ENCORE: TRAINING-FREE FEW-STEP DIFFUSION SAMPLING WITH TERMINAL MOMENT MATCHING

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion models achieve state-of-the-art quality in visual and audio generation, but sampling requires many evaluations of an expensive denoising network, and fewer steps severely degrade sample quality. Existing approaches either design fast ODE solvers or rely on auxiliary trained networks for post-hoc refinement. We show that even with optimized ODE solvers, few-step generation deviates systematically from many-step runs in two distributional ways: (i) a loss of power across all frequency bands and (ii) a mean shift in a feature space. Because both are distribution-level properties, they cannot be precisely measured or corrected along a single sampling trajectory: the large per-sample variance masks the overall mean shift and distorts frequency measurements. To address this issue, we propose **ENCORE**, a training-free, post-hoc correction that reuses the differences between denoising outputs at consecutive steps, which the sampler otherwise discards, and thus requires no additional network evaluation. In a spectral-then-mean (S2M) pipeline, ENCORE first restores the per-band power via magnitude scaling and then realigns the sample centroid in a pre-trained feature space by ridge regression over these differences. The two stages match a second moment per frequency band and a first moment in feature space, after the last step. Because it operates after sampling, ENCORE is plug-and-play and yields further gains on top of sampler-level optimizations. Evaluations across pixel-space, image latent, and audio latent diffusion models demonstrate that ENCORE improves sample quality with negligible overhead.

1 INTRODUCTION

Diffusion models (Ho et al., 2020; Song et al., 2021b) have emerged as the prevailing framework for generative modeling, achieving state-of-the-art performance across unconditional and class-conditional image generation (Karras et al., 2022; Peebles & Xie, 2023), text-to-image synthesis (Rombach et al., 2022), and audio generation (Liu et al., 2024). However, sampling from a trained model requires numerically solving a differential equation, where each iteration involves an expensive evaluation of a large denoising network. Because high-fidelity generation traditionally demands hundreds to thousands of function evaluations (NFEs) (Ho et al., 2020; Song et al., 2021b), slow inference remains the primary bottleneck for real-world applications. To accelerate sampling, most existing efforts focus on designing fast ODE solvers (Lu et al., 2022b; Zhao et al., 2023; Zhang & Chen, 2023; Liu et al., 2022) that reduce discretization errors per step, on placing the steps along a better schedule (Chen et al., 2025), or on reusing intermediate denoising predictions (Zhang et al., 2023). Nevertheless, generating high-quality samples in the few-step regime (e.g., 4–10 steps) remains a major challenge, as such short runs often fail to recover the fine details present in many-step samples (Figure 1).

In this work, we ask a fundamental question: *Even if an ODE solver minimizes discretization errors along the trajectory, what deficits remain at the end of a few-step run?* We investigate this by comparing a few-step run with a much longer run of the same sampler (the *target*), starting from identical noise seeds. As shown in Figure 2, we identify two systematic, distribution-level discrepancies that a few-step set exhibits relative to its target: (i) a severe power loss across all frequency bands, and (ii) a mean shift in a pre-trained feature space. Both are statistics of the sample set (the band power ratio is an average over the set, and the feature mean is its centroid), so neither can be evaluated or corrected from within a single sampling trajectory.

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.



Figure 1: **Qualitative comparison on ImageNet-256 with DiT-XL/2.** Each column shows one class, and all rows share the same seeds. With four network evaluations (NFE), ENCORE (middle) restores fine structural details lost by the baseline (top), as highlighted in the zoomed crops at right. The 64-step run (bottom) serves as a reference.

Existing acceleration methods operate on orthogonal axes. Trajectory-level optimizations (Lu et al., 2022b; Zhao et al., 2023; Zhang et al., 2023; Choi et al., 2025) focus on minimizing discretization errors along individual sampling paths. Because only a single trajectory is observable at each iteration, a quantity defined across samples cannot be measured from within such a method. Making each path more accurate can still move the set, and methods that fit their updates against a long run do exactly that. What has not been optimized is the terminal moments of the set themselves: under every solver we tested, the band power and the feature mean of a few-step set still fall short of the target. On the other hand, existing post-hoc refinement approaches (Peng & Gao, 2026) attempt to refine samples after generation, but they require training an auxiliary refiner network and incur one additional NFE on top of the sampler.

To bridge this gap, we propose **ENCORE**, a training-free, post-hoc correction framework applied after the final sampling step. Our key insight is that the information required to correct this distributional degradation is already computed by the sampler during generation and then discarded. At every step, the denoiser predicts the clean data, and we show that the differences between these discarded denoising outputs recover a large portion of the feature-space mean shift. Reusing these differences at no additional NFE, ENCORE resolves both deficits in a spectral-then-mean (S2M) pipeline. The first stage (Stage **S**) rescales only the frequency magnitudes to restore spectral power, leaving the sampler's phase untouched; the second stage (Stage **M**) adds a closed-form, ridge-regularized combination of these differences that realigns the centroid in a pre-trained feature space (e.g., Inception). Because sampling trajectories of the same model share a similar low-dimensional structure (Chen et al., 2024), both stages are calibrated once per setting on 1,000 synthetic pairs drawn by the model itself, requiring no real data.

Figure 2 shows that, on CIFAR-10, ENCORE moves the few-step samples toward the target in both properties. Stage **S** matches the band power, and Stage **M** then adds these differences on top, so the corrected samples end up above the target level while remaining closer to it than the baseline in every band. Their feature centroid also moves toward the target.

Our contributions are as follows:

- We identify two distribution-level discrepancies that few-step sampling leaves behind (spectral power decay and a feature-space mean shift) and show that the latter can be reduced using the differences between the denoising predictions that the sampler discards.
- Building on this observation, we develop ENCORE, a closed-form correction applied once after the last step. It restores band power and realigns the feature centroid with no additional network evaluation, no training, and no real data.
- We demonstrate that, across pixel-space, image latent, and audio latent diffusion models, the correction improves sample quality and composes with existing fast samplers.

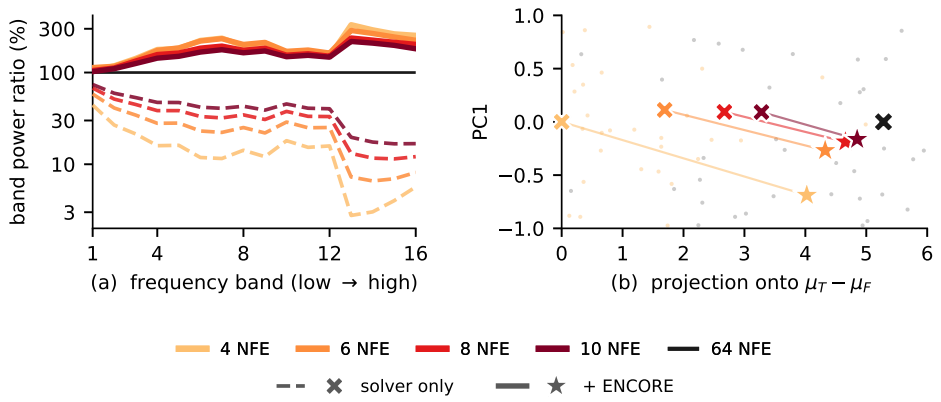


Figure 2: **Distributional properties of few-step sampling and the ENCORE correction.** Results use the Euler sampler on CIFAR-10 at NFE $\in \{4, 6, 8, 10\}$, against a 64-step target. **(a)** Band power ratio to the target in each frequency band. Dashed lines denote the baseline solver and solid lines denote the same solver with ENCORE. **(b)** Inception features projected onto $\mu_T - \mu_F$, the direction between the two centroids, and onto the leading orthogonal principal component (PC1). Dots denote individual samples. Each NFE has its own centroid ($\times$), and ENCORE ($\star$) moves it toward the target. The displacement shrinks as the number of steps grows.

2 Related Work

Few-step ODE solvers and distillation. Accelerating diffusion sampling generally involves designing faster numerical ODE solvers or retraining the model weights. High-order solvers (Lu et al., 2022b; Zhao et al., 2023; Zhang & Chen, 2023; Liu et al., 2022) reduce the local truncation error of a step, and STORK (Tan et al., 2026) stabilizes the update at large step sizes. Complementary strategies optimize time-step schedules (Chen et al., 2025), fit update coefficients (Zheng et al., 2023; Zhang et al., 2024; Frankel et al., 2025), or train a lightweight predictor for the intermediate time step (Zhou et al., 2024). In parallel, training-based methods such as progressive distillation (Salimans & Ho, 2022) and consistency models (Song et al., 2023) modify model parameters to enable generation in very few steps. Because these methods operate along individual trajectories or retrain the model, they do not address the distribution-level discrepancies that ENCORE corrects.

Reusing intermediate predictions and momentum. To reduce discretization errors without additional NFEs, another line of work reuses the predictions the sampler has already computed. For instance, LA-DPM (Zhang et al., 2023) applies lookahead extrapolation to clean-image estimates, RX-DPM (Choi et al., 2025) leverages Richardson extrapolation across steps, and PFDiff (Wang et al., 2025a) reuses past scores to construct a "springboard" state before applying Nesterov-inspired foresight updates. Alternatively, methods such as DC-Solver (Zhao et al., 2024) and PAS (Wang et al., 2025b) precompute trajectory adjustments offline, such as step compensation ratios or PCA-based coordinate offsets. Nevertheless, because these corrections operate within individual sampling paths, they address per-sample trajectory errors rather than distribution-level discrepancies.

Post-hoc distributional correction. Beyond trajectory-level solver design, post-hoc methods adjust generated sample statistics directly. Guidance rescaling (Lin et al., 2024) mitigates overexposure under classifier-free guidance, whereas covariance-aware sampling (Schioppa & Salimans, 2026) models the reverse-process covariance in the frequency domain at the cost of one extra Jacobian-vector product per step. Similarly, colored noise sampling (Davidson et al., 2026) shapes the noise spectrum but relies on 50–250 steps. For sample refinement, BFR (Peng & Gao, 2026) trains an auxiliary flow network that adds one extra NFE. Alternatively, optimizing samples through a differentiable sampler (Watson et al., 2022) risks overfitting to specific evaluation metrics without genuine perceptual gains (Song & Dhariwal, 2024; Kynkäänniemi et al., 2023). While ENCORE also targets a distribution-level shift, it avoids these additional costs: its band gains and coefficients are solved in closed form from model-generated pairs and applied once after the final step, requiring no extra NFEs, no real data, and no auxiliary networks.

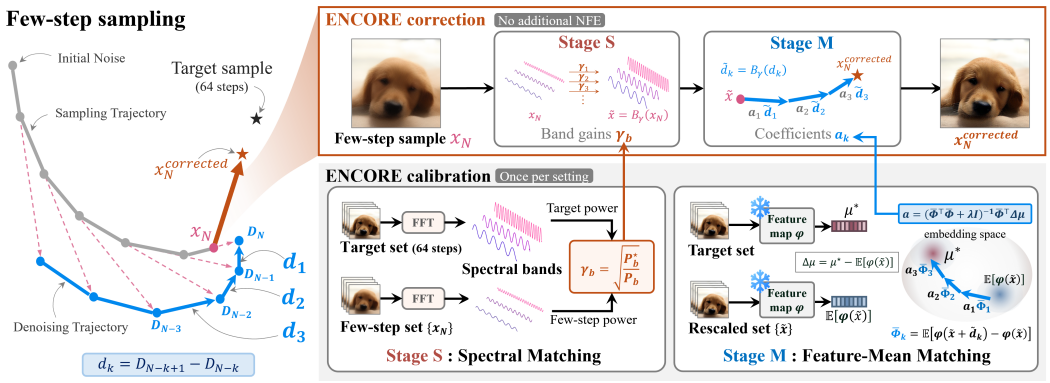


Figure 3: **Overview of ENCORE. Left:** A few-step run produces the sampling trajectory (gray) and the denoising outputs D_i (blue). ENCORE reuses their last differences d_k. **Top right:** At inference, Stage **S** rescales the band power of x_N with the band gains γ_b, and Stage **M** adds the rescaled differences $\tilde{d}_k$ with the coefficients a_k. **Bottom right:** Calibration runs once per setting on self-generated samples: γ_b from the ratio of target to few-step band power, and a_k by ridge regression on the feature-mean displacement toward the target mean $\mu^\star$. Shown with K=3.

3 Preliminaries

Diffusion sampling. Generating data from a diffusion model amounts to integrating the probability-flow ODE (Lu et al., 2022a) backward from noise, using a network $\epsilon_\theta(x_t, t)$ to approximate the intractable scaled score. Because the ODE has no closed-form solution, solvers discretize the trajectory over noise levels $\sigma_{t_1} > \cdots > \sigma_{t_{N+1}} = \sigma_{\min}$: N network calls cut the noise range into N intervals. At each step $i = 1, \ldots, N$ the solver evaluates the network once to form the current denoising output,

$$D_i = \frac{x_{t_i} - \sigma_{t_i}\, \epsilon_\theta(x_{t_i}, t_i)}{\alpha_{t_i}}, \tag{1}$$

where α_t and σ_t define the noise schedule. Models parameterized by data prediction, such as the EDM checkpoints we correct in pixel space (Karras et al., 2022), output D_i directly. For multistep integrators, several recent D_i are reused to advance the trajectory, but all of them are discarded once the terminal state is reached; we write x_N for that terminal state $x_{t_{N+1}}$.

Terminology and setup. We designate the solver targeted for correction, with network weights strictly frozen, as the *few-step sampler*. The objective is to recover the *target distribution*: the empirical output of the *target sampler*, the identical model and solver executed at 64 steps, which approximates the long-run trajectory rather than ground-truth data (Appendix B). *Calibration* is our one-time offline procedure that determines the correction parameters, the band gains γ_b and the coefficients a_k. Both are derived via gradient-free, closed-form solutions on a small generated subset, while the history depth K is chosen by scanning every candidate against the feature-mean discrepancy on a held-out half. Sampling cost is measured in NFEs. The computational cost of our method is reported separately in wall-clock time: as a one-time offline calibration duration per setting, and as a per-sample overhead during generation.

4 Method

An N-step run produces N denoising outputs $D_1, \ldots, D_N$. Rather than discarding these intermediate outputs, ENCORE reuses the final K consecutive differences as correction directions, $d_k = D_{N-k+1} - D_{N-k}$ for $k = 1, \ldots, K$, where $k = 1$ is the last difference $D_N - D_{N-1}$. Individual denoising outputs do not serve as directions on their own: the last update moves x_N to within a factor $\sigma_{\min}/\sigma_{t_N}$ of D_N for a first-order step, and to within the residual noise of scale $\sigma_{\min}$ in general, so D_N lies along a direction the sample has already traversed. We correct the final sample

Algorithm 1 ENCORE calibration (offline, once per setting)

Require: few-step sampler at NFE N; target sampler at 64 steps; 1,000 calibration seeds; frequency bands $b = 1, \dots, 16$; feature map φ; ridge rule λ

for all calibration seeds s **do**
  $(x_N,\ D_{1:N}) \leftarrow \textsc{FewStep}(s;\ N),\ \ x^\star \leftarrow \textsc{Target}(s;\ 64)$ ▷ same seed
end for
Partition the pairs $\{(x_N, D_{1:N}, x^\star)\}$ into two halves $\mathcal{F}$ and $\mathcal{H}$ ▷ $\mathcal{F}$ fits γ, a; $\mathcal{H}$ selects K
$\gamma_b \leftarrow \sqrt{P_b^\star / P_b}$ for $b = 1, \dots, 16$, with $P_b, P_b^\star$ the band powers over $\mathcal{F}$
$\tilde{x} \leftarrow B_\gamma(x_N),\ \ \tilde{d}_k \leftarrow B_\gamma(D_{N-k+1} - D_{N-k})$ on both halves
$\mu_{\mathcal{F}}^\star \leftarrow \mathbb{E}_{\mathcal{F}}[\varphi(x^\star)],\ \ \mu_{\mathcal{H}}^\star \leftarrow \mathbb{E}_{\mathcal{H}}[\varphi(x^\star)]$
$\bar{\Phi}_k \leftarrow \mathbb{E}_{\mathcal{F}}[\varphi(\tilde{x} + \tilde{d}_k) - \varphi(\tilde{x})],\ \ \Delta\mu \leftarrow \mu_{\mathcal{F}}^\star - \mathbb{E}_{\mathcal{F}}[\varphi(\tilde{x})]$
for $K = 1, \dots, N-1$ **do**
  $a^{(K)} \leftarrow (\bar{\Phi}_{1:K}^\top \bar{\Phi}_{1:K} + \lambda I)^{-1} \bar{\Phi}_{1:K}^\top \Delta\mu$ ▷ λ from the trace rule at this K
  $J(K) \leftarrow \left\| \mathbb{E}_{\mathcal{H}}\left[\varphi\left(\tilde{x} + \sum_{k \le K} a_k^{(K)} \tilde{d}_k\right)\right] - \mu_{\mathcal{H}}^\star \right\|_2$
end for
$K^\star \leftarrow \arg\min_K J(K),\ \ a \leftarrow a^{(K^\star)}$
return $(\gamma,\ a,\ K^\star)$ ▷ frozen for inference

Algorithm 2 ENCORE correction (applied once after the last step)

Require: terminal sample x_N and its denoising outputs $D_{1:N}$ from one few-step run; calibrated $(\gamma, a, K^\star)$ from Algorithm 1

$d_k \leftarrow D_{N-k+1} - D_{N-k},\ \ k = 1, \dots, K^\star$ ▷ no network evaluation
return $B_\gamma(x_N) + \sum_{k=1}^{K^\star} a_k\, B_\gamma(d_k)$

as

$$x_N^{\text{corrected}} \ = \ B_\gamma(x_N) \ + \ \sum_{k=1}^{K} a_k\, B_\gamma(d_k), \tag{2}$$

where B_γ is a linear operator scaling the Fourier coefficients of each of the 16 frequency bands by a band gain γ_b, $b = 1, \dots, 16$, and $a_k \in \mathbb{R}$ are scalar coefficients. Stage **S** applies the band gains to the sample and to the differences, and Stage **M** then adds the rescaled differences with the coefficients a_k. For $\gamma = 1$ and $a = 0$, the baseline sampler is recovered exactly. The band gains and the coefficients are determined offline by matching the second and the first moments of the output distribution to those of the target. Figure 3 illustrates the correction pipeline, and Algorithms 1 and 2 state the procedure. Applying Equation 2 at inference incurs no additional NFEs, since the few-step sampler already computes every D_i.

4.1 Spectral Matching (Stage S)

Stage **S** matches the power of the samples in each frequency band to that of the target. We apply a discrete Fourier transform to the generated terminal sample, group the frequency axis into 16 bands, and scale the Fourier coefficients within band b by a band gain γ_b, which rescales that band's power from P_b to $\gamma_b^2 P_b$. We therefore set

$$\gamma_b \ = \ \sqrt{P_b^\star / P_b}\,, \tag{3}$$

where P_b and $P_b^\star$ are the average band powers of the baseline few-step samples x_N and of the target samples $x^\star$, evaluated over the calibration set. The rescaling matches the target's pooled band power while preserving the generated phase. This is the only stage that matches a second moment, targeting pooled power rather than full covariance.

4.2 Feature-Mean Matching (Stage M)

Stage **M** aligns the feature mean of the output distribution with that of the target. Let φ be a feature map (Inception for the pixel and image latent models, CLAP for audio). We denote the rescaled sample by $\tilde{x} = B_\gamma(x_N)$ and the rescaled directions by $\tilde{d}_k = B_\gamma(d_k)$, with $d_k = D_{N-k+1} - D_{N-k}$ as above, and the target feature mean over the calibration set by $\mu^\star$. Each direction $\tilde{d}_k$ displaces the feature mean by $\bar{\Phi}_k = \mathbb{E}[\varphi(\tilde{x} + \tilde{d}_k) - \varphi(\tilde{x})]$. These displacement vectors form the columns of $\bar{\Phi}$.

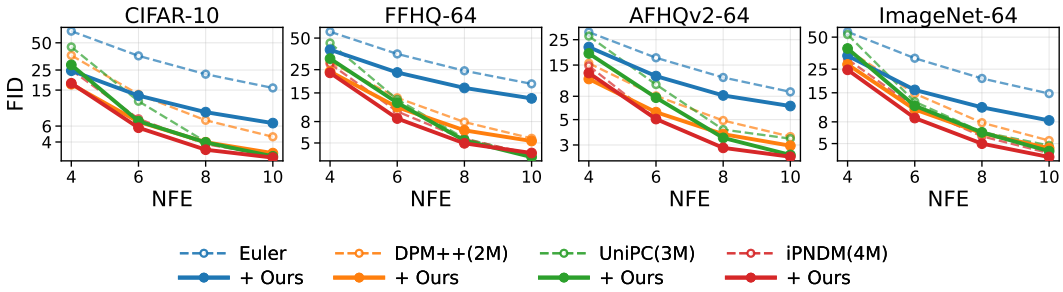


Figure 4: **Sample quality across four pixel-space datasets and four ODE solvers.** FID (↓) across NFE $\in \{4, 6, 8, 10\}$ on 50k samples. Dashed lines denote the baseline solver and solid lines denote the same solver with ENCORE.

We then solve

$$a \;=\; (\bar{\Phi}^\top \bar{\Phi} + \lambda I)^{-1}\, \bar{\Phi}^\top \Delta\mu, \qquad \Delta\mu = \mu^\star - \mathbb{E}[\varphi(\tilde{x})], \tag{4}$$

a ridge-regularized least-squares solution in closed form, where $\lambda = 10^{-3}\ \mathrm{tr}(\bar{\Phi}^\top \bar{\Phi})/K$ is a ridge term and $\Delta\mu$ is the residual mean discrepancy. Neither constructing $\bar{\Phi}$ nor solving for a requires a forward pass through the diffusion network.

The solve rests on the *linear-displacement assumption*: the feature-mean shift is linear in the coefficients,

$$\mathbb{E}\Big[\varphi\big(\tilde{x} + \textstyle\sum_k a_k \tilde{d}_k\big)\Big] \;\approx\; \mathbb{E}\big[\varphi(\tilde{x})\big] \;+\; \textstyle\sum_k a_k\, \bar{\Phi}_k, \qquad \bar{\Phi}_k \;=\; \mathbb{E}\big[\varphi(\tilde{x} + \tilde{d}_k) - \varphi(\tilde{x})\big]. \tag{5}$$

Each column $\bar{\Phi}_k$ is a finite difference over the *full* step $\tilde{d}_k$ rather than a Jacobian product. It therefore needs no derivative of the network and costs a single forward pass of the feature map per direction, and Equation 5 is exact for a single direction at unit coefficient. The linearity holds exactly only where φ is affine on the span of $\{\tilde{d}_k\}$. Solving a globally over the calibration set, rather than per sample, preserves the diversity of the generated distribution.

4.3 Order of the Two Stages and Calibration

Applying the band gains and solving for the coefficients in the other order gives a different result, so the order has to be fixed. We solve γ first on the baseline few-step samples, apply B_γ to both the samples and the directions, and then solve a in the scaled space (**S2M**, spectral-then-mean). Because B_γ is linear, the deployed sample $B_\gamma(x_N) + \sum_k a_k B_\gamma(d_k)$ is the state on which Stage **M** was solved, so no operation follows the solve. In the reverse order (M2S, mean-then-spectral), the feature mean is matched first, and the band gains applied afterward perturb the mean they have just matched. The two orders differ in the state on which γ and a are solved, not in the formula: in S2M, a is solved on the rescaled samples and directions; in M2S, a is solved on the baseline samples and γ on the samples after the coefficients are added, and the two solutions differ.

Because both stages have closed-form solutions, the history depth $K \in \{1, \ldots, N-1\}$ is the only hyperparameter selected by validation, on a held-out half of the calibration set (Algorithm 1). Both stages are solved once per setting and applied unchanged at inference. On CIFAR-10 with Euler at NFE $\in \{4, 6, 8, 10\}$, the corrected band power equals the target on the calibration half it was solved on and falls within 97.9–109.2% of it on 50,000 samples drawn with a different seed.

5 Experiments

We evaluate ENCORE across pixel-space, image latent, and audio latent diffusion models in Sections 5.2 and 5.3, including settings where the correction is stacked on published methods at no additional NFE. Section 5.4 ablates the contributions of Stage **S** and Stage **M** and the order in which they are determined.

Table 1: Pixel-space generation quality and stacked baselines. FID (↓) at 50k samples on CIFAR-10 and ImageNet-64 at NFE $\in \{4, 6, 8, 10\}$. Bold indicates the lower FID per pair. PAS reports ImageNet-64 from NFE 5 but provides official coordinates for NFE 4.

	CIFAR-10				*ImageNet-64*			
	4	6	8	10	4	6	8	10
Euler (Song et al., 2021a)	67.31	35.86	22.49	15.85	56.11	31.67	20.46	14.75
+ Ours	**24.58**	**13.04**	**8.55**	**6.47**	**33.45**	**15.96**	**10.95**	**8.25**
DPM++(2M) (Lu et al., 2022b)	36.32	13.45	6.95	4.57	37.99	14.74	7.85	5.34
+ Ours	**17.43**	**6.69**	**3.98**	**3.01**	**27.72**	**10.46**	**6.26**	**4.52**
UniPC(3M) (Zhao et al., 2023)	45.01	11.30	4.02	2.96	52.88	12.37	6.42	4.80
+ Ours	**28.49**	**6.80**	**3.96**	**2.78**	**39.09**	**11.33**	**6.37**	**4.25**
iPNDM(3M) (Zhang & Chen, 2023)	29.70	9.95	5.38	3.78	31.43	11.40	6.07	4.24
+ Ours	**17.63**	**5.81**	**3.48**	**2.71**	**23.93**	**8.14**	**4.86**	**3.70**
iPNDM(3M) + PAS (Wang et al., 2025b)	27.78	7.63	3.96	2.90	59.07	11.15	5.82	4.07
+ Ours	**25.02**	**5.31**	**3.48**	**2.78**	**54.96**	**8.90**	**4.94**	**3.70**
iPNDM(4M) (Zhang & Chen, 2023)	24.98	7.23	3.78	2.84	30.99	11.02	5.85	4.06
+ Ours	**17.84**	**5.75**	**3.29**	**2.69**	**24.65**	**8.69**	**4.99**	**3.74**
iPNDM(4M) + GITS (Chen et al., 2025)	15.10	4.91	3.29	2.51	16.60	7.06	**4.82**	3.77
+ Ours	**13.48**	**4.89**	**3.28**	**2.43**	**12.58**	**7.00**	4.92	**3.56**

5.1 Experimental Setup

Models and datasets. We evaluate ENCORE across three distinct diffusion domains. *Pixel-space diffusion*: CIFAR-10, FFHQ-64, AFHQv2-64, and ImageNet-64, each with four ODE solvers of increasing order, Euler (1st), DPM++(2M) (2nd), UniPC(3M) (3rd), and iPNDM(4M) (4th); iPNDM(3M) additionally serves as the base solver for PAS. *Image latent diffusion*: DiT-XL/2 on ImageNet-256 and Stable Diffusion 1.4 on MS-COCO captions, both with DPM-Solver++(2M). *Audio latent diffusion*: AudioLDM 2 on AudioCaps with DDIM.

Protocol and calibration. All runs within a given (model, dataset, solver, NFE) *setting* share identical random seeds and schedules to isolate the post-hoc correction as the sole independent variable. Each setting is calibrated on 1,000 self-generated samples against a 64-step target generated by the same solver, with 16 frequency bands, without requiring real data. The calibration pairs are split into two halves: band gains γ and coefficients a are solved on $\mathcal{F}$, while the history depth $K^\star$ is selected on the held-out $\mathcal{H}$ via feature-mean discrepancy (Algorithm 1). Calibration never computes any reported metric and never observes a real or evaluation sample, and a single rule fixes K per setting without per-cell tuning.

Baselines. We adopt official checkpoints and default configurations (Appendix A). When ENCORE is combined with a published baseline, the 64-step target is generated with that method's own underlying solver to preserve sampler consistency.

Evaluation metrics. For image generation, we report FID against official reference statistics: 50,000 samples against EDM references (Karras et al., 2022) for pixel-space models and the ADM reference batch (Dhariwal & Nichol, 2021; Peebles & Xie, 2023) for DiT-XL/2. For SD 1.4, FID is measured at 30,000 samples against the first 30,000 MS-COCO val2014 images (Rombach et al., 2022). For audio generation, we report Fréchet Audio Distance (FAD), KL divergence, and CLAP score over 957 AudioCaps clips, all computed with the evaluation package released with AudioLDM 2 (Liu et al., 2024).

5.2 Main Results on Pixel-Space Diffusion Models

Figure 4 and Table 1 demonstrate that ENCORE improves generation quality across settings, lowering FID in 63 of 64 evaluated cells. In particular, at four steps on CIFAR-10, first-order Euler with ENCORE reaches a lower FID than DPM++(2M), UniPC(3M), and iPNDM(4M). A higher-order solver combines the denoising outputs of previous steps to follow the curve of the trajectory, and

Table 2: Sample quality in image latent diffusion. FID (↓) at 50k samples for DiT-XL/2 on ImageNet-256 and at 30k for Stable Diffusion 1.4 on MS-COCO. Bold indicates the better FID per pair. The two blocks use distinct references and protocols and are not comparable across.

Model	NFE			
	4	6	8	10
Stable Diffusion v1.4 (Rombach et al., 2022)	23.49	18.00	16.97	16.55
+ Ours	**21.76**	**16.48**	**14.46**	**14.61**
DiT-XL/2 (Peebles & Xie, 2023)	47.16	11.82	5.68	4.60
+ Ours	**16.10**	**4.70**	**3.68**	**3.43**

Table 3: Sample quality in audio latent diffusion. AudioLDM 2 on AudioCaps, 957 clips. Coefficients are solved in CLAP, while FAD (↓) and KL (↓) are measured in the networks of the evaluation package released with AudioLDM 2 (Liu et al., 2024). Bold indicates the better score per pair.

	NFE 4			NFE 8			NFE 16		
Method	FAD ↓	KL ↓	CLAP ↑	FAD ↓	KL ↓	CLAP ↑	FAD ↓	KL ↓	CLAP ↑
AudioLDM 2 (Liu et al., 2024)	14.91	3.16	0.12	8.86	2.01	0.33	4.18	1.71	0.40
+ Ours	**11.30**	**3.00**	**0.17**	**3.24**	**1.85**	**0.39**	**2.30**	**1.64**	**0.43**

ENCORE reuses the same outputs after the trajectory ends. This result indicates that a discrepancy visible only across samples, and not along any single trajectory, is comparable in effect to an increase in solver order.

Composition with existing acceleration methods. ENCORE is applied after the last sampling step, so it can be combined with methods that optimize the sampling process itself. Applied on top of PAS and GITS, ENCORE lowers FID in 15 of 16 settings of Table 1. An optimized sampler followed by ENCORE refines the sampling process and then its output, and the effect is largest at four steps. The sole exception is iPNDM(4M)+GITS on ImageNet-64 at eight steps, where the optimized time schedule appears to leave little of the discrepancy for the correction to remove.

5.3 Generalization to Image Latent and Audio Latent Diffusion Models

For latent models, both stages operate on the latent, which is the tensor the sampler returns. An image latent preserves the spatial axes of the image, so the band split and the calibration protocol are the same as for the pixel-space models. Table 2 shows that the correction lowers the FID at every NFE on both image latent models. On DiT-XL/2, the improvement is largest at the smallest NFE, as on the pixel-space grid, and the corrected runs at NFE 6 and 8 reach a lower FID than the baseline run with two more steps. On Stable Diffusion 1.4, the baseline FID decreases slowly as the NFE grows, and the corrected run at NFE 6 reaches a lower FID than the baseline at NFE 10. AudioLDM 2 encodes a log-mel spectrogram, so the values in its latent are logarithms of band energies rather than amplitudes, and we therefore solve and apply the band gains on the decoded waveform while the coefficients stay on the latent. Table 3 shows that the correction improves FAD, KL, and CLAP at every NFE. The distribution-level correction identified on pixel-space models therefore holds in both latent families, with the same two stages and the same calibration.

5.4 Ablations

Stage necessity and order. Table 4 compares each stage on its own with the two orders. S2M reaches the lowest FID in every column. M2S comes second in seven of the eight columns, so the two stages together account for most of the improvement in either order. Across the full grid, S2M reaches a lower FID than M2S in 49 of the 64 settings. The exception is FFHQ-64, where M2S reaches a lower FID than S2M in 12 of the 16 settings. Each stage alone also lowers the FID below the baseline, and the coefficients lower it more than the band gains in every column. The improvement from the band gains is largest at NFE 4 and shrinks as the NFE grows, which matches the band deficit in Figure 2.

Table 4: Ablation of the two stages and their order. FID (↓) with Euler on CIFAR-10 and ImageNet-64. *S* and *M* denote each stage alone, and **S2M** (deployed) and **M2S** the two orders. Bold and underline denote the lowest and second-lowest corrected FID per column. Appendix C reports the remaining solvers and datasets.

		NFE						NFE			
Dataset	Stage	4	6	8	10	Dataset	Stage	4	6	8	10
	none	67.31	35.86	22.49	15.85		none	56.11	31.67	20.46	14.75
	S only	40.72	22.84	14.88	10.92		S only	46.87	24.14	15.45	11.24
CIFAR-10	M only	32.59	17.79	10.17	<u>7.35</u>	*ImageNet-64*	M only	34.79	17.63	11.78	8.92
	M2S	<u>27.33</u>	<u>17.15</u>	<u>9.84</u>	7.75		M2S	<u>33.58</u>	<u>16.67</u>	<u>11.47</u>	<u>8.87</u>
	S2M (ours)	**24.58**	**13.04**	**8.55**	**6.47**		**S2M** (ours)	**33.45**	**15.96**	**10.95**	**8.25**

Table 5: Coefficients solved in different feature spaces. Top: FID (↓) with Euler on CIFAR-10 and ImageNet-64, always scored with Inception. Bottom: FAD (↓) and KL (↓) for AudioLDM 2 on AudioCaps. CLAP is the deployed choice. Bold indicates the best feature space per column.

EDM

	CIFAR-10			ImageNet-64		
Feature map φ	NFE 4	6	8	4	6	8
Baseline	67.31	35.86	22.49	56.11	31.67	20.46
Inception	**24.58**	**13.04**	**8.55**	**33.45**	15.96	10.95
DINOv2	24.64	15.47	9.02	34.24	**15.87**	**10.71**

AudioLDM 2

	NFE 4		NFE 8		NFE 16	
Feature map φ	FAD	KL	FAD	KL	FAD	KL
Baseline	14.91	3.16	8.86	2.01	4.18	1.71
VGGish	11.32	3.05	4.08	**1.85**	2.52	1.65
CLAP (deployed)	**11.30**	3.00	**3.24**	**1.85**	**2.30**	**1.64**
PANNs	11.47	**2.99**	3.81	**1.85**	2.33	**1.64**

Feature space of the coefficients. Table 5 solves them in other spaces: on the pixel-space models, DINOv2 instead of Inception lowers the Inception FID by nearly as much, and on ImageNet-64 at NFE 6 and 8 the DINOv2 solve reaches a lower FID. On audio, the deployed CLAP computes neither FAD nor KL. Yet the CLAP solve reaches a lower FAD at all three NFEs than the solve in VGGish, the FAD network, and the three solves differ in KL by at most 0.06, so the improvement is consistent across image and audio.

Computational cost. All wall-clock times are measured on one RTX 4090. Calibration has two parts. The 64-step target set is drawn once per model and reused for every NFE. It takes 66 s on CIFAR-10 and 10 min on AudioLDM 2. For each NFE, one few-step set is drawn and γ and a are solved on it. At NFE 4 these take 9 s on CIFAR-10 and 416 s on AudioLDM 2. Applying the correction adds 0.26–2.10% to the time of one sample.

6 Conclusion

We present ENCORE, a training-free, post-hoc correction that operates on the sample set rather than on a single trajectory. Few-step samples deviate from many-step samples in two distributional ways: a power deficit in every frequency band, and a shift of the feature mean. Neither is observable within one trajectory, yet both are statistics of the set and can therefore be measured and corrected after sampling. ENCORE solves both in closed form using the differences between denoising outputs that the sampler has already computed, at no additional network evaluation per sample, without training, and without real data. It improves sample quality across pixel-space, image latent, and audio latent diffusion models. Beyond these two moments, a localized transform such as a wavelet and a kernel discrepancy such as maximum mean discrepancy are promising directions for future work.

AI USE STATEMENT

We used generative AI tools to implement and run the experiment code, and to draft and revise the prose of this paper. We formulated the hypothesis, designed the experiments, and interpreted the results. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Generative models of images and audio can be used to synthesize inappropriate or misleading content, and any work that improves what they return inherits that potential. This paper contributes a correction applied after sampling to checkpoints that already exist. It trains nothing, releases no data, and does not change what a checkpoint is able to generate. We therefore see no hazard specific to this work beyond the one it inherits. All datasets we use are public benchmarks.

REPRODUCIBILITY STATEMENT

Section 4 gives the method and its hyperparameters. Algorithms 1 and 2 state the calibration and what is applied at generation, in pseudo-code. Section 5.1 gives the evaluation protocol, including the sample counts, the reference statistics of each family and the separation between calibration and evaluation samples. Anonymized source code for the method, the calibration and every experiment in this paper is submitted as supplementary material.

REFERENCES

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

Defang Chen, Zhenyu Zhou, Can Wang, and Siwei Lyu. Geometric regularity in deterministic sampling dynamics of diffusion-based generative models. *Journal of Statistical Mechanics: Theory and Experiment*, 2025(12):124002, 2025.

Jinyoung Choi, Junoh Kang, and Bohyung Han. Enhanced diffusion sampling via extrapolation with multiple ODE solutions. In *International Conference on Learning Representations*, 2025.

Hadar Davidson, Noam Issachar, and Sagie Benaim. Colored noise diffusion sampling. *arXiv preprint arXiv:2605.30332*, 2026.

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

Eric Frankel, Sitan Chen, Jerry Li, Pang Wei Koh, Lillian J. Ratliff, and Sewoong Oh. S4S: Solving for a fast diffusion model solver. In *International Conference on Machine Learning*, 2025.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 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*, 2022.

Tuomas Kynkäänniemi, Tero Karras, Miika Aittala, Timo Aila, and Jaakko Lehtinen. The role of imagenet classes in Fréchet inception distance. In *International Conference on Learning Representations*, 2023.

Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2024.

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. AudioLDM 2: Learning holistic audio generation with self-supervised pretraining. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2871–2883, 2024.

Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *International Conference on Learning Representations*, 2022.

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*, 2022a.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022b.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Xin Peng and Ang Gao. Rethinking refinement: Correcting generative bias without noise injection. *arXiv preprint arXiv:2601.21182*, 2026.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

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

Andrea Schioppa and Tim Salimans. Covariance-aware sampling for diffusion models. *arXiv preprint arXiv:2605.13910*, 2026.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021a.

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In *International Conference on Learning Representations*, 2024.

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 (ICLR)*, 2021b.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, 2023.

Zheng Tan, Weizhen Wang, Andrea L. Bertozzi, and Ernest K. Ryu. STORK: Faster diffusion and flow matching sampling by resolving both stiffness and structure-dependence. In *International Conference on Learning Representations (ICLR)*, 2026.

Guangyi Wang, Yuren Cai, Lijiang Li, Wei Peng, and Songzhi Su. PFDiff: Training-free acceleration of diffusion models combining past and future scores. In *International Conference on Learning Representations*, 2025a.

Guangyi Wang, Wei Peng, Lijiang Li, Wenyu Chen, Yuren Cai, and Songzhi Su. Diffusion sampling correction via approximately 10 parameters. In *International Conference on Machine Learning (ICML)*, 2025b.

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.

Guoqiang Zhang, Kenta Niwa, and W. Bastiaan Kleijn. Lookahead diffusion probabilistic models for refining mean estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Guoqiang Zhang, Kenta Niwa, and W. Bastiaan Kleijn. On accelerating diffusion-based sampling processes via improved integration approximation. In *International Conference on Learning Representations*, 2024.

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

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

Wenliang Zhao, Haolin Wang, Jie Zhou, and Jiwen Lu. DC-Solver: Improving predictor-corrector diffusion sampler via dynamic compensation. In *European Conference on Computer Vision (ECCV)*, 2024.

Kaiwen Zheng, Cheng Lu, Jianfei Chen, and Jun Zhu. DPM-Solver-v3: Improved diffusion ODE solver with empirical model statistics. In *Advances in Neural Information Processing Systems*, 2023.

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

A Implementation and Evaluation Details

Models and evaluation. The pixel-space models are the EDM checkpoints, and FID is measured against the reference statistics released with them. DiT-XL/2 runs at guidance 1.5, and FID is measured against the ADM reference batch. SD 1.4 runs at guidance 7.5 on one caption for each of the first 30,000 images of MS-COCO val2014, and FID is measured against those images. AudioLDM 2 runs at its default guidance of 3.5 on the 957 AudioCaps test clips. FAD is measured on VGGish features and KL on PANNs CNN14 logits, the defaults of the evaluation package released with AudioLDM 2 (Liu et al., 2024). The coefficients are solved in CLAP on captions from the AudioCaps validation split. The band split uses 16 radial bands of equal width for images and 16 log-spaced bands on the waveform for audio.

Baselines. The published methods we stack on, PAS and GITS, run from their official code at their default settings on the EDM checkpoint, and every row of theirs is our own run.

B Model Target versus Real-Data Target

The calibration target of ENCORE is a 64-step run of the same model, and no real data is used. We provide an ablation on this choice in Table 6, where the target is replaced by real data: the band gains are solved to match the power spectrum of real images, and the coefficients are solved to match the Inception mean of the reference images. On the four pixel-space datasets with Euler at NFE 4, the real-data target scores better than the 64-step target on FFHQ and scores worse on the other three datasets. The choice of target is therefore not critical, and we keep the 64-step target because it requires no real data.

Table 6: Ablation on the calibration target. FID at 50,000 samples with Euler at NFE 4 on the four pixel-space datasets. The 64-step target is the one ENCORE uses. The real-data target replaces it with the power spectrum of real images and the Inception mean of the reference images.

Dataset	Baseline	64-step target	Real-data target
CIFAR-10	67.31	24.58	25.61
FFHQ	57.47	38.73	38.62
AFHQ	29.12	21.54	21.79
ImageNet-64	56.11	33.46	35.10

C Additional Experimental Results

This appendix collects the tables behind the main text and the remaining ablations. Unless stated otherwise in a caption, the checkpoints, samplers, and evaluation protocol are those of Section 5.1 and Appendix A.

C.1 All Four Datasets

Table 7 reports the four solvers with and without the correction on all four pixel-space datasets, under the calibration and protocol of Table 1.

Table 7: Pixel-space grid, all four datasets. FID at 50,000 samples, the same calibration and protocol as Table 1. Each pair is the baseline run and the same run with ENCORE, and bold marks the lower of the two. The published methods of Table 1 are stacked on CIFAR-10 and ImageNet-64 only.

	NFE					NFE			
Solver	4	6	8	10	Solver	4	6	8	10
CIFAR-10					*ImageNet-64*				
Euler	67.31	35.86	22.49	15.85	Euler	56.11	31.67	20.46	14.75
+ Ours	**24.58**	**13.04**	**8.55**	**6.47**	+ Ours	**33.45**	**15.96**	**10.95**	**8.25**
DPM++(2M)	36.32	13.45	6.95	4.57	DPM++(2M)	37.99	14.74	7.85	5.34
+ Ours	**17.43**	**6.69**	**3.98**	**3.01**	+ Ours	**27.72**	**10.46**	**6.26**	**4.52**
UniPC(3M)	45.01	11.30	4.02	2.96	UniPC(3M)	52.88	12.37	6.42	4.80
+ Ours	**28.49**	**6.80**	**3.96**	**2.78**	+ Ours	**39.09**	**11.33**	**6.37**	**4.25**
iPNDM(4M)	24.98	7.23	3.78	2.84	iPNDM(4M)	30.99	11.02	5.85	4.06
+ Ours	**17.84**	**5.75**	**3.29**	**2.69**	+ Ours	**24.65**	**8.69**	**4.99**	**3.74**
FFHQ-64					*AFHQv2-64*				
Euler	57.47	35.31	24.40	18.31	Euler	29.12	17.37	11.68	8.75
+ Ours	**38.72**	**23.49**	**16.74**	**13.30**	+ Ours	**21.54**	**12.00**	**8.13**	**6.57**
DPM++(2M)	29.57	13.47	7.92	5.53	DPM++(2M)	15.56	8.06	4.91	3.55
+ Ours	**23.97**	**10.96**	**6.60**	**5.22**	+ Ours	**11.34**	**5.81**	**3.73**	**2.96**
UniPC(3M)	44.87	12.99	5.53	3.82	UniPC(3M)	26.91	10.13	4.09	3.41
+ Ours	**31.87**	**12.10**	**5.28**	**3.67**	+ Ours	**18.95**	**7.78**	**3.46**	**2.46**
iPNDM(4M)	28.43	9.98	5.44	**3.93**	iPNDM(4M)	14.78	5.08	2.84	2.38
+ Ours	**23.31**	**8.55**	**4.96**	4.01	+ Ours	**12.78**	**5.06**	**2.83**	**2.37**

C.2 Stacking on LA-DPM, RX-DPM, and STORK-4

We also stack the correction on three further acceleration methods, LA-DPM, RX-DPM, and STORK-4, which report from NFE 9 or 10 upward in their own papers and are therefore compared at NFE 10, 12, and 15. LA-DPM extrapolates the clean-image estimates one step ahead, RX-DPM applies Richardson extrapolation across steps, and STORK-4 is a stabilized fourth-order Runge–Kutta solver. RX-DPM and STORK-4 run from their official code, LA-DPM is ported from its official code to the EDM checkpoint, and all three use the settings their papers fix for CIFAR-10. STORK-4 runs on the DDPM checkpoint released with PNDM, so its pair is read against its own baseline row, and its NFE here counts the three start-up calls and the closing Tweedie call that its paper leaves out. Table 8 reports the results: the correction lowers the FID in all nine pairs, and the largest improvement is on LA-DPM at NFE 10.

Table 8: Stacking on LA-DPM, RX-DPM, and STORK-4 at NFE 10, 12, and 15. FID on CIFAR-10 at 50,000 samples. Bold marks the lower of a pair.

	NFE		
	10	12	15
LA-DPM (Euler) (Zhang et al., 2023), λ=0.1	15.57	9.78	7.19
+ Ours	**6.46**	**4.64**	**3.92**
RX-DPM (Euler) (Choi et al., 2025)	5.22	4.18	3.36
+ Ours	**3.96**	**3.32**	**3.05**
STORK-4 (Tan et al., 2026)	8.78	6.94	5.53
+ Ours	**6.25**	**4.67**	**4.32**

C.3 Stage Ablation on All Four Datasets

Table 9 extends Table 4 to all four datasets and four solvers, with the baseline, each stage alone, and the two orders under the calibration and protocol of Table 1. The counts of Section 5.4 are read from this table.

Table 9: Stage ablation on all four datasets. FID at 50,000 samples. *S* and *M* denote each stage alone, and **S2M** and **M2S** denote the two orders. Bold and underline denote the lowest and second-lowest corrected FID per column.

Solver	Stage	NFE 4	6	8	10	Solver	Stage	NFE 4	6	8	10
CIFAR-10						*FFHQ-64*					
Euler	none	67.31	35.86	22.49	15.85	Euler	none	57.47	35.31	24.40	18.31
	S only	40.72	22.84	14.88	10.92		S only	49.43	31.29	21.89	16.56
	M only	32.59	17.79	10.17	7.35		M only	34.49	**20.93**	**15.39**	**11.68**
	M2S	27.33	17.15	9.84	7.75		M2S	**34.19**	21.39	16.13	12.14
	S2M (ours)	**24.58**	**13.04**	**8.55**	**6.47**		**S2M** (ours)	38.72	23.49	16.74	13.30
DPM++(2M)	none	36.32	13.45	6.95	4.57	DPM++(2M)	none	29.57	13.47	7.92	5.53
	S only	23.82	9.69	5.70	4.12		S only	28.00	12.76	7.76	5.50
	M only	21.58	7.08	4.13	3.06		M only	**23.57**	**9.58**	**6.11**	**5.11**
	M2S	19.01	7.60	4.57	3.25		M2S	23.67	9.63	6.15	5.13
	S2M (ours)	**17.43**	**6.69**	**3.98**	**3.01**		**S2M** (ours)	23.97	10.96	6.60	5.22
UniPC(3M)	none	45.01	11.30	4.02	2.96	UniPC(3M)	none	44.87	12.99	5.53	3.82
	S only	35.44	10.57	4.80	3.16		S only	41.19	13.72	5.57	3.77
	M only	32.15	7.52	4.11	**2.68**		M only	**31.37**	11.92	**5.20**	3.73
	M2S	29.82	7.83	3.97	2.79		M2S	32.00	**11.92**	5.27	3.70
	S2M (ours)	**28.49**	**6.80**	**3.96**	2.78		**S2M** (ours)	31.87	12.10	5.28	**3.67**
iPNDM(4M)	none	24.98	7.23	3.78	2.84	iPNDM(4M)	none	28.43	9.98	5.44	3.93
	S only	19.63	6.58	3.64	2.79		S only	27.62	9.88	5.42	3.95
	M only	23.21	6.32	3.30	**2.69**		M only	**22.88**	**8.48**	**4.94**	3.95
	M2S	21.09	6.50	3.50	2.79		M2S	23.28	8.70	4.96	**3.94**
	S2M (ours)	**17.84**	**5.75**	**3.29**	2.69		**S2M** (ours)	23.31	8.55	4.96	4.01
ImageNet-64						*AFHQv2-64*					
Euler	none	56.11	31.67	20.46	14.75	Euler	none	29.12	17.37	11.68	8.75
	S only	46.87	24.14	15.45	11.24		S only	23.68	14.31	9.99	7.70
	M only	34.79	17.63	11.78	8.92		M only	23.24	13.11	9.14	6.66
	M2S	33.58	16.67	11.47	8.87		M2S	21.85	12.76	9.04	**6.54**
	S2M (ours)	**33.45**	**15.96**	**10.95**	**8.25**		**S2M** (ours)	**21.54**	**12.00**	**8.13**	6.57
DPM++(2M)	none	37.99	14.74	7.85	5.34	DPM++(2M)	none	15.56	8.06	4.91	3.55
	S only	30.51	11.50	6.66	4.87		S only	12.77	7.49	4.91	3.60
	M only	34.03	12.32	6.78	4.71		M only	13.21	5.91	3.73	**2.94**
	M2S	27.95	11.42	6.67	4.79		M2S	12.36	5.93	3.77	2.96
	S2M (ours)	**27.72**	**10.46**	**6.26**	**4.52**		**S2M** (ours)	**11.34**	**5.81**	**3.73**	2.96
UniPC(3M)	none	52.88	12.37	6.42	4.80	UniPC(3M)	none	26.91	10.13	4.09	3.41
	S only	42.93	11.89	6.94	4.81		S only	22.84	10.37	3.96	3.27
	M only	43.16	11.81	**6.02**	**4.18**		M only	20.86	7.97	3.55	**2.45**
	M2S	40.25	12.46	6.25	4.26		M2S	20.52	8.06	3.49	2.49
	S2M (ours)	**39.09**	**11.33**	6.37	4.25		**S2M** (ours)	**18.95**	**7.78**	**3.46**	2.46
iPNDM(4M)	none	30.99	11.02	5.85	4.06	iPNDM(4M)	none	14.78	5.08	2.84	2.38
	S only	26.69	9.84	5.51	3.98		S only	13.49	5.02	2.83	2.38
	M only	27.29	9.56	5.25	3.75		M only	14.27	5.11	**2.83**	2.37
	M2S	25.81	9.67	5.39	3.79		M2S	13.26	**5.00**	2.83	2.44
	S2M (ours)	**24.65**	**8.69**	**4.99**	**3.74**		**S2M** (ours)	**12.78**	5.06	2.83	**2.37**

C.4 Number of Calibration Pairs

We vary the number of calibration pairs from 100 to 1,000 with DPM++(2M) at NFE 8 and report the FID in Table 10. For each count, both stages are solved and K is picked again from that many pairs, which are split into two halves as in Algorithm 1. The count is not critical, because 100 pairs already recover most of the improvement. We therefore keep 1,000 pairs.

Table 10: Ablation on the number of calibration pairs. FID at 50,000 samples with DPM++(2M) at NFE 8. Columns denote the number of pairs, and baseline denotes the sampler without the correction.

Dataset	baseline	100	250	500	1,000
CIFAR-10	6.95	4.31	4.04	4.19	3.98
FFHQ-64	7.92	6.51	6.99	6.76	6.60
AFHQv2-64	4.91	3.87	3.72	3.70	3.73
ImageNet-64	7.85	6.20	6.15	6.12	6.26

C.5 Computational Cost

Calibration is run once per setting and has two parts. The 64-step target set is drawn once per model and is shared by every NFE and solver of that model. The few-step set is drawn at the deployed NFE, and γ and a are solved on it. Table 11 reports the two parts, and Table 12 reports the time the correction adds to each generated sample.

Table 11: Cost of calibration, run once per setting. Wall-clock in seconds on one RTX 4090 for 1,000 pairs. Generation denotes drawing the samples, and solve denotes solving for γ and a.

		NFE 4		NFE 8		target (64 steps)
Model (solver)	batch	generation	solve	generation	solve	generation
CIFAR-10 (Euler)	250	4	5	8	7	66
ImageNet-64 (Euler)	100	15	4	31	7	240
DiT-XL/2 (DPM-Solver++(2M))	50	17	59	34	112	274
AudioLDM 2 (DDIM)	64	128	288	72	400	623

Table 12: Cost of applying the correction, paid for every sample. Milliseconds to generate one sample at batch size one on one RTX 4090, as a median over 50 runs. The left block denotes the baseline sampler, and the right block denotes the time the correction adds to it, as a percentage. The $\pm$ denote the bootstrap 95% interval of the median.

	baseline (ms) at NFE				added by ENCORE (%)			
Model, dataset	4	6	8	10	4	6	8	10
EDM, CIFAR-10	70.8 ± 0.1	130.3 ± 0.4	141.4 ± 1.8	174.9 ± 1.5	1.87	2.10	1.84	1.83
EDM, FFHQ-64	88.2 ± 0.2	132.6 ± 0.4	221.0 ± 2.5	275.1 ± 1.7	1.68	1.64	1.79	1.80
EDM, AFHQv2-64	88.5 ± 0.1	162.5 ± 1.0	177.6 ± 2.1	224.1 ± 1.6	1.66	1.82	1.65	1.62
EDM, ImageNet-64	136.4 ± 0.2	262.8 ± 0.5	274.2 ± 1.6	427.2 ± 2.4	1.06	1.18	1.05	1.15
DiT-XL/2, ImageNet-256	160.1 ± 2.0	192.4 ± 0.9	257.7 ± 1.3	306.8 ± 1.6	1.20	1.10	1.14	1.11
SD 1.4, MS-COCO	152.7 ± 0.1	214.6 ± 0.5	267.2 ± 0.7	329.3 ± 2.7	1.05	1.13	1.17	1.17
AudioLDM 2, AudioCaps	487.8 ± 2.7	628.7 ± 3.0	904.4 ± 4.2	923.1 ± 6.9	0.45	0.35	0.29	0.26

C.6 Choice of the History Depth K

We sweep the history depth K on the image and audio latent models on 5,000 samples and report the scores in Tables 13 and 14, together with the K that scores best (oracle) and the K the held-out rule picks without seeing these scores.

Table 13: History depth K on image latent diffusion. FID at 5,000 samples. The last three columns denote the oracle K, the K the held-out rule picks, and their gap.

			FID at fixed K									rule vs. oracle		
Model	NFE	none	1	2	3	4	5	6	7	8	9	oracle	picked	Δ
DiT-XL/2	4	53.17	22.51	29.99	29.49	–	–	–	–	–	–	1	1	+0.00
DiT-XL/2	6	18.36	13.29	13.67	12.04	11.54	11.49	–	–	–	–	5	5	+0.00
DiT-XL/2	8	12.42	11.52	10.81	10.52	10.47	10.47	10.45	10.46	–	–	6	5	+0.02
DiT-XL/2	10	11.39	11.25	10.58	10.51	10.47	10.46	10.44	10.20	10.19	10.20	8	9	+0.01
SD 1.4	4	30.58	28.80	32.36	37.12	–	–	–	–	–	–	1	1	+0.00
SD 1.4	6	24.52	22.99	21.90	21.24	21.38	21.25	–	–	–	–	3	1	+1.74
SD 1.4	8	23.46	21.93	20.87	20.57	20.85	20.95	20.82	21.11	–	–	3	4	+0.27
SD 1.4	10	22.96	21.76	21.26	21.04	21.13	21.08	21.29	21.33	21.17	21.19	3	4	+0.09

Table 14: History depth K in audio latent diffusion. AudioLDM 2 on AudioCaps, 957 clips at guidance 3.5, FAD against the real clips, computed with our own VGGish implementation, so the scale differs from Table 3. One column per K, then the oracle K and the K the held-out rule picks. Blank where K exceeds the $N-1$ differences available.

NFE	none	K=1	K=2	K=3	K=4	K=6	K=8	K=12	K=15	oracle K	picked K
4	3.82	2.85	2.85	2.96	–	–	–	–	–	1	2
8	2.34	1.18	1.19	1.20	1.32	1.33	–	–	–	1	1
16	1.25	0.83	0.84	0.84	0.85	0.84	0.84	0.85	0.84	1	1

C.7 Feature Space of the Coefficients, All Solvers

Table 15 extends Table 5 to all four solvers at NFE 6. The coefficients can be solved in either network, and the Inception FID falls in both cases.

Table 15: Feature space of the coefficients, all four solvers. Inception FID (lower is better) at 50,000 samples at NFE 6. Columns indicate the network the coefficients were solved in, with none the baseline and Inception the deployed choice. Bold marks the lower of the two networks.

	NFE 6		
Dataset / Solver	none	Inception	DINOv2
CIFAR-10 / Euler	35.86	**13.04**	15.47
CIFAR-10 / DPM++(2M)	13.45	**6.69**	7.27
CIFAR-10 / UniPC	11.30	**6.80**	7.45
CIFAR-10 / iPNDM	7.23	**5.75**	6.34
FFHQ-64 / Euler	35.31	**23.49**	23.59
FFHQ-64 / DPM++(2M)	13.47	**10.96**	11.86
FFHQ-64 / UniPC	12.99	**12.10**	12.40
FFHQ-64 / iPNDM	9.98	8.55	**8.54**
AFHQv2-64 / Euler	17.37	**12.00**	12.37
AFHQv2-64 / DPM++(2M)	8.06	**5.81**	5.96
AFHQv2-64 / UniPC	10.13	**7.78**	8.43
AFHQv2-64 / iPNDM	5.08	**5.06**	5.45
ImageNet-64 / Euler	31.67	15.96	**15.87**
ImageNet-64 / DPM++(2M)	14.74	10.46	**10.21**
ImageNet-64 / UniPC	12.37	11.33	**10.90**
ImageNet-64 / iPNDM	11.02	**8.69**	8.69

C.8 Order of the Two Stages in Audio

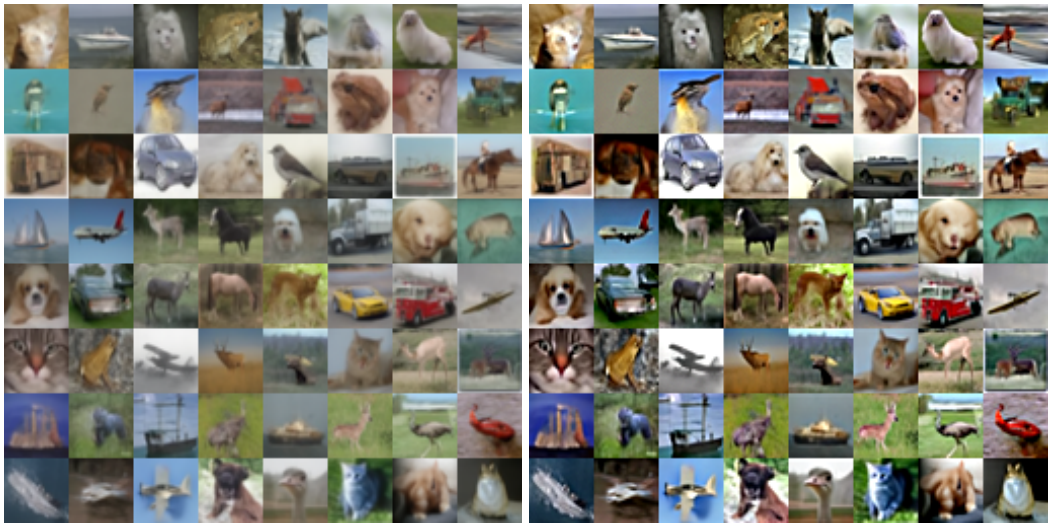
Table 16 compares the two orders on AudioLDM 2, each calibrated from scratch. The deployed order reaches a lower FAD at every NFE.

Table 16: Order of the two stages in audio. FAD (lower is better) for AudioLDM 2 on AudioCaps, 957 clips, target at 64 steps. Each order is calibrated from scratch. FAD is computed with our own VGGish implementation, as in Table 14. Bold marks the lower of each pair.

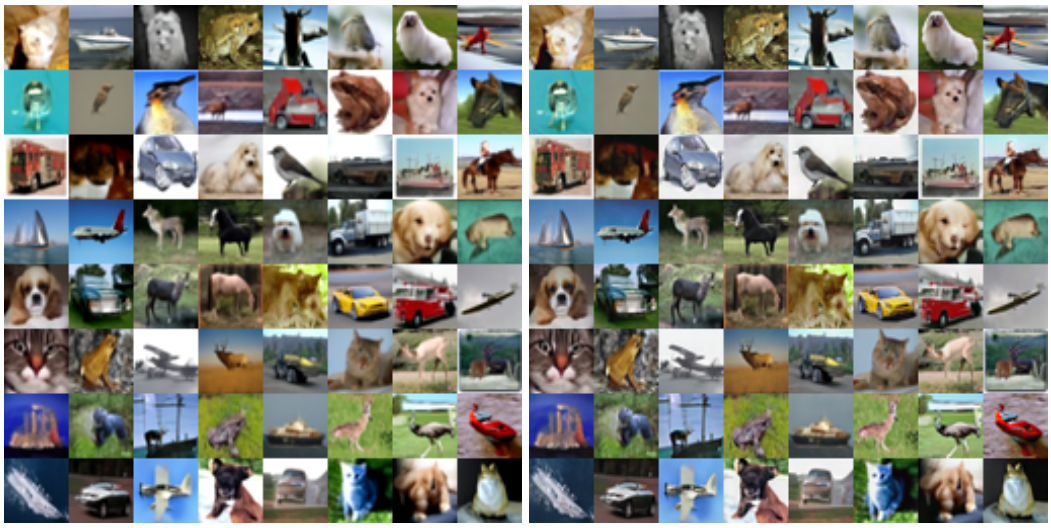
	FAD against real audio		
NFE	none	S2M	M2S
4	3.822	**2.818**	2.840
8	2.343	**1.055**	1.116
16	1.252	**0.791**	0.843

D QUALITATIVE SAMPLES

Figures 5 to 8 show samples from the four EDM checkpoints, Figure 9 shows the correction stacked on PAS, and Figure 10 shows DiT-XL/2.



Euler (NFEs = 8, FID: 22.49) | Euler + ENCORE (NFEs = 8, FID: 8.55)



DPM++(2M) (NFEs = 8, FID: 6.95) | DPM++(2M) + ENCORE (NFEs = 8, FID: 3.98)

Figure 5: Qualitative comparison on CIFAR-10 with the EDM backbone. The two panels in each row start from the same initial noise, and each FID is measured on 50,000 samples.



Euler (NFEs = 8, FID: 11.68)

Euler + ENCORE (NFEs = 8, FID: 8.13)



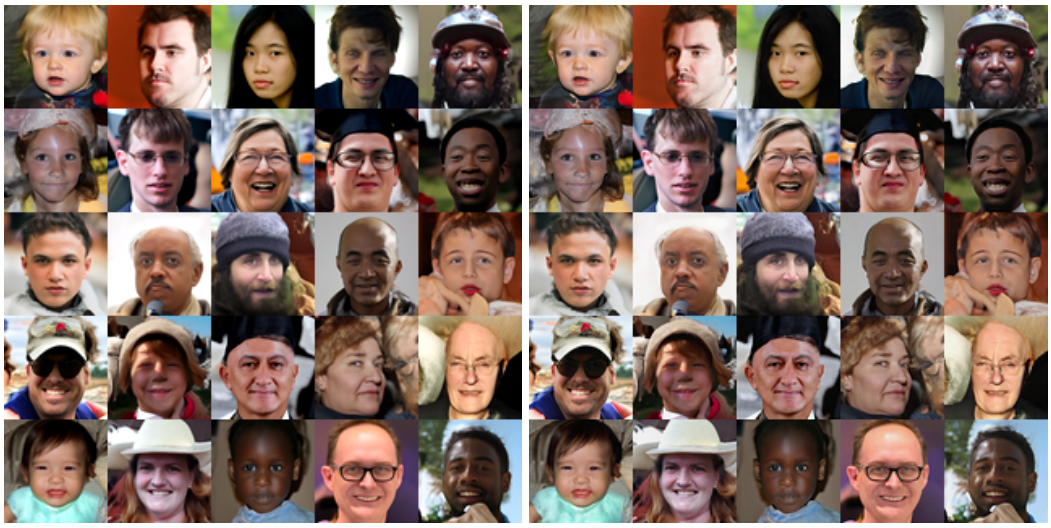
DPM++(2M) (NFEs = 8, FID: 4.91)

DPM++(2M) + ENCORE (NFEs = 8, FID: 3.73)

Figure 6: Qualitative comparison on AFHQv2-64 with the EDM backbone. The two panels in each row start from the same initial noise, and each FID is measured on 50,000 samples.

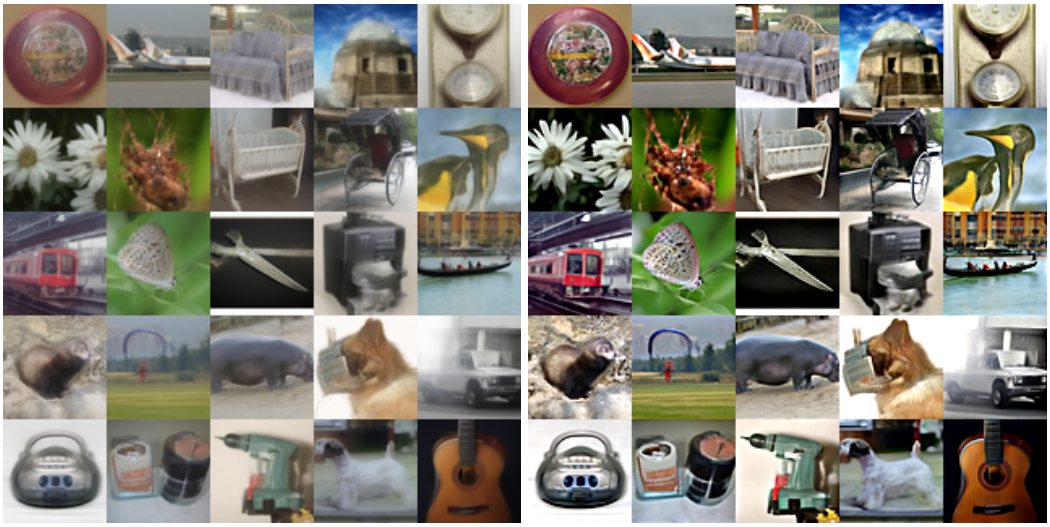


Euler (NFEs = 8, FID: 24.40) Euler + ENCORE (NFEs = 8, FID: 16.74)

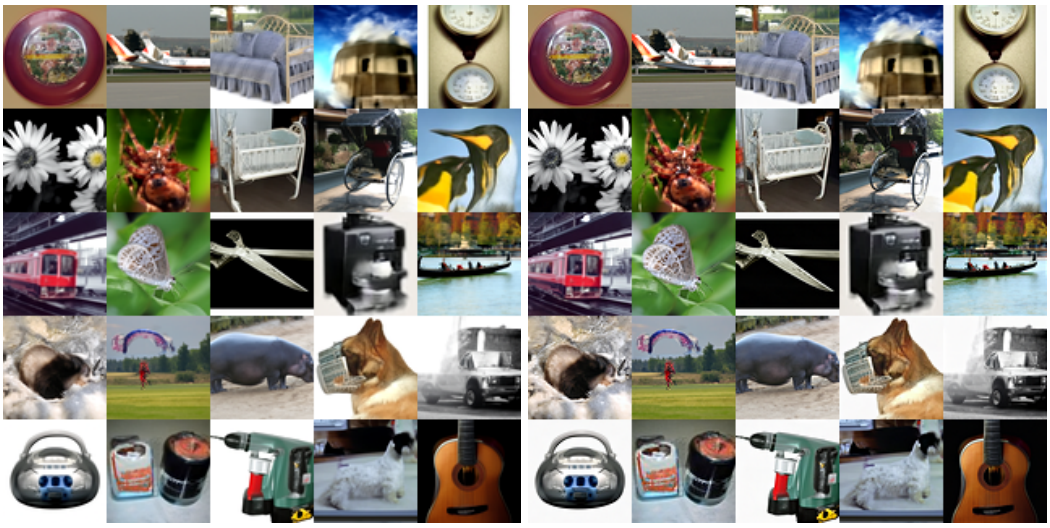


DPM++(2M) (NFEs = 8, FID: 7.92) DPM++(2M) + ENCORE (NFEs = 8, FID: 6.60)

Figure 7: Qualitative comparison on FFHQ-64 with the EDM backbone. The two panels in each row start from the same initial noise, and each FID is measured on 50,000 samples.

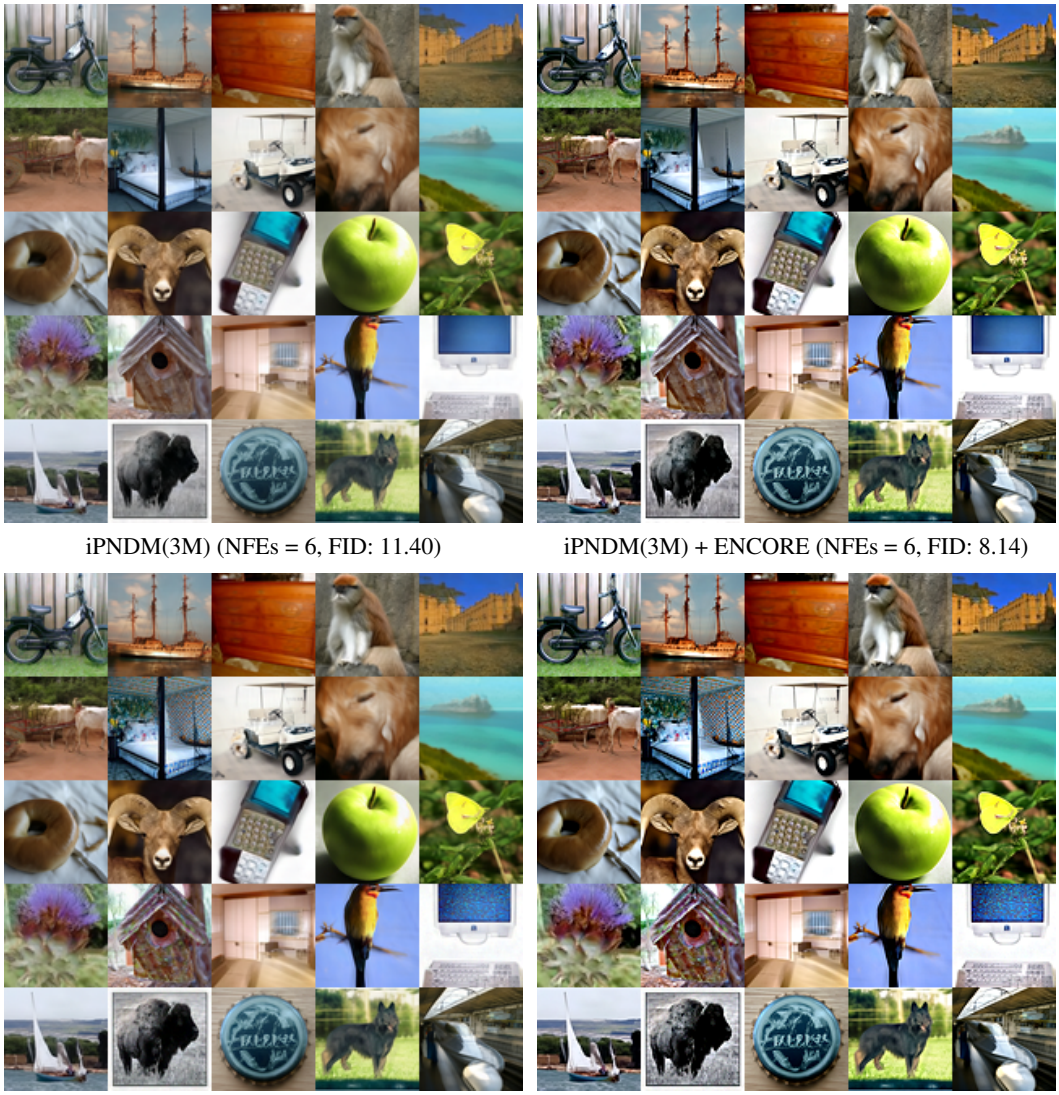


Euler (NFEs = 8, FID: 20.46) Euler + ENCORE (NFEs = 8, FID: 10.95)



DPM++(2M) (NFEs = 8, FID: 7.85) DPM++(2M) + ENCORE (NFEs = 8, FID: 6.26)

Figure 8: Qualitative comparison on ImageNet-64 with the EDM backbone. The two panels in each row start from the same initial noise, and each FID is measured on 50,000 samples.



iPNDM(3M) (NFEs = 6, FID: 11.40)

iPNDM(3M) + ENCORE (NFEs = 6, FID: 8.14)

iPNDM(3M) + PAS (NFEs = 6, FID: 11.15)

iPNDM(3M) + PAS + ENCORE (NFEs = 6, FID: 8.90)

Figure 9: Qualitative comparison on ImageNet-64 with the EDM backbone, with the correction stacked on PAS in the lower pair. The two panels in each row start from the same initial noise, and each FID is measured on 50,000 samples.



NFEs = 4 (FID 47.16 → 16.10)



NFEs = 6 (FID 11.82 → 4.70)

Figure 10: Qualitative comparison on ImageNet 256×256 with the DiT-XL/2 backbone at two NFEs. Within a block, the two rows use the same noise, so a cell differs from the one above it only by the correction. The same six classes appear in both blocks, but a sample need not match across them, because each NFE follows a different trajectory. FID is measured on 50,000 samples against the ADM reference batch with a guidance scale of 1.5.