matched comparison would test a reimplementation rather than the published method. AYS (Sabour et al., 2024) releases numerical schedules only for four text-to-image and video models at ten steps; the ten-step Stable Diffusion 1.5 schedule lies outside our four-to-seven-NFE range, its CIFAR-10 and FFHQ schedules appear only as plots, and its optimization code is not public. HSO (Zhu et al., 2026) releases code but not its final schedules, and evaluates Stable Diffusion 2.1 and PixArt-α with UniPC and DDIM. The released code of Xue et al. (2024) computes schedules for third-order DPM-Solver++ and UniPC, and its CIFAR-10 and ImageNet-64 experiments use

checkpoints different from ours. DDSS (Watson et al., 2022) learns a different sampler family on DDPM checkpoints, and its paper provides neither code nor learned parameters; we instead evaluate its objective under our protocol (Appendix C.1).

Interpretation of the released schedules. The LD3 artifacts provide separate denoiser-query and solver-step time sequences. At four NFE, using either released sequence on its own gives FID 15.03 or 15.33, whereas using the released pair gives 9.632 in our pipeline. At ten NFE, corresponding nodes in the two sequences differ by less than 0.9%, yet the paired schedule improves FID from 3.351 to 2.369. The comparison therefore uses both released sequences rather than treating either as a single-sequence baseline.

We evaluate the released LD3 schedules but do not retrain LD3 with our reference, optimization samples, or measurement protocol. The reported comparison covers one checkpoint, one solver, and two NFE budgets. It shows that the reference-mean objective can exploit this two-sequence parameterization and improve the released schedules in a matched evaluation pipeline; it does not bound the outcome of retraining LD3 under our protocol. LD3 pays a training cost and backpropagates through the solver, whereas our method pays a derivative-free search cost per setting, so these results do not establish a general cost ranking.

E.4 Additional Ablations

Adding the Bures covariance term. To assess the effect of optimizing the full reference FID, we replace M_R by $\mathrm{FID}(S, R)$, keeping the reference, initialization protocol, schedule parameterization, latent set, and search allowance matched within each of four cells. Relative to optimizing M_R, the resulting data FID is 1.62% higher for four-step DPM-Solver++ (three free parameters), 0.67% higher for four-step iPNDM (three), 43.7% higher for four-step decoupled iPNDM (seven), and 1.87% higher for ten-step decoupled iPNDM (nineteen). The seven-parameter cell attains FID 11.633 with $\mathrm{FID}(S, R)$ versus 8.093 with M_R. Its poor result does not imply that the covariance term is intrinsically harmful to the population target: remeasured at $K = 10^4$, the schedule found by M_R also has lower $\mathrm{FID}(S, R)$ (10.98 versus 13.23) than the schedule returned by the direct $\mathrm{FID}(S, R)$ search. The finite-sample covariance distortions in Appendix B coincide with a harder search in these experiments, but these tests do not identify estimator bias as its sole cause.

Direct FID search in the two-sequence space. At four NFE, the direct-FID search in the two-sequence space starts, like every other search in Table 2, from its own single-sequence optimum, and ends at FID 13.767. Over three rounds its own objective, FID_K at $K = 2000$, falls only from 27.40 to 27.00, and the third round stops on its own tolerance after 610 evaluations. On the same latents, the schedule found by M_R already has $\mathrm{FID}_K = 21.36$, and at $K = 10^4$ it has 10.16 against 15.64 for the direct-FID schedule. Restarted at that schedule, direct-FID optimization reaches FID 7.859, the best-found value in this space; the 8.093 of M_R lies 2.98% above it. This is evidence of sensitivity to the starting point in this finite-budget search, not evidence that a surrogate can beat the minimum of its own target.

In the ten-NFE cell, optimizing both sequences reaches FID 2.201 against 2.275 for the query-only restriction, with a 47.5% lower value of M_R; on two further disjoint sets of $50\,000$ seeds the full search remains 3.6% and 3.3% lower in FID. At four NFE, the query-only search already reaches FID 8.140, compared with 8.093 when both sequences are optimized. Their objective values move from 4.497 at the single-sequence starting schedule to 2.346 after query-only fitting and 2.212 after jointly fitting both sequences (versus 2.944 for the released LD3 pair). The search budgets for each parameter count are listed in Appendix A.

Finally, the remaining ablations concern two implementation choices not analyzed elsewhere: the number of samples used to optimize the finite-K objective and the convention used for the final denoising evaluation.

Sample count K. We repeat the four-step CIFAR-10 optimization from the Karras initialization using

$$K \in \{500, 1000, 2000\}. \tag{72}$$

Under direct FID optimization, the resulting schedules obtain final FIDs

$$17.039, \qquad 16.577, \qquad 16.594, \tag{73}$$

while minimizing the feature-mean objective gives

$$17.649, \qquad 16.763, \qquad 16.927, \tag{74}$$

respectively.

Direct FID optimization is at least as good at every tested K, including the smallest sample count, where the empirical 2048×2048 covariance is strongly rank-deficient. Thus the experiments do not support the hypothesis that the mean-only objective gains a particular optimization advantage at low sample counts.

The optimized schedule also changes non-monotonically with K. Relative to the $K = 2000$ solution, the $K = 500$ schedule differs by 13.7% per node and the $K = 1000$ schedule by 18.1% per node, while the lowest FID among the three occurs at $K = 1000$. We therefore use the same $K = 2000$ optimization sample count throughout the main comparisons rather than selecting it separately for each cell.

Terminal-denoise convention. At a fixed budget of B network evaluations, two conventions can be used. The sampler can take B solver evaluations and stop at $\sigma_{\min}$, or use $B-1$ solver evaluations and reserve the final network evaluation for an explicit denoising step to zero.

On the Karras $\rho = 7$ baseline at $K = 10^4$, stopping at $\sigma_{\min}$ gives lower FID by 37.3%, 18.5%, and 4.1% at budgets $B = 5$, 9, and 17, respectively. Across the five schedules tested, appending the explicit terminal denoising call worsens FID by 1.8–2.5% while consuming one additional network evaluation.

For the EDM models used here, $\sigma_{\min} = 0.002$, corresponding to residual noise of approximately 0.255 in $\{0, \ldots, 255\}$ image units. Based on this controlled comparison, all EDM experiments in the paper use the convention that spends the available evaluations on integration to $\sigma_{\min}$ and does not append an additional denoising evaluation.