BEYOND MORE SAMPLES: TEST-TIME SCALING OF TRAINING-FREE GUIDANCE IN GENERATIVE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work on test-time scaling has demonstrated that additional computation during inference can improve the quality of generative model outputs. In parallel, training-free guidance has enabled pretrained diffusion and flow-matching models to pursue specific target objectives during sampling without retraining. These two inference-time approaches can complement one another in improving a pretrained sampler's outputs for the target objective. We study their intersection through three recurring scaling strategies applied to guided sampling: (1) averaging K local draws, (2) propagating P parallel sampling trajectories, and (3) selecting among N final samples. Yet alongside inconsistencies reported in past research, we find that these strategies often fail to reliably improve final outputs and can even severely degrade them. The strategy that improves outputs most also changes, even between samplers that share the same checkpoint and objective. In this work, we develop a common characterization of how additional compute acts in each scaling strategy: what increasing the sampling budget actually refines, why that refinement can still fail to improve final outputs, and where to intervene when further scaling stalls. Building on this characterization, we also show that conventional summaries used to monitor these strategies, such as weight concentration or rank agreement, can omit information needed to predict the effects of scaling.

1 INTRODUCTION

Test-time compute scaling (Snell et al., 2025; Ramesh & Mardani, 2025) has recently emerged as a powerful mechanism for improving sample quality without retraining the underlying generative models. In diffusion and flow-matching frameworks, this paradigm can further enhance training-free guidance, which steers a pretrained generative process toward a target objective defined on clean outputs (Ye et al., 2024; Chung et al., 2023; Yu et al., 2023; Bansal et al., 2024). Existing approaches in this domain allocate additional computation in three recurring ways (Figure 1): (1) averaging, which computes the mean across K local Monte Carlo draws at each sampling step (Song et al., 2023; Ye et al., 2024), (2) particle propagation, which maintains P parallel particles across sampling steps using importance resampling (Wu et al., 2023; Cardoso et al., 2024; Kim et al., 2025b), and (3) output selection, which generates N completed samples and selects the top candidate at the final step (Stiennon et al., 2020; Nakano et al., 2021).

Despite the wide variety of scaling strategies proposed, their empirical effectiveness remains highly inconsistent across different models and benchmark tasks. Recent work (Zheng et al., 2025) demonstrates similar empirical inconsistencies in diffusion-based inverse problems. We observe the same phenomenon in our experiments across 18 released checkpoints and 29 task settings. Figure 2(a) shows the median performance gain over the single-sample guidance baseline, denoted as the *plug-in*, against increasing computational budget. Selection shows relatively steady improvements, though largely within the shared rerun band, while particle propagation shows little aggregate improvement and averaging can even degrade performance. Together, these trends show that more samples alone are not enough to ensure better output quality.

Prior work has partially explained these inconsistencies within individual strategies, identifying factors that govern the returns to additional sampling (Song et al., 2023; Geyfman et al., 2026; Ye et al., 2024; Ren et al., 2026a; Kim et al., 2025a; Gao et al., 2023). What remains missing is a

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.

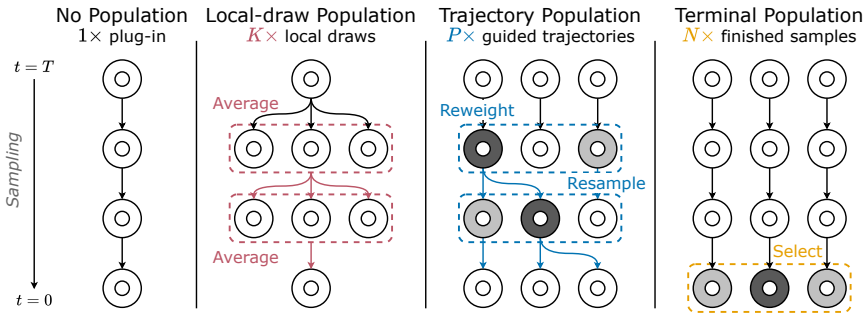

Figure 1: **Plug-in guidance and three test-time scaling strategies.** Dashed boxes mark the populations pooled by each strategy; fill darkness indicates weight or score.

common characterization across strategies: what additional samples change, why that change does not reliably improve the final output, and what underlying component must change when scaling is ineffective. With established statistical identities, we build this characterization on the population each strategy acts on, which we call its *consumed population*: the K local draws, the P particles, or the N completed samples (dashed boxes in Figure 1). More draws can estimate the local mean guidance with less variance, but the draw construction determines that mean. More particles can approximate a fixed target output law more faithfully, but the sampler and weighting rule determine that law. More candidates push selection toward higher-scoring outputs, where gains continue only while the selector remains reliable. In each case, whether a larger budget improves the output depends on a component that the budget leaves unchanged. Our controlled comparisons show this dependence directly: changing draw construction recovers much of averaging's lost performance, replacing only the sampler changes particle gains on the same checkpoint, and switching to a reliable selector enables further improvement.

This perspective also reveals limits in the information commonly used to assess scaling. While researchers frequently use population-level statistics to quantify weight concentration, sample diversity, and score alignment (Doucet & Johansen, 2011; Corso et al., 2024; Xu et al., 2023a), we show that these metrics can be misleading. Specifically, we construct matched counterexamples where summary statistics remain unchanged despite substantial shifts in the key quantities driving each sampling strategy. We further test if these blind spots can matter in practice, where we find the summary-based diagnostics to often mispredict scaling gains.

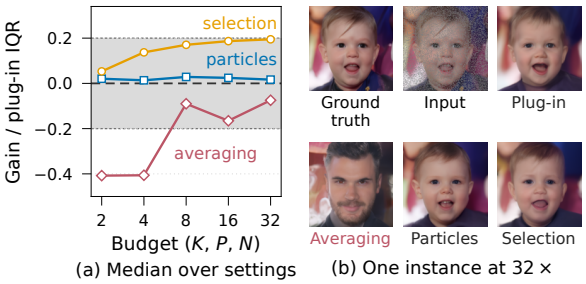

Figure 2: **Returns to test-time scaling.** (a) Median gains; gray band: pooled seed-range reference (not a confidence interval). (b) 32× averaging returns a different person. Figure 3 shows every setting.

We summarize our contributions as follows:

1. **Scaling characterization**: we provide a common characterization of test-time scaling for training-free guidance, identifying what additional sampling actually changes, why those changes fail to reliably improve final-output quality, and what needs to change when further scaling stalls.
2. **Limits of population summaries**: we construct counterexamples showing that common population summaries can remain unchanged even when the quantities relevant to scaling accuracy or final-output quality differ.
3. **Empirical evaluation**: we demonstrate heterogeneous scaling returns in broad experiments across tasks and released checkpoints and use controlled interventions to test the factors identified by our characterization. We further find that common summaries provide limited guidance on when additional compute will pay off.

2 SETUP

2.1 TRAINING-FREE GUIDANCE

Diffusion and flow-based generative models establish a continuous-time transport map between a data distribution $p_0(x_0)$ and a Gaussian prior $p_1(x_1) = \mathcal{N}(0, I)$. Along a prescribed interpolation path $x_t = \alpha_t x_0 + \sigma_t \varepsilon$ for $t \in [0, 1]$, where $\varepsilon \sim \mathcal{N}(0, I)$, the marginal probability path satisfies a probability flow ordinary differential equation (ODE) governed by a marginal velocity field $v^*(x_t, t)$. The velocity model $v_\theta(x_t, t)$ targets the conditional rate of change along the path, $v^*(x_t, t) = \mathbb{E}[\dot{\alpha}_t x_0 + \dot{\sigma}_t \varepsilon \mid x_t]$. Under the optimal velocity field, Tweedie's formula enables exact closed-form parameter conversions between the velocity v_θ, the predicted clean data $\hat{x}_0(x_t, t) = \mathbb{E}[x_0 \mid x_t]$, and the predicted noise $\hat{\varepsilon}(x_t, t) = \mathbb{E}[\varepsilon \mid x_t]$.

Training-free objective guidance steers the generative sampling trajectory toward task-specific objectives without fine-tuning the velocity model. Formally, given a differentiable loss function $J(x_0)$, the target distribution $\tilde{p}_0$ is modeled as an energy-based model:

$$\tilde{p}_0(x_0) \propto p_0(x_0) e^{-\lambda J(x_0)}$$

where $\lambda > 0$ is a guidance scale parameter. Sampling from $\tilde{p}_0(x_0)$ is achieved by steering the sampling trajectory with the exact guidance gradient $g^*(x_t)$,

$$g^*(x_t) = \nabla_{x_t} \log \mathbb{E}_{p(x_0|x_t)}[e^{-\lambda J(x_0)}]. \tag{1}$$

To make this expectation tractable, standard approaches evaluate the objective directly on the posterior-mean prediction $\hat{x}_0(x_t)$, yielding the gradient approximation:

$$g_{\text{plug}}(x_t) = -\lambda \nabla_{x_t} J(\hat{x}_0(x_t)). \tag{2}$$

We refer to g_{plug} as the *plug-in guide*; computing it requires backpropagating through $\hat{x}_0(x_t)$. Throughout, a *guide* is an estimator of g^* used at a sampling step.

This formulation approximates the posterior expectation of $e^{-\lambda J}$ by evaluating the objective directly at the estimated conditional mean $\hat{x}_0(x_t)$. We formalize the resulting guidance approximation bias as $b(x_t) = g_{\text{plug}}(x_t) - g^*(x_t)$.

2.2 TEST-TIME SCALING

We examine three prevalent strategies for scaling training-free guidance, each a *family* of implementations that share a *combination operator*, applied to a *consumed population* of exchangeable elements: averaging over local guidance gradients, weighting and resampling over trajectories, or selection over finished samples. Figure 1 contrasts these strategies against the single-evaluation plug-in baseline, showing which population each consumes, where it arises during sampling, and how its operator combines it.

Plug-in ($1\times$) As our baseline, the plug-in method evaluates a single clean prediction per timestep (Equation 2) without combining candidate samples.

Averaging local draws ($K\times$) As a natural first improvement over the plug-in estimator, local averaging evaluates K clean output samples, $(x_0^{(1)}, \ldots, x_0^{(K)})$, at a given sampling state x_t. These samples are drawn independently via $x_0^{(i)} \sim \mathcal{N}\big(\hat{x}_0(x_t),\, \sigma_{0|t}^2 I\big)$ using the posterior variance scale $\sigma_{0|t}$. Defining raw weights $w_i = \exp\big(-\lambda J(x_0^{(i)})\big)$ and normalized weights $\tilde{w}_i = w_i / \sum_j w_j$, we consider the following guidance estimators:

$$g_K^{\text{mean}} = \frac{1}{K} \sum_{i=1}^{K} \nabla_{x_t} \big[-\lambda J(x_0^{(i)})\big], \qquad g_K^{\text{tilt}} = \sum_{i=1}^{K} \tilde{w}_i \, \nabla_{x_t} \big[-\lambda J(x_0^{(i)})\big]. \tag{3}$$

Propagating particles ($P\times$) Sequential Monte Carlo (SMC) methods propagate P parallel sampling particles, updating path weights with factors G_t (Wu et al., 2023; Cardoso et al., 2024).

$$W_t^{(p)} \propto W_{t+1}^{(p)} \, G_t(x_t^{(p)}, x_{t+1}^{(p)}). \tag{4}$$

Resampling duplicates high-weight trajectories, while subsequent stochastic transitions allow duplicated paths to diverge. We select the returned particle using the final particle weights. Appendix A.1.2 specifies the weighting variants, and Appendix E.4 details path separation under deterministic and stochastic samplers.

Selection of outputs ($N\times$) Best-of-N selection generates N final outputs and selects the candidate with the minimum objective value (Stiennon et al., 2020; Nakano et al., 2021):

$$n^\star = \arg\min_{n\le N} J(x_0^{(n)}). \tag{5}$$

As such, we view each scaling family through the lens of a population–operator relationship: which samples it consumes and how it uses them.

3 What Test-Time Scaling Changes

Each family spends its extra compute on a different quantity. This section characterizes what that quantity is, and why improving it does not necessarily translate into a better output. Building on this characterization, we then show that commonly used summaries of those quantities are insufficient to determine this connection. We first define the final utility gain G_{scale} as the expected improvement achieved by the scaling strategy relative to the plug-in baseline:

$$G_{\text{scale}} := \mathbb{E}[U(\Pi_{\text{scale}}(\varepsilon))] - \mathbb{E}[U(\Pi_0(\varepsilon))] \tag{6}$$

where ε denotes the source of randomness (e.g., initial latent noise $\varepsilon \sim \mathcal{N}(0, I)$), Π_{scale} and Π_0 represent the generative process under the scaling strategy and the plug-in, respectively, and U is the task-specific utility function evaluated on the final outcome.

3.1 Effects of Scaling Within Each Family

Prior work has explained scaling behavior within individual families, including why additional samples may saturate. We provide a unified population–operator characterization that places these mechanisms under the same structure and separates what additional budget actually improves from what it leaves unchanged. We identify these quantities explicitly for each family and trace their connection to final utility gain G_{scale}.

Averaging ($K\times$)**.** Under this characterization, scaling K (more local Monte Carlo draws) controls the finite-sample accuracy of the local-draw average around its population mean. This exposes two limitations for final utility: (1) increasing K reduces sampling variance without necessarily correcting estimator bias, and (2) its per-step error reduction fails to guarantee downstream utility gains. For the unweighted estimator g_K^{mean}, each draw $x_0^{(i)}$ yields a local guidance gradient $g^{(i)} = \nabla_{x_t}\left[-\lambda J(x_0^{(i)})\right]$. The error between g_K^{mean} and the exact guidance direction g^* admits a classic bias–variance decomposition:

Lemma 3.1 (Bias–variance decomposition)**.** *Fix a sampling state and a deterministic reference guidance g^*. If $g^{(1)}, \dots, g^{(K)}$ are i.i.d. with $\mathbb{E}\|g^{(i)}\|^2 < \infty$, mean $\bar{g} = \mathbb{E}[g^{(i)}]$, and covariance matrix $\mathrm{Cov}(g^{(i)})$, then for every $K \ge 1$:*

$$\mathbb{E}\|g_K^{\text{mean}} - g^*\|^2 = \|\bar{g} - g^*\|^2 + \frac{\operatorname{tr}\mathrm{Cov}(g^{(i)})}{K}. \tag{7}$$

Equation 7 explicitly shows that allocating extra compute ($K \to \infty$) suppresses only the variance term at an $\mathcal{O}(1/K)$ rate. The bias term $\|\bar{g} - g^*\|^2$ remains entirely invariant to K. Consequently, scaling simply yields an increasingly precise estimate of the local mean guidance direction, even when it is misaligned with g^*. Even when additional compute lowers this per-step gradient MSE, local accuracy improvements do not directly translate into higher downstream performance. Because local errors propagate dynamically throughout the sampling trajectory, two constructions with identical per-step MSEs across all K can produce opposing effects on terminal metrics such as PSNR (Proposition A.5). Thus, neither finite-K estimation error nor per-step gradient MSE alone determines final utility; the local guide and its interaction with the sampling trajectory remain consequential. Tilt-weighted averaging likewise converges to a population-dependent mean that can differ from exact guidance (Appendix A.1.1).

Particles ($P\times$)**.** For particle-based strategies with fixed sampler transitions and weighting rules, increasing the particle count P can improve the approximation of the associated Feynman–Kac weighted output law ν (Del Moral, 2004, Ch. 2–3). This law is determined by the sampler's trajectory law q and the cumulative weights $W = \prod_t G_t$:

Lemma 3.2 (Feynman–Kac Output Law). *If $0 < \mathbb{E}_q[W] < \infty$, then for any bounded measurable output metric f, the target output distribution ν satisfies:*

$$\mathbb{E}_\nu[f(X_{\text{out}})] = \frac{\mathbb{E}_q[W\, f(X_{\text{out}})]}{\mathbb{E}_q[W]}. \tag{8}$$

Notice the particle budget P does not appear in Equation 8. The weighted output law ν is fixed by the sampler's trajectory law q and the cumulative weights W. Increasing P can improve the particle approximation of ν, but does not change q or what W favors. Consequently, even an accurate particle approximation can represent a law that favors low-utility outputs. Increasing P alone cannot correct this population-level misalignment; changing it requires changing q or W.

Selection ($N\times$). For selection-based strategies (e.g., Best-of-N), scaling test-time compute by generating more candidates N merely allows the selector to pick further into its highest-ranked outputs. Consequently, if the selector's ranking criterion is misaligned with the true utility U, allocating extra compute to increase N can be fundamentally ineffective. Formally, let X denote a finished output candidate and $V \in [0, 1]$ its rank quantile under the selector score, where $V = 0$ represents the selector's top candidate. Define $m(v) = \mathbb{E}[U(X) \mid V = v]$ as the expected utility of a candidate ranked at quantile v. The top candidate among N independent draws achieves the minimum quantile, which follows a $\text{Beta}(1, N)$ distribution (David & Nagaraja, 2003). The expected utility of the selected output is thus governed by this tail distribution:

Lemma 3.3 (Beta-Weighted Tail Selection). *For N i.i.d. candidate outputs with a continuous selector score and integrable utility U_{sel}, the expected utility of the selected candidate satisfies:*

$$\mathbb{E}[U_{\text{sel}}] = N \int_0^1 m(v)(1-v)^{N-1}\, \mathrm{d}v. \tag{9}$$

As $N \to \infty$, the weighting function $N(1-v)^{N-1}$ sharply concentrates its measure on $v \to 0$. Thus, additional budget changes how strongly selection exploits the selector's extreme tail, but not the selector–utility relationship $m(v)$ within that tail. If the selector score is misaligned with the true utility U (i.e., $m(v)$ does not achieve its maximum at $v = 0$), generating more candidates ($N \to \infty$) cannot guarantee better output quality. In such cases, additional compute can amplify the selection of over-optimized candidates that score high on a flawed metric but deliver low actual utility. The return to further scaling is therefore limited by the fidelity of the selector in the increasingly extreme region it selects.

Synthesis across scaling dimensions. Across the three families, additional budget therefore refines a different population–operator interaction: K refines a per-step gradient whose errors still propagate; P sharpens the particle approximation of a fixed target law ν; and N filters terminal outputs. Whether these refinements improve final utility depends on the properties of the consumed population that the budget itself does not change. Section 4.2 tests this distinction through controlled interventions.

3.2 Limits of Population Summaries

To monitor the behavior of averaging and particle systems, practitioners routinely track scalar summaries of the consumed populations. One common measure is the *Effective Sample Size* (ESS) (Doucet & Johansen, 2011; Kong, 1992), which measures weight concentration across M candidates ($M = K$ for draws, $M = P$ for particles). For nonnegative weights with positive total, let $\tilde{w}_i = w_i / \sum_j w_j$; the relative ESS is given by:

$$\frac{\text{ESS}}{M} = \frac{1}{M \sum_{i=1}^{M} \tilde{w}_i^2} \in \left[\frac{1}{M}, 1\right]. \tag{10}$$

For i.i.d. candidates with finite weight variance, ESS/M converges to the overlap with the reweighted sampling law (Appendix A.4.2). A value near 1 indicates uniform weighting, whereas a value near $1/M$ indicates severe weight collapse onto a single candidate. ESS measures weight concentration, not which candidates receive that weight; the same reading can therefore accompany beneficial or harmful concentration.

Averaging. Extending this diagnostic limitation to local averaging reveals an even stronger theoretical boundary: even possessing the complete joint probability distribution of draw weights is insufficient to determine whether averaging improves or degrades guidance accuracy. The weights w_i measure how strongly the objective favors specific draws, but guidance accuracy depends on the directional alignment between the resulting gradient vectors $\nabla_{x_t}\left[-\lambda J(x_0^{(i)})\right]$ and the exact guidance g^*. Because a scalar weight carries no direction, two objectives can produce identical weight statistics yet move gradient MSE in opposite directions:

Proposition 3.4. *There exist two smooth objective functions J_A and J_B, a Gaussian denoising model, and a sampling state such that:*

1. *For every $K \geq 1$, the draw-weight vector $(w_1, \ldots, w_K)$ has the identical joint distribution under J_A and J_B;*
2. *For some $K \geq 2$, both averaging estimators in Equation 3 achieve strictly lower gradient MSE than the plug-in baseline under J_A, but strictly higher gradient MSE under J_B.*

The detailed derivation is provided in Appendix A.2.1. This divergence stems from permuting the spatial coordinates x of the objective function $J(x)$. Because diffusion proposal noise is isotropic (spherical in data space), permuting feature dimensions leaves the joint distribution of draw weights $(w_1, \ldots, w_K)$ unchanged. However, because the true target posterior is anisotropic (directional), that same spatial permutation dramatically alters the alignment between the estimated gradient vectors and the exact guidance g^*. Consequently, scalar draw weights are fundamentally decoupled from spatial gradient directions, proving that weight-based diagnostics cannot determine whether local averaging successfully refines guidance.

Particles. In particle systems, weights directly drive trajectory resampling, making weight concentration (ESS) a natural candidate for diagnostic monitoring. However, tracking ESS alone fails because it measures how concentrated the weights are without revealing which trajectories receive those weights, whether they represent helpful state corrections or harmful distortions.

Proposition 3.5. *There exist two samplers sharing a common prior and likelihood with identical asymptotic normalized* ESS*, such that exact likelihood reweighting moves the output mean strictly closer to the true posterior mean under one sampler, but strictly farther away from it under the other.*

The detailed derivation is provided in Appendix A.2.2. In the first sampler, the unweighted trajectory distribution matches the prior, so likelihood reweighting correctly shifts the population toward the posterior mean. In the second sampler, the proposal already centers on the correct posterior mean; applying the same likelihood reweighting causes over-correction, pulling the output mean away from the truth. By tuning the proposal variance, both samplers yield the exact same asymptotic ESS, proving that identical weight overlap can accompany opposite reweighting outcomes.

Pairwise geometric measures provide a complementary view of particle diversity (Corso et al., 2024). Pairwise separation detects collapse that ESS misses: identical particles have zero separation but can carry uniform weights and maximal ESS. Swapping unequal weight values between two fixed particles preserves every pairwise distance, yet dramatically alters their weighted mean. Thus, weight concentration (ESS) and particle geometry reflect two decoupled dimensions: neither individually nor in naive combination do they capture the joint association between a trajectory's weight and its actual quality.

Selection. In selection-based strategies, practitioners often monitor candidate utility spread (to evaluate candidate diversity) or global rank correlation metrics such as Kendall's τ (to assess alignment between the selector score and true utility U). However, Best-of-N selection does not evaluate the candidate population globally: as N grows, the operator concentrates on the selector's extreme top-ranked tail. Consequently, global metrics like Kendall's τ fail to capture the local alignment in the precise quantile region that selection actually isolates.

Proposition 3.6. *There exist two candidate populations of selector-utility pairs (S, U) with continuous joint distributions such that:*

1. *Both populations share identical selector and utility marginal distributions, hence identical utility spread, and have the same Kendall's τ;*
2. *For every candidate budget $N \geq 3$, Best-of-N selection strictly increases expected utility relative to a single unselected candidate under the first population, but strictly decreases expected utility under the second.*

The detailed derivation is provided in Appendix A.2.3. This divergence arises because global rank correlation integrates over the entire candidate distribution, masking localized misalignment in the top quantiles. As N grows, selection places all its measure on $v \to 0$. If the selector's top-quantile favorite is misaligned with U, a high global Kendall's τ will fail to prevent Best-of-N from picking an over-optimized, low-utility candidate.

Across all three compute-scaling strategies, diagnostic failures stem from a unified structural flaw. Population-level summaries strip away the specific joint relationship that the scaling operator depends on:

- **Averaging (K):** Scalar draw weights discard spatial gradient directions in data space.
- **Particles (P):** Relative ESS and geometry decouple particle weights from trajectory quality.
- **Selection (N):** Global rank correlation (Kendall's τ) masks local misalignment in the selector's extreme tail.

Identical diagnostic readings can therefore conceal differences in guidance accuracy, reweighting effects, or final selection utility. Section 4.3 tests whether these theoretical blind spots are consequential in practice.

4 Results

Section 3 identified the quantities additional sampling changes in each family. We now empirically examine how these quantities affect final-output quality. Our experiments comprise budget sweeps, controlled comparisons, and evaluations of population summaries across complementary model families and tasks. Collectively, the testbeds span pixel- and latent-space diffusion and flow matching across inverse imaging, structural design, and text-to-image generation, including individual testbeds providing deterministic–stochastic sampler and learned–exact evaluator contrasts (Table 2). Throughout, a *cell* is one testbed condition, and a *setting* is a cell carried through a sweep of population budgets K, P, or N. Experimental details and robustness checks are provided in Appendices C–H.

4.1 Returns Across Budgets

We first examine the overall scaling picture across 29 settings on 18 released checkpoints, sweeping across population budgets K, P, and N. We report median per-setting gains over the plug-in, normalized by its interquartile range (IQR) across instances to compare quality metrics. The gray rerun band marks the median three-seed rerun spread of this gain estimate. These sweeps use representative implementations based on established constructions and should be interpreted as studies of within-family scaling behavior, rather than as a universal ranking among methods.

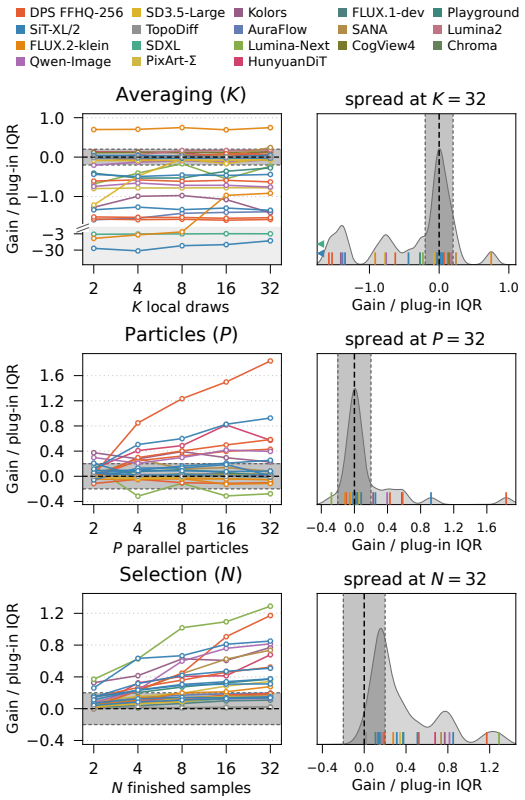

Figure 3: **Compute vs gain; spread of settings at 32× population budget.** Gains are normalized by the plug-in's IQR; budgets are on log scale. The gray band shows the median rerun spread of the gain estimate across cells (±0.20 IQR).

Figure 3 shows gains ranging from substantial losses to large improvements across settings. The budget curves trace how final-output gain changes as K, P, or N increases, while the companion spread plots show the distribution of setting-level gains at the largest tested budget of 32. Although the three distributions have different shapes and tails, none shifts uniformly toward positive gain: settings with substantial improvements coexist with settings showing little change or degradation. At this budget, 15 family–setting pairs fall below the band, 48 inside, and 24 above it.

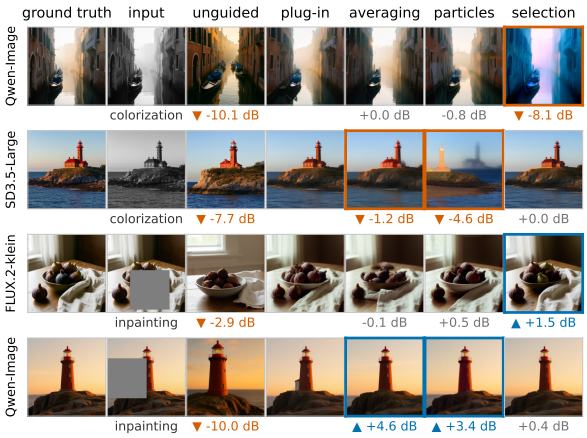

Figure 4: **Sample guided outputs scaled at** $16\times$ **population budget.** Numbers give each tile's PSNR gain (in dB; higher is better) over the plug-in (full image for colorization, inside the hole for inpainting). Tiles are highlighted when the gain's magnitude exceeds the plug-in's median seed change in that setting (red means worse; blue means better).

The family-specific curves further show that increasing budget does not produce uniformly increasing returns. For averaging, increasing the draw count sixteen-fold, from $K = 2$ to $K = 32$, raises the gain in 19 of 29 settings, but by only 0.02 plug-in IQRs in the median setting (well within rerun variability). Of the 16 settings that lose at $K = 2$, 14 still lose at $K = 32$, with the median setting recovering only 5% of its deficit. Particle returns vary substantially across settings without a consistent improvement with P, whereas selection responds most consistently to increasing N, although many of its gains remain within rerun variability.

Figure 4 illustrates this variability qualitatively across four text-to-image settings. Rows share a checkpoint (first/last), text prompt (second/last), or task (top two colorization; bottom two inpainting), yet the magnitude and relative pattern of gains noticeably differ. For example, selection performs poorly in the first row, while it produces the closest reconstruction to the ground truth in the third row. These examples show that the observed inconsistency in gain corresponds to visibly different final outputs even when important components of the generation setting are shared. Together, these sweeps characterize how each family responds to budget alone, but do not isolate what causes those responses.

4.2 Controlled Interventions on Scaling Mechanisms

The budget sweeps show that more samples alone do not reliably translate into higher final utility. Section 3 characterizes what sits between additional sampling and the final output: what each budget refines, and what must remain favorable for that refinement to be useful. We now intervene on these factors directly to test how changing them affects scaling effectiveness.

Averaging. Lemma 3.1 separates finite-sample variability from the local mean guidance: increasing K reduces the former without changing the latter. We therefore intervene on the draw population directly. On two FFHQ inpainting conditions, holding K and the averaging procedure fixed, we replace the Gaussian draws of Equation 3 with draws made by one stochastic sampler step (one-step transition draws) or by running the unguided sampler to the endpoint (differentiable continuations), so that the population-mean guidance itself changes. At every tested K, one-step transition draws remove most of the deficit of averaging Gaussian draws, showing that in these settings, changing the population being averaged has a significantly larger effect on final utility than increasing its size. Thus, changing the draw construction can make additional averaging budget useful when the population being averaged is limiting. Full experimental details are in Appendix E.3.

Particles. The target-law characterization in Lemma 3.2 identifies the sampler and weighting rule as the components that determine what increasing P approximates. We therefore intervene on these two components directly. First, holding the objective, particle operator, and P fixed, we only change the sampler. Resampling creates duplicate high-weight particles; stochastic transitions can separate these copies, whereas deterministic transitions cannot. On the SiT-XL/2 flow-matching checkpoint, particles beat the plug-in under the stochastic sampler in all three tested cells (two inverse cells and

one classifier-reward cell), but show no significant gain under the deterministic sampler. Consistent with this mechanism, in the two inverse cells, particles are separated at every recorded sampling step under the stochastic sampler, but at no more than 3.9% of recorded steps under the deterministic sampler. Second, we isolate the weighting rule while holding the sampler and P fixed. Importance weighting improves final utility beyond budget-matched greedy search in most inverse imaging cells, but provides no additional benefit on the flow-matching stack. Together, these interventions show that the sampler or weighting rule can determine whether scaling P is useful.

Selection. Lemma 3.3 shows that increasing N pushes selection further into the selector's top-ranked tail without changing the selector–utility relationship within that tail. We therefore test selector fidelity directly by holding the candidate population fixed while changing only the selector. In topology optimization, an exact oracle and a learned surrogate rank the same candidate populations. Oracle selection continues to improve as N grows, while surrogate selection saturates. This isolates top-tail fidelity as the limiting factor: a larger candidate pool helps only while the selector can reliably identify high-utility outputs in the increasingly extreme tail it selects from. Phase retrieval provides a contrasting case: despite low global rank agreement, final utility continues to improve as N increases, showing that global rank agreement may be too coarse to identify when selector fidelity becomes limiting. Together, these comparisons show that the return to scaling N depends on top-tail fidelity.

Across the three families, these controlled interventions identify the components that can make additional sampling budget translate more reliably into final-utility gains.

4.3 Empirical Consequences of Summary Insufficiency

Section 3.2 shows that common population summaries are generally insufficient to determine scaling outcomes. These constructions establish a theoretical blind spot, but do not imply that the omitted information is critical in realistic guided samplers. We therefore retrospectively evaluate rules that use these summaries to predict gain or no gain from additional sampling. Table 1 reports the fraction of correct predictions within each predicted class. A gain denotes an improvement exceeding the shared rerun band; no gain includes within-band improvements and losses. Diagnostic definitions, prediction rules, and evaluation details are provided in Appendix D.

For averaging and particles, the summary-based diagnostic rules predict no gain more precisely than gain, but much of this reflects the prevalence of no gain outcomes. Averaging's no-gain precision is slightly below that of always predicting no gain; particle separation is correct when it predicts no gain but captures only 6/12 of such cases. Meanwhile, all six of selection's gain predictions are correct, but it also misses just as many. Overall, these rules do not consistently identify which settings benefit from additional sampling, motivating analysis beyond these summaries.

Table 1: **Summary-based prediction precision.**

Summary	Precision for	
	No gain	Gain
averaging	22/23 (96%)	0/5 (0%)
particles	6/6 (100%)	3/9 (33%)
selection	11/17 (65%)	6/6 (100%)

5 Conclusion

We studied how averaging local draws, particle propagation and output selection spend test-time compute on training-free guidance, and found that scaling alone does not reliably translate into final-utility gain. In each strategy, a larger budget uses the consumed population more fully, while components that the budget leaves unchanged decide whether the final output improves. Our findings suggest that effective scaling begins with examining what limits the current sampling procedure. For averaging, this means checking how local draws are constructed; for particles, whether transitions maintain diversity and weights favor useful outputs; for selection, whether the highest-ranked candidates satisfy the intended quality criterion. When returns stall, changing these components may be more productive than increasing the sample count. Common population summaries may provide partial evidence about these limitations, while comparisons of final-output quality and computational cost remain essential for deciding whether to scale further.

Our analysis and experiments are limited to training-free guidance, and the recurring strategies we study do not exhaust test-time scaling. The distinction between what additional samples refine and what the underlying procedure leaves unchanged may extend beyond this scope. Testing this broader applicability across a wider range of population-based scaling methods and developing calibrated rules for when to scale versus modify the procedure are natural directions for future work.

AI USE STATEMENT

In this work, we used generative AI tools for coding assistance on experiment implementation and assistance in the validation of proofs. We have not used generative AI tools for developing theoretical models or conceptual frameworks, proposing or refining hypotheses, formulating mathematical claims, providing critical ingredients for proving mathematical claims, designing research methodology or experiments, interpreting results, and supporting qualitative and thematic data analysis. Generating synthetic datasets, assisting with translation, and cleaning and reformatting datasets are not applicable to this work. Additionally, we used generative AI tools for editing sections to improve readability and sourcing/searching for information. We have reviewed all AI-assisted work. At least two different authors have critically validated, tested, and further edited all generated code, writing and mathematical claims. We also manually verified the existence and content of all cited references. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M. Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, 32(3):405–431, 2017.

Aradhye Agarwal, Ayan Sengupta, and Tanmoy Chakraborty. The art of scaling test-time compute for large language models, 2025.

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Roni Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Jan-Hendrik Bastek, WaiChing Sun, and Dennis M. Kochmann. Physics-informed diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V. Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling, 2024.

Gabriel Cardoso, Yazid Janati El Idrissi, Sylvain Le Corff, and Eric Moulines. Monte Carlo guided denoising diffusion models for Bayesian linear inverse problems. In *International Conference on Learning Representations (ICLR)*, 2024.

George Casella and Roger L. Berger. *Statistical Inference*. Duxbury, Pacific Grove, CA, 2nd edition, 2002. ISBN 978-0-534-24312-8.

Sourav Chatterjee and Persi Diaconis. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Muthu Chidambaram, Khashayar Gatmiry, Sitan Chen, Holden Lee, and Jianfeng Lu. What does guidance do? a fine-grained analysis in a simple setting. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations (ICLR)*, 2023.

Gabriele Corso, Yilun Xu, Valentin De Bortoli, Regina Barzilay, and Tommi S. Jaakkola. Particle guidance: non-I.I.D. diverse sampling with diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Chenguang Dai, Jeremy Heng, Pierre E. Jacob, and Nick Whiteley. An invitation to sequential Monte Carlo samplers. *Journal of the American Statistical Association*, 117(539):1587–1600, 2022. doi: 10.1080/01621459.2022.2087659.

Herbert A. David and Haikady N. Nagaraja. *Order Statistics*. Wiley-Interscience, 3rd edition, 2003.

Pierre Del Moral. *Feynman–Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Probability and its Applications. Springer, New York, NY, 2004. doi: 10.1007/978-1-4684-9393-1.

Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1019–1028, 2017. URL https://proceedings.mlr.press/v70/dinh17b.html.

Arnaud Doucet and Adam M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In Dan Crisan and Boris Rozovskii (eds.), *The Oxford Handbook of Nonlinear Filtering*, pp. 656–705. Oxford University Press, Oxford, 2011. ISBN 9780199532902.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B. Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and MCMC. In *International Conference on Machine Learning (ICML)*, 2023.

Luca Eyring, Shyamgopal Karthik, Karsten Roth, Alexey Dosovitskiy, and Zeynep Akata. ReNO: Enhancing one-step text-to-image models through reward-based noise optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10835–10866. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23h.html.

Daniel Geyfman, Felix Draxler, Jan Niklas Groeneveld, Hyunsoo Lee, Theofanis Karaletsos, and Stephan Mandt. Calibrated test-time guidance for Bayesian inference. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=NZvWGf2XXM.

Aidan Gleich and Scott C. Schmidler. Multilevel and sequential Monte Carlo for training-free diffusion guidance. *arXiv preprint arXiv:2601.21104*, 2026.

Shivam Gupta, Ajil Jalal, Aditya Parulekar, Eric Price, and Zhiyang Xun. Diffusion posterior sampling is computationally intractable. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17020–17059. PMLR, 2024.

Jisung Hwang, Yunhong Min, Jaihoon Kim, I-Chao Shen, and Minhyuk Sung. NoiseTilt: Noise-tilted reverse kernels for diffusion reward alignment. In *European Conference on Computer Vision (ECCV)*, 2026.

Yazid Janati, Badr Moufad, Alain Durmus, Eric Moulines, and Jimmy Olsson. Divide-and-conquer posterior sampling for denoising diffusion priors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=SJgIPJBFvH.

Jaihoon Kim, Taehoon Yoon, Jisung Hwang, and Minhyuk Sung. Inference-time scaling for flow models via stochastic generation and rollover budget forcing. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025a.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In *International Conference on Learning Representations (ICLR)*, 2025b.

Augustine Kong. A note on importance sampling using standardized weights. Technical Report 348, Department of Statistics, University of Chicago, 1992.

Yi Liu. When can voting help, hurt, or change course? exact structure of binary test-time aggregation, 2026.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. RePaint: Inpainting using denoising diffusion probabilistic models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11461–11471, 2022.

Nanye Ma, Shangyuan Tong, Haolin Jia, Hexiang Hu, Yu-Chuan Su, Mingda Zhang, Xuan Yang, Yandong Li, Tommi Jaakkola, Xuhui Jia, and Saining Xie. Scaling inference time compute for diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2523–2534, 2025.

Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Ankur Moitra, Andrej Risteski, and Dhruv Rohatgi. The tractability landscape of diffusion alignment: Regularization, rewards, and computational primitives. *arXiv preprint arXiv:2605.11361*, 2026.

Vaishnavh Nagarajan and J. Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/05e97c207235d63ceb1db43c60db7bbb-Paper.pdf.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. WebGPT: Browser-assisted question-answering with human feedback, 2021.

Art B. Owen. *Monte Carlo Theory, Methods and Examples*. https://artowen.su.domains/mc/, 2013.

Vignav Ramesh and Morteza Mardani. Test-time scaling of diffusion models via noise trajectory search. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, 2025.

Minsi Ren, Wenhao Deng, Ruiqi Feng, and Tailin Wu. One step further with monte-carlo sampler to guide diffusion better. In *International Conference on Learning Representations (ICLR)*, 2026a.

Yinuo Ren, Wenhao Gao, Lexing Ying, Grant M. Rotskoff, and Jiequn Han. Driftlite: Lightweight drift control for inference-time scaling of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2026b.

Shivanshu Shekhar, Sagnik Mukherjee, Jia Yi Zhang, and Tong Zhang. Vasr: Variance-aware systematic resampling for reward-guided diffusion. *arXiv preprint arXiv:2604.06779*, 2026.

Yifei Shen, Xinyang Jiang, Yifan Yang, Yezhen Wang, Dongqi Han, and Dongsheng Li. Understanding and improving training-free loss-based diffusion guidance. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Raghav Singhal, Zachary Horvitz, Ryan Teehan, Mengye Ren, Zhou Yu, Kathleen McKeown, and Rajesh Ranganath. A general framework for inference-time scaling and steering of diffusion models. In *International Conference on Machine Learning (ICML)*, 2025.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2408.03314.

Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In *International Conference on Machine Learning (ICML)*, pp. 32483–32498, 2023.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 2020.

Bram Wallace, Akash Gokul, Stefano Ermon, and Nikhil Naik. End-to-end diffusion latent optimization improves classifier guidance. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7246–7256, 2023.

Nick Whiteley, Anthony Lee, and Kari Heine. On the role of interaction in sequential Monte Carlo algorithms. *Bernoulli*, 22(1):494–529, 2016. doi: 10.3150/14-BEJ666.

Luhuan Wu, Brian L. Trippe, Christian A. Naesseth, David M. Blei, and John P. Cunningham. Practical and asymptotically exact conditional sampling in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Yangzhen Wu, Zhiqing Sun, Shanda Li, Sean Welleck, and Yiming Yang. Inference scaling laws: An empirical analysis of compute-optimal inference for LLM problem-solving. In *International Conference on Learning Representations (ICLR)*, 2025.

Zihui Wu, Yu Sun, Yifan Chen, Bingliang Zhang, Yisong Yue, and Katherine L. Bouman. Principled probabilistic imaging using diffusion models as plug-and-play priors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a.

Yilun Xu, Mingyang Deng, Xiang Cheng, Yonglong Tian, Ziming Liu, and Tommi Jaakkola. Restart sampling for improving generative processes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023b.

Haotian Ye, Haowei Lin, Jiaqi Han, Minkai Xu, Sheng Liu, Yitao Liang, Jianzhu Ma, James Zou, and Stefano Ermon. TFG: Unified training-free guidance for diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. FreeDoM: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23174–23184, 2023.

Bingliang Zhang, Wenda Chu, Julius Berner, Chenlin Meng, Anima Anandkumar, and Yang Song. Improving diffusion inverse problem solving with decoupled noise annealing. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20895–20905, 2025.

Hongkai Zheng, Wenda Chu, Bingliang Zhang, Zihui Wu, Austin Wang, Berthy T. Feng, Caifeng Zou, Yu Sun, Nikola Kovachki, Zachary E. Ross, Katherine L. Bouman, and Yisong Yue. InverseBench: Benchmarking plug-and-play diffusion priors for inverse problems in physical sciences. In *International Conference on Learning Representations (ICLR)*, 2025.

APPENDIX CONTENTS

A Theory **15**

B Numerical checks of the theoretical statements **25**

C Experimental design and statistical analysis **27**

D Population measurements and small-scale evaluations **28**

E Results **31**

F Analysis specifications and deviations **47**

G Reproducibility details **48**

H Gain scale and variability band **49**

I Scope and limitations **51**

J Related Work **51**

A THEORY

This appendix follows Section 3: population-specific quantities, matched counterexamples, and the connection from per-step gradient MSE to terminal utility. Supporting derivations and attribution follow. Numerical checks and tests of predictions motivated by these results are in Appendix B.

The new results are Propositions 3.4–3.6 and Proposition A.5; Lemmas 3.1–3.3 and the remaining propositions restate or specialize established results, as Appendix A.5 details.

Notation and shared conventions. Throughout, q is a proposal distribution, $\ell \geq 0$ a measurable weight function with $0 < \mathbb{E}_q[\ell^2] < \infty$, and $\tilde{\pi}$ the tilted target with $\mathrm{d}\tilde{\pi}/\mathrm{d}q = \ell/\mathbb{E}_q[\ell]$. The order-two Rényi divergence and relative weight variance are $D_2(\tilde{\pi}\|q) = \log \mathbb{E}_q[(\mathrm{d}\tilde{\pi}/\mathrm{d}q)^2]$ and $\mathrm{RelVar} = \mathrm{Var}_q(\ell)/\mathbb{E}_q[\ell]^2$. Draws $x_1, \ldots, x_N$ are i.i.d. from q unless stated otherwise. Write $\mu_1 = \mathbb{E}_q[\ell]$, $\mu_2 = \mathbb{E}_q[\ell^2]$, $m_1 = \frac{1}{N}\sum_i \ell(x_i)$, and $m_2 = \frac{1}{N}\sum_i \ell(x_i)^2$. The ESS statistic of equation 10 is invariant to rescaling ℓ, so computations may use raw weights: $\mathrm{ESS}/N = m_1^2/m_2$. On the event that every weight is zero we set $\mathrm{ESS} = 0$; its probability decays geometrically in N and is absorbed into every $O(1/N)$ term below. Averaging uses the unweighted mean guide g_K^{mean} or the self-normalized tilt-weighted guide g_K^{tilt} in equation 3. Appendix A.1.1 characterizes both population means; Lemma 3.1's exact variance-over-K formula applies to the unweighted guide. Proposition 3.4 covers both guides. These gradient-MSE comparisons fix the sampling state and concern the guides before norm clipping or guidance-scale calibration; Appendix A.4.1 specifies those implementation details.

A.1 POPULATION-SPECIFIC QUANTITIES

A.1.1 AVERAGING: POPULATION MEANS AND GRADIENT MSE

Lemma 3.1 (restated). *Fix a sampling state and a deterministic reference guidance g^*. If $g^{(1)}, \ldots, g^{(K)}$ are i.i.d. with $\mathbb{E}\|g^{(i)}\|^2 < \infty$, mean $\bar{g} = \mathbb{E}[g^{(i)}]$, and covariance matrix $\mathrm{Cov}(g^{(i)})$, then for every $K \geq 1$:*

$$\mathbb{E}\|g_K^{\mathrm{mean}} - g^*\|^2 = \|\bar{g} - g^*\|^2 + \frac{\mathrm{tr}\,\mathrm{Cov}(g^{(i)})}{K}. \tag{11}$$

Proof. The second-moment condition makes every expectation below finite. Write $g_K^{\mathrm{mean}} - g^* = (\bar{g} - g^*) + (g_K^{\mathrm{mean}} - \bar{g})$. Expanding the squared norm,

$$\mathbb{E}\|g_K^{\mathrm{mean}} - g^*\|^2 = \|\bar{g} - g^*\|^2 + 2\big\langle \bar{g} - g^*,\ \mathbb{E}[g_K^{\mathrm{mean}} - \bar{g}]\big\rangle + \mathbb{E}\|g_K^{\mathrm{mean}} - \bar{g}\|^2,$$

where $\bar{g} - g^*$ is deterministic. The cross term vanishes because $\mathbb{E}[g_K^{\mathrm{mean}}] = \bar{g}$. For the last term, independence gives $\mathrm{Cov}(g_K^{\mathrm{mean}}) = K^{-1}\,\mathrm{Cov}(g^{(1)})$, and $\mathbb{E}\|Y - \mathbb{E}Y\|^2 = \mathrm{tr}\,\mathrm{Cov}(Y)$ for any square-integrable Y, which yields $\mathrm{tr}\,\mathrm{Cov}(g^{(1)})/K$. □

The unweighted identity is the standard bias–variance decomposition (Casella & Berger, 2002). Its mean $\bar{g}$ is also its almost-sure limit by the strong law of large numbers.

For the tilt-weighted extension, let $(w_i, g^{(i)})$ be i.i.d. under the draw distribution q, with $w > 0$, $0 < \mathbb{E}_q[w] < \infty$, $\mathbb{E}_q\|g\|^2 < \infty$ and $\mathbb{E}_q[w\|g\|] < \infty$. Then

$$g_K^{\mathrm{tilt}} \longrightarrow \bar{g}_w = \frac{\mathbb{E}_q[wg]}{\mathbb{E}_q[w]} \quad \text{almost surely.} \tag{12}$$

Indeed, the strong law gives $K^{-1}\sum_i w_i g^{(i)} \to \mathbb{E}_q[wg]$ and $K^{-1}\sum_i w_i \to \mathbb{E}_q[w] > 0$ almost surely. Dividing proves the claim, the standard ratio-estimator argument (Owen, 2013, Sec. 9.2). For this guide, define its finite-sample normalization bias by $b_K = \mathbb{E}[g_K^{\mathrm{tilt}}] - \bar{g}_w$. The corresponding exact identity is

$$\mathbb{E}\|g_K^{\mathrm{tilt}} - g^*\|^2 = \|\bar{g}_w - g^* + b_K\|^2 + \mathrm{tr}\,\mathrm{Cov}(g_K^{\mathrm{tilt}}). \tag{13}$$

Indeed, positive normalized weights make the guide a convex combination, so $\|g_K^{\mathrm{tilt}}\|^2 \leq \sum_i \|g^{(i)}\|^2$ and its second moment is finite. Expanding around its expectation then proves equation 13. Both b_K

and the covariance can depend on K. Almost-sure convergence alone does not assert convergence of MSE or a monotone reduction of either term.

The two limiting means depend on the draw distribution and the weighting rule. With exact posterior draws from a differentiable reparameterization, and differentiation justified under the expectation, the tilt-weighted mean equals g^*. An approximate draw distribution can leave a nonzero gap, as Proposition 3.4 demonstrates.

The following fixed-vector bound measures how much disagreement a nonnegative guidance-scale change can remove.

Lemma A.1 (Nonnegative scalar rescaling). *Let $u, w \in \mathbb{R}^d$ with $w \neq 0$. Then*

$$\min_{\lambda \geq 0} \frac{\|\lambda u - w\|^2}{\|w\|^2} = 1 - \max\big(\cos(u, w), 0\big)^2, \tag{14}$$

where $\cos(u, w) = \langle u, w\rangle / (\|u\| \, \|w\|)$ for $u \neq 0$ and $\max(\cos(u, w), 0)$ is read as 0 when $u = 0$. The minimizer is $\lambda^\star = \max(\langle u, w\rangle, 0)/\|u\|^2$ when $u \neq 0$, and every λ attains the value 1 when $u = 0$.

Proof. If $u = 0$ the ratio equals 1 for every λ, which is the right-hand side under the stated convention. Let $u \neq 0$ and put $\phi(\lambda) = \|\lambda u - w\|^2 = \lambda^2 \|u\|^2 - 2\lambda\langle u, w\rangle + \|w\|^2$, a strictly convex quadratic with unconstrained minimizer $\lambda_0 = \langle u, w\rangle / \|u\|^2$ and value $\|w\|^2 - \langle u, w\rangle^2/\|u\|^2 = \|w\|^2(1 - \cos^2(u, w))$. If $\langle u, w\rangle > 0$ then $\lambda_0 > 0$ is feasible and $\cos(u, w) > 0$, giving $\|w\|^2(1 - \max(\cos, 0)^2)$. If $\langle u, w\rangle \leq 0$ then ϕ is nondecreasing on $[0, \infty)$, so the constrained minimum is at $\lambda = 0$ with value $\|w\|^2$, and $\max(\cos(u, w), 0) = 0$ gives the same right-hand side. Dividing by $\|w\|^2 > 0$ completes the proof. □

The restriction $\lambda \geq 0$ matches deployed guidance scales; allowing negative scales instead gives $1 - \cos^2(u, w)$.

On released models, g^* is unknown. The logs measure disagreement with the plug-in, $\hat{B} = \|\bar{g} - g_{\text{plug}}\|^2 / \|g_{\text{plug}}\|^2$, which measures normalized squared bias when $g_{\text{plug}} = g^*$. Lemma A.1 bounds this fixed-vector disagreement after rescaling, without determining which direction improves utility. Numerical checks with known g^* and logged gradients are in Appendix B.1.

A.1.2 Particles: the Feynman–Kac output law

Lemma 3.2 is the general form of the statement below, which fixes the deployed increments and the final transition. The implementation evaluates potentials *before* its guided move. Index the guided Markov proposal without resampling by $X_0 \sim q_0$, $X_{k+1} \mid X_k \sim Q_k$, $k = 0, \ldots, S-1$, and let $\gamma_k(X_k) > 0$ be that pre-move potential. The simplified increment, attached to the moved particle with the previous potential carried through resampling, is

$$G_0^{\text{simp}} = 1, \qquad G_k^{\text{simp}} = \frac{\gamma_k(X_k)}{\gamma_{k-1}(X_{k-1})}, \quad 1 \leq k \leq S-1. \tag{15}$$

The transition-corrected variant additionally multiplies by the unguided-over-guided density ratio $p_k(X_{k+1} \mid X_k)/Q_k(X_{k+1} \mid X_k)$ where defined.

Proposition A.2 (Kernel-relative weighting and the final transition). *Assume $0 < Z = \mathbb{E}_q W < \infty$, where $W = \gamma_{S-1}(X_{S-1})/\gamma_0(X_0)$. The exact simplified-weight law just before the final move is*

$$\eta^-(\mathrm{d}x) = Z^{-1}\gamma_{S-1}(x)\mathbb{E}_q[\gamma_0(X_0)^{-1} \mid X_{S-1} = x] q_{S-1}(\mathrm{d}x).$$

The Feynman–Kac output law is its pushforward $\nu_S = \eta^- Q_{S-1}$, or equivalently

$$\nu_S(\mathrm{d}y) = Z^{-1}\mathbb{E}_q\left[\frac{\gamma_{S-1}(X_{S-1})}{\gamma_0(X_0)} \,\middle|\, X_S = y\right] q_S(\mathrm{d}y). \tag{16}$$

Any final deterministic clamp is a further pushforward, or part of Q_{S-1}. A state-independent initial potential simplifies the pre-move *law to $\eta^- \propto \gamma_{S-1} q_{S-1}$.*

Proof. The increments telescope along every ancestral path to W: $\prod_{k=1}^{S-1} G_k^{\mathrm{simp}} = \gamma_{S-1}(X_{S-1})/\gamma_0(X_0)$, the intermediate potentials cancelling in pairs, and $G_0^{\mathrm{simp}} = 1$ contributing the initial factor. Conditionally unbiased resampling on the augmented state (location and carried previous potential) preserves the unnormalized functional $\mathbb{E}_q[W\varphi(X_S)]$ by tower-property induction. The same holds when the resampling times are chosen adaptively, provided the trigger is measurable with respect to the particle filtration (Whiteley et al., 2016, Thm. 1); Dai et al. (2022, Sec. 2.4) state this for the ESS threshold. This is where the carried previous potential is needed, since the ratio in equation 15 refers to a state a resampling step may have replaced. Conditioning on X_{S-1} yields η^-; the last potential does not depend on the last innovation, so Q_{S-1} pushes it to ν_S. Conditioning instead on X_S gives equation 16. No finite-particle normalized unbiasedness or terminal-selection identity is inferred. □

Factoring an output likelihood $L(y)$ is justified when $\gamma_{S-1}(X_{S-1}) = L(X_S)$ almost surely, or under an explicitly declared endpoint approximation; an identity final kernel with a matching likelihood is sufficient. In general, the pre-move potential remains inside the conditional expectation given the post-move output.

Relation to the finite-P implementation. The law in equation 16 depends on the guided sampler's terminal marginal q_S, as well as its weights. The deployed finite-P population and its argmax-running-weight terminal pick require a separate analysis; Proposition A.2 identifies the exact reweighted law. Appendix B.2 gives numerical checks.

A.1.3 Selection: expected utility of the selected candidate

Lemma 3.3 (restated). *Let $(X_1, S_1), \ldots, (X_N, S_N)$ be i.i.d. pairs of a finished candidate and its selector score, with S_1 having a continuous distribution function F_S and with $\mathbb{E}|U(X_1)| < \infty$ for the evaluation utility U. Let $V_i = F_S(S_i)$ be the selector-rank quantile, so that V_i is uniform on $[0,1]$ and lower values of V are the selector's better ones, and let $m(v) = \mathbb{E}[U(X_1) \mid V_1 = v]$. Selecting the candidate with the smallest selector score returns $U_{\mathrm{sel}} = U(X_I)$ with $I = \arg\min_i V_i$, and*

$$\mathbb{E}[U_{\mathrm{sel}}] = N \int_0^1 m(v)\,(1-v)^{N-1}\,\mathrm{d}v. \tag{17}$$

The weight $N(1-v)^{N-1}$ is the $\mathrm{Beta}(1, N)$ density of the selected quantile, and it concentrates on the selector's best quantiles as N grows.

Proof. Continuity of F_S makes V_1 uniform on $[0,1]$ and gives $\Pr(V_i = V_j) = 0$ for $i \neq j$, so the minimizer I is almost surely unique and U_{sel} is well defined. The pairs are exchangeable, so

$$\mathbb{E}[U_{\mathrm{sel}}] = \sum_{i=1}^{N} \mathbb{E}\big[U(X_i)\,\mathbf{1}\{V_i < V_j\ \forall j \neq i\}\big] = N\,\mathbb{E}\big[U(X_1)\,\mathbf{1}\{V_1 < V_j\ \forall j \neq 1\}\big],$$

each summand being finite because $\mathbb{E}|U(X_1)| < \infty$, which also licenses the interchanges below. Condition on (X_1, V_1). The pairs (X_j, V_j), $j \neq 1$, are independent of (X_1, V_1) and each has $\Pr(V_j > v) = 1 - v$, so $\Pr(V_1 < V_j\ \forall j \neq 1 \mid X_1, V_1) = (1 - V_1)^{N-1}$. Hence

$$\begin{aligned}\mathbb{E}[U_{\mathrm{sel}}] &= N\,\mathbb{E}\big[U(X_1)(1-V_1)^{N-1}\big] = N\,\mathbb{E}\big[\mathbb{E}[U(X_1) \mid V_1]\,(1-V_1)^{N-1}\big] \\ &= N \int_0^1 m(v)(1-v)^{N-1}\,\mathrm{d}v,\end{aligned}$$

using the tower property and the uniform law of V_1. That $N(1-v)^{N-1}$ is the density of $\min_i V_i$, the first order statistic of N uniforms, is the $\mathrm{Beta}(1, N)$ law (David & Nagaraja, 2003, Ch. 2). □

Subtracting the expected utility of one unselected candidate, $\mathbb{E}[U(X_1)] = \int_0^1 m(v)\,\mathrm{d}v$, gives the selection gain at fixed test condition; averaging over test conditions gives the family's entry in equation 6. The comparator in equation 17 is therefore one draw from the same candidate population, not an arbitrary deployed baseline.

Oracle selection is the special case in which the selector is the utility itself.

Corollary A.3 (Oracle selection reads a tail quantile)**.** *In the setting of Lemma 3.3, let the candidate be summarized by a reward r with continuous, strictly increasing distribution function F on its support, lower values better, and let selection be oracle, $S_1 = r$. Then $V_1 = F(r)$, the selected reward is the sample minimum $r_{(1:N)}$, its distribution function is $1-(1-F(t))^N$, and its median is the tail quantile $F^{-1}\big(1-2^{-1/N}\big)$. The median gain over one unselected candidate, $F^{-1}(\frac{1}{2}) - F^{-1}\big(1-2^{-1/N}\big)$, is therefore positive for $N > 1$ whenever F separates those two quantiles, that is, whenever the terminal population has spread, and it grows with N through that quantile. No per-step quantity appears in these expressions. For higher-is-better scores replace the minimum by the maximum and $F^{-1}(u)$ by $F^{-1}(1-u)$, so that the quantile reads $F^{-1}(2^{-1/N})$.*

Proof. With $S_1 = r$ and F continuous, $F_S = F$ and $V_1 = F(r)$, so the selector's best quantile is the smallest reward and $\arg\min_i V_i$ selects $r_{(1:N)}$. Independence gives $\Pr(r_{(1:N)} > t) = (1-F(t))^N$, hence the stated distribution function; the median solves $1-(1-F(t))^N = \frac{1}{2}$, that is, $F(t) = 1-2^{-1/N}$, and strict monotonicity makes $t = F^{-1}(1-2^{-1/N})$ the unique solution. Since $1-2^{-1/N} = \frac{\log 2}{N}(1+O(1/N))$, the median selected reward lies at the $\approx (\log 2)/N$ quantile of F, and the median of one unselected candidate is $F^{-1}(\frac{1}{2})$. Every quantity is a functional of F alone. The higher-is-better statement follows by applying the same argument to $-r$. □

With an imperfect selector, returns depend on conditional utility $m(v)$, rather than the marginal spread alone.

A.2 Limits of diagnostic summaries

The following constructions hold population summaries fixed while changing gradient MSE, the effect of exact reweighting, or selection gain. Each isolates information missing from the summary.

Shared preliminaries. The summaries under test are built from the effective sample size ESS of equation 10 and from rank agreement. For i.i.d. draws from q with weights ℓ and $\mathbb{E}_q[\ell^4] < \infty$, the normalized ESS obeys $\mathrm{RelVar} = e^{D_2(\tilde\pi\|q)} - 1$ exactly and $\mathbb{E}[\mathrm{ESS}/N] = e^{-D_2(\tilde\pi\|q)} + O(1/N)$, as stated in Proposition A.6 and proved in Appendix A.4.2, which also records where the i.i.d. hypothesis fails. Two populations with the same complete weight law necessarily share every weight-only reading, including ESS at every sample size; two populations with the same D_2 share the asymptotic normalized ESS but not necessarily the same finite-sample distribution. The averaging construction below matches the stronger of these, the particle construction the weaker. Kendall's τ is the rank statistic $\tau = 4\iint C\,\mathrm{d}C - 1$ for a continuous copula C.

A.2.1 Averaging: the same weight law, opposite gradient-MSE orderings

Proposition 3.4 (restated). *There exist two smooth objectives J_A and J_B, a Gaussian denoising model and a sampling state such that (i) for every integer $K \geq 1$, the draw-weight vector $(w_1, \ldots, w_K)$ has the same joint law under J_A and J_B; and (ii) for some integer $K \geq 2$, both unclamped guides in equation 3 have strictly lower gradient MSE than the posterior-mean plug-in under J_A and strictly higher under J_B.*

Proof. Let $X_0 \sim N(0, \mathrm{diag}(9, 1/99))$ and $U = X_0 + \epsilon$, with $\epsilon \sim N(0, I_2)$ independent of X_0. Then $X_0 \mid U = u \sim N(Vu, V)$ with $V = \mathrm{diag}(0.9, 0.01)$. Both instances use the same posterior-mean denoiser Vu, state $u = 0$ and isotropic proposal $D(u, Z) = Vu + Z$, $Z \sim N(0, I_2)$. Set $\lambda = 1$ and $a = b = 0.1$, $J_A(x) = a - a\sin x_1 + (b/2)x_2^2$, and $J_B(x) = J_A(x_2, x_1)$. These objectives are smooth and nonnegative; $L_c = e^{-J_c}$ lies in $(0, 1]$. Coordinate swapping preserves the proposal and couples the two raw-weight vectors entry by entry, hence their entire weight laws and every weight-only reading coincide.

Define all derivatives in the same state coordinate: $g_c^\star = \nabla_u \log \mathbb{E}[L_c(X_0) \mid U = u]$, $p_c = \nabla_u \log L_c(Vu)$ and $g_{c,k} = \nabla_u \log L_c(Vu + Z_k)$, where the Z_k are i.i.d. $N(0, I_2)$ proposal draws held fixed when differentiating. The two estimators are

$$\hat g_{c,K} = K^{-1}\sum_k g_{c,k}, \qquad \tilde g_{c,K} = \frac{\sum_k L_c(Vu + Z_k) g_{c,k}}{\sum_k L_c(Vu + Z_k)}.$$

All expectations and MSEs below are evaluated at $u = 0$ against the corresponding $g_c^\star(0)$. For sinusoidal coordinate i and quadratic coordinate j, put

$$r(v) = \frac{\mathbb{E}[\cos X\, e^{a \sin X}]}{\mathbb{E}[e^{a \sin X}]}, \quad X \sim N(0, v), \qquad c_0 = e^{-1/2}, \quad s^2 = (1 + e^{-2})/2 - e^{-1}.$$

Differentiating the Gaussian location family and using symmetry gives $g_c^\star(0) = v_i a r(v_i) e_i$, $p_c(0) = v_i a e_i$, and $\mathbb{E}\hat{g}_{c,K}(0) = v_i a c_0 e_i$. Thus

$$R_{\text{plug}} = (v_i a)^2 [1 - r(v_i)]^2,$$
$$R_{\text{avg}} = (v_i a)^2 [c_0 - r(v_i)]^2 + \big((v_i a)^2 s^2 + (v_j b)^2\big)/K.$$

With $c_v = e^{-v/2}$ and $h = \cosh(a \sin X) - 1$, odd terms vanish, $0 \le h \le \delta = \cosh(a) - 1$, and

$$r(v) - c_v = \frac{\mathbb{E}[(\cos X - c_v)h]}{1 + \mathbb{E}h}, \qquad |r(v) - c_v| \le (1 + c_v)\delta, \qquad r(v) \le 1.$$

Set $K = 16$. These elementary bounds give $R_{\text{avg},A} < 1.14 \cdot 10^{-4} < 1.01 \cdot 10^{-3} < R_{\text{plug},A}$, whereas $R_{\text{avg},B} \ge 5.0625 \cdot 10^{-4}$ and $R_{\text{plug},B} < 2.25 \cdot 10^{-10}$.

For the weighted guide, write $Y_k = Z_{ki}$, $T_k = Z_{kj}$ and $t_k = \exp(a \sin Y_k - bT_k^2/2)$, so the raw weights are $w_k = e^{-a} t_k$. The gradients and their weighted population mean are

$$g_k = v_i a \cos Y_k\, e_i - v_j b T_k\, e_j, \qquad m = \frac{\mathbb{E}[tg]}{\mathbb{E}[t]} = v_i a r(1) e_i.$$

Let $h_k = g_k - m$, $\bar{t} = K^{-1} \sum_k t_k$ and $D_K = \mathbb{E}\|\tilde{g}_{c,K} - m\|^2$. The arithmetic–geometric mean inequality gives

$$\bar{t}^{-2} \le e^{2a} \exp\left(\frac{b}{K} \sum_k T_k^2\right), \quad D_K \le e^{2a} \mathbb{E}\left[e^{(b/K)\sum_k T_k^2} \left\| K^{-1} \sum_k t_k h_k \right\|^2\right].$$

For $K > 2b$, every off-diagonal term in this expansion vanishes: $\mathbb{E}[e^{bT^2/K} t(g - m)] = 0$. Its ith coordinate is zero by the definition of $r(1)$, and its jth coordinate is odd in T. Bounding the remaining diagonal terms using $t^2 \le e^{2a}$ and Gaussian exponential moments yields

$$D_K \le \frac{e^{4a}(1 - 2b/K)^{-K/2}}{K} \left[(v_i a)^2 \{s^2 + (c_0 - r(1))^2\} + \frac{(v_j b)^2}{1 - 2b/K}\right].$$

In case A, where $(v_i, v_j) = (0.9, 0.01)$, this gives $D_{16} < 1.7 \cdot 10^{-4}$. Applying the bound on $r(v)$ at $v = 1$ and $v = 0.9$ gives

$$|r(1) - r(0.9)| \le |e^{-1/2} - e^{-0.45}| + (2 + e^{-1/2} + e^{-0.45})\delta < 0.0474.$$

Hence $\|m - g_A^\star\|^2 < 0.09^2 (0.0474)^2 < 1.9 \cdot 10^{-5}$ and

$$R_{\text{tilt},A} \le 2D_{16} + 2\|m - g_A^\star\|^2 < 3.8 \cdot 10^{-4} < 1.01 \cdot 10^{-3} < R_{\text{plug},A}.$$

This bound does not require the finite-sample weighted estimator to have expectation m.

In case B, where $(v_i, v_j) = (0.01, 0.9)$, the exact correction's jth coordinate is zero. Since $0 < w_k \le 1$, the denominator $\sum_k w_k$ is at most K. Independence and oddness therefore give

$$R_{\text{tilt},B} \ge (v_j b)^2 \mathbb{E}\left[\left(K^{-1} \sum_k w_k T_k\right)^2\right]$$
$$= \frac{(v_j b)^2 e^{-2a} \mathbb{E}[e^{2a \sin Y}]}{K(1 + 2b)^{3/2}} \ge \frac{(v_j b)^2 e^{-2a}}{K(1 + 2b)^{3/2}} > 3.1 \cdot 10^{-4} > R_{\text{plug},B},$$

where $\mathbb{E}[e^{2a \sin Y}] = \mathbb{E}[\cosh(2a \sin Y)] \ge 1$. Thus the same objectives and $K = 16$ give both strict comparisons for both averaging guides.

For a variance-preserving state $X_t = U/\sqrt{2}$ the isotropic proposal scale is one and every gradient is multiplied by $\sqrt{2}$. The weights are unchanged, so all three MSEs double without changing their order. □

The coupling preserves every weight-only statistic at every K, yet both guides reverse their accuracy ordering against the posterior-mean plug-in. The weighted limit $v_i a r(1) e_i$ still reflects the isotropic proposal, whereas the exact correction is $v_i a r(v_i) e_i$. The opposite orderings therefore persist when weights enter the estimator.

A.2.2 PARTICLES: MATCHED ASYMPTOTIC OVERLAP, OPPOSITE WEIGHTING EFFECTS

Proposition 3.5 (restated). *There exist two samplers for a common prior and likelihood with the same asymptotic normalized* ESS*, for which exact likelihood reweighting moves the output mean strictly closer to the posterior mean in one case and strictly farther from it in the other.*

Proof. Take prior $p_0 = N(0,1)$ and likelihood $L(x) = e^{-(2-x)^2/2}$, giving posterior $\pi = N(1, 1/2)$. A guided kernel returns $X = (1-\alpha)Z + 2\alpha + \tau\varepsilon$, with independent standard normal Z, ε. Its output law is $q = N(m, v)$, where $m = 2\alpha$ and $v = (1-\alpha)^2 + \tau^2$. Weight this population by L, followed by an identity final transition. The exactly reweighted law $\eta \propto Lq$ has mean $(m + 2v)/(1 + v)$. Define its improvement in posterior-mean error by

$$H(q) = |m - 1| - \left|\frac{m + 2v}{1 + v} - 1\right|.$$

Gaussian integration gives $\mathbb{E}_q L^k = (1+kv)^{-1/2} \exp[-k(2-m)^2/(2(1+kv))]$. Hence the asymptotic normalized ESS is $e^{-D(m,v)}$, with

$$D(m, v) = \log \frac{1 + v}{\sqrt{1 + 2v}} + \frac{(2 - m)^2 v}{(1 + v)(1 + 2v)}.$$

For $\alpha = \tau = 0$, $q_A = N(0, 1)$ and $\eta_A = \pi$, so $H(q_A) = 1$ and $D(0, 1) = \log(2/\sqrt{3}) + 2/3 > 2/3$. For $\alpha = 1/2$, choose any $v \geq 1/4$ by setting $\tau = \sqrt{v - 1/4}$. Then $q_B = N(1, v)$ already has the posterior mean, and $H(q_B) = -v/(1 + v) < 0$. At the smallest variance,

$$D(1, 1/4) = \log \frac{5}{2\sqrt{6}} + \frac{2}{15} < \frac{1}{4} + \frac{2}{15} < \frac{2}{3} < D(0, 1).$$

The function $D(1, v)$ is continuous and tends to infinity as v grows. The intermediate value theorem therefore gives $v_* > 1/4$ with $D(1, v_*) = D(0, 1)$. These kernels have exactly equal asymptotic normalized ESS, distinct reweighted means, and strictly opposite improvements in posterior-mean error. □

This construction matches asymptotic normalized ESS and compares exact reweighted means. It varies the kernel's output variance, so particle geometry is not held fixed. A separate observation addresses what distances leave out.

Remark A.4 (Distances do not determine a weighted mean). Resampling creates copies, which carry uniform weights and maximal ESS even when all particles coincide, so pairwise separation adds information that ESS does not carry. It is not sufficient either. Take two distinct particle states $x_1 \neq x_2$ and normalized weights $(w, 1 - w)$ with $w \neq \frac{1}{2}$. Exchanging the weights to $(1 - w, w)$ leaves the multiset of pairwise distances unchanged and moves the weighted mean from $wx_1 + (1 - w)x_2$ to $(1 - w)x_1 + wx_2$, a displacement of $(1 - 2w)(x_1 - x_2) \neq 0$. Separation is therefore a statistic of the states alone, and the weighted output depends on the pairing of weights to states. A minimum pairwise separation near zero detects one close pair, which does not necessarily mean the population has collapsed. This concerns a weighted mean; its usefulness for predicting deployed gain is tested in Section 4.3.

A.2.3 SELECTION: MATCHED MARGINALS AND RANK AGREEMENT, OPPOSITE GAINS

Proposition 3.6 (restated). *There exist two populations of selector–utility pairs with continuous joint laws such that (i) the two share the same selector and utility marginals, hence the same utility spread, and also the same Kendall's τ; and (ii) for every $N \geq 3$, best-of-N selection strictly raises expected utility relative to a single candidate from the same population in one population and strictly lowers it in the other.*

Proof. On $[0, 1]^2$, define the density $c_\theta(u, v) = 1 + \theta f(u)g(v)$, where $f(u) = 6u^2 - 6u + 1$, $g(v) = 2v - 1$, and $|\theta| \leq 1$. Both functions integrate to zero and have absolute value at most one, so the density is nonnegative with uniform marginals. The selector U is lower-is-better and quality is $R = V$. For N i.i.d. candidate pairs, the regression is $m_\theta(u) = 1/2 + (\theta/6)f(u)$, hence

$$\Delta_N = \frac{\theta}{6}\mathbb{E}_{\mathrm{Beta}(1,N)} f(U) = \frac{\theta(N - 1)(N - 2)}{6(N + 1)(N + 2)}.$$

Thus $\theta = \pm 1/2$ gives strict opposite gains for $N \geq 3$. To verify exact rank agreement, set $F(u) = 2u^3 - 3u^2 + u$ and $G(v) = v^2 - v$; the copula is $C_\theta(u, v) = uv + \theta F(u)G(v)$. For continuous copulas $\tau = 4 \iint C_\theta c_\theta - 1$. The constant integral is $1/4$. The linear coefficient vanishes because $\int_0^1 uf(u)\,\mathrm{d}u = \int_0^1 F(u)\,\mathrm{d}u = 0$; the quadratic coefficient vanishes because $\int Ff = \int Gg = 0$, their primitives being one-half of squared functions that vanish at both endpoints. Thus $\tau = 0$ for every θ, while the finite-budget gains have opposite signs. □

Both populations have uniform marginals and random-candidate utility $1/2$. Their difference lies in conditional utility within the selector's best quantiles, increasingly emphasized by the $\mathrm{Beta}(1, N)$ weight.

What the three constructions establish. The matched summaries omit gradient information for averaging, the association between weights and outputs for particles, and utility in the selector's best-ranked tail. The comparisons concern fixed-state gradient MSE, exact reweighted means, and selected utility, respectively. Appendix B.4 gives numerical checks; Section 4.3 evaluates predictive usefulness.

A.3 From per-step gradient MSE to terminal utility

The next construction holds both terms of the per-step gradient MSE fixed at every step, yet reverses the terminal gain in a linear error model.

The error model. Fix a horizon $T \geq 1$, a step $\Delta t > 0$ and a rate $\lambda > 0$. Let the error $e_k \in \mathbb{R}^2$ of a sampling run relative to its target evolve by

$$e_0 = 0, \qquad e_{k+1} = Ae_k + \Delta t\,(\beta + \xi_k^{(K)}), \qquad A = \mathrm{diag}\big(e^{\lambda\Delta t},\, e^{-\lambda\Delta t}\big), \tag{18}$$

for $k = 0, \ldots, T-1$, where $\beta \in \mathbb{R}^2$ is the deterministic per-step guidance bias. Let $\xi_{k,j}$ be independent, mean-zero Gaussian draw noises with common covariance Σ, and define their average $\xi_k^{(K)} = K^{-1}\sum_{j=1}^K \xi_{k,j}$. The first coordinate is amplified and the second contracted. Averaging divides the per-step noise covariance by K and leaves β unchanged, so the two terms of equation 11 are

$$\|\beta\|^2 \quad \text{and} \quad \mathrm{tr}\,\Sigma/K. \tag{19}$$

Solving equation 18 gives $e_T = \sum_{k<T} A^{T-1-k}\Delta t(\beta + \xi_k^{(K)})$, so with $\varphi_k^{\pm} = e^{\pm\lambda\Delta t(T-1-k)}\Delta t$ the terminal error in the amplified coordinate is $\varphi^+ \cdot (\beta_1 \mathbf{1} + \xi_{\cdot,1}^{(K)})$ and in the contracted coordinate $\varphi^- \cdot (\beta_2 \mathbf{1} + \xi_{\cdot,2}^{(K)})$. The terminal metric is $\Psi(e) = -10\log_{10}\|e\|^2$, which is peak signal-to-noise ratio up to an additive constant fixed by the peak value and the dimension; that constant cancels in every gain below. We compare the K-draw-averaged run with a single noisy draw, $K = 1$, of the same construction. That comparator is not the deployed posterior-mean plug-in.

Proposition A.5 (The per-step gradient MSE trace does not identify the terminal sign)**.** *Let $\sigma, \beta_\star > 0$ and consider two instances of equation 18 with the same $T, \Delta t, \lambda$:*

> ***Cell A**: $\beta = (0, \beta_\star)$ and $\Sigma = \mathrm{diag}(\sigma^2, 0)$, so the floor lies along the contracted direction and the spread along the amplified one;*
>
> ***Cell B**: $\beta = (\beta_\star, 0)$ and $\Sigma = \mathrm{diag}(\sigma^2, 0)$, so both lie along the amplified direction.*

The two cells share both terms of equation 19 at every step and for every K, hence the same per-step gradient MSE trace $\beta_\star^2 + \sigma^2/K$. Write $V = \sigma^2\sum_k(\varphi_k^+)^2$ for the propagated spread, $B = \beta_\star\sum_k \varphi_k^+$ and $c = \big(\beta_\star\sum_k \varphi_k^-\big)^2$ for the propagated floors, and $G_\bullet(K) = \mathbb{E}[\Psi(e_T) \mid K] - \mathbb{E}[\Psi(e_T) \mid K = 1]$ for the gain of K-draw averaging over a single noisy draw. Then, for every $K > 1$:

(i) averaging strictly reduces expected squared terminal error in both cells, by $V(1 - 1/K)$;

(ii) $0 < G_A(K) \leq 10\log_{10} K$, and $G_A(K) \to 10\log_{10} K$ as $c/(V/K) \to 0$;

(iii) with $\rho = V/B^2$, if $\rho < \rho_\star := 1/(2t_\star^2) = 0.5854\ldots$, where $t_\star = 0.92413\ldots$ is the unique maximizer of Dawson's function $F(t) = e^{-t^2}\int_0^t e^{x^2}\,\mathrm{d}x$, then $G_B(K) < 0$: averaging strictly loses expected Ψ while, by (i), strictly winning expected squared error. As $\rho \to 0$,

$$G_B(K) = -\frac{10}{\ln 10}\Big[\Big(1-\frac{1}{K}\Big)\rho + \frac{3}{2}\Big(1-\frac{1}{K^2}\Big)\rho^2\Big] + O(\rho^3),$$

uniformly in $K > 1$.

The per-step gradient MSE trace therefore does not fix the terminal sign in this error model.

Proof. The shared trace is immediate: both cells have $\|\beta\|^2 = \beta_\star^2$ and $\operatorname{tr}\Sigma/K = \sigma^2/K$ at every step, by construction, and neither quantity records which coordinate carries which term.

Write $F_K = \sum_k \varphi_k^+ \xi_{k,1}^{(K)}$ for the propagated noise under K-draw averaging, a centered Gaussian with variance V/K, and note that the contracted coordinate carries no noise in either cell. Then

$$\|e_T\|^2 = \begin{cases} F_K^2 + c, & \text{cell A},\\ (B+F_K)^2, & \text{cell B}. \end{cases}$$

(i) $\mathbb{E}\|e_T\|^2$ equals $V/K + c$ in cell A and $B^2 + V/K$ in cell B, since $\mathbb{E}F_K = 0$ and $\mathbb{E}F_K^2 = V/K$. Both decrease in K by exactly $V(1-1/K)$ relative to $K = 1$.

(ii) In cell A, F_1 and $\sqrt{K}F_K$ have the same law, so

$$G_A(K) = \mathbb{E}\Big[10\log_{10}\frac{KF_K^2 + c}{F_K^2 + c}\Big].$$

The integrand is finite almost surely and bounded: for $c > 0$ and any $x \geq 0$, $1 \leq (Kx+c)/(x+c) \leq K$, the upper bound because $Kx + c \leq K(x+c)$. Hence $0 \leq G_A(K) \leq 10\log_{10} K$, and the lower bound is strict because the ratio inside the logarithm exceeds 1 on $\{F_K \neq 0\}$, an event of probability one. For the limit, set $w = F_K^2/(V/K) \sim \chi_1^2$ and $t = c/(V/K)$; the integrand is $10\log_{10}\big((Kw+t)/(w+t)\big)$, which increases to $10\log_{10} K$ pointwise on $\{w > 0\}$ as $t \downarrow 0$ and is dominated by $10\log_{10} K$, so monotone (or dominated) convergence gives $G_A(K) \to 10\log_{10} K$.

(iii) In cell B, $\Psi(e_T) = -10\log_{10}(B+F_K)^2$, so with $f(\rho) = \mathbb{E}\log\big((1+\sqrt{\rho}\,Z)^2\big)$ for $Z \sim N(0,1)$,

$$G_B(K) = \frac{10}{\ln 10}\big(f(\rho) - f(\rho/K)\big), \tag{20}$$

because $\mathbb{E}\log\big((B+sZ)^2\big) = \log B^2 + f(s^2/B^2)$ on factoring B out of the square. Since $\rho/K < \rho$, the sign claim is that f is strictly decreasing on $(0, \rho_\star)$, and the expansion is a Taylor expansion of f at 0. We establish both.

Put $\delta = 1/\rho$. Then $(1+\sqrt{\rho}\,Z)^2/\rho = (Z + \sqrt{\delta})^2$ is noncentral chi-square with one degree of freedom and noncentrality δ, which is χ^2_{1+2J} in law with $J \sim \mathrm{Poisson}(\delta/2)$, and $\mathbb{E}\log\chi_\nu^2 = \log 2 + \psi_{\mathrm{dg}}(\nu/2)$ for the digamma function ψ_{dg}. Hence

$$f(\rho) = \log\rho + \log 2 + \mathbb{E}_{J\sim\mathrm{Poisson}(\delta/2)}\big[\psi_{\mathrm{dg}}(J + \tfrac{1}{2})\big], \qquad \delta = 1/\rho. \tag{21}$$

The Poisson mixture is an entire function of δ, because $\psi_{\mathrm{dg}}(j+\frac{1}{2})$ grows logarithmically in j, so term-by-term differentiation is valid. Writing $m = \delta/2$ and using $\frac{\mathrm{d}}{\mathrm{d}m}\mathbb{E}[\varphi(J)] = \mathbb{E}[\varphi(J+1) - \varphi(J)]$ for $J \sim \mathrm{Poisson}(m)$ together with $\psi_{\mathrm{dg}}(x+1) - \psi_{\mathrm{dg}}(x) = 1/x$,

$$\frac{\mathrm{d}}{\mathrm{d}\delta}\mathbb{E}\big[\psi_{\mathrm{dg}}(J+\tfrac{1}{2})\big] = \tfrac{1}{2}\,\mathbb{E}\Big[\tfrac{1}{J+1/2}\Big], \qquad \text{so} \qquad f'(\rho) = \delta - \frac{\delta^2}{2}\mathbb{E}\Big[\tfrac{1}{J+1/2}\Big] = \delta\Big(1 - m\,\mathbb{E}\Big[\tfrac{1}{J+1/2}\Big]\Big).$$

With $m = t^2$, $\int_0^t x^{2j}\,\mathrm{d}x = t^{2j+1}/(2j+1)$ gives

$$m\,\mathbb{E}\Big[\tfrac{1}{J+1/2}\Big] = 2t^2 e^{-t^2}\sum_{j\geq 0}\frac{t^{2j}}{j!\,(2j+1)} = 2t\,e^{-t^2}\int_0^t e^{x^2}\,\mathrm{d}x = 2t\,F(t).$$

Dawson's function satisfies $F'(t) = 1 - 2tF(t)$ and has a unique stationary point $t_\star$, which is its maximum, so $2tF(t) > 1$ exactly for $t > t_\star$. Since $\delta > 0$, $f'(\rho) < 0$ exactly when $t = \sqrt{\delta/2} = \sqrt{1/(2\rho)}$ exceeds $t_\star$, that is exactly when $\rho < 1/(2t_\star^2) = \rho_\star$. So f is strictly decreasing on $(0, \rho_\star)$ and equation 20 is negative there.

For the expansion, write $s = \sqrt{\rho}$, $u = sZ$ and split on $\mathcal{A} = \{|Z| \le 1/(2s)\}$, on which $|u| \le \frac{1}{2}$ and $1 + u > 0$. There $\log(1+u)^2 = 2\log(1+u) = 2u - u^2 + \frac{2}{3}u^3 - \frac{1}{2}u^4 + \frac{2}{5}u^5 + r(u)$ with $|r(u)| = \left|2\sum_{k\ge6}(-1)^{k+1}u^k/k\right| \le \frac{1}{3}\sum_{k\ge6}|u|^k \le \frac{2}{3}|u|^6$. The odd moments of u vanish and $\mathbb{E}u^2 = \rho$, $\mathbb{E}u^4 = 3\rho^2$, $\mathbb{E}u^6 = 15\rho^3$, so the terms on $\mathcal{A}$ contribute $-\rho - \frac{3}{2}\rho^2 + O(\rho^3)$. The two contributions from $\mathcal{A}^c$ are $o(\rho^n)$ for every n: $\Pr(\mathcal{A}^c) \le 2e^{-1/(8\rho)}$ by the Gaussian tail bound, and Cauchy–Schwarz bounds each of them by $\Pr(\mathcal{A}^c)^{1/2}$ times a factor growing at most like $\log(1/\rho)$, since $\mathbb{E}\log^2\big((1+sZ)^2\big)$ is finite for every $s > 0$ and grows like $O(\log^2(1/s))$ as $s \downarrow 0$. Hence $f(\rho) = -\rho - \frac{3}{2}\rho^2 + O(\rho^3)$, and substituting into equation 20 gives the stated expansion, the remainder being uniform in $K > 1$ because $\rho/K \le \rho$. □

The second-order term is a fraction $\frac{3}{2}\rho(1 + K^{-1})$ of the leading term, which sets the numerical tolerance in Appendix B.5.

The trace omits the directions of bias and noise. In cell A, averaging reduces noise along the amplified direction while bias is contracted. In cell B, both lie along the amplified direction. Averaging reduces expected squared terminal error in both cells, but expected PSNR also depends on the error distribution through its logarithm. When $\rho < \rho_\star$ as in part (iii), the single draw's wider error distribution in cell B has higher expected PSNR, giving the opposite sign.

This is a linear error-model construction, with a single noisy draw as the comparator. It does not instantiate a posterior and objective or identify the cause of any measured reversal.

A.4 Supporting derivations

A.4.1 Guidance in the score and velocity parameterizations

A flow sampler consumes a velocity rather than a score. Its linear interpolant is $x_t = (1-\sigma_t)x_0 + \sigma_t\varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \boldsymbol{I})$ independent of x_0, $\sigma_0 = 0$, $\sigma_1 = 1$, and its velocity in the σ clock is $v(x_t) = \mathbb{E}[\varepsilon - x_0 \mid x_t] = -(x_t + \sigma_t\, s_t(x_t))/(1-\sigma_t)$, with s_t the score of the intermediate marginal. Shifting s_t by equation 1 therefore shifts the velocity by exactly $h^*(x_t) = -\sigma_t(1-\sigma_t)^{-1}\, g^*(x_t)$ for $\sigma_t < 1$; a schedule traversing σ_t in another time contributes its rate $\dot\sigma_t$ as a factor.

For the variance-preserving path $x_t = \sqrt{\bar\alpha_t}x_0 + \sqrt{1-\bar\alpha_t}\varepsilon$, the draw scale is $\sigma_{0|t} = \sqrt{1-\bar\alpha_t}$; on the linear flow path it is $\sigma_{0|t} = \sigma_t/\sqrt{(1-\sigma_t)^2 + \sigma_t^2}$. Both are the scale of Song et al. (2023): writing the path in its variance-exploding form $x_t = x_0 + s_t\varepsilon$, a standard normal prior on x_0 makes the posterior standard deviation $s_t/\sqrt{1+s_t^2}$, which is the expression above in each parameterization. These define the Gaussian approximation used by the arms, not exact posterior samples. The plug-in clamp rescales g_{plug} by $\min(1, r/\|g_{\text{plug}}\|)$, where r is the median unclamped norm over guided steps on validation conditions. Released arms displace the state along this calibrated approximation after each step, rather than directly integrating h^*; the scale combines step size, schedule factor and tilt strength. These are implementation details of the deployed samplers and are not assumptions of any statement in Appendices A.1–A.3.

A.4.2 Overlap, the favorable-set ceiling and the hit budget

Proposition A.6 (Per-population overlap identity)**.** *For the proposal and weights defined above,* $\mathrm{RelVar} = e^{D_2} - 1$ *exactly, and if in addition* $\mathbb{E}_q[\ell^4] < \infty$ *then* $\mathbb{E}[\mathrm{ESS}/N] = e^{-D_2(\tilde\pi\|q)} + O(1/N)$. *For tilts* $\ell = e^{-\lambda U}$ *with energy* $U \ge 0$ *vanishing exactly on a favorable set of q-mass* $p > 0$*, every strength* λ *obeys the ceiling* $\mathrm{RelVar} \le (1-p)/p$*, with equality in the sharp limit* $\lambda \to \infty$*; and* $N \ge \log(1/\eta)/p$ *draws contain a favorable one with probability at least* $1 - \eta$.

The divergence and importance-sampling connections are classical (Kong, 1992; Agapiou et al., 2017; Chatterjee & Diaconis, 2018). Lemmas A.7–A.10 prove the four claims, including the expectation bound for normalized ESS.

Lemma A.7 (Exact identity). $e^{D_2(\tilde{\pi}\|q)} = \mu_2/\mu_1^2 = 1 + \mathrm{RelVar}$, *so* $\mathrm{RelVar} = e^{D_2} - 1$.

Proof. $\mathrm{d}\tilde{\pi}/\mathrm{d}q = \ell/\mu_1$, so $\mathbb{E}_q[(\mathrm{d}\tilde{\pi}/\mathrm{d}q)^2] = \mu_2/\mu_1^2$, and $\mu_2/\mu_1^2 = 1 + \mathrm{Var}_q(\ell)/\mu_1^2 = 1 + \mathrm{RelVar}$. □

Lemma A.8 (ESS convergence, expectation form). $\mathrm{ESS}/N \to e^{-D_2}$ *almost surely. If additionally* $\mathbb{E}_q[\ell^4] < \infty$, *then* $\mathbb{E}[\mathrm{ESS}/N] = e^{-D_2} + O(1/N)$, *with the implied constant depending on the first four moments of* ℓ *under* q.

Proof. Almost-sure convergence is the strong law applied to m_1 and m_2: $m_1^2/m_2 \to \mu_1^2/\mu_2 = e^{-D_2}$.

For the expectation, let $\phi(a, b) = a^2/b$, so $\mathrm{ESS}/N = \phi(m_1, m_2)$, and note two a priori bounds: $\phi(m_1, m_2) \le 1$ always (Cauchy–Schwarz gives $m_1^2 \le m_2$), and $\mathbb{E}[m_i] = \mu_i$. Let $G = \{\, |m_2 - \mu_2| \le \mu_2/2 \,\}$. On G the pair (m_1, m_2) lies in the convex region $R = [0, \sqrt{3\mu_2/2}\,] \times [\mu_2/2,\ 3\mu_2/2]$, using $m_1 \le \sqrt{m_2}$ again, and on R the second derivatives of ϕ are bounded by a constant $C(\mu_2)$. Taylor's theorem with Lagrange remainder on R gives, on G,

$$\phi(m_1, m_2) = \frac{\mu_1^2}{\mu_2} + \frac{2\mu_1}{\mu_2}(m_1 - \mu_1) - \frac{\mu_1^2}{\mu_2^2}(m_2 - \mu_2) + Q, \qquad |Q| \le C(\mu_2)\big[(m_1 - \mu_1)^2 + (m_2 - \mu_2)^2\big].$$

Now take expectations of $\phi \mathbf{1}_G$ term by term. The linear terms have mean zero, so their contribution on G equals minus their contribution on G^c, which Cauchy–Schwarz bounds by $\sqrt{\mathrm{Var}(m_i)\ \Pr(G^c)}$; Chebyshev gives $\Pr(G^c) \le 4\,\mathrm{Var}(m_2)/\mu_2^2 = O(1/N)$, and $\mathrm{Var}(m_1) = \mathrm{Var}_q(\ell)/N$ and $\mathrm{Var}(m_2) = \mathrm{Var}_q(\ell^2)/N$ are $O(1/N)$ under the fourth moment, so each linear term contributes $O(1/N)$. The remainder satisfies $\mathbb{E}[|Q|\mathbf{1}_G] \le C(\mu_2)[\mathrm{Var}(m_1) + \mathrm{Var}(m_2)] = O(1/N)$. Finally $\mathbb{E}[\phi \mathbf{1}_{G^c}] \le \Pr(G^c) = O(1/N)$ since $\phi \le 1$, and $(\mu_1^2/\mu_2)\Pr(G^c) = O(1/N)$. Summing, $\mathbb{E}[\phi(m_1, m_2)] = \mu_1^2/\mu_2 + O(1/N)$. □

Lemma A.9 (Ceiling). *Let* $U \ge 0$ *have favorable set* $A = \{U = 0\}$ *with* $p = q(A) > 0$, *and let* $\ell_\lambda = e^{-\lambda U}$ *for* $\lambda \ge 0$. *Then: (i)* $\mathrm{RelVar}(\ell_\lambda) \le (1-p)/p$ *for every* λ; *(ii)* $\mathrm{RelVar}(\ell_\lambda) \to (1-p)/p$ *as* $\lambda \to \infty$; *(iii) conversely, any weight vanishing* q*-a.e. off* A *obeys* $\mathrm{RelVar} \ge (1-p)/p$, *with equality exactly for weights* q*-a.e. constant on* A.

Proof. (i) $0 \le \ell_\lambda \le 1$ with $\ell_\lambda = 1$ on A, so $\mu_1 \ge p$ and $\mu_2 \le \mu_1$. Hence $e^{D_2} = \mu_2/\mu_1^2 \le 1/\mu_1 \le 1/p$, and Lemma A.7 converts this to $\mathrm{RelVar} \le (1-p)/p$. (ii) $\ell_\lambda \to \mathbf{1}_A$ pointwise because $U > 0$ off A, and $0 \le \ell_\lambda \le 1$, so dominated convergence gives $\mu_1 \to p$ and $\mu_2 \to p$, hence $e^{D_2} \to 1/p$. (iii) If $\ell = \ell\,\mathbf{1}_A$, Cauchy–Schwarz gives $\mu_1^2 = \mathbb{E}_q[\ell\,\mathbf{1}_A]^2 \le p\,\mu_2$, so $e^{D_2} \ge 1/p$, with equality exactly in the constant case. □

The ceiling implies $p \le 1/(1 + \mathrm{RelVar})$ for these bounded tilts, the inversion used in the Darcy example (Appendix D.1.1).

Lemma A.10 (Favorable-hit budget). $\Pr(\textit{no draw among } N \textit{ lands in } A) = (1-p)^N \le e^{-Np}$, *which is at most* η *whenever* $N \ge \log(1/\eta)/p$.

Proof. Independence gives the product form; $1 - p \le e^{-p}$ gives the bound; solving $e^{-Np} \le \eta$ gives the budget. □

Remark A.11 (Instrument cap). With K normalized weights, $\sum_i \tilde{w}_i = 1$ and $\sum_i \tilde{w}_i^2 \in [1/K, 1]$, so $\mathrm{ESS} = 1/\sum_i \tilde{w}_i^2$ lies in $[1, K]$ as equation 10 says and the plug-in estimate $K/\mathrm{ESS} - 1$ of RelVar lies in $[0, K-1]$. A domain whose true relative variance exceeds $K - 1$ therefore cannot read above $K - 1$: the value is a censoring boundary, not a measurement. This is the cap at $K - 1 = 15$ behind the saturated readings of Appendix C.3 and the saturation diagnostics of Appendix D.

Remark A.12 (Where the identity applies, and where it does not). The overlap and ESS identities require i.i.d. samples and the specified tilt weights. Independent local draws with bounded reward weights satisfy these conditions. Independent terminal restarts do so only when supplied with tilt weights; rank-and-keep alone defines no ESS. Resampled trajectories are dependent, and their simplified weights define a kernel-relative Feynman–Kac law (Proposition A.2), rather than established posterior-importance weights.

Appendix B.6 checks these statements numerically, on the closed-form grid and, per population, on the two-population toy.

A.5 Relation to established results

The bias–variance decomposition (Casella & Berger, 2002), self-normalized importance-sampling consistency (Owen, 2013), Feynman–Kac weighting (Del Moral, 2004), order statistics (David & Nagaraja, 2003), and ESS identities (Kong, 1992; Agapiou et al., 2017) are established results. Our contributions here are the matched counterexamples and terminal-sign construction. Proposition A.2 specializes the reweighting identity to the implemented potentials and final transition.

Related analyses study posterior-sampling intractability (Gupta et al., 2024), reward-alignment tractability (Moitra et al., 2026), and departures from the intended target (Chidambaram et al., 2024). Our simplified particle weights differ from transition-corrected SMC (Wu et al., 2023); Appendix A.1.2 specifies their output law. Related guidance methods and cost comparisons are discussed in Appendix J.

B Numerical checks of the theoretical statements

This appendix follows the theory's order, separating numerical checks of identities and constructions from empirical predictions motivated by them. Failed predictions are reported with their checks; specifications and corrections are recorded in Appendix F.

B.1 Gradient MSE decomposition

In the toy model, where the exact correction g^* is known, the identity verifies to a worst case of 2.40 standard errors over nine (step, K) cells with a variance-term log-log slope of -1.003, and the draw population's Rényi-2 depth ranks the excess risk backwards (Spearman -0.94, against the geometry term's $+0.76$). On released models, the logs instead give the sampling variance and disagreement with the plug-in, $\hat{B}$, under the schedule-scale draw construction; they do not measure bias relative to g^*. A post-hoc comparison finds that at $K = 16$ the cell-median sampling-variance term is at most 6.1×10^{-3} of the squared plug-in gradient, while cell-median disagreement with the plug-in ranges over 0.52–1.02 of it; the smallest cell-median disagreement-to-variance ratio is 67.7. For the fixed gradients in Lemma A.1, the cell-median irreducible share of squared disagreement under nonnegative scalar rescaling ranges from 42.7% to 99.8% across all 10 logged cells. This is a fixed-vector comparison, not a bound on a retuned sampler's final output. The draw-weight summary enters neither term of the decomposition.

One of four prespecified predictions failed: excess MSE at the largest finite K did not equal the limiting bias difference. It omitted the remaining variance and uncertainty in the reference pool. A post-hoc check including both places every tested step within two standard errors.

B.2 Trajectory estimand

The numerical comparison supports the kernel-relative output law: the weighted population lands on the Feynman–Kac output law's moments to a worst case of 1.66 standard errors (the γ_{k_0} correction shifts the moment vector by 1.1×10^{-4}), while the transition-corrected bookkeeping, run on the same trajectories, lands on the posterior to 1.92 standard errors. Its reference weights use the pre-move terminal potential, while evaluation uses the post-move state, as in equation 16. The reported initial-potential correction does not justify replacing that pre-move potential by an output likelihood.

A post-hoc comparison illustrates the effect of the weighting convention: raising the guidance scale moves the simplified reading up (Spearman $+0.90$) and the corrected reading down (Spearman -0.90) on the same trajectories, the two coinciding identically at scale zero. The corrected weights reach the $1/P$ floor where the simplified weights give higher normalized ESS: the two readings measure different divergences on the same trajectories. Four of the check's six prespecified predictions failed. A post-hoc analysis attributes two failures to tail degeneracy of the normalizing-constant estimator under deep telescoping potentials; a higher ESS under simplified weights does not resolve that estimation problem. The other two failures concerned an expected log-likelihood ordering not implied by the identity and a kernel-relative ESS contrast tested in a regime where the readings repeatedly saturate.

B.3 TAIL-FIDELITY SELECTION LAW

In the numerical check, at $N = 16$ the top eighth of the selector's own ordering carries 88.2% of the selection weight (representation verified to 0.80 standard errors), and the discrete pool identity evaluates any stored pool deterministically. The comparative operational claim, that tail statistics out-predict a global rank correlation, is supported *in-sample only*. The held-out comparison did not support it: the prespecified tail statistic equalled 1.0 on all three held-out cells and could not distinguish them. Across the nine cell–budget combinations ($N = 4, 8, 16$), global τ had Spearman correlation 0.820 with gain efficiency; predictor values repeat across budgets within each cell. Mean-aggregated tail statistics computed post hoc matched that correlation at 0.820, without establishing a predictive advantage.

B.4 THE THREE MATCHED CONSTRUCTIONS

Numerical checks complement the constructions of Appendix A.2. The analytic witnesses and the terminal-sign construction of Appendix A.3 were developed after the empirical results.

Averaging (Proposition 3.4). A separate check evaluates the analytical bounds for the proof's exact-guidance witness. For the fixed-state construction at $K = 16$, analytical MSE bounds are below 1.14×10^{-4} for unweighted averaging and 3.8×10^{-4} for tilt-weighted averaging under J_A, versus a plug-in MSE above 1.01×10^{-3}. Under J_B, both averaging MSEs exceed 3.1×10^{-4} and the plug-in MSE is below 2.25×10^{-10}. Deterministic Gaussian quadrature independently checks the exact corrections. It also checks the weight-preserving coordinate swap and the variance-preserving reparameterization.

Particles (Proposition 3.5). Gaussian quadrature checks the proof's likelihood moments and reweighted mean; bisection checks the equal-overlap root and opposite signs. A separate numerical construction has one posterior and one likelihood, and changing only the guidance kernel moves the trajectory estimand to a total variation of 0.579. Its matched-overlap companion gives two guidance kernels on one posterior whose asymptotic trajectory readings e^{-D_2} match to machine precision (expected finite-P readings differ at $O(1/P)$), under which exact reweighting gains 0.687 and -0.725. The restricted-family check reports that the reading varies monotonically with the kernel parameter over 2001 tested points in the deterministic family; the matched-reading construction also varies injected noise. The companion matches only the asymptotic reading, and its payoff is the Feynman–Kac output law's mean, so the deployed resample-and-select operator lies outside its scope.

Selection (Proposition 3.6). Exact rational-polynomial tests verify the uniform marginals, $\tau = 0$, and the gain formula at $N = 1, 2, 3, 4, 8, 16, 32$; the proofs establish the claims over their full stated ranges.

B.5 TERMINAL-SIGN CONSTRUCTION

A fixed-seed simulation of equation 18 checks the identical per-step terms and opposite terminal signs: cell A gains within the bound $10\log_{10} K$, while cell B loses despite lower expected squared error. The numerical acceptance criteria are six Monte Carlo standard errors for each sign and a 25% relative tolerance for the leading-order cell-B loss. At the tested $\rho = 0.0434 < \rho_\star$, the second-order fraction $\frac{3}{2}\rho(1 + K^{-1})$ is 6.9%. The simulation checks this concrete instance; the proof establishes the signs over the full stated range.

B.6 THE OVERLAP IDENTITY

In the closed-form check, the population identity $(\mathbb{E}w)^2/\mathbb{E}[w^2] = e^{-D_2}$ has a maximum absolute error of 1.1×10^{-16} over 56 grid points (7 overlap levels $\times$ 8 guidance strengths). The ceiling of Proposition A.6 appears in the predicted windows, and the relative-variance ceiling matches $(1-p)/p$ exactly. The two-population toy verifies the same identity per population (population identity residual 1.7×10^{-17}; population ESS fraction 0.97956 vs mean Monte Carlo fraction 0.97955, relative error 4.7×10^{-6}).

C EXPERIMENTAL DESIGN AND STATISTICAL ANALYSIS

This appendix describes the testbeds and the paired-comparison protocol applied to them; experiment-specific analyses appear with their results. A prespecified analysis fixes its rule before examining the relevant evaluation outcomes. We use *registered* for rules recorded in a versioned pre-evaluation protocol; this does not imply public preregistration. Post-hoc analyses are chosen after outcomes are available. Appendix F collects specifications and deviations. Table 1 uses the retrospective binary assessment defined in Appendix E.7.1, with prediction rules defined in Appendix D.1. Comparisons lacking the required measurements are reported separately.

C.1 TESTBEDS

Table 2 summarizes the testbeds and the evidence each supplies.

Table 2: **Testbeds and evaluation roles.** Settings counts the budget-sweep settings in Figure 3. Prediction rules were developed on the first three testbeds and tested on the last two; trajectory recalibration and the post-hoc InverseBench statistic are described in Appendices E.6.3 and E.5.

Testbed	Domain	Models	Settings	Rule tests	Role in the evaluation
topology optimization	structural design, solver-evaluated compliance	1	1	developed	an exact oracle, so fixing the candidate pools and changing the selector attributes selection's gain (Appendix E.1)
FFHQ faces	pixel-space inverse problems, 256^2	1	4	developed	an independently released pipeline; the full arm matrix (Appendix E.2) and the draw-replacement tests (Section 4.2)
SiT-XL/2	class-conditional flow matching, 256^2	1	6	developed	two released samplers on one checkpoint, so a particle population can be sustained or collapsed (Section 4.2)
InverseBench	published benchmark: face inpainting, inverse scattering, Navier–Stokes	3	0	held out	externally specified tasks, operators and metrics (Appendix E.5)
text-to-image	latent inpainting and reward, 1024^2	15	18	held out	scale and breadth: three 8–20B text-to-image models and twelve further released checkpoints (Appendix E.6)
total		21	29		

C.2 COMPARISONS AND EVALUATION UNITS

Draw constructions. *Schedule-scale draws* perturb the clean prediction at the sampling schedule's noise scale and, on FFHQ, clip the result to the data range. *LGD-scale draws* use the smaller scale from loss-guided diffusion (Song et al., 2023); FFHQ draws are unclipped, while topology draws remain bounded to the design space. These names describe the draws; weighting and guidance calibration are specified separately for each comparison. In the arm labels, schedule-scale arms are `I2-`K or `A5-`K, and LGD-scale arms are `I2LU-`K on FFHQ, `A5L-`K in topology and `I2L-`K elsewhere. Figure 3 reports the LGD-scale configuration. Transition and continuation draws are defined in Appendix E.3.

Table 1 evaluates measured draw-floor predictions against LGD-scale outcomes. The measurement protocols and eligible settings are specified in Appendices D.1 and E.7.1. Schedule-scale and LGD-scale comparisons each use their own plug-in baseline.

A pilot is a small-scale evaluation that estimates gain before committing a full sampling budget; its samples count toward the compute budget. The primary topology comparison is the equal-one-call

test: one forward and backward surrogate evaluation per guided step, spent at the plug-in prediction or at one draw ($K = 1$), with the same guidance window.

For topology optimization, the paired unit is the test condition, summarized by the median over five shared seeds. Both arms use the same conditions and seeds. In the inverse-imaging and flow-matching comparisons, the paired unit is the image, with seeds aggregated first. Seed rows from one image are not independent test instances. The intervention bootstrap likewise resamples images with their seeds together (Appendix E.3); terminal-pilot coverage uncertainty uses the underlying candidate pool as its independent unit (Appendix D.2.1).

C.3 Hyperparameter selection and inference

For topology optimization, guidance strength and the guidance-active window are selected per arm by median objective on 200 validation conditions from the training split. Guidance strengths use a log grid around the published value. Test evaluation follows this selection; calibration procedures for the other testbeds are specified in their experimental subsections.

In topology optimization and inverse imaging, the paired comparisons use a two-sided Wilcoxon signed-rank test at $\alpha = 0.05$, with Holm correction within a comparison family. A paired sign test is reported alongside it; seed-majority and prediction-rule evaluations use the criteria stated with their results. Primary and sensitivity analyses are reported separately.

Pre-evaluation checks. The inverse-imaging and Darcy-flow studies used weight-variance checks before allocating sampling runs. Here relative variance is $K/\mathrm{ESS} - 1$, bounded by $K - 1$. In these checks, all 13 inverse-imaging cells reached the relative-variance cap of 15, with none below 1. The two initial Darcy configurations had maximum median relative variances 0.082 and 0.358. Their required threshold of 32 exceeded the instrument's 15 cap, making the gate infeasible. No upgrade was evaluated after the Darcy or molecular measurements, so those measurements contribute no scored gain predictions.

D Population measurements and small-scale evaluations

D.1 Population summaries and prediction rules

Table 3 summarizes the binary prediction rules assessed in Table 1, followed by the small-scale evaluation restrictions. Measurement definitions, logging schedules and costs follow; outcomes and baseline comparisons are in Appendix E.7.1. The rules use draw-weight concentration, particle separation and terminal spread as indicators of whether additional samples can help. The thresholds below specify a no-gain region and a gain region for each applicable reading. Eligibility requires a measured summary and an applicable numerical criterion; Appendix E.7.1 specifies the included settings.

D.1.1 Draw-weight concentration

Every mth step, evaluate K_d local draws $\hat{x}_0 + \sigma_{0|t}\epsilon$, weight them by the reward and record ESS/K_d. Defaults are $K_d = 16$ and $m = 25$, costing $K_d\lceil T/m \rceil$ objective evaluations for T steps. The scale $\sigma_{0|t}$ is the schedule-scale draw perturbation in the probe runs; the production reward settings use the LGD-scale calibration readings described below. The draw-floor criterion predicts no gain for $\mathrm{ESS}/K_d \le 1.5/K_d$ and predicts gain for $\mathrm{ESS}/K_d > 1.5/K_d$. The cutoff allows a 50% tolerance above the floor $1/K_d$, which corresponds to one effective draw; its no-gain region contains at most 1.5 effective draws at the reported aggregation level. This is an operational near-floor criterion. In the 38 logged measurements examined in Appendix E.6.2, no reading falls strictly above the floor and at or below the cutoff. Table 1 applies this criterion to 28 measured settings: four FFHQ, six SiT, twelve checkpoint-transfer and six production settings. The production reward readings come from the matched non-depicting settings' LGD-scale calibration runs, with $K_d = 16$ and reward-tilt temperature 1. The reading is the median across two calibration instances of the time-mean normalized ESS at the smallest valid tested guidance coefficient. All three exceed the draw-floor threshold and predict gain. For FFHQ, normalized ESS is recovered from the recorded relative weight variance as $1/(1 + \mathrm{RelVar})$ and summarized across the logged profile. The cutoff depends on the effective draw

Table 3: **Population summaries and small-scale gain evaluations.** The first three rows define the retrospective binary predictions assessed in Appendix E.7.1; the remaining rows specify pilot restrictions.

Measurement	Tested prediction rule or restriction
Draw-weight concentration (ESS/K_d)	no gain for $\mathrm{ESS}/K_d \leq 1.5/K_d$; gain for $\mathrm{ESS}/K_d > 1.5/K_d$. The applicable measurement cohort is specified below
Particle separation	fixed threshold: no gain for $a < 0.10$; gain for $a \geq 0.10$. Clone-calibrated: for $c > 0$, no gain for $d \leq 2c$ and gain for $d > 2c$. The readings a, d, c and positive-reference requirement are defined below
Terminal spread (relative IQR)	no gain for $r \leq 0.05$; gain for $r > 0.05$. The reading r and applicable PSNR cohort are defined below
Terminal pilot: bootstrap lower bound on gain	use $M \geq 12$ samples and $N \leq M/2$; otherwise abstain. The bound is calibrated empirically
Particle pilot: confidence sequence on paired gains	evaluate a fixed pilot size against held-out outcomes. The confidence sequence is asymptotic

count, so the same numerical criterion applies to these normalized ESS measurements; their different measurement protocol remains part of the evaluation. The three topology budgets lack matched draw-floor measurements and are excluded. The separate reward-loss hypothesis (Appendix E.6) does not supply a draw-floor prediction.

Overlap illustration. The extended Darcy likelihood-scale grid supplies the numerical example in Lemma A.9: $p_{\mathrm{fav}} \leq 1/(1 + \mathrm{RelVar}_{\max}) = 1/(1 + 9.64) = 0.094$. This grid extends the two initial configurations in Appendix C.3; it measures overlap without evaluating guidance upgrades. The two-level likelihood approximation makes the bound an illustrative estimate, rather than a measured gain.

D.1.2 Particle separation

Run P guided trajectories under the sampler's own propagation and resampling rules, with default $P = 8$. At each logged step, separation is the smallest max-absolute difference between two particle states, divided by the mean max-absolute state magnitude across particles (with denominator floor 10^{-12}). A small reading detects at least one near-duplicate pair; it does not establish whole-population collapse. Two separation criteria are used. For the fixed-threshold criterion, let a be the fraction of logged steps with separation above 10^{-4}. The criterion predicts no gain for $a < 0.10$ and predicts gain for $a \geq 0.10$. The separation cutoff 10^{-4} lies above the approximately 10^{-7} relative separation produced by numerical clone drift on the single-precision SiT stack and below its macroscopic stochastic separation. The 0.10 cutoff requires a near-duplicate pair at more than 90% of logged steps for a no-gain prediction; it is an operational persistence criterion.

For the clone-calibrated criterion, let d be median particle separation and c the median separation of initially identical particles propagated by the same sampler and numerical precision. For $c > 0$, the criterion predicts no gain for $d \leq 2c$ and predicts gain for $d > 2c$. The factor 2 treats separation within twice the numerical-clone level as near-duplication and adapts the cutoff to each sampler and numerical precision. The classifications on the four checkpoints with positive clone references are unchanged for multipliers in $\{1.5, 2, 5, 10\}$ (Appendix E.6.3). The criterion requires $c > 0$; the zero-reference SANA setting is therefore excluded from Table 1 (Appendix E.7.1).

The fixed criterion is used for the four SiT inverse settings and seven checkpoint-transfer settings; clone calibration is used for four further checkpoints with positive reference separation. These criteria assign no gain to the two SiT ODE inverse settings and to CogView4, Playground, Lumina2 and Chroma. The four FFHQ settings, five topology budgets and six production text-to-image settings lack measurements supporting these numerical separation criteria and are excluded from this prediction assessment. Obtaining this measurement requires particle trajectories and their backbone evaluations; the draw-weight measurement instead adds objective evaluations to an existing trajectory.

D.1.3 TERMINAL SPREAD AND RANK AGREEMENT

From available finished samples, compute the median across instances of metric IQR divided by absolute median. For a minimized selector objective S and maximized utility U, rank agreement is $\tau(-S, U)$, so positive values indicate agreement. Let r denote the relative-IQR reading. The spread-based criterion predicts no gain for $r \leq 0.05$ and predicts gain for $r > 0.05$. The cutoff uses 5% of the absolute median as the reference scale for a small interquartile width, summarized by the median relative IQR across instances. It is a heuristic for limited candidate variation under the stated metric convention. Rank agreement is measured separately to assess selector fidelity and does not set these binary predictions. The same threshold applies to all 23 PSNR settings in Table 1: four FFHQ settings, four SiT inverse settings, twelve checkpoint-transfer settings and three production inpainting settings. The three production reward settings use log-probability, for which this common PSNR cutoff is not applicable, and are excluded from selection scoring. FFHQ, SiT and production inpainting use three-seed plug-in candidate pools. For the twelve checkpoint-transfer settings, the reading is the median of three seed-level readings from the same 16-candidate pools as their selection outcomes. Pool sizes differ across cohorts; this descriptive retrospective assessment is not held-out calibration of a common threshold. These measurements require no new sampling cost only when their required samples are already available.

Dependence on the metric's zero point. Relative terminal spread is invariant to rescaling the metric but changes when a constant is added, because its denominator is an absolute median. An affine-invariant alternative did not reproduce all high-spread classifications across the 20 evaluated cells. The tested thresholds therefore apply to the stated metric conventions.

D.2 SMALL-SCALE GAIN EVALUATIONS

A pilot uses samples counted toward the compute budget to estimate gain before choosing a larger evaluation budget. The particle pilot compares paired whole-run outcomes on a subset of instances (Appendix E.7.2). The terminal pilot estimates selection gain from a candidate pool and abstains when its lower bound does not clear the prespecified threshold or the pool does not meet Table 3's size restrictions.

D.2.1 TERMINAL-SELECTION EVALUATION

The pilot uses $M = 12$ samples and selection budgets $N \in \{4, 6\}$. Its 640 resampled evaluations reuse 16 pools from 4 held-out implementations: each pool is one image's finished candidates, evaluated at both budgets, and each pool–budget case is piloted 20 times on a fresh random subsample, so a count of resampled evaluations is that many times a count of cases; empirical coverage is 0.953 at nominal 0.90. Appendix H gives pool-clustered uncertainty. Coverage by held-out implementation ranged from 0.919 to 0.975, and the median gap between the point estimate and lower bound was 0.089 dB. The bound was correct on 72 of 73 gain predictions, with 1 incorrect. The full-pool reference gain exceeded the 0.75 dB threshold in 200 resampled evaluations. The bound identified 36% of qualifying gains, compared with 5.5% for the relative-IQR comparator using the same pilot samples.

Under the operating restriction, best-of-16 needs $M \geq 32$ pilot samples. The prespecified global-τ comparator was not scored in the held-out run, a deviation from the plan. Separately, the held-out test of whether tail statistics predict gain better than global τ was unsuccessful, so that advantage remains an in-sample finding (Appendix B.3).

D.3 LOGGING AND MEASUREMENT CHECKS

Logged quantities. Instrumented runs log, per step and condition, the draw reading ESS/K (a softmax over the K proposals' log-likelihoods), the fraction of draws clipped to the data range and, on guided steps, the gradient norm. Trajectory runs also log the pre-resample ESS/P, the resampling pattern and the particles' minimum relative pairwise separation. Terminal statistics are computed afterwards from finished samples.

Distinct-particle rule. Trajectory ESS is recorded on distinct-particle steps only. The step immediately after a resample is excluded, because the resampled particles are bit-identical there and $\mathrm{ESS}/P \equiv 1$ measures the resampler rather than the population. The exclusion lasts one step.

Instrumentation subset. Every instrumented analysis that does not use the full test split uses the first 50 test conditions in the released ordering, as specified in advance.

Clipping and saturation. Clipping creates point masses at the data boundary and can change agreement statistics, so the clipped fraction is logged at each step. At the $\mathrm{ESS} = 1$ floor, relative weight variance $K/\mathrm{ESS} - 1$ reaches its $K - 1$ upper bound and cannot distinguish more concentrated weight laws. The variance of the log weights supplies an additional measurement without this bound.

Terminal measurement inputs. The terminal reading uses plug-in outputs only: in topology optimization, the plug-in arm's fixed test seeds, rescored by the clean surrogate on the stored designs; in inverse imaging, three independent plug-in seeds per instance. It reads no selection-arm output. Simulated selection from these plug-in samples is checked against the selection-arm median measured independently from its stored outputs.

E Results

E.1 Topology optimization

The topology-optimization testbed supplies a solver-evaluated compliance oracle. Its stored terminal populations let us compare learned and oracle selectors on the same candidates. Table 5 lists the schedule-scale per-step upgrades and controls on 1,000 test conditions. Terminal selection and LGD-scale averaging are separate comparisons. Appendix G describes the data and evaluator.

E.1.1 Per-step upgrades

Table 5 reports the schedule-scale arms and their validation-selected strengths and guidance windows. Larger draw budgets do not produce a monotone improvement in median compliance error. At $K = 4$, both paired tests favor averaging (sign $p = 7.2\times10^{-3}$; Wilcoxon $p = 4.6\times10^{-4}$), although its condition-first median compliance error is higher than the plug-in's. The Wilcoxon result survives Holm correction within the averaging family, but neither test clears the record-wide Bonferroni reference 4.1×10^{-4}. The paired sign and Wilcoxon tests disagree at the nominal level for four arms: averaging (K=16) (sign 0.21, Wilcoxon 0.021); particles (P=8) (sign 0.082, Wilcoxon 0.034); particles (P=32) (sign 0.046, Wilcoxon 0.077); no net material (sign 0.039, Wilcoxon 0.57). No particle arm has a significant Holm-corrected advantage over the plug-in.

A separate sensitivity study crosses guidance strength and draw budget for the schedule-scale construction on the 50-condition instrumentation subset: 30 cells crossing 5 tilt strengths λ from 0.25 to 4 with 6 budgets from $K = 1$ to $K = 32$, none of which shows a significant difference in either direction after Holm correction. The predicted crossover favoring averaging was not observed. The corresponding equivalence analysis finds that in every one of the 30 cells the 90% interval rules out an effect larger than 0.0097 CE%, which is 0.16 of the plug-in's own gain over unguided sampling (0.0603 CE%). These are pointwise 90% boundary-condition-cluster bootstrap intervals, rather than a simultaneous bound over the grid. Within this tested grid, extra draws add little to the improvement already obtained by plug-in guidance.

E.1.2 Selection

The measured arms show selection's ceiling: best-of-4 beats the plug-in on 56.6% of conditions (sign $p = 3.4\times10^{-5}$) and best-of-16 on only 53.7% ($p = 0.02$). In a separate analysis of subsets of the plug-in's five stored candidates per condition, the paired gap between the surrogate's pick and the oracle's increases: from 0.159 $[0.134, 0.182]$ at $N = 2$ to 0.230 $[0.202, 0.270]$ at $N = 4$ (paired 90% intervals). An oracle re-selection sweep holds the population fixed at each N (Table 4): selecting with the surrogate turns back at the largest N, while selecting with the exact oracle on the same populations keeps improving, so the gap measures selector fidelity alone. The prespecified reproduction of the pilot readings (spread 0.854 and fidelity $\tau = 0.140$) failed: rank agreement

Table 4: Selection from fixed topology candidate pools (CE%, lower is better). Uniform pick is the median across conditions of each pool's mean CE; the other columns summarize the surrogate and oracle picks from the same pools. Pools differ across N. Spread is relative IQR; τ is pooled rank agreement.

N	Uniform pick	Surrogate	Oracle	Spread	τ
2	0.458	0.350	0.196	0.456	0.220
8	0.547	0.304	−0.001	1.057	0.139
32	0.542	0.291	−0.120	1.214	0.134
64	0.540	0.310	−0.193	1.216	0.130

Table 5: Schedule-scale arms and controls on the topology-optimization benchmark, 1,000 test conditions, condition-first medians. CE and VFE use the released 0.5-threshold evaluation (Appendix E.1.3). Window and λ are selected per arm on validation; Steps counts guidance-active steps, and rows have different costs. The two plug-in comparisons use the sign and primary Wilcoxon tests; the unguided comparison uses the sign test.

	Arm	CE%	VFE%	Window	λ	Steps	p vs plug-in		p vs unguided
							sign	Wilcoxon	sign
baselines	unguided	0.443	-0.116	all	0	100	5.9×10^{-24}	9.0×10^{-21}	
	noisy-state guidance	0.440	-0.106	all	1	100	2.2×10^{-20}	3.6×10^{-20}	0.87
	plug-in (unclipped)	0.433	-0.058	last 71	0.25	71	2.8×10^{-9}	4.4×10^{-11}	9.2×10^{-14}
published	published AMORT	0.445	-0.116	last 71	0.25	71	2.7×10^{-21}	3.6×10^{-23}	0.55
plug-in	plug-in	0.404	-0.046	all	1	100			5.9×10^{-24}
averaging	averaging (K=1)	0.412	-0.024	all	1	100	0.72	0.55	8.4×10^{-21}
	averaging (K=2)	0.415	-0.038	last 71	1	71	0.39	0.40	6.6×10^{-21}
	averaging (K=4)	0.415	-0.031	all	4	100	7.2×10^{-3}	4.6×10^{-4}	6.5×10^{-21}
	averaging (K=8)	0.415	-0.046	last 71	0.25	71	0.47	0.44	1.4×10^{-18}
	averaging (K=16)	0.409	-0.034	all	1	100	0.21	0.02	5.1×10^{-26}
	averaging (K=32)	0.414	-0.046	last 71	2	71	0.73	0.13	4.1×10^{-20}
particles	particles (P=2)	0.436	-0.008	all	2	100	0.43	0.71	8.6×10^{-3}
	particles (P=4)	0.408	-0.062	last 71	0.25	71	0.97	1.00	0.15
	particles (P=8)	0.447	-0.008	last 71	0.25	71	0.08	0.03	0.68
	particles (P=16)	0.440	-0.013	last 71	1	71	0.55	0.95	0.19
	particles (P=32)	0.441	-0.055	last 71	4	71	0.05	0.08	0.97
controls	volume-only	0.412	-0.038	all	0.25	100	0.55	0.47	7.8×10^{-19}
	no net material	0.407	-0.046	all	1	100	0.04	0.57	4.1×10^{-17}

The noisy-state control evaluates the clean surrogate on the noisy input. The unclipped plug-in omits the update's norm clip. Volume-only retains only the volume correction; no net material removes net material change from the update. Window specifies all 100 steps or the last 71. Objective and backbone counts are in Appendix G.

reproduced within tolerance, but spread did not. The measured best-of-4 median, 0.3116, falls inside its simulated 90% interval, $[0.260, 0.387]$, in the arm's own estimand.

E.1.3 Surrogate and metric checks

Both compliance surrogates were trained on an augmented corpus, chosen before sampling began, and checked for sensitivity to the design. Without augmentation, the one-design-per-problem corpus permits a conditioning shortcut: accurate compliance predictions with almost no gradient with respect to the design.

Compliance error (CE) and volume-fraction error (VFE) use the released 0.5-threshold evaluation. Projecting designs to a fixed volume before scoring can sever load paths and produce a bimodal error distribution. This motivated changing the primary metric from volume-projected compliance to thresholded compliance; Appendix F records the timing and original specification.

E.2 Image inverse problems

Protocol. We use the released DPS FFHQ-256 ADM checkpoint at its pinned commit, with the released guidance rule, step sizes, 100-step respacing and operators. Weight measurements cover

Table 6: Primary inverse-imaging arm matrix: summary medians of the measurement residual (res.) and of PSNR in dB, with model-generated ground truths, 64 images and three seeds per cell. Calls are backbone calls, not objective evaluations. The lower block gives the weighted-particle arms' realized diagnostics.

		S		M		D		PR	
arm	Calls	Res. ↓	PSNR ↑	Res. ↓	PSNR ↑	Res. ↓	PSNR ↑	Res. ↓	PSNR ↑
I0	100	131.46	9.22	238.83	9.22	290.78	9.22	168.79	9.22
I1	100	95.12	18.11	47.95	24.09	45.45	25.37	57.99	11.17
I2-4	100	124.92	10.02	195.65	10.93	232.11	11.18	177.93	8.05
I2-16	100	125.48	9.96	193.74	11.02	227.95	11.33	178.25	8.14
I3-4	400	93.26	19.52	44.59	24.87	42.46	25.96	56.30	9.89
I3-16	1600	92.25	20.52	42.16	25.47	39.75	26.53	55.01	9.83
I4-4	400	95.17	17.99	47.69	24.13	44.72	25.53	57.00	10.19
I4-16	1600	95.26	18.00	48.42	24.02	45.74	25.31	57.63	10.75
LGD-scale draws; separately paired plug-in									
I1 (LGD scale)	100	95.42	17.91	48.56	23.94	45.47	25.35	57.57	10.46
I2LU-4	100	97.48	16.69	77.16	19.34	85.14	19.89	75.49	9.68
I2LU-16	100	97.37	16.73	77.09	19.36	84.61	19.94	75.59	10.90

diagnostic	arm	S	M	D	PR
median ESS/P	I3-4	0.4999	0.5000	0.5000	0.5000
median ESS/P	I3-16	0.2500	0.2153	0.1278	0.1989
resampling rate	I3-4	0.4998	0.4899	0.4861	0.4935
resampling rate	I3-16	0.5842	0.6028	0.6130	0.5985

Arm codes: I0 unguided; I1 plug-in; I2-K averaging over K projected draws and I2LU-K over K LGD-scale draws; I3-P weighted particles, P particles; I4-P greedy replicate-after-selection control, P particles. S, M and D are inpainting cells of increasing measurement informativeness, and PR is phase retrieval. Schedule-scale and LGD-scale configurations have separate plug-in baselines; comparisons pair arms within each block, never across experiments. Realized ESS/P and resampling rates describe the unchanged particle arms.

13 conditions: a 4×3 inpainting grid of mask-keep fraction and measurement noise, plus phase retrieval. The grid uses keep fractions $\{0.1, 0.3, 0.6, 0.9\}$ and noise levels $\{0.01, 0.05, 0.2\}$. The evaluated settings S, M and D use (keep fraction, noise) $(0.1, 0.2)$, $(0.6, 0.05)$ and $(0.9, 0.01)$, respectively; PR uses phase retrieval with noise 0.05. These four settings span 6.6 orders of magnitude in measured log-weight variance: three inpainting conditions and nonlinear phase retrieval. Each reuses the measured condition's mask and noise realization at the same seed. Additional terminal-spread measurements on super-resolution, motion blur and saturation appear in Table 21.

Ground truth and metric audit. The primary matrices use 64 model-generated ground truths, unconditional samples at a fixed seed, so absolute PSNR is not directly comparable to published DPS evaluations. The comparisons use paired gains on the same reference images. An audit re-runs every headline comparison on PSNR with the same paired tests: 36 comparisons agree in direction out of 40; linear cells 29/30; PR cell 7/10; 4 direction flips, in M/PR. Because the flips are confined to the phase-retrieval regime and its neighbor, phase retrieval's per-step weighting results are stated as optimization-landscape findings.

Primary arm matrix. Table 6 reports medians, backbone costs and realized particle statistics for the primary comparisons.

Averaging arms. With the projected-draw construction fixed, even after retuning λ on a logarithmic grid for each K, averaging's best PSNR over (λ, K) is 2.46 to 28.33 dB below the plug-in's best over λ across the four cells. This tests guidance-scale sensitivity; draw replacements are evaluated separately in Appendix E.3. Averaging losses concern the projected-draw configuration; the LGD-scale rows in Table 6 use their own plug-in baseline.

Estimator controls. Table 7 compares three averaging rules on the projected-draw construction. For each image, let

$$z_i = \mathrm{clip}_{[-1,1]}\big(\hat{x}_0(x_t) + \tau_t \varepsilon_i\big), \qquad \varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, I), \qquad \tau_t = \sqrt{\frac{1-\bar{\alpha}_t}{\max(\bar{\alpha}_t, 10^{-8})}}.$$

Write $\rho_i = \|y - A(z_i)\|_2$ and $\alpha_i = \exp[-\rho_i^2/(2\sigma_y^2)]/\sum_j \exp[-\rho_j^2/(2\sigma_y^2)]$. The likelihood-weighted residual-norm control and the negative log-mean-likelihood gradient are, respectively,

$$h_R = \sum_i \alpha_i \nabla_{x_t} \rho_i,$$

$$h_L = -\nabla_{x_t} \log\left[\frac{1}{K}\sum_i e^{-\rho_i^2/(2\sigma_y^2)}\right] = \sum_i \alpha_i \frac{\rho_i}{\sigma_y^2} \nabla_{x_t} \rho_i.$$

The weights in h_R are held fixed during differentiation; the unweighted control uses $K^{-1}\sum_i \nabla_{x_t}\rho_i$. For a common objective in both the weights and gradients, Equation 3's tilt-weighted guide is exactly the gradient of the log-mean-exponential objective. Thus $-h_L$ instantiates that guide for $\lambda J(z) = \|y - A(z)\|_2^2/(2\sigma_y^2)$, while h_R weights residual-norm gradients with squared-residual likelihood weights.

The third rule matches h_L to the plug-in's per-image gradient norm:

$$h_L^{\mathrm{n}} = h_L \frac{\|h_{\mathrm{plug}}\|_2}{\max(\|h_L\|_2, 10^{-12})}, \qquad h_{\mathrm{plug}} = \nabla_{x_t} \|y - A(\hat{x}_0(x_t))\|_2.$$

Each rule uses the update $x_{t-1} = x_{t-1}^{\mathrm{prop}} - sh$, with the released step coefficient $s = 0.5$ for inpainting and $s = 1$ for phase retrieval. The arms share the draw construction but use separate trajectory and draw streams. Matching to the plug-in norm does not match the norm of h_R, so differences between h_L^{n} and h_R do not isolate a directional effect. The likelihood-weighted residual-norm control h_R wins against the plug-in in 0 of 64 images in all 8 rows (four cells $\times$ $K \in \{4, 16\}$; Holm p at most 8.7×10^{-19} throughout), with median residual deficits of 31 to 189. Against the unweighted mean using the same draw construction, no row reaches statistical significance: all 8 rows have Holm $p = 1.00$, with win counts of 26 to 34 of 64. The plug-in-norm-matched control h_L^{n} wins against the plug-in on 0 of 64 images in each of 4 cells at both budgets (Holm p at most 8.7×10^{-19}). The unnormalized control h_L also loses in all 8 comparisons; these unmatched outcomes are reported without prediction scoring (Appendix F). The h_L and h_R controls are statistically separated in all 8 comparisons, reported without prediction scoring. Matching h_L to the plug-in norm improves its outcome on 100% of images in every comparison. The resulting h_L^{n} is not statistically separated from h_R in S and M but remains separated in D and PR. These comparisons concern different gradient objectives and normalization rules, not two forms of the same estimator.

Particle controls. These controls distinguish the contribution of weight assignment from resampling. Against the greedy control (Table 7), the weighted arm wins in every linear cell at both budgets (Holm $p \le 2.8\times10^{-17}$) and in phase retrieval at $16\times$; at $4\times$ the phase-retrieval result is marginal, significant on the sign test but not on the Wilcoxon test; that last phase-retrieval comparison favors greedy search on PSNR at both budgets (Table 6). Shuffling the weights across particles at every selection keeps their values but breaks their link to the particles, and against that control the weights carry information: the weighted arm wins in all six linear-cell comparisons (Holm p at most 2.8×10^{-17}) and, where no direction had been predicted, in both phase-retrieval comparisons (p at most 7.1×10^{-6}). The shuffled control follows the weighted arm's resampling schedule only up to its first selection, so a replay control that reproduces the schedule exactly was also run; against it the weighted arm again wins in all six linear-cell comparisons (Holm $p \le 1.4\times10^{-17}$), by 2.0 to 6.2 in median measurement residual, so the direction is unchanged once the confound is removed. Phase retrieval favours the weights too, although no prediction was made for it, since the greedy comparison there was equivocal. Against the plug-in, at $P = 16$, median paired reductions in measurement residual are 3.08 (S), 5.90 (M) and 6.24 (D), pooling image–seed pairs. This aggregation differs from the per-arm medians in Table 6, and 64 of 64 images improve from $4\times$ to $32\times$. Two variants of the weights show how sensitive the gain is to their form. Replacing the incremental potential with an absolute one leaves the arm statistically unchanged, at nearly identical realized ESS.

Table 7: **Estimator controls in image inverse problems.** Counts among 64 images where the first estimator has lower seed-averaged measurement residual, at budgets 4 and 16. Directions h_R, h_L and h_L^{n} are defined above; h_L^{n} matches the plug-in gradient norm. S, M and D denote inpainting; PR denotes phase retrieval. Bold: Holm-corrected sign-test $p < 0.05$ in either direction; ties are not wins. †Reported without a scored prediction.

	S		M		D		PR	
comparison	4	16	4	16	4	16	4	16
h_R − plug-in	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
h_R − unweighted	33	30	33	29	34	28	26	29
h_L^{n} − plug-in	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
h_L − plug-in†	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
h_L − h_R†	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
h_L^{n} − h_L†	**64**	**64**	**64**	**64**	**64**	**64**	**64**	**64**
h_L^{n} − h_R†	34	35	22	27	**19**	**20**	**1**	**0**
particles − greedy control	**64**	**64**	**64**	**64**	**63**	**63**	**42**	**55**
particles − permuted weights	**64**	**64**	**64**	**64**	**63**	**63**	**50**†	**58**†
particles − replay control	**64**	**64**	**64**	**64**	**63**	**63**	**52**†	**57**†

Table 8: **The terminal reading on the four dial cells, and what each family then gained.** Spread and fidelity τ come from three plug-in seeds, taken before any best-of-N result was read; PR is phase retrieval and the other three cells are linear. PR's spread is 12–25× that of the linear cells. The four right-hand columns give the change in median PSNR over the plug-in at the 16× budget, with 95% cluster bootstraps over the 64 images. Averaging is the projected-draw arm at $K = 16$ (`I2-16`).

			Δ median PSNR vs plug-in (dB)			
Cell	Spread	τ	Averaging	Greedy selection	Particles	Best-of-N
S	0.021	0.926	−8.15 [−8.5, −7.6]	−0.11 [−0.3, +0.1]	+2.41 [+2.2, +2.8]	+0.62 [+0.4, +0.9]
M	0.011	0.957	−13.06 [−13.8, −12.2]	−0.07 [−0.3, +0.3]	+1.38 [+1.2, +1.7]	+0.40 [+0.3, +0.9]
D	0.010	0.990	−14.04 [−14.8, −13.1]	−0.05 [−0.3, +0.3]	+1.16 [+1.0, +1.6]	+0.37 [+0.2, +0.9]
PR	0.255	0.403	−3.04 [−4.1, −1.9]	−0.42 [−1.6, +0.6]	−1.34 [−2.4, −0.3]	+7.41 [+6.1, +8.8]

Transition-corrected weights, which are formally proper, pull the arm down to or below the greedy control in the deep cells, where their realized ESS sits at the $1/P$ floor.

In the linear cells, disrupting the association between particles and weights worsens the measurement residual even when the resampling schedule is replayed. Weight assignment therefore contributes beyond the availability of alternative trajectories in these comparisons. The flow-matching results below provide a contrasting case: particles improve over the plug-in without a significant advantage over greedy selection. A particle gain can thus arise through different mechanisms, which a greedy control helps distinguish.

Terminal selection arms. The terminal reading separates phase retrieval from the linear cells. Terminal spread: linear cells 0.010–0.021 (D 0.010, M 0.011, S 0.021); PR 0.255. Fidelity: linear cells 0.93–0.99 (S 0.93, M 0.96, D 0.99); PR 0.40, 95% CI 0.24–0.56. The spread and rank measurements were taken from plug-in seeds before selection outcomes were read (Table 8). Despite low global rank agreement, phase retrieval's gain continued to grow through $N = 16$, contradicting the predicted saturation (Figure 5).

Saturation of the draw reading. On the schedule-scale FFHQ draws, the same logged draws give identical median ESS/K across cells (largest/smallest ratio 1.000×), although their median log-weight variances span 6.11 orders of magnitude. For one-step transition draws, the prediction was $\mathrm{ESS}/M \approx 1$ at $M = 16$, failing at or below 0.5. The reading is 0.0625 in both inverse-imaging cells probed (D and PR), exactly the $1/M$ floor and equal to the tilted-draw reading on the same states, over 144 logged readings in each, so the prediction failed. These measurements show that

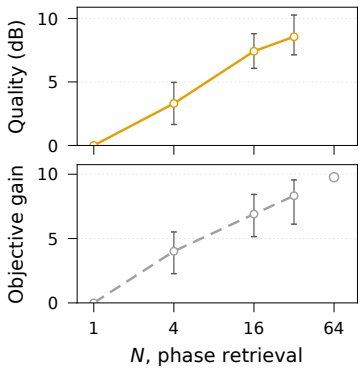

Figure 5: **Selection returns in phase retrieval.** Top: median PSNR gain over the plug-in; bottom: the selector's objective. Intervals are 95% cluster bootstraps over 64 images. The open $N = 64$ marker comes from a prespecified rule and has no interval. Diagnostic measurements and family gains are in Table 8.

draw ESS can remain saturated when the draw construction changes. The whole-run transition-draw comparisons below evaluate final PSNR on S and M, a different pair of cells.

E.3 Draw-construction and update-length comparisons

This experiment changes how clean-output draws are constructed while retaining the checkpoint, objective, unweighted averaging operator, guidance scale and final PSNR evaluation. It compares Gaussian projected draws, one-step transition draws, and differentiable unguided continuations to the endpoint. The latter are learned finite-budget continuations, not exact posterior samples. The two FFHQ inpainting conditions, S and M, share the checkpoint and image set; they are not independent model replications. All three draw constructions are evaluated at $K = 2, 4, 8$ on 64 images with three seeds per condition.

Changing the draw construction includes changing its Jacobian with respect to the current state. The experiment does not separately isolate the clean-output marginal distribution or match guidance-update length. Nor does it match computation: at $K = 8$, the recorded primary-forward counts are 100, 892, and 39,700 for projected draws, transition draws and full continuations, respectively. These counters exclude backward work and checkpoint recomputation.

E.3.1 Whole-run effects and inference

Tables 9 and 10 report mean PSNR effects in native units. For each image the three paired seed effects are averaged before image identities are resampled. Intervals are pointwise and conditional on these three seeds.

Transition draws remove most of the projected-averaging loss at every tested K. Full continuations score below transition draws at each K, with the same direction in every seed. Longer lookahead therefore does not monotonically improve final utility here. The transition's $K = 8$ baseline-relative intervals include zero in both conditions, which establishes neither superiority nor equivalence to the plug-in.

Transitions match or exceed continuations, contrary to the predicted intermediate ordering. An exact-cost prediction remains unscored because its comparator is unavailable. Aggregation was chosen after outcomes were available (Appendix F).

Table 9: Whole-run mean paired PSNR differences (dB), with pointwise 95% image-cluster bootstrap intervals. Each cell has 64 image clusters and three fixed seeds, not 192 independent observations.

Cell	K	Projected − plug-in	Transition − plug-in	Continuation − plug-in
S	2	−8.180 [−8.580, −7.788]	+0.006 [−0.098, 0.108]	−0.321 [−0.427, −0.219]

Cell	K	Projected − plug-in	Transition − plug-in	Continuation − plug-in
S	4	−8.243 [−8.654, −7.834]	−0.013 [−0.120, 0.109]	−0.320 [−0.438, −0.210]
S	8	−8.315 [−8.718, −7.925]	+0.032 [−0.066, 0.127]	−0.271 [−0.381, −0.161]
M	2	−13.165 [−13.663, −12.661]	−0.162 [−0.269, −0.067]	−1.474 [−1.645, −1.320]
M	4	−13.145 [−13.674, −12.614]	−0.092 [−0.155, −0.030]	−1.350 [−1.488, −1.217]
M	8	−13.263 [−13.739, −12.785]	−0.020 [−0.085, 0.043]	−1.285 [−1.424, −1.151]

All contrasts are higher-is-better. The 10,000-draw bootstrap resamples images with their three seed outcomes; intervals are approximate and conditional on these seeds and this checkpoint. Arms use different random streams despite shared image and seed labels, costs are not matched, and an interval crossing zero does not establish equivalence.

Table 10: Direct whole-run population contrasts: mean paired PSNR differences (dB) and pointwise 95% image-cluster bootstrap intervals. These intervals are computed from each paired contrast, not obtained by subtracting baseline-relative interval endpoints.

Cell	K	Transition − projected	Continuation − transition
S	2	+8.186 [7.805, 8.574]	−0.327 [−0.436, −0.220]
S	4	+8.230 [7.812, 8.658]	−0.307 [−0.478, −0.159]
S	8	+8.346 [7.958, 8.744]	−0.303 [−0.425, −0.184]
M	2	+13.003 [12.498, 13.499]	−1.312 [−1.490, −1.147]
M	4	+13.053 [12.525, 13.577]	−1.258 [−1.403, −1.116]
M	8	+13.243 [12.750, 13.730]	−1.266 [−1.409, −1.121]

Each cell has 64 image clusters with three fixed seeds; the pairing, conditional inference scope and unmatched-cost qualifications of Table 9 also apply.

E.3.2 Single-update comparisons

To distinguish isolated update effects from changes accumulated along a trajectory, we replace one update at five saved plug-in states and then resume plug-in guidance with shared subsequent noise. Each state uses the same 16 images and three seeds. The comparisons increase projected draw count, match update length, calibrate the projected guide, or substitute continuation draws.

Increasing draw count stabilizes projected updates at most tested states, but the predicted alignment with the plug-in and large continuation benefit fail in both conditions. Continuation draws yield small positive final-PSNR effects at several states (Table 11). The native effects are too small to evaluate the specified half-gap recovery and retained-advantage tests; full recovery remains unscored because its criterion was ambiguous. These interventions begin on plug-in trajectories and resume plug-in guidance. Their small effects therefore leave the much larger whole-run deficit unresolved, including how repeated updates interact with states visited by projected guidance.

Table 11: Single-update population contrasts at K=8: mean paired final-PSNR differences (dB) and pointwise 95% image-cluster bootstrap intervals. Each comparison uses 16 image clusters and three fixed seeds, not 48 independent observations.

Cell	State	Continuation − projected	Both lengths matched
S	90	+0.0069 [0.0032, 0.0108]	+0.0017 [−0.0023, 0.0059]
S	70	+0.0660 [0.0577, 0.0745]	+0.0449 [0.0360, 0.0529]
S	50	+0.0274 [0.0243, 0.0306]	+0.0197 [0.0177, 0.0216]
S	30	+0.0063 [0.0058, 0.0068]	+0.0048 [0.0043, 0.0053]
S	10	+0.0009 [0.0008, 0.0010]	+0.0003 [0.0003, 0.0004]
M	90	+0.0152 [0.0053, 0.0263]	Unavailable (46/48)
M	70	+0.0082 [−0.0012, 0.0202]	+0.0008 [−0.0059, 0.0103]
M	50	+0.0488 [0.0426, 0.0554]	+0.0305 [0.0260, 0.0352]
M	30	+0.0463 [0.0400, 0.0522]	+0.0362 [0.0307, 0.0415]
M	10	+0.0236 [0.0199, 0.0273]	+0.0064 [0.0043, 0.0090]

Both branches start from the same saved plug-in state and use common finish noise, then resume plug-in guidance. Matched branches both have the plug-in update length. Unavailable entries report valid/expected pairs; no available-case substitution is used. These small continuous effects supplement, rather than change, the median-based prediction criteria. Intervals are conditional and approximate.

E.3.3 WHOLE-TRAJECTORY UPDATE-LENGTH CONTROL

A prespecified whole-trajectory comparison controls the *length* of the applied guidance update (Table 12). At each image and step, the controlled projected and transition arms rescale their own update to the length that the same run's plug-in arm applied, keeping their own direction and evolving state; the schedule is open loop, so both controlled arms receive the plug-in's length although their states differ. Two crossover arms run projected guidance for the first half of the applied updates and the plug-in for the second, or the reverse. Streams are keyed by image, step and draw, so every arm shares its initial and outer noise.

The transition–projected gap remains +7.80 and +13.37 dB under this schedule (Table 12). Using projected guidance only during the first half of sampling reproduces most of its full-run deficit.

The controlled transition arm matched its prescribed length on 100.00% of scheduled steps in S and 99.88% in M, and the controlled projected arm on 99.45% and 97.49%. An unmatched step is one whose own update has zero length while the reference is positive, so it has no direction to rescale; a zero reference is matched by applying nothing. Unmatched steps occur early and touch 39 of 192 and 85 of 192 projected trajectories. The control tests one open-loop length schedule, not every step-size adjustment, and it fixes the draw count rather than compute.

Table 12: Whole-trajectory update-length control: mean paired final-PSNR differences in dB with image-cluster intervals over 64 images carrying three seeds each. Intervals are pointwise and conditional on these seeds.

Contrast	Cell S	Cell M
transition - projected, controlled	+7.80 [+7.48, +8.13]	+13.37 [+12.91, +13.84]
transition - projected, native	+8.04 [+7.62, +8.48]	+13.01 [+12.48, +13.55]
controlled projected - plug-in	-7.78 [-8.11, -7.46]	-13.37 [-13.84, -12.91]
controlled transition - plug-in	+0.02 [+0.01, +0.03]	+0.00 [-0.01, +0.01]
native projected - plug-in	-8.03 [-8.47, -7.62]	-13.04 [-13.57, -12.52]
controlled - native projected	+0.25 [-0.24, +0.77]	-0.32 [-0.74, +0.09]
controlled - native transition	+0.01 [+0.00, +0.02]	+0.04 [+0.01, +0.05]
projected first half only - plug-in	-7.68 [-8.12, -7.26]	-11.47 [-12.01, -10.94]
projected second half only - plug-in	-0.34 [-0.35, -0.33]	-2.52 [-2.66, -2.37]

E.3.4 SEPARATELY CALIBRATED AVERAGING REFERENCES

The preceding transition-draw comparisons use unweighted averaging at a shared guidance scale. This additional comparison gives two reference rules their own weighting and their own calibration: the released DPS `ps+` rule (Chung et al., 2023) and the weighted LGD-MC estimator (Song et al., 2023), both at $K = 16$ on two FFHQ inpainting conditions. S retains 10% of pixels with measurement noise $\sigma_y = 0.2$; M retains 60% with $\sigma_y = 0.05$. Each uses 64 evaluation images, seeds $0, 1, 2$ and 100 sampling steps.

The released `ps+` rule perturbs the predicted clean image by independent uniform noise in $[0, 0.05)$ and averages residual-norm gradients. We use 16 draws rather than its default five, supply the inpainting mask and preserve the released single-image gradient when batching. LGD-MC differentiates the log-mean-exponential likelihood, retaining likelihood weights rather than an unweighted gradient mean. This is a paper-based implementation, with gradient and limiting-case tests; `ps+` additionally passes an update-equality test against the pinned released code. These reference configurations are specific to this two-condition comparison.

Calibration uses eight separate images, trajectory seed 100 and median PSNR. Plug-in and `ps+` each search a multiplier $\rho \in \{1/4, 1/2, 1, 2, 4\}$ relative to the released inpainting scale. LGD-MC searches likelihood scale $s \in \{0.25, \sigma_y\}$ and $\rho \in \{1/256, 1/64, 1/16, 1/4, 1\}$; a model-free

update-magnitude check sets this grid. Plug-in and ps+ select $\rho = 4$ in both conditions, at the upper boundary with calibration PSNR still increasing. LGD-MC selects $(s, \rho) = (\sigma_y, 1/4)$ in S and $(0.25, 1/16)$ in M. Its median normalized weight ESS is $1/16$ throughout the calibration grid. Comparisons below concern these selected configurations; the plug-in and ps+ optima beyond the grid remain unresolved.

Table 13: Reference averaging at $K = 16$: paired mean PSNR differences (dB), with descriptive pointwise 95% image-bootstrap intervals. P is the released plug-in and P_c the separately calibrated plug-in. Each image carries all three seeds.

Contrast	Condition S	Condition M
ps+$-P$	$+2.97$ [$+2.80$, $+3.14$]	$+5.01$ [$+4.66$, $+5.34$]
LGD-MC$-P$	$+0.50$ [-0.13, $+1.10$]	-5.60 [-6.36, -4.88]
Calibration alone $P_c - P$	$+3.06$ [$+2.86$, $+3.24$]	$+5.85$ [$+5.46$, $+6.20$]
ps+$-P_c$	-0.09 [-0.24, $+0.07$]	-0.84 [-1.19, -0.48]
LGD-MC$-P_c$	-2.56 [-3.15, -2.00]	-11.46 [-12.24, -10.68]

The ps+ reference improves on the released plug-in in both conditions (Table 13). The LGD-MC interval includes zero in S and lies below zero in M. Recalibrating the plug-in itself matches or exceeds the ps+ gain. Against the calibrated baseline, the ps+ interval overlaps zero in S and lies below zero in M, and both LGD-MC intervals lie below zero. Neither reference establishes a benefit over the separately calibrated plug-in on these two conditions.

Effects are means over images of seed-averaged paired differences. Intervals use 10,000 image-bootstrap resamples, condition on the three observed seeds and are unadjusted for multiple comparisons; overlap with zero does not establish equivalence. The reference arms use different draw streams from the schedule-scale comparison. Validation checks comparator hashes, frozen code, calibration and saved configuration identities; pairing follows the frozen runner because checkpoint/data hashes and explicit instance IDs were not saved. At each cell and seed, the calibrated plug-in uses 6,400 objective evaluations and each reference uses 102,400; all use 6,400 backbone forward and backward rows, so the objective-evaluation multiplier is not a wall-time multiplier.

Taken together, these comparisons identify draw construction and guidance calibration as consequential choices alongside draw count. Replacing draws changes whole-run utility substantially, and the deficit survives the tested plug-in-derived length schedule. Separately calibrating the baseline also changes the apparent benefit of the reference estimators. A useful averaging comparison therefore evaluates the draw rule and calibration against a correspondingly calibrated plug-in, then measures their effects over complete trajectories.

E.4 Flow-matching extension

We evaluate SiT-XL/2 with its released deterministic probability-flow ODE and stochastic SDE on super-resolution, inpainting and classifier reward (log-probability). Each sampler uses a 64-point schedule, giving 63 transitions, on 64 instances and three seeds per task. Averaging uses $K = 16$, particles $P = 8$ and terminal selection $N = 16$. Zero-guidance runs reproduce released unguided sampling. Population measurements and prediction thresholds precede evaluation of the upgrades.

Population measurements. In Table 14, the draw column tests the draw-floor rule on a flow model, the separation column measures the population mechanism of Section 2.2, and the trajectory-ESS column shows what that reading fails to separate. The reward cell's terminal-spread predictions are reported to show the reading itself, not scored, because of a threshold-unit defect: their gain thresholds are in decibels, while that cell's gain is a log-probability, so a verdict would change under a rescaling of the log-probability that changes nothing physical.

Output quality. Table 15 compares upgrades with the plug-in and particles with a greedy control. Plug-in guidance improves on unguided sampling in every setting. Under the stochastic sampler particles beat the plug-in in all three cells; under the deterministic sampler, with the same checkpoint, cells and budgets, none is significant. Each particle arm is also compared with its own greedy control, with no significant advantage from weighting on this checkpoint. The one significant difference

Table 14: Flow-matching population measurements under the schedule-scale draw construction. Re-measured at the LGD draw scale, the draw column changes only on the reward cells, from 0.945 and 0.949 to 0.917. The draw reading ESS/K, the trajectory reading ESS/P on distinct-particle steps, the minimum relative particle separation, and the terminal relative-IQR spread. Terminal spread here is measured across particle-selected outputs; the SiT entries in Table 1 use plug-in candidate spreads.

Cell	Sampler	Draw ESS/K	Traj. ESS/P	Rel. separation	term. spread
super-resolution	ODE	0.0625	0.663	5.2×10^{-6}	0.017
super-resolution	SDE	0.0625	0.125	1.8×10^{-1}	0.021
inpainting	ODE	0.0625	0.125	1.1×10^{-6}	0.093
inpainting	SDE	0.0625	0.125	1.5×10^{-1}	0.036
reward	ODE	0.9452	0.866	7.1×10^{-2}	0.242
reward	SDE	0.9486	0.815	5.6×10^{-1}	0.272

Table 15: Flow-matching arm matrix under the schedule-scale draw construction: median paired differences over 64 images with seeds aggregated, in dB for the two inverse cells and in class log-probability for the reward cell, with the number of images on which the first arm wins in parentheses. Guidance is the plug-in minus unguided sampling; the average and particle columns are each arm minus the plug-in; the last column is particles minus their greedy replicate-after-selection control. Bold marks Holm-corrected $p < 0.05$.

Cell	Sampler	Guidance	Average	Particles	Particles − greedy
super-resolution	ODE	**10.49 (64)**	**−3.12 (0)**	0.00 (32)	0.02 (34)
super-resolution	SDE	**8.18 (64)**	**−2.29 (0)**	**1.08 (63)**	0.02 (33)
inpainting	ODE	**9.35 (64)**	**−0.40 (22)**	0.29 (37)	0.23 (34)
inpainting	SDE	**12.52 (64)**	**−1.13 (15)**	**0.60 (48)**	0.00 (32)
reward	ODE	**7.24 (64)**	**−1.97 (2)**	0.00 (33)	0.00 (34)
reward	SDE	**5.22 (64)**	**−3.21 (0)**	**1.24 (56)**	**−0.10 (19)**

is a loss in the reward cell under the stochastic sampler, which is the mechanism split Section 4.2 reports. Under the LGD-scale configuration, averaging loses to the plug-in in the reward cell under both samplers and in super-resolution under the deterministic one, and stays within the rerun band in the other three cells. Table 15 reports the schedule-scale configuration used to score this testbed's prespecified predictions.

Averaging predictions. On this checkpoint both inverse cells read exactly at the $1/K$ floor, as every diffusion cell did, on a different model family, tokenizer and sampler, while the reward cell reads essentially at the ceiling with the same code (Table 14). Averaging lost under both samplers, in both reward configurations. In Table 1's binary assessment, these two above-threshold settings receive gain predictions, neither of which is correct. High normalized ESS therefore did not identify a useful upgrade. Appendix D.1 specifies the prediction rules.

Particle separation and weight concentration. The same-checkpoint separation contrast is explicit: on the two inverse cells the particles are separated by more than 10^{-4} of their own scale in at most 3.9% of deterministic-sampler readings and in 100% of stochastic ones, a median separation of 5.2×10^{-6} and 1.1×10^{-6} against 0.182 and 0.155. The mechanism runs through resampling, so a cell whose weights remain diffuse keeps distinct particles under either sampler: the reward cell does not collapse under either sampler (95.2% of deterministic readings clear the same threshold), as the mechanism implies: its weights read 0.866, so resampling seldom fires and there are few duplicates for a deterministic map to preserve. Meanwhile the trajectory ESS does not distinguish the two samplers here: it reads 0.1252 under the deterministic sampler against 0.1250 under the stochastic one on the inpainting cell, and higher under the deterministic sampler on super-resolution (0.663 against 0.125), because clones carry identical weights. A checkpoint shipping both samplers therefore sustains and collapses the trajectory population with everything else held fixed, which is the intervention Section 4.2 runs. Figure 6 shows the corresponding per-image particle gains.

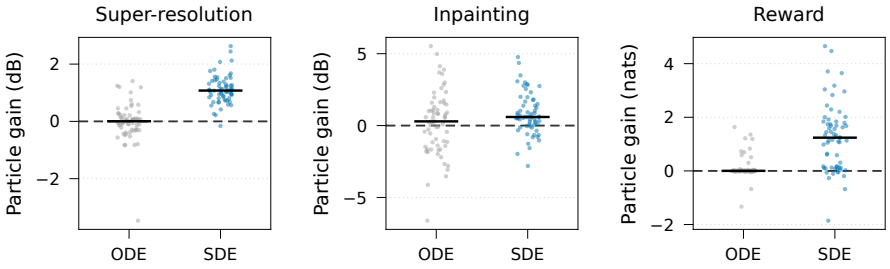

Figure 6: **Particle gains under deterministic and stochastic sampling.** Dots show paired gains of $P = 8$ particles over the plug-in on 64 images per task, averaged over three seeds; bars mark medians. Gray: deterministic ODE; blue: stochastic SDE. Table 15 reports medians and significance.

Table 16: **Diagnostic measurements on InverseBench.** Draw max is the maximum per-step median normalized ESS over the data-scale window. Trajectory median uses distinct-particle steps in the final quarter (a post-hoc statistic). Log-w range is the mid-trajectory log-weight range in nats; inst. counts instances.

Problem	Inst.	K	P	Draw max	Log-w range	Traj. median	$1/P$	Diverged	Resamp. gap
ffhq256-inpaint	32	16	8	0.0625	2.9×10^{5}	0.1250	0.1250	0	1
inv-scatter	32	16	8	0.0625	5.2×10^{6}	0.1250	0.1250	0	1
navier-stokes	8	16	4	0.0625	6.3×10^{6}	0.2500	0.2500	0	1

Draws use the schedule-scale construction with clipping. Draw ESS reaches the $1/K$ floor on all three problems. The draw and trajectory ESS columns use different temporal aggregations. Window: scheduler noise $\sigma_t \leq 1$. The noiseless Navier–Stokes likelihood-scale sensitivity grid is $\{0.005, 0.01, 0.05\}$ RMS(y). Trajectory sensitivity replays the primary run's resampling events and is therefore approximate. Verdicts that change across the sensitivity grid: none.

E.5 InverseBench transfer

Sampler and task configurations. The benchmark checkout is `InverseBench@1db5932`, with three problems: FFHQ box inpainting (VP-preconditioned ADM), inverse scattering and Navier–Stokes inversion (both EDM-preconditioned Dhariwal U-Nets). The checkpoints and the inverse-scattering and Navier–Stokes test sets are the benchmark's released ones; the FFHQ problem uses the first 32 images of a public FFHQ-256 copy with the benchmark's box-mask operator, since the release omits its FFHQ images. The closed-form problems use the released first-order DPS rule (scale 10, stochastic sampler, 1000 steps). The Navier–Stokes solver has no autograd path, so its released zero-order DPS rule estimates each gradient from 2048 Gaussian queries in batches of 512 (scale 0.1, smoothing radius 0.01, central differences, 1000 steps); this is the released configuration, not a tuned intervention.

Sampling and likelihood scales. Each problem uses its first test instances, as many as Table 16 lists; FFHQ uses the pinned real-image set because its images are absent from the benchmark release. Draws are centered on the denoised estimate with scheduler-scale perturbations, clipped to the declared data range and logged every 25 steps. Likelihood noise is 0.05 for FFHQ and 10^{-4} for scattering; the noiseless PDE problem uses $0.01\,\mathrm{RMS}(y)$, with its sensitivity grid in the table note. Non-finite states or losses end an instance and count as divergence events.

Trajectory weights and analysis scope. Particles use coupled likelihood increments, systematic resampling and weight reset at $\mathrm{ESS}/P < 0.5$. The trajectory statistic excludes steps immediately after resampling, when clones have identical weights. This statistic is post hoc; the weighting amendment and exclusion are recorded in Appendix F. Including clone steps makes the predicted above-floor reading appear to hold; excluding them reveals its failure and fails to separate the problem classes. It does not predict InverseBench's published method ordering. The FFHQ implementation injects independent transition noise before evaluating weights, and a separate check rules out the same clone artifact there.

Failure of the trajectory prediction. Under the InverseBench kernel the longest stretch without a resample is 1 step on every ffhq256-inpaint instance and the post-reset single-increment ESS reads 0.87 of P, against resampling on 53% of steps and 0.37 of P under ours. The flow arms gained despite the floored reading, so the proposed prediction fails rather than counted as a correct no-gain prediction.

E.6 TEXT-TO-IMAGE CHECKPOINT TRANSFER

This study tests the draw rule and a separate reward-loss hypothesis on three released text-to-image checkpoints: FLUX.2-klein (9B), Qwen-Image (20B) and SD3.5-Large (8B). Each runs at 1024^2 with text conditioning on inpainting and two class-reward cells, whose prompts either depict or do not depict the scored class. Each condition has eight instances and three evaluation seeds. Within each setting, all configuration choices are shared by the plug-in and all population budgets such that the sweep measures scaling effects under a fixed sampling configuration. Five arms run: unguided sampling, the plug-in, averaging with $K = 16$, particles with $P = 4$ and best-of-16 selection. Schedule-scale and LGD-scale averaging results are reported for their respective configurations.

The twelve further checkpoints use inpainting with keep fraction 0.6, measurement noise 0.05 and checkpoint-specific calibrated guidance. Their per-setting gains appear in Figure 3. Table 17 identifies the fifteen released checkpoints.

Table 17: Released text-to-image checkpoints, by repository identifier. The first three are the production stacks of the three-checkpoint study; the remaining twelve are the transfer checkpoints of Figure 3.

Checkpoint	Identifier
FLUX.2-klein	`black-forest-labs/FLUX.2-klein-base-9B`
Qwen-Image	`Qwen/Qwen-Image`
SD3.5-Large	`stabilityai/stable-diffusion-3.5-large`
AuraFlow	`fal/AuraFlow-v0.3`
Chroma	`lodestones/Chroma`
CogView4	`THUDM/CogView4-6B`
FLUX.1-dev	`black-forest-labs/FLUX.1-dev`
HunyuanDiT	`Tencent-Hunyuan/HunyuanDiT-v1.2-Diffusers`
Kolors	`Kwai-Kolors/Kolors-diffusers`
Lumina-Next	`Alpha-VLLM/Lumina-Next-SFT-diffusers`
Lumina2	`Alpha-VLLM/Lumina-Image-2.0`
PixArt-Σ	`PixArt-alpha/PixArt-Sigma-XL-2-1024-MS`
Playground	`playgroundai/playground-v2.5-1024px-aesthetic`
SANA	`Efficient-Large-Model/Sana_1600M_1024px_diffusers`
SDXL	`stabilityai/stable-diffusion-xl-base-1.0`

E.6.1 OUTCOMES

Table 18 gives every arm on all nine cells. On their inpainting cells, all 32 logged states on each checkpoint read $\mathrm{ESS}/K = 1/16$, despite different averaging outcomes. Under the schedule-scale construction, averaging beats the plug-in at every seed on FLUX.2 inpainting, which the draw rule records as a miss, loses at every seed on Qwen-Image, and wins one seed of three on SD3.5-Large. A prespecified seed extension found no stable direction on SD3.5-Large. The opposite directions on FLUX.2 and Qwen-Image remain unexplained.

With the schedule-scale draw construction, averaging underperformed the plug-in in all six reward cells. The separately registered reward-loss hypothesis held in the three scorable cells; the other three were ceiling-limited. These verdicts do not test the draw-floor rule. The median loss on SD3.5-Large's non-depicting cell is $51\times$ the whole headroom left above the plug-in. The reward score is a class log-probability with a hard maximum at zero, and on that cell guidance alone consumes 7.63 of the 7.71 nats between unguided sampling and the maximum, 99.0% of the range, leaving only that remainder for further improvement. At the three evaluation seeds the SD3.5 non-depicting cell clears the eligibility criterion for remaining reward headroom and is scored.

Table 18: **Production-checkpoint outcomes with schedule-scale draws.** Median PSNR (dB) for inpainting and class log-probability (maximum zero) for reward; higher is better and the best arm is bold. Δ_s is averaging ($K = 16$) minus the plug-in at seed s.

Cell	Unguided	Plug-in	Average	Particles	Selection	Δ_0	Δ_1	Δ_2	Outcome
FLUX.2-klein									
inpainting (dB)	10.35	20.73	**21.40**	20.69	21.11	0.73	0.47	0.71	incorrect
reward, depicted	—	−0.010	—	—	—	−0.23	−0.15	−0.19	not scored
reward, not depicted	—	−0.027	—	—	—	−4.78	−4.30	−4.28	not scored
Qwen-Image									
inpainting (dB)	10.31	23.75	12.67	23.58	**25.12**	−12.13	−11.08	−10.30	correct
reward, depicted	−0.886	−0.244	−0.470	−0.213	**−0.118**	−0.12	−0.24	−0.39	correct
reward, not depicted	−7.772	−3.040	−7.691	−0.310	**−0.080**	−5.13	−3.40	−4.74	correct
SD3.5-Large									
inpainting (dB)	10.40	23.61	23.48	23.81	**24.51**	−0.49	0.67	−0.14	correct
reward, depicted	—	−0.014	—	—	—	−0.18	−0.13	−0.12	not scored
reward, not depicted	−7.707	−0.076	−3.958	−0.016	$\mathbf{-3}\times10^{-4}$	−3.27	−3.85	−5.90	correct

Outcomes score the inpainting draw-floor rule and the separate reward-loss hypothesis, each using its pre-specified seed-majority criterion. Ceiling-limited cells are unscored; only the plug-in median and averaging's direction are shown. Averaging costs $8.5\times$, particles $4\times$ and selection $16\times$ the plug-in's model calls on every cell.

Under the LGD-scale configuration the inpainting picture differs: averaging beats the plug-in on FLUX.2 at every budget, and on Qwen-Image and SD3.5-Large the effect stays inside the rerun band for $K \geq 4$; Qwen-Image falls below it at $K = 2$. Table 1 evaluates the numerical draw-floor criterion against the LGD-scale outcomes at $K = 16$. For the non-depicting reward settings, this uses their matched LGD-scale calibration measurements, not the separate reward-loss hypothesis (Appendix E.7.1). Under the LGD-scale configuration, averaging loses on the FLUX.2-klein and Qwen-Image reward cells. SD3.5-Large stays within the rerun band for $K \geq 4$, but falls below it at $K = 2$.

E.6.2 Checks

- *Floor tolerance.* Of the 38 logged measurements, 30 equal $1/K$ to recorded precision; the remaining 8 range from 0.943 to 0.991. None lies above the hard floor and at or below the threshold.
- *Port.* The SD3.5 implementation agrees with its released pipeline above 30 dB, while a deliberately mis-scaled timestep falls below 25 dB; gradients remain finite.
- *Guidance and calibration.* Guidance uses the dimensionless update $\eta\|x\|g/\|g\|$, normalized after any draw averaging. The step size is the grid point that minimizes the plug-in's mean terminal objective on at most four instances at seed 99, disjoint from the evaluation seeds $0, 1, 2$, and consults neither arm comparisons nor the quality metric. The grid is $\{0.00125, 0.0025, 0.005, 0.01, 0.02, 0.04\}$. Every arm reads the same calibration for its cell, and comparisons using different calibrations are excluded from scoring.
- *Precision.* Reductions are computed in float32: bf16 accumulation can place the gradient-coherence ratio $\|\sum_i g_i\| / \sum_i \|g_i\|$ above one, violating its mathematical bound.
- *Runtime.* Table 19 gives the process wall time of all 135 arm runs, each upgrade paired with the plug-in for the same cell and seed. The timer covers everything from model loading to evaluation except queue wait. The 15 seed-zero inpainting rows include runs repeated to save their output images, and that time cannot be separated out.

Table 19: Production-stack process wall times. Plug-in seconds are the median over three seeds of one process run on eight instances; upgrade columns are median paired process-time ratios to the same cell and seed's plug-in, with the three-seed range in brackets. These include startup/reference overhead, not pure sampling throughput.

Stack	Cell	Plug-in (s)	Average $K = 16$	Particles $P = 4$	Selection $N = 16$
FLUX.2-klein	inpainting	316.4	8.48 [5.47, 9.24]	3.83 [2.66, 4.23]	12.05 [8.54, 13.06]
FLUX.2-klein	reward, depicted	294.5	8.58 [4.73, 8.90]	3.95 [2.16, 4.93]	12.89 [6.57, 14.19]
FLUX.2-klein	reward, not depicted	604.7	4.28 [4.25, 9.35]	2.22 [1.77, 3.43]	6.22 [6.18, 12.79]
Qwen-Image	inpainting	677.9	7.15 [6.05, 7.26]	3.22 [3.04, 3.34]	9.41 [8.14, 10.24]
Qwen-Image	reward, depicted	738.7	6.39 [6.05, 8.40]	2.61 [2.28, 3.25]	8.39 [8.31, 12.83]
Qwen-Image	reward, not depicted	701.0	6.28 [5.19, 9.45]	2.64 [2.41, 4.85]	8.91 [7.54, 13.36]
SD3.5-Large	inpainting	290.6	6.96 [6.73, 7.30]	2.68 [2.66, 3.22]	9.27 [9.20, 9.84]
SD3.5-Large	reward, depicted	279.8	6.98 [6.77, 9.18]	3.07 [3.01, 3.74]	9.75 [9.54, 13.29]
SD3.5-Large	reward, not depicted	213.4	9.32 [7.10, 9.41]	3.71 [3.02, 4.19]	13.08 [9.51, 13.52]

All 135 records report NVIDIA H100 80GB hardware, but matching labels do not control machine load or cache state, and optional image serialization and cache/I/O overhead are not separately timed.

E.6.3 Trajectory-rule transfer

A fixed minimum-separation threshold based on single-precision SiT assigns gain on all seven held-out text-to-image checkpoints. Their bfloat16 rounding floor was 39 times above that threshold, so near-identical particles still read as separated. Only HunyuanDiT and Kolors injected noise that could separate resampled copies. Particles gained beyond rerun spread on HunyuanDiT (1.57 dB against 1.05 dB spread), but not on the other six checkpoints.

The clone-calibrated variant sets the threshold from initially identical particles propagated on each checkpoint (Appendix D.1). Four of the five evaluated checkpoints satisfy the no-gain criterion for every tested threshold multiplier from 1.5 to 10. All five settings show no gain. On SANA, clone separation is exactly zero, so the positive-reference criterion is undefined and Table 1 excludes that setting. On the four settings with positive clone references, both the calibrated criterion and always predicting no gain are correct.

This transfer test shows why separation must be interpreted relative to numerical precision and the sampler's own clone dynamics. Calibration distinguishes rounding-scale differences from macroscopic separation, but successful no-gain calls on an all-no-gain cohort do not establish discrimination. Testing that distinction requires settings with both useful and unhelpful upgrades.

E.7 Population-summary predictions and small-scale gain evaluations

We compare summary-based predictions with family-level no-gain rates, then assess direct gain estimates from small-scale evaluations.

E.7.1 Summary-based predictions

Evaluation cohort and eligibility. Table 1 evaluates predictions computed from measured population summaries using the numerical criteria in Appendix D.1. An eligible setting has a finite standardized outcome, an available summary measurement, and a criterion applicable to that measurement. These requirements make the reported precision an assessment of the summary-based rule: a separate reward-loss hypothesis or a group assignment cannot substitute for a numerical reading. Eligibility depends on measurement support and the criterion's domain, not on whether the observed outcome is a gain.

The resulting cohorts differ across families. Averaging includes four FFHQ, six SiT, twelve checkpoint-transfer and six production settings. The three topology budgets lack matched draw-floor readings and are excluded. Averaging outcomes use LGD-scale draws at $K = 16$. Predictions use schedule-scale measurements except for the three production reward settings, which use their matched LGD-scale calibration readings (Appendix D.1). The latter all predict gain under the ESS threshold and all observe no gain; their predictions are not supplied by the reward-loss hypothesis.

Production inpainting measurements were taken along unguided probe trajectories. These testbed-specific protocols assess the available measurements rather than a single uniformly calibrated instrument. Particles include 15 settings: four SiT inverse settings, seven fixed-threshold transfer settings and four clone-calibrated settings with positive reference separation. The four FFHQ, five topology and six production settings lack measurements supporting the tested separation criteria. SANA has a zero clone reference, outside the positive-reference criterion, and is also excluded. Selection includes 23 PSNR settings: four FFHQ, four SiT inverse, twelve checkpoint-transfer and three production inpainting settings. Production reward settings have no applicable common PSNR cutoff; topology selection lacks a standardized outcome. These exclusions concern prediction scoring only; the associated sampling results remain reported in their testbed evaluations.

Scoring and interpretation. Every eligible setting receives either a gain or a no-gain prediction. An observed gain exceeds 0.20 plug-in IQR; within-band improvements and losses count as no gain. Precision is the fraction of correct predictions among settings assigned a label. The numerical rules and this assessment are retrospective, with outcomes already available. Selection includes checkpoint settings whose summary measurements and outcomes share candidate samples. The comparison evaluates the stated thresholds under the recorded measurement protocols; it does not identify why a prediction fails or establish calibration on new settings.

No-gain class and base rates. Table 20 compares no-gain precision with the no-gain rate within each eligible cohort. Averaging's no-gain precision is slightly below its baseline, and none of its gain predictions are correct. Particle separation is correct on all its no-gain predictions but identifies only part of the no-gain cohort; its gain predictions are less precise (Table 1). Selection's gain predictions have precision 6/6 in this small sample but identify only half of the observed gains; low-spread no-gain predictions have limited precision. Precision therefore needs to be read together with coverage and class prevalence. For averaging, most settings additionally test transfer from schedule-scale measurements to LGD-scale outcomes.

Table 20: **No-gain class precision and family base rates.** Settings with applicable summary measurements and standardized outcomes, including in-sample terminal settings. Ground truth is the overall no-gain fraction within each eligible cohort.

Diagnostic summary	Predicted no gain	Correct predictions	Ground truth
averaging: draw-weight concentration (ESS/K)	23/28	22/23	27/28
particles: particle separation	6/15	6/6	12/15
selection: terminal spread	17/23	11/17	11/23

Transfer across inverse operators. Table 21 shows what a terminal-population measurement can add to a coarse problem description. Motion blur and saturation share the same observation count and noise scale, yet their measured spreads accompany markedly different selection gains. Super-resolution has low spread and an intermediate gain. Spread was measured from plug-in runs before selection outcomes were evaluated, using PSNR against model-generated reference images. Those plug-in runs also supply the gain baseline, so the statistics are not independent. This small comparison illustrates how terminal spread varies with selection gain within one checkpoint.

E.7.2 Small-scale gain evaluations

Estimating gain from small runs retains candidate-level outcome information that the summaries discard. In the terminal evaluation (Appendix D.2.1), a lower bound on gain finds more qualifying selection upgrades than a spread-based rule using the same pilot samples. This is evidence for a more informative use of those samples within the evaluated pools; performance on new pools and larger budgets remains unestablished. The whole-run particle evaluation below tests a different use of small-scale sampling: deciding whether to run the upgrade on further instances.

Table 21: **Terminal spread and selection gains on three held-out operators.** SR4: $4\times$ super-resolution; MB: motion blur; HDR: saturation. The observation-count estimate is $d_{\mathrm{obs}}/\sigma_y^2$. Spread is the median across images of PSNR IQR divided by absolute median over plug-in seeds. Gain is the best-of-16 median PSNR improvement over the plug-in.

Operator	Obs.-count estimate	Spread	Gain (dB)
SR4	4.9×10^6	0.020	0.78
MB	7.9×10^7	0.018	0.47
HDR	7.9×10^7	0.199	6.00

Primary particle pilot and sensitivity. The particle pilot evaluates whole-run outcomes, covering 30 cell–budget units across seven inverse-imaging and flow-matching cells, with 16 pilot instances per unit and the remainder held out. It uses an asymptotic confidence sequence and has no finite-sample guarantee. It predicted gain on 10, abstained on 20, predicted no gain on 0 and achieved a decision rate of 0.33. All gain predictions were confirmed; always upgrading also made 0 errors on those same decided units. The primary advantage over the better fixed policy is +0.038 dB per instance [-0.096, +0.200], an interval that includes zero even at zero compute price; superiority after pilot cost is unestablished. The setting-specific advantage is +0.511 dB in inverse imaging versus -0.316 in flow matching. It depends on one avoided inverse-imaging loss: leaving that cell out gives -0.207 [-0.291, -0.118]. The prespecified larger-pilot sensitivity, at 24 pilot instances, gives +0.153 dB [+0.028, +0.305] and meets the criterion at decision rate 0.47.

The primary result illustrates the importance of the fixed-policy comparison. The larger-pilot sensitivity suggests that more pilot data can improve decisions in this cohort, but also increases the cost paid before any upgrade is chosen. Both the fixed-policy comparison and this acquisition cost matter when evaluating a useful compute decision.

Separation-based filtering before evaluation. A secondary comparison used particle separation to decide whether to run the pilot and skipped the pilot on two deterministic-sampler cells. It saved 29% of pilot cost but wrongly skipped the pilot in 4 of 10 units. On the flow inpainting cell, 2 particles give a held-out interval of [+1.24, +2.27] dB, with positive effects on all three seeds; four and more show no gain. Collapse late in a run therefore does not rule out gains made earlier.

E.8 Practical implications

The experiments motivate several checks before enlarging a sampling budget. Across families, the central issue is whether the information being averaged, reweighted or ranked supports the desired output quality. The controls show why the sampling construction deserves evaluation alongside its budget.

For averaging, compare draw constructions at a modest fixed budget and calibrate both the upgrade and the plug-in on separate calibration data. The draw-replacement experiments show substantial sensitivity to how draws are generated, while the reference comparison shows that retuning the plug-in can match gains measured against its released configuration (Appendix E.3). Report whether calibration reaches a grid boundary. Evaluate final outputs after the full sampling trajectory: stabilizing a local gradient or improving one isolated update provides incomplete evidence about repeated use of that update.

For particles, inspect whether resampled copies separate under the actual transition and numerical precision. Compare weighted propagation with greedy selection at the same particle budget, and report their measured costs. The inverse-imaging and SiT controls show why both checks matter: maintaining alternative trajectories and assigning useful weights can contribute differently. Assess the utility of the particle actually returned, since a diverse or well-weighted population alone does not specify that output.

For terminal selection, evaluate output quality among the highest-ranked candidates as the budget grows, using a criterion distinct from the selector where feasible. The topology oracle comparison directly separates available candidate quality from the selector's ability to recover it; phase retrieval shows that low global rank agreement can coexist with increasing selection gains. These results

motivate checking the selected tail across budgets rather than extrapolating a global correlation. Reference-based checks require an appropriate evaluation utility, which may be unavailable during deployment.

A possible selective use of a population summary would be to flag likely no-gain settings and abstain on the rest. Table 1's classwise results motivate investigating this use, but do not establish the utility of such a policy. Evaluating it would require measuring both prediction precision and coverage, together with gains missed by skipping upgrades and the cost of acquiring the summary.

Finally, evaluate a compute recommendation against simple alternatives. Report the fraction of settings receiving a prediction, its accuracy, and the no-gain rate in the same cohort. For small-scale gain evaluations, include the samples already spent and compare with always running or always skipping the upgrade. The terminal and particle evaluations illustrate why correct predictions, coverage and net utility must be assessed together. This evaluation sequence turns the population-specific analysis into concrete checks while keeping the final decision tied to measured output gains and cost.

F Analysis specifications and deviations

This appendix records each analysis's specification and its deviations, distinguishing predictions made before evaluation from analyses developed after outcomes were available. The retrospective binary rules and common scoring convention for Table 1 are described in Appendix E.7.1.

Topology metric and primary comparison. Volume-projected compliance was replaced by the released thresholded metric after identifying projection-induced load-path failures and before the deciding arm's test evaluation. Before evaluation, the primary comparison was defined as the equal-one-call test (Appendix C.2). The original protocol required a significant plug-in advantage before proceeding with its proposed write-up; this hypothesis was not supported and the criterion was not met. The final analysis reports this comparison without claiming plug-in superiority. The draw-construction, weighting and selection conclusions rely on their respective comparisons in Appendix E.

Inverse-imaging and transfer analyses. The image-level paired unit was adopted after inverse-imaging outcomes were available. In InverseBench, the post-measurement weighting amendment replaced independent cumulative weights with coupled likelihood increments and resampling. The exclusion of clone steps was adopted after outcomes were seen, so the reported trajectory statistic is post hoc. The clone-calibrated separation criterion was set after the fixed-threshold transfer results and before evaluation on five further checkpoints.

Draw interventions. Whole-run and single-update aggregation conventions were fixed after outcomes were available. The unmatched log-sum-exp control was excluded from prediction scoring after two of its twelve runs had completed. The intermediate transition-draw ordering failed; exact-cost and single-update full-recovery predictions remain unscored because the required comparator and an unambiguous criterion, respectively, were unavailable. The whole-trajectory length control was specified before it ran but was not registered. Results and their scope are in Appendix E.3.

Summary-based evaluations. Table 1's binary classification and common scoring convention are retrospective. Predictions are computed from the measured summaries and numerical criteria in Appendix D.1; settings without applicable measurements receive no prediction and are excluded from precision calculations. Averaging is scored against LGD-scale outcomes. Appendix E.7.1 defines eligibility and the scoring convention. The terminal pilot's global-rank comparator was not scored in the held-out run (Appendix D.2.1).

Numerical predictions and correction. For Appendix B, the 15 prespecified predictions span gradient, trajectory and selection studies, including auxiliary selection models and the held-out comparison; 9 were supported and 6 failed. Failed predictions are explained beside the corresponding checks. The trajectory-estimand statement was corrected after its check to attach the last potential to the state before the final move rather than to the output.

G REPRODUCIBILITY DETAILS

Two full re-runs measured run-to-run variation: median relative drift was 0.011 in inverse imaging (maximum 0.50) and 0.010 on SiT across 56 summary statistics (maximum 0.73). In the inverse-imaging reruns, median relative drift was 2.0×10^{-4} at $P = 4$ and 0.021 at $P = 16$. Inverse-imaging comparisons pair arms within one process; the twelve checkpoint-transfer implementations run each arm in a separate job, so their comparisons are paired by instance and seed across processes.

Rerunning sampling. Rerunning sampling requires the released weights, the datasets and the configurations given with each testbed's protocol (Appendix E). The text-to-image particle samplers draw injected noise from the global random stream, so the recorded run seed alone does not specify those noise realizations.

Topology-optimization data and evaluator. Topology optimization uses the 64×64 subset of OpenTO's `NITO` split for training and `NITO_test` for evaluation. All 42 test boundary-condition configurations occur in training; this is not an unseen-boundary-condition generalization experiment. Reference compliance is re-solved with the released OptimizeAnyTopology/pyEDGE evaluator rather than imported from a different solver. Signed compliance and volume-fraction errors, thresholding, and refinement variants retain their released definitions.

Illustrative figure. Figure 4 comes from a separate run using each checkpoint's released deployment classifier-free guidance (CFG) scale, whereas the quantitative text-to-image sweeps use CFG-off sampling. Its other settings also differ from the quantitative runs; in particular, every upgrade shown has budget 16, while the three-checkpoint text-to-image study (Appendix E.6) uses $P = 4$. The examples were selected to illustrate cases where upgrades help in one setting and hurt in another, not to estimate how often these outcomes occur. Figure 4 is therefore used only as a qualitative illustration of how scaling strategies can produce visibly different final outcomes under a deployment configuration; it is not pooled with, or used to support the frequencies reported by, the CFG-off budget sweeps.

Compute accounting. Objective-evaluation counts include the evaluations needed to generate and return one output, including scores required for terminal ranking. Scores computed only for the subsequent audit are excluded. Objective evaluations, backbone evaluations, forward-equivalent passes and wall-clock are different quantities, and each table identifies the cost it records. Runtime fields that were not recorded are not imputed, and a candidate pool keeps its acquisition cost when a later analysis reuses it.

G.1 RUNNER-SPECIFIC COMPUTE ACCOUNTING

Figure 3 uses the nominal population budget B: K local draws, P trajectories, or N terminal samples. Equal values of B do not imply equal backbone work, objective evaluations, or wall-clock time.

Pixel-space averaging sums the per-draw objectives before one shared backward. The topology runner accumulates objective derivatives at the predicted clean state, then applies one backbone vector–Jacobian product. The SiT, three-checkpoint text-to-image and transfer runners instead backpropagate each draw through the backbone; the SiT and three-checkpoint text-to-image implementations also evaluate the sampler velocity separately. The normal-deployment sequential implementation accumulates decoder derivatives first and uses one backbone backward per classifier-free-guidance branch, independently of K.

Particles and terminal selection replicate trajectories, but some runners incur extra work for weights. In particular, the SiT particle implementation evaluates the clean-state predictor again after the update, giving three rather than two forward passes per particle at each guided step. Guidance windows must also be included: topology arms with different active-step counts do not necessarily cost the same at equal nominal population size. Its schedule-scale tilt-weighted averaging implementation evaluates the surrogate once for weights and again for derivatives; counting draws alone does not count both passes.

Table 22 totals this work per finished sample. Diffusion test-time scaling commonly prices compute as the number of function evaluations (NFE), which counts network forward passes (Ma et al., 2025). Averaging adds no forward passes, so its NFE equals the plug-in's at every K. Its extra work is objective evaluations and, in the SiT, three-checkpoint text-to-image and transfer runners, one backward pass per draw instead of one per step. Particles and selection repeat whole trajectories, so all three counts grow with P or N. Four of the six particle implementations also evaluate the objective again to update their weights. NFE compares cost only on the same network, and the production, transfer and deployment runners each use several networks.

Table 22: **Compute per finished sample.** Plug-in entries are absolute counts; upgrade entries are multiples of the plug-in. Columns count network forward passes (fwd), input-gradient backward passes (bwd), and objective evaluations (obj). Topology averaging uses LGD-scale draws. ≈ marks rounded entries.

	plug-in			averaging			particles			selection		
runner	fwd	bwd	obj	fwd	bwd	obj	fwd	bwd	obj	fwd	bwd	obj
Pixel-space imaging	100	100	100	1	1	K	P	P	$2P$	N	N	N
Topology, 100 steps	100	100	100	1	1	K	P	P	P	N	N	N
Topology, 71 steps	–	–	–	–	–	–	P	$0.71P$	$0.71P$	–	–	–
SiT flow	126	63	63	1	K	K	$1.5P$	P	$2P$	N	N	N
Text-to-image	56–60	28–30	28–30	1	K	K	P	P	$\approx 2P$	N	N	N
Checkpoint transfer	28	28	28	1	K	K	P	P	P	N	N	N
Deployment, Figure 4	436–468	218–234	109–117	1	1	K	P	P	$\approx 1.25P$	N	N	N

Forward counts exclude activation-checkpoint recomputation. Objective counts include particle-weight evaluations and, in latent-space runners, decoder work. Logging-only evaluations are excluded; terminal ranking adds at most 4% to the displayed selection count. Topology uses all 100 guided steps except particles at $P \in \{4, 8, 16, 32, 64\}$, which use the final 71; both rows use the 100-step plug-in as denominator. The $P = 64$ entry accounts for an additional run outside Table 5's primary matrix. LGD-scale averaging evaluates the objective once per draw; schedule-scale weighted averaging used $2K$ evaluations.

These counts describe operation scaling within each runner; they are not FLOP estimates or comparisons at matched runtime across families. In implementations that share a backward pass, increasing K adds objective evaluations without multiplying backbone work.

H Gain scale and variability band

H.1 Gain metric

Each instance and seed gives one paired difference: the arm's quality minus the plug-in's on the same instance and seed, negated for compliance error so that higher is better. Figure 3 plots the median of these differences divided by the plug-in's interquartile range (IQR), computed from its per-instance scores with seeds pooled. For LGD-scale averaging, paired differences use the plug-in rerun alongside that arm; the normalizing IQR is retained from the schedule-scale experiment, as in the figure.

Using the plug-in's IQR gives every family in a setting the same normalization, independently of the upgrade's variability. The resulting dimensionless scale allows decibels, class log-probabilities and compliance errors to share an axis. The IQR is less sensitive to extreme scores than the standard deviation.

The normalized gain depends on test-set heterogeneity: the same native-unit effect is smaller on this scale when baseline scores vary more across instances. Equal normalized gains on different metrics can also have different practical value. Across-instance variability is distinct from rerun uncertainty. Classifications against each setting's own rerun spread are reported separately from the shared-band counts.

H.2 VARIABILITY BAND

The tables below use the configurations plotted in Figure 3: LGD-scale averaging, and the unchanged particle and selection runs. The shared ± 0.20 band remains fixed, computed from the schedule-scale averaging rows together with particles and selection; these analyses test sensitivity rather than redefine it.

The shared band drawn in Figure 3 is the median normalized seed range across 88 family–setting–budget rows: 80 with three seeds and eight topology rows with five seeds, including multiple topology budgets. For each row, the per-seed *gain estimate* is the median of the upgrade over instances minus the median of the plug-in. The range of these estimates across that row's seeds is divided by the plug-in's IQR, and the median of the normalized ranges is rounded to two decimals. This per-seed estimate is the difference of medians that the fixed rules use, which can differ from the median paired difference on skewed cells. The band summarizes rerun variability of the estimated effect across these rows and depends on both arms; it is not the plug-in's own seed-to-seed variability. Table 23 measures both spreads at the largest plotted budget on settings with complete three-seed data: 28 settings per family, 84 family–setting pairs. Averaging includes topology but excludes SiT's stochastic-reward $K = 32$ point, whose seed-0 run is missing. Particles and selection exclude topology, which lacks paired seed arrays for these checks. Table 24 compares the band definitions on those settings.

Table 23: Three-seed spreads of the effect estimate and the plug-in's median score at the largest plotted budget, in plug-in IQRs.

Spread	Median	First quartile	Third quartile	Maximum
effect	0.23	0.10	0.46	14.6
plug-in	0.22	0.08	0.40	0.8

Table 24: Top-budget counts (above / inside / below) under three bands, using settings with complete three-seed data and the plotted median paired gain. These sensitivity counts do not rescore registered predictions.

Band	Averaging	Particles	Selection
the shared band	2 / 13 / 13	8 / 19 / 1	14 / 14 / 0
each setting's own effect spread	2 / 17 / 9	6 / 22 / 0	21 / 7 / 0
each setting's own plug-in spread	3 / 13 / 12	7 / 20 / 1	20 / 8 / 0

H.3 THRESHOLD SENSITIVITY AND SEED AGREEMENT

Table 25 recomputes these counts over bands from 0.10 to 0.50 baseline IQRs on all plotted settings, including the two-seed SiT averaging endpoint. Selection has the most above-band settings throughout this range. Its settings above the band go from 27 at the narrowest band to 9 at the widest, because its median gain, +0.19 IQRs, sits just inside the edge of the shared band. Counts for averaging and particles also change.

Three seeds per setting give limited information about rerun uncertainty. We report how often all three seeds agree on the sign of the paired median gain (Table 26); this is descriptive agreement, not a confidence guarantee.

H.4 POOL-LEVEL UNCERTAINTY FOR THE TERMINAL EVALUATION

The held-out coverage of the terminal bound (Appendix D.2.1) rests on 640 draws, but those draws are 20 subsamples of each of 16 pools (four implementations, four instances each) at two values of N, 40 draws per pool, so the pool is the independent unit. With the pool as the unit the pooled coverage 0.953 has standard error 0.0123, against a binomial 0.0119 that treats draws as independent, a cluster-bootstrap 90% interval of $[0.934, 0.972]$, and 2 pools below the nominal 0.90 (the lowest at 0.875). This interval does not certify coverage beyond these pools and four held-out implementations.

Table 25: Threshold sensitivity across all 29 plotted settings per family (above / inside / below). The shared band is in bold.

Band (IQRs)	Averaging	Particles	Selection
0.10	5 / 10 / 14	8 / 18 / 3	27 / 2 / 0
0.15	3 / 12 / 14	8 / 20 / 1	24 / 5 / 0
0.20	2 / 13 / 14	8 / 20 / 1	14 / 15 / 0
0.25	1 / 15 / 13	7 / 21 / 1	14 / 15 / 0
0.30	1 / 16 / 12	6 / 23 / 0	13 / 16 / 0
0.40	1 / 16 / 12	5 / 24 / 0	9 / 20 / 0
0.50	1 / 17 / 11	4 / 25 / 0	9 / 20 / 0

Table 26: Settings at the top budget in which all three seeds' effects share a sign, and the direction of that sign.

Family	Settings	All seeds agree	Positive	Negative
averaging	28	20	7	13
particles	28	13	12	1
selection	28	28	28	0

I Scope and Limitations

This appendix summarizes limitations shared across the studies. Experiment-specific restrictions accompany their results; protocol deviations and corrections are collected in Appendix F.

The testbeds and checkpoints form a purposive corpus, so observed frequencies describe these settings rather than prevalence across all guidance problems. The experiments evaluate utility under particular sampling configurations; target-distribution correctness and comparisons with training-time methods are separate questions. Production-checkpoint budget sweeps and transfer tests use CFG-off sampling. Deployment-CFG examples are selected to show what the upgrades look like under a deployment configuration; they do not establish the sweep frequencies there.

These evaluations do not establish general gain guarantees from population summaries. Such guarantees would require summary-to-gain bounds or prediction errors calibrated across tasks.

J Related Work

J.1 Training-free guidance

Training-free guidance appears under several names, among them plug-and-play diffusion priors (Wu et al., 2024; Zheng et al., 2025), zero-shot or inference-time guidance, and diffusion posterior sampling (Chung et al., 2023). The compared scaling strategies build on that framework and are measured against a standard plug-in guide: local-draw averaging (Song et al., 2023), particle methods (Wu et al., 2023; Cardoso et al., 2024; Kim et al., 2025b), and terminal selection (Stiennon et al., 2020; Nakano et al., 2021).

Other inference-time approaches change the sampling procedure itself: decoupled annealing (Zhang et al., 2025), variational reconstruction (Mardani et al., 2024), inpainting-specific resampling (Lugmayr et al., 2022), input-latent optimization (Wallace et al., 2023; Eyring et al., 2024), and schedule restarts (Xu et al., 2023b). Compositional sampling (Du et al., 2023) changes the conditioning problem, while physics-informed diffusion (Bastek et al., 2025) incorporates constraints during training. These are complementary choices outside our three population-budget comparisons.

J.2 Test-time scaling

Recent works reveal that expanding test-time compute yields highly inconsistent performance gains across generative sampling paradigms. For averaging strategies, final performance with perturbed clean predictions remains largely insensitive to sample count in the tested configurations (Ye et al.,

2024). Conversely, averaging predictions after a stochastic sampling step yields clear benefits (Ren et al., 2026a). However, sampling clean endpoints through full inner diffusion rollouts can reduce approximation bias but incurs substantial additional cost (Geyfman et al., 2026). Similarly, in particle propagation, deterministic scaling offers only marginal gains, whereas introducing stochastic sampling yields meaningful compute-scaling benefits in flow models (Kim et al., 2025a).

Related explanations include proposal–posterior support mismatch in loss-based guidance (Shen et al., 2024), while weight degeneracy and lineage collapse motivate particle corrections (Ren et al., 2026b; Shekhar et al., 2026).

Compute allocation also arises in language-model scaling, where rankings vary with model, difficulty and budget (Snell et al., 2025; Brown et al., 2024; Wu et al., 2025; Agarwal et al., 2025), and in verifier-guided diffusion search (Ma et al., 2025). Voting theory already connects aggregation to answer distributions (Liu, 2026). Prior work calls for a better understanding of inference-time compute–performance trade-offs (Singhal et al., 2025). Our cross-family account connects these findings by distinguishing what additional sampling changes from the population and operator properties that shape its returns.

Cost comparisons also use different currencies: denoiser function evaluations (Hwang et al., 2026), runtime and memory (Janati et al., 2024), or cost per successful output (Gleich & Schmidler, 2026). We report objective and backbone work separately (Appendix G).

J.3 Population summaries and predictive evaluation

Population summaries inform sampling and evaluation: effective sample size monitors weight concentration (Doucet & Johansen, 2011), pairwise similarity measures sample diversity (Corso et al., 2024), and rank agreement assesses scoring models against human preferences (Xu et al., 2023a). Methodological precedents distinguish insufficiency from predictive usefulness: counterexamples expose limits of sharpness and two-sided uniform convergence as explanations of generalization (Dinh et al., 2017; Nagarajan & Kolter, 2019); controlled evaluations test which measures predict it (Jiang et al., 2020).

Our matched constructions show that identical weight laws, asymptotic normalized effective sample sizes, or rank summaries can conceal differences in gradient MSE, weighting effects and selection gains. Empirical evaluations compare the predictive usefulness of these summaries with simple baselines. Appendix A.5 details the established statistical ingredients.