When ODE Fidelity Misleads Diffusion Timestep Search

Anonymous authors
Paper under double-blind review

Abstract

Diffusion models can generate images by solving an ordinary differential equation (ODE) that transforms noise into an image. A numerical solver follows this process through repeated model evaluations, using each prediction to update the current state. Fast samplers use fewer evaluations to reduce computation. A schedule specifies when these evaluations occur. We examine choosing among candidate schedules by minimizing the distance from the final state to a reference computed with many more steps. We change the evaluation times while keeping the model, sampling procedure, and total number of evaluations the same. The selected schedules can bring the final state closer to the reference yet score worse on automated preference metrics and on metrics comparing generated and real image distributions. Under strong classifier-free guidance, which increases the text description's influence, the search can leave long gaps in the low-noise region near the end of generation. For a schedule with such gaps, smaller steps early in generation reduce the distance to the reference more, whereas smaller steps near the end improve the learned image-preference score PickScore more. Using fewer evaluations early and more near the end can improve PickScore without increasing the total computation, although the distance to the reference increases. Constrained endpoint search limits the largest low-noise gap and recovers most of the PickScore loss in the tested candidate sets on Stable Diffusion 1.5, Stable Diffusion XL (SDXL), and PixArt-Σ. The failure lies in how the search allocates computation: evaluation times that improve numerical accuracy can leave too few evaluations where they matter more for image quality.

1 Introduction

A diffusion model can generate an image by solving an ordinary differential equation (ODE) that gradually transforms noise into an image. A numerical solver follows this process through repeated model evaluations, using each prediction to update the current state. Faster sampling uses fewer evaluations, making their timing an important choice. A *schedule* specifies when these model evaluations occur.

One way to choose a schedule is to compare its final state with a reference computed using many more steps. Optimal Stepsize for Diffusion Sampling (OSS) searches for evaluation times that match such a reference endpoint (Pei et al., 2025). Related methods learn evaluation times by matching teacher outputs with a perceptual distance (Tong et al., 2025). Here we ask about numerical accuracy relative to the same ODE. But does a schedule that better matches the reference also produce better images? Our main experiment tests selection from fixed candidates. Additional OSS comparisons use model adaptations, rather than reproducing the published methods.

We test this question by changing only the evaluation times. The pretrained model, the ODE it defines, the numerical solver, and the number of model evaluations stay fixed. For a schedule S, let x_S be its final state and x^* the reference starting from the same noise. We call their relative Euclidean distance $d(S) = \|x_S - x^*\|/\|x^*\|$ the *endpoint error*, where $\|\cdot\|$ is the Euclidean (L_2) norm. Lower endpoint error means greater *ODE fidelity*, or numerical

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.

accuracy relative to this reference. For the models studied here, these states are compressed image representations, called latents, which a decoder converts into visible images. We compare schedules with few evaluations against a reference with many more, calling them *coarse* schedules.

Under strong classifier-free guidance, which increases the influence of the text description or *prompt*, minimizing endpoint error can select worse-scoring images. The selected schedule may stop evaluating the model early, or continue near the image end while leaving long gaps between the later evaluations. Both leave the later, low-noise part of generation sparsely sampled. Figure 1 illustrates the resulting disagreement between endpoint error and PickScore, a learned image-preference score. The reference itself scores highest in our interpolation test, so the failure concerns how distance ranks its fast approximations.

With a fixed number of evaluations, spending more early means leaving fewer near the image end. This choice can favor numerical accuracy at the expense of image quality: early refinement can remove more endpoint error, while late refinement can improve image scores more. A solvable two-interval model derives this difference in preferred allocation. We test the explanation by changing where guidance acts and where evaluations are placed. In particular, moving evaluations toward the image end can improve PickScore while increasing endpoint error, with no extra computation. This leads to a simple intervention: limit the largest low-noise gap before minimizing endpoint error. We call this procedure *constrained endpoint search*, and the limit its *spacing bound*. It preserves evaluations near the image end while retaining the endpoint-error objective.

Contributions.

1. **Endpoint error can misrank image quality.** With the model, solver, and evaluation count fixed, endpoint-error selection can produce worse image scores under strong guidance. Preference and distribution metrics establish the disagreement on Stable Diffusion 1.5 (SD1.5), and preference scores reproduce it on Stable Diffusion XL (SDXL) and PixArt-Σ. Higher-precision sampling and reference computation preserve the PixArt result.
2. **A low-noise spacing constraint reduces the quality loss.** Constrained endpoint search limits the largest gap between evaluations near the image end before minimizing endpoint error. It recovers most of the PickScore loss in the studied strong-guidance settings on these three models. The two-interval model and controlled experiments explain the reason for this intervention: numerical accuracy and image quality can favor spending the same computation in different parts of generation.

2 Related work

Numerical accuracy and image quality. Xue et al. (2024) minimize an analytic bound on ODE error using only the relation between time and noise level. Their schedule reaches low noise (Appendix A.6). Other objectives compare states along a reference trajectory, the sequence of states during generation, or approximate how step errors affect the final state (Chen et al., 2024a; Wang et al., 2025; Huang et al., 2026). When a fast student model is trained to imitate a slower teacher, it can follow the teacher's ODE more accurately and produce worse samples (Vouitsis et al., 2024). We instead vary evaluation times within one pretrained model and solver at the same evaluation count.

Guidance and the generation path. Classifier-free guidance combines predictions with and without the prompt, changing the ODE (Ho & Salimans, 2022). Its effect on the sampling distribution has motivated changes to when and how guidance is applied (Chidambaram et al., 2024; Bradley & Nakkiran, 2025; Chung et al., 2025; Kynkäänniemi et al., 2024; Wang et al., 2024; Sadat et al., 2025; Karras et al., 2024). ERK-Guid uses embedded Runge–Kutta error estimates to correct guidance (Kong et al., 2026). We keep this ODE fixed and change its evaluation times. The perception–distortion trade-off (Blau & Michaeli, 2018) also distinguishes distance from quality. We ask whether distance to a high-scoring reference orders fast approximations correctly.

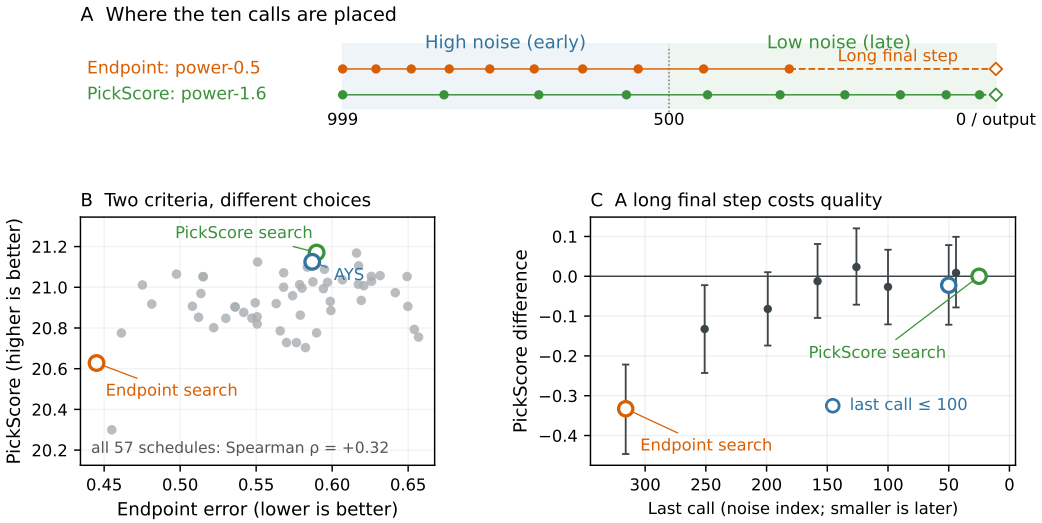

Figure 1: Timestep placement separates numerical accuracy from image quality (SD1.5, guidance 7.5, ten model calls). **A:** Dots show the ten calls of power-0.5 (endpoint search) and power-1.6 (PickScore search). Indices label training noise levels, decreasing toward the output. Diamonds mark the output without another call. **B:** The two criteria select different schedules among 57 candidates on 16 calibration prompts. **C:** On 128 separate prompts, PickScore falls as the last call moves away from the image end. Differences are relative to power-1.6, with paired-prompt 95% confidence intervals. Blue marks endpoint search restricted to a late last call ($t \leq 100$).

Faster sampling. Faster solvers change the numerical update (Zhang & Chen, 2023; Zheng et al., 2023) or jointly search its coefficients and evaluation times (Wang et al., 2025). RX-DPM reduces error by extrapolating multiple ODE solutions (Choi et al., 2025). Align Your Steps (AYS) is a schedule-search method based on bounds on distribution error (Sabour et al., 2024). GITS instead assigns a cost to each possible step and searches for the lowest-cost sequence (Chen et al., 2024a). Other methods use the geometry of the generation path (Zhou et al., 2026).

These searches differ in what they optimize. Zhang et al. (2026) directly optimize a final image metric. OSS minimizes squared distance between fast and reference endpoints (Pei et al., 2025, Eq. 8). Tong et al. (2025) learn both solver times and the times supplied to the model. Their training uses LPIPS, a perceptual image distance, to match teacher outputs while allowing small changes to the initial noise. This differs from our fixed-noise, relative-L_2 selection. These objectives motivate our test, but the algorithms and allowed schedules differ from our fixed candidates.

GITS can sum the Euclidean sizes of individual step errors, charging directly for long jumps. Our implementation of this cost preserves low-noise spacing without a constraint. Other searches score final samples (Liu et al., 2024), bound the error introduced by each step (Jo & Choi, 2026), or allocate evaluations using an Euler-error surrogate (Huang et al., 2026). These costs need not rank schedules like endpoint error: later propagation and cancellation can change what a step error contributes to the endpoint. Appendix A.13 examines this distinction.

3 When endpoint accuracy selects worse images

A schedule can reduce numerical error by spending evaluations in intervals that matter less for image quality. We first establish this disagreement under a fixed model and sampling procedure. Sections 4 and 5 then explain the allocation conflict and test a bound on low-noise spacing.

3.1 Only the evaluation times change

Our main experiment uses Stable Diffusion 1.5 (Rombach et al., 2022) at 512×512 with ten model calls (number of function evaluations, NFE). DPM-Solver++ 2M is the numerical solver: it uses the current and previous predictions to advance the state (Lu et al., 2022). With no fresh noise added during sampling, it is deterministic. For each prompt, schedules start from the same latent noise. Sampling moves from large to small native indices $t \in \{0, \ldots, 999\}$, which label the training noise levels.

We compare 57 schedules, including uniform, log-SNR (log signal-to-noise ratio), Karras (Karras et al., 2022), AYS, GITS, power-law schedules, and mixtures. Power-p denotes the exponent controlling spacing; $p < 1$ places more calls early and $p > 1$ places more late. Appendix A.2 lists the evaluation times, also called nodes. We choose schedules on 16 prompts (the calibration set), then evaluate them on separate prompts (the held-out set). *Endpoint search* chooses the schedule with the lowest mean endpoint error. *PickScore search* chooses the highest mean PickScore (Kirstain et al., 2023). We call their outputs the endpoint-selected and PickScore-selected schedules. Our *constrained endpoint search* first excludes schedules with large low-noise gaps, then applies endpoint search. We also compare with the published AYS schedule, which requires no calibration on our prompts. Together, these comparisons test the objective, the constraint, and an existing schedule.

We report 95% confidence intervals for the fixed selections. These use paired prompt resampling within four styles: anime, concept art, paintings, and photographs. An interval containing zero leaves the direction of a difference uncertain. A *low-noise gap* is a step entering or within the final region $t \leq 500$, including the step to the output. Thus it can be large even when the last model call is near the image end.

The reference x^* uses DDIM, a solver that advances the state from each current model prediction (Song et al., 2021), with 1000 steps and the same noise, prompt, and guidance. Reference refinement changes the endpoint much less than the selected schedules differ. The reference retains a small amount of noise, while candidates output at zero noise. Repeating the comparison with a fine DPM-Solver++ reference and matching zero-noise output preserves the calibration ranking and the held-out ordering (Appendix A.4).

3.2 Closer endpoints, lower image scores

At guidance 7.5, the two objectives choose visibly different allocations: endpoint selection leaves one long final step, whereas PickScore search continues evaluating near the image end (Table 1). The endpoint-selected schedule is closer to the reference on 119 of 128 held-out prompts, but the PickScore-selected schedule gains $0.333\,[0.222, 0.446]$ PickScore and wins on 91. Human Preference Score v2.1 (HPSv2.1), another learned image-preference score (Wu et al., 2023), gives the same ordering. All four styles favor the PickScore-selected schedule on average. The exact score winner varies when the small calibration set is resampled, while endpoint search consistently favors early stopping (Appendix A.2).

Table 1 follows three steps: select among 57 schedules, add one candidate and select again, then redistribute three evaluations of the new winner at fixed cost (Section 4). Its values come from a common image rerun, so the means differ slightly from the original comparison above.

Distribution metrics compare groups of generated images with real images. Using captions from the COCO image–caption dataset (Lin et al., 2014), Fréchet Inception Distance (FID) (Heusel et al., 2017) is 32.15 for the endpoint-selected schedule and 22.23 for the PickScore-selected schedule (lower is better). Kernel Inception Distance (KID) (Bińkowski et al., 2018) and a distance computed with the DINOv2 image encoder (Oquab et al., 2024) agree. So do ImageReward, a learned preference score (Xu et al., 2023), and CLIP similarity, which measures agreement between image and text (Radford et al., 2021). The schedule selected by endpoint error remains closer to the reference even when distance is measured after decoding the images.

Table 1: Schedule choice and redistribution on SD1.5: ten calls, guidance 7.5, DPM-Solver++ 2M, and 128 prompts. Block 1 shows selections and AYS from the same 57 candidates. Block 2 adds times found by an OSS adaptation and transferred to our solver; this candidate becomes the endpoint-error winner. Block 3 moves three of its early calls into late gaps, preserving the first call, last call, and total count. Last-call indices decrease toward the image end; arrows indicate better metric values. With the OSS adaptation's own DDIM-equivalent update, its PickScore difference from PickScore search has an interval containing zero (Appendix A.7).

Selection method and resulting schedule	Last call	Largest low-noise gap	Endpoint error ↓	PickScore ↑
1. Choose from 57 schedules, including AYS				
Endpoint search (power-0.5)	316	316	0.467	20.30
PickScore search (power-1.6)	25	124	0.597	20.65
Constrained endpoint search				
uniform–Karras 0.25	74	124	0.533	20.65
AYS (published schedule)	24	112	0.587	20.71
2. Add OSS-transferred times: choose again from 58 schedules				
Endpoint search (OSS-transferred)	45	181	0.456	20.52
3. Move three early calls of the new winner into late gaps				
Same schedule after moving three calls	45	91	0.598	20.66

3.3 Guidance matters, but the failure is not universal

The disagreement also appears with two other solvers, DDIM and UniPC (Zhao et al., 2023), on SDXL (Podell et al., 2024), and at twenty-four evaluations, but its confidence interval includes zero at sixteen. SDXL endpoint errors use 32 anime prompts, while scores use all 128 prompts. PickScore and HPSv2.1 sometimes disagree (Appendices A.2 and A.3).

PixArt-Σ (Chen et al., 2024b), a text-conditioned diffusion transformer, also shows the disagreement at both strong and default guidance. At the default setting, PickScore search gains roughly half a PickScore point over endpoint search, and HPSv2.1 agrees. Computation in 32-bit floating-point arithmetic (FP32) preserves the result with validated reference solutions (Appendix A.10). Without guidance, the earlier lower-precision sweep instead gives the highest mean PickScore to endpoint search (Figure 4).

The flow-matching model SD3.5-Medium (Esser et al., 2024), which learns a velocity from noise toward images, gives mixed results across prompt sets and lacks a validated strong-guidance reference (Appendix A.9).

Search design also matters. With their own first-order update, both OSS adaptations reach low noise: SD1.5 has no distinguishable PickScore deficit to the PickScore-selected schedule, and PixArt gains no distinguishable preference benefit from the constraint (Appendix A.7). These comparisons change the update, reference, and allowed schedules together, so they do not isolate why OSS allocates evaluations differently.

4 Why accuracy and quality favor different intervals

At a fixed evaluation count, a smaller step in one part of generation leaves less computation for another. We ask why endpoint search can favor early steps even when later steps matter more for image quality. We first separate the conditions for improving endpoint

error and image score, then derive their consequences in a simple model and test them through interventions.

4.1 Reducing error and improving a score require different corrections

Fix the prompt, initial noise, and guidance. Let $x = x_S$, $e = x - x^*$, and let an alternative schedule or refinement of selected intervals produce $x' = x - r$, where r is the resulting change in the endpoint. Let Q be a continuously differentiable image score, with decoding included when the state is latent. Its gradient points toward a higher score. Define the average gradient along the correction as $\bar{g}_r = \int_0^1 \nabla Q(x - ur)\, du$. Expanding the squared norm and applying the fundamental theorem of calculus gives the exact gains

$$\begin{aligned} \|e\|^2 - \|e - r\|^2 &= 2\langle e, r\rangle - \|r\|^2, \\ Q(x - r) - Q(x) &= -\langle \bar{g}_r, r\rangle. \end{aligned} \tag{1}$$

Endpoint error decreases while the score falls precisely when $2\langle e, r\rangle > \|r\|^2$ and $\langle \bar{g}_r, r\rangle > 0$. Reducing the error norm improves numerical accuracy. Improving quality instead depends on whether the same correction points toward a higher score. The correction r includes changes in later states and in predictions stored by the solver (its history). For fixed $x^* \neq 0$, reducing the squared norm reduces $d(S)$; calibration averages $d(S)$ across prompts.

This also explains why the reference can score best while distance to it ranks schedules poorly: errors in different directions need not have the same effect on the score. In the interpolation test, the two directions even exchange their score ordering as we move away from the reference (Appendix A.2). Appendix B.1 shows that unequal directional sensitivity can cause misranking arbitrarily close to a score maximum.

4.2 Larger early errors can draw evaluations away from the image end

Classifier-free guidance combines the velocity predicted with the prompt, v_{cond}, and without it, v_{uncond}, as $v_w = v_{\text{uncond}} + w(v_{\text{cond}} - v_{\text{uncond}})$, where w is the guidance strength. To isolate a possible consequence, consider one early and one late interval. Suppose stronger guidance increases the error made in the early interval, while image score remains more sensitive to late error. The following model shows how the two objectives then allocate the same number of evaluations differently.

Proposition 1 (allocation in a two-interval model)**.** Consider the two-dimensional ODE in Appendix B.1, with an early and a late interval of unit length. Its velocity changes at rates $a, b > 0$ in the two intervals, affecting separate coordinates. The exact endpoint is $z^* = (1, 1)$, and the score is $Q(z) = -\lambda_H(z_1 - 1)^2 - \lambda_L(z_2 - 1)^2$, where $0 < \lambda_H < \lambda_L$. Explicit Euler uses the velocity at the start of each step. Its schedule has N calls and includes the interval boundary. Let z_N be the numerical endpoint. For n early calls, uniform spacing within each interval minimizes both squared endpoint error E and score loss L:

$$E(n) = \frac{a^2}{4n^2} + \frac{b^2}{4(N-n)^2}, \qquad L(n) = -Q(z_N) = \frac{\lambda_H a^2}{4n^2} + \frac{\lambda_L b^2}{4(N-n)^2}.$$

Treating n as continuous, their optimal early-call counts n_E and n_Q satisfy

$$\frac{n_E}{N - n_E} = \left(\frac{a}{b}\right)^{2/3}, \qquad \frac{n_Q}{N - n_Q} = \left(\frac{\lambda_H}{\lambda_L}\right)^{1/3} \left(\frac{a}{b}\right)^{2/3} < \frac{n_E}{N - n_E}.$$

A bound $h \leq 1/m$ on each late step width, with integer m such that $n_Q \leq N - m < n_E$, moves the continuous endpoint optimum to $n = N - m$, increasing endpoint error and improving the score.

Larger early errors draw calls early, while greater score sensitivity to late error keeps calls near the end. In the ten-call example (Table 17), minimizing endpoint error gives eight early and two late calls, maximizing score gives four early and six late, and the bound gives six early and four late. These are toy-model optima, not diffusion measurements. They require no later amplification and do not predict which diffusion settings exhibit the conflict.

In the fixed candidate sets, endpoint-selected schedules also shift toward earlier calls as guidance increases: from five to eight early calls on SD1.5 and from five to seven on PixArt-Σ. PickScore-selected schedules keep four to six early calls. Here "early" means above native index 500. Table 5 and Figure 4 give the guidance sweeps. Appendix A.8 reports finite propagation measurements on SD1.5.

4.3 Refining different intervals separates the two benefits

Where guidance acts. Applying strong guidance only during high-noise generation reproduces the early-stop selection and quality gap. Applying it only at low noise gives a gap whose confidence interval includes zero. These interventions connect the allocation shift to high-noise guidance (Table 5), complementing work on guidance restricted to selected intervals (Kynkäänniemi et al., 2024). The guidance sweep uses a common 24-schedule set for strengths 1–5; restricting the 7.5 selection to that set preserves both choices.

Where smaller steps help. Starting from power-0.5, we refine either the high-noise intervals ending above index 500 or the low-noise portion beginning with the step that crosses this boundary. On 64 prompts, both refinements improve PickScore (Figure 2). High-noise refinement removes more endpoint error, whereas low-noise refinement improves PickScore more, by a paired difference of $0.23\,[0.09, 0.37]$. The contrast repeats on additional prompts. These refinements use extra computation.

The controls show when this contrast changes. For two schedules that already cover low noise, the score-change intervals include zero (Table 10). At guidance 1, low-noise refinement gives both the larger endpoint-error reduction and the score gain. Refinement changes error direction as well as its size, so these gains do not decompose the original error. Appendix A.6 specifies the fine integration and solver-history handling. We next test the allocation tradeoff without extra computation by moving calls between intervals while keeping their total fixed.

Improving quality without adding evaluations. The OSS-transferred schedule selected from 58 candidates in Table 1 reaches low noise but leaves large intermediate gaps. We keep the same DPM-Solver++ system and change where three evaluations occur.

We replace three high-noise evaluations with one inside each of its three large low-noise gaps. The first call, last call, and ten-call total stay fixed. We choose among rearrangements using calibration endpoint error. Across 256 evaluation prompts, the rearrangement gains $0.124\,[0.051, 0.197]$ PickScore while also increasing endpoint error. The gain appears in both the original prompt set and a second set.

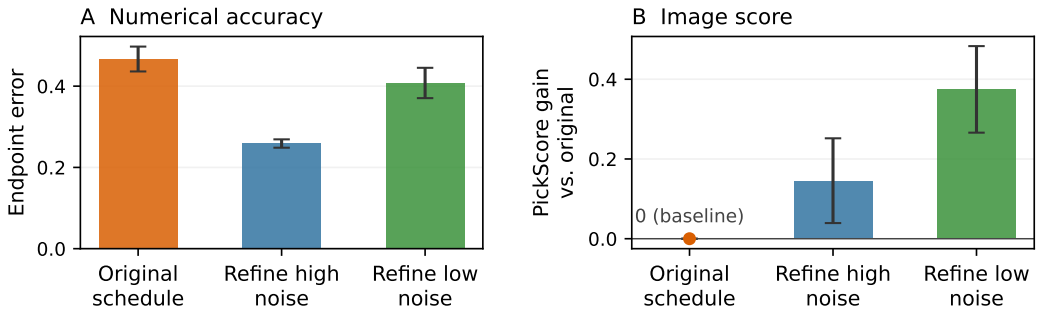

Figure 2: Refining different intervals improves different outcomes (SD1.5, guidance 7.5). Starting from power-0.5, we refine the high-noise or low-noise intervals, including the boundary-crossing step in low-noise refinement. **A:** High-noise refinement reduces endpoint error more. **B:** Low-noise refinement improves PickScore more. Both use extra computation. Bars show paired-prompt 95% bootstrap intervals on 64 prompts; the original schedule's score gain is zero by definition.

On the second set alone, the unrearranged schedule's deficit to PickScore search is smaller, with a confidence interval containing zero. Appendix A.7 reports each comparison.

5 Keeping enough evaluations near the image end

The allocation experiments suggest a direct way to protect the late part of generation. *Constrained endpoint search* excludes schedules with overly large low-noise gaps, then minimizes endpoint error among those that remain. The constraint changes which schedules are eligible, while the original objective still chooses between them.

5.1 A maximum gap preserves low-noise evaluations

Write the model-call schedule as $S = (s_0, \ldots, s_{N-1})$, with integer indices $s_0 > \cdots > s_{N-1} \geq 0$. For gap accounting, set $q_i = s_i$ $(i < N)$ and append $q_N = 0$ for the candidate's output event. This final denoising update uses the last prediction without another model call. Native index 0 has positive noise; the appended event has zero noise but shares its gap coordinate. Let $G_{\text{low}}(S)$ be the largest gap entering or within $t \leq b = 500$, bounded by B_N with dimensionless factor γ:

$$G_{\text{low}}(S) = \max_{0 \leq i < N:\, q_{i+1} \leq b} (q_i - q_{i+1}), \qquad G_{\text{low}}(S) \leq B_N, \qquad B_N = \text{round}(\gamma\, 999/N). \quad (2)$$

The average spacing of a uniform schedule is $999/N$, so γ sets the allowed multiple. Here round means nearest-integer rounding, and $t = 500$ corresponds to noise level $\sigma = 1.62$ on the scaled-linear schedule shared by both SD models (Appendix A.6 gives the conversion). We choose $\gamma = 1.25$ on SD1.5 calibration prompts, giving $B_{10} = 125$ and $B_{24} = 52$, and reuse it for guidance 5, SDXL, and NFE 24. This is an empirical choice. In the original 57-schedule set, changing the boundary to 400 or 600 leaves the selected schedule unchanged. Tested values of γ from 1.1 to 1.5 select different schedules, but each has a PickScore difference from PickScore search whose confidence interval includes zero (Appendix A.6). These checks support a range of usable settings here; a new noise schedule requires its own calibration.

5.2 The bound recovers much of the score loss

We test schedules with freely placed evaluations and schedules whose last model call is fixed at index 24. In the original set, the remaining mean PickScore difference from PickScore search is close to zero, with confidence intervals containing zero in both the original and enlarged evaluations. Table 2 shows the results across models, solvers, and final-call restrictions. Recovery is incomplete in some settings: a smaller but distinguishable gap remains at moderate guidance on SD1.5, and at default guidance on PixArt. The bound also excludes some schedules that score well, so it should be read as a conservative spacing check rather than a sharp quality threshold. Appendix A.6 reports these exceptions and the threshold sensitivity tests.

Requiring only a late final call also restores PickScore in the main candidate set, but it leaves the preceding gaps uncontrolled. To separate these features, we fix every candidate's last call at 24. Endpoint search still chooses a long preceding gap, with a quality loss at guidance 5 and on SDXL. At guidance 7.5, the confidence interval includes zero (Table 5). These controls, together with the fixed-count redistribution in Section 4, motivate checking the whole low-noise region.

Expanding the set with 512 random schedules preserves the early-stopping endpoint choice. The spacing bound selects uniform and improves both preference scores. This uses a new 16/112 selection/evaluation split of previously studied prompts (Appendix A.6).

Distribution metrics show the same improvement without full recovery. On COCO, the spacing bound closes about four fifths of the FID gap, but the PickScore-selected schedule still performs better. The remaining difference shows that recovery depends on which aspect of image quality is measured (Table 9).

Table 2: Limiting low-noise gaps recovers PickScore across settings. The middle column compares constrained endpoint search with endpoint search, using the same candidate set. The right column compares PickScore search with constrained endpoint search: positive means PickScore search is better. Brackets are 95% confidence intervals. Ten calls and guidance 7.5 unless stated. Schedule identities and endpoint errors are in Appendix A.6.

Setting	PickScore gained by adding the bound	Remaining advantage of PickScore search
Evaluation times are free		
SD1.5	+0.318 [+0.223, +0.416]	+0.014 [−0.078, +0.102]
SD1.5, 24 calls	+0.176 [+0.107, +0.246]	+0.023 [−0.036, +0.082]
SD1.5, UniPC solver	+0.255 [+0.155, +0.361]	+0.033 [−0.054, +0.119]
PixArt-Σ, 7.5 (FP32)	+0.391 [+0.282, +0.498]	+0.006 [−0.011, +0.023]
PixArt-Σ, 4.5 (FP32)	+0.402 [+0.346, +0.460]	+0.084 [+0.016, +0.153]
Every candidate's last call is fixed at index 24		
SD1.5	+0.097 [+0.015, +0.179]	−0.024 [−0.109, +0.060]
SD1.5, guidance 5	+0.222 [+0.137, +0.312]	+0.023 [−0.052, +0.093]
SDXL	+0.120 [+0.029, +0.210]	same schedule

5.3 When constrained endpoint search helps

Benefits depend on the model and search. On SD1.5, constrained endpoint search has lower PickScore than AYS (Table 8). On PixArt-Σ at default guidance, it gains about 0.09 PickScore over uniform and also exceeds SD AYS times transferred without re-optimization (Appendix A.10). Published schedules need no calibration, and direct PickScore search uses fewer generation calls than endpoint search in our SD1.5 setup (Appendix A.6). When matching a reference remains the search objective, the bound provides a way to guard against large low-noise gaps.

6 Limitations

The loss depends on the candidates: expanding the set reduces it to about 0.1 PickScore. The OSS comparisons use model adaptations rather than published configurations, and learned-timestep training (Tong et al., 2025) is untested. The spacing bound is calibrated on SD1.5 and scaled by the evaluation count. Several schedules exceed it without a clear PickScore loss, while published GITS has an intermediate gap and a small loss whose confidence interval excludes zero (Table 8). It is a conservative bound, not a sharp quality threshold.

SD3.5's strong-guidance reference remains unverified, and its disagreement uses 64 previously studied prompts (Appendix A.9). PixArt's reference checks cover 32 of the 128 evaluation prompts (Appendix A.10). The main quality comparisons use automatic metrics. Human evaluation covers 100 pairs per comparison on SDXL (Appendix A.11).

Regional refinements use 64 prompts for early stopping and 16 per already-covered schedule, where the score-change intervals include zero. The 16-prompt propagation measurements are exploratory. Local-error theory excludes multistep and stochastic updates; Proposition 2's derivative constants are not estimated.

7 Conclusion

Under strong guidance, endpoint search can leave too few evaluations near the image end. Redistributing evaluations can improve PickScore at fixed cost even while increasing endpoint error. Limiting low-noise gaps recovers much of the observed loss. When searching by endpoint error, check low-noise spacing and evaluate the selected images separately.

Reproducibility Statement

Appendix A documents model settings, prompt splits, schedule construction, reference solvers, scoring, and paired analyses. Appendix B gives mathematical definitions, assumptions, and proofs. Experiment configurations, per-prompt results, and scripts reproduce the reported tables and figures.

AI Use Disclosure

In this work, we used generative AI tools (Codex) for formulate mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, interpret results. We have not used generative AI tools for Generate synthetic data sets, help develop theoretical models or conceptual frameworks, provide critical ingredients for proving mathematical claims, assist with translation, clean and reformat dataset, and support qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools for draft parts of a research paper (appendix, related works), sourcing/searching for information, research paper to improve readability, format references. We have reviewed all AI-assisted work. E.g., we checked the correctness LLM-generated mathematical proofs, LLM-generated code was verified and tested for correctness, etc. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

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

Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6228–6237, 2018.

Arwen Bradley and Preetum Nakkiran. Classifier-free guidance is a predictor-corrector. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=zrWNtzSZsf.

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

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-Σ: Weak-to-strong training of diffusion transformer for 4K text-to-image generation. In *European Conference on Computer Vision (ECCV)*, 2024b.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru R. Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *International Conference on Learning Representations (ICLR)*, 2023.

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.

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

Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. CFG++: Manifold-constrained classifier free guidance for diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning (ICML)*, 2024.

Dhruba Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Ernst Hairer, Syvert P. Nørsett, and Gerhard Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer, 2nd edition, 1993.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Yilie Huang, Wenpin Tang, and Xun Yu Zhou. Art for diffusion sampling: Continuous-time control and actor-critic learning. *arXiv preprint arXiv:2607.02137*, 2026.

Sangwoo Jo and Sungjoon Choi. Formalizing the sampling design space of diffusion-based generative models via adaptive solvers and wasserstein-bounded timesteps. *arXiv preprint arXiv:2602.12624*, 2026.

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

Tero Karras, Miika Aittala, Tuomas Kynkäänniemi, Jaakko Lehtinen, Timo Aila, and Samuli Laine. Guiding a diffusion model with a bad version of itself. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023.

Inho Kong, Sojin Lee, Youngjoon Hong, and Hyunwoo J. Kim. Error as signal: Stiffness-aware diffusion sampling via embedded runge-kutta guidance. In *International Conference on Learning Representations (ICLR)*, 2026.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Yangming Li and Mihaela van der Schaar. On error propagation of diffusion models. In *International Conference on Learning Representations (ICLR)*, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Enshu Liu, Xuefei Ning, Huazhong Yang, and Yu Wang. A unified sampling framework for solver searching of diffusion probabilistic models. In *International Conference on Learning Representations (ICLR)*, 2024.

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

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *European Conference on Computer Vision (ECCV)*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

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

Jianning Pei, Han Hu, and Shuyang Gu. Optimal stepsize for diffusion sampling. *arXiv preprint arXiv:2503.21774*, 2025.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations (ICLR)*, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

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

Amirmojtaba Sabour, Sanja Fidler, and Karsten Kreis. Align your steps: Optimizing sampling schedules in diffusion models. In *International Conference on Machine Learning (ICML)*, 2024.

Seyedmorteza Sadat, Otmar Hilliges, and Romann M. Weber. Eliminating oversaturation and artifacts of high guidance scales in diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

Lawrence F. Shampine and H. A. Watts. Global error estimates for ordinary differential equations. *ACM Transactions on Mathematical Software*, 2(2):172–186, 1976. doi: 10.1145/355681.355687.

Robert D. Skeel. Thirteen ways to estimate global error. *Numerische Mathematik*, 48:1–20, 1986. doi: 10.1007/BF01389440.

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

Hans J. Stetter. The defect correction principle and discretization methods. *Numerische Mathematik*, 29:425–443, 1978. doi: 10.1007/BF01432879.

Chenxi Tao and Seung-Kyum Choi. On the role of strain and vorticity in numerical integration error for flow matching. *arXiv preprint arXiv:2605.06680*, 2026.

Vinh Tong, Trung-Dung Hoang, Anji Liu, Guy Van den Broeck, and Mathias Niepert. Learning to discretize denoising diffusion ODEs. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xDrFWUmCne.

Noël Vouitsis, Rasa Hosseinzadeh, Brendan Leigh Ross, Valentin Villecroze, Satya Krishna Gorti, Jesse C. Cresswell, and Gabriel Loaiza-Ganem. Inconsistencies in consistency models: Better ODE solving does not imply better samples. *NeurIPS Workshop on Attributing Model Behavior at Scale*, 2024.

Shuai Wang, Zexian Li, Qipeng Zhang, Tianhui Song, Xubin Li, Tiezheng Ge, Bo Zheng, and Limin Wang. Differentiable solver search for fast diffusion sampling. In *International Conference on Machine Learning (ICML)*, 2025.

Xi Wang, Nicolas Dufour, Nefeli Andreou, Marie-Paule Cani, Victoria Fernandez Abrevaya, David Picard, and Vicky Kalogeiton. Analysis of classifier-free guidance weight schedulers. *Transactions on Machine Learning Research*, 2024.

Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023.

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)*, 2023.

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

Pedro E. Zadunaisky. On the estimation of errors propagated in the numerical integration of ordinary differential equations. *Numerische Mathematik*, 27(1):21–39, 1976. doi: 10.1007/BF01399082.

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

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

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

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

Zhenyu Zhou, Defang Chen, Siwei Lyu, Chun Chen, and Can Wang. Analyzing and improving fast sampling of text-to-image diffusion models. In *European Conference on Computer Vision (ECCV)*, 2026.

A Experimental details

The sections below document schedule selection, reference checks, allocation interventions, and cross-model comparisons. Appendix B contains the theoretical derivations.

A.1 Models and paired measurements

Stable Diffusion v1.5 uses its public checkpoint at 512×512 resolution. SDXL-base-1.0 uses 1024×1024. PixArt-Σ XL/2 uses its public 512×512 checkpoint (Appendix A.10). The main search fixes the pretrained model, guidance, solver, initial noise, and number of calls. Calibration prompts are disjoint from the main held-out prompts. Unless stated otherwise, confidence intervals use 10,000 bootstrap resamples: we sample prompts with replacement within each style and keep all schedules for a prompt together. ImageNet experiments instead resample images. Correlations use Spearman ρ, which compares rankings; reported p-values test zero correlation.

The local-error tests use public DiT-XL/2 (Peebles & Xie, 2023) and SiT-XL/2 (Ma et al., 2024) checkpoints on ImageNet at 256^2, with guidance 1. DiT uses deterministic DDIM; SiT uses Euler or Heun to integrate its learned velocity field. Their state has no solver history, allowing individual local errors to be isolated.

A.2 Selecting schedules by endpoint error and image preference

The main SD1.5 experiment uses 512×512 images, guidance 7.5, and ten-call deterministic DPM-Solver++ 2M with midpoint updates and a lower-order final step. Its 57 unique rounded schedules combine uniform native index, log-SNR, Karras ($\rho = 7$), AYS, published GITS, powers $t_i = 999(1 - i/10)^p$ for $i = 0, \ldots, 9$ and $p = 0.4, 0.5, \ldots, 2.5$, and coordinate-wise interpolations of the named schedules. Each starts at 999 with ten distinct decreasing nodes.

After excluding prompts from the DDIM study, we draw four calibration and 32 held-out prompts per style from the HPSv2 benchmark, pairing initial latents across schedules. Calibration selects power-0.5 by mean relative endpoint distance and power-1.6 by PickScore. Denote these schedules by S_{ode} and S_{pref}, respectively. Their evaluation indices are

$$S_{\text{ode}} = (999, 948, 894, 836, 774, 706, 632, 547, 447, 316),$$
$$S_{\text{pref}} = (999, 844, 699, 565, 441, 330, 231, 146, 76, 25).$$

The reference uses all 1000 native DDIM nodes, inheriting `set_alpha_to_one=False`: its terminal signal-retention factor is $\bar{\alpha}_0 \approx 0.99915$, not 1, and its last native-0 update leaves the state unchanged. Candidate DPM-Solver++ uses `final_sigmas_type=zero`, so the two terminal conventions differ. Reference convergence is checked in Appendix A.4. Replaying common model outputs through an independent DPM-Solver++ scheduler is bit-identical.

Post-hoc selection stability uses 10,000 style-stratified bootstrap samples. Endpoint search chooses power-0.4, 0.5, or 0.6 in 97.79% of draws. Its modal choice is power-0.5 (69.69%). The preference winner is less stable: power-1.6 is selected in 16.60%, AYS in 20.91%, and the AYS–GITS midpoint in 34.18%. The two objectives select different schedules in every draw. The original pair remains fixed for all reported tests.

Sensitivity to the error definition. A retrospective check reuses all 57 schedules and 16 calibration prompts from the SD1.5 rerun in Appendix B.1. Mean relative and raw L_2 select power-0.5 (last call 316); mean squared L_2 selects power-0.4 (last call 398). Early stopping persists, but the winner depends on aggregation. This calibration-only check has no new held-out quality comparison and does not reproduce OSS's update, reference, or search space.

Sensitivity to calibration prompts. On 64 new prompts (16 per style, disjoint from earlier experiments), the 1000-step reference again selects power-0.5. Powers 0.4–0.6 win 99.5% of 10,000 stratified resamples; every selected schedule stops at 199 or above. PickScore selects a uniform–AYS mixture (last call 80, gap 103), with resampled choices also including powers 1.1 and 1.2. The new pair differs by 0.278 [0.147, 0.412] PickScore on calibration, with the endpoint choice 0.077 closer. The original pair's gap is 0.088 [−0.128, 0.296]. Early stopping persists, but the preference winner varies; the new pair has no separate held-out evaluation.

Scoring runs. All eleven schedules in Table 8, including both bound-optimized schedules, use stored JPEGs scored together on the same 128 prompts with a fixed model revision, processor, and package version. This replaces scores whose processor settings were unrecorded. Artifacts identify schedules by ID and integer nodes. Tables 3 (schedule comparisons), 7, and 8 use this cache; Table 1 uses regenerated images and a separate common scoring run (Appendix A.7). Reference-image comparisons retain the interpolation cache.

Evaluation cost. Raising the failing schedule to 24 evaluations gains 0.301 [0.222, 0.379] PickScore over its ten-evaluation version, but does not resolve an advantage over ten-evaluation AYS, GITS, uniform, or Karras. Search costs are given with the named-schedule comparison in Appendix A.6.

Table 3: SD1.5, DPM-Solver++, ten calls, 128 evaluation prompts. Each difference is first minus second: positive endpoint error means farther from the reference; positive scores mean better images. "Endpoint" and "score" identify the selections. Brackets are paired 95% intervals; NR means unreported. Schedule comparisons use the common scoring cache, reference comparisons the original interpolation cache.

Comparison	Endpoint error	PickScore	HPSv2.1
Score − endpoint	+0.1305 [0.1123, 0.1495]	+0.333 [+0.222, +0.446]	+0.0172 [+0.0127, +0.0219]
AYS − endpoint	NR	+0.4101 [0.2985, 0.5216]	+0.01990 [0.01534, 0.02467]
Reference − endpoint	NR	+0.4779 [0.3767, 0.5807]	+0.02075 [0.01703, 0.02460]
Reference − score	NR	+0.1452 [0.0318, 0.2591]	+0.00358 [−0.00016, 0.00725]

The interpolation experiment decodes $x^* + \alpha(x_S - x^*)$ at $\alpha = 0, .25, .5, .75, 1$ for both selected schedules. Ordinary least-squares slopes are fitted within prompt. The PickScore slopes are $-0.511\,[-0.621, -0.403]$ for S_{ode} and $-0.172\,[-0.297, -0.047]$ for S_{pref}; their paired difference is $+0.339\,[0.215, 0.462]$. HPSv2.1 gives slopes -0.02184 and -0.00463, with difference $+0.01721\,[0.01211, 0.02229]$. Both measured error directions lower the mean score relative to the reference endpoint.

We also compare the decoded images, using Learned Perceptual Image Patch Similarity (LPIPS) with AlexNet features and pixel root-mean-square error (RMSE). Relative to S_{ode}, S_{pref} is farther by $0.0544\,[0.0345, 0.0745]$ under LPIPS and by $0.0453\,[0.0400, 0.0508]$ under RMSE. AYS is farther by $0.0358\,[0.0161, 0.0563]$ and $0.0423\,[0.0373, 0.0477]$, respectively. Thus the endpoint-error ordering used in the main comparison agrees across Euclidean latent distance, decoded LPIPS, and pixel RMSE.

To check whether simple image properties account for the preference difference, we regress the paired PickScore difference (S_{pref} minus S_{ode}) on the paired differences of five image statistics computed on the held-out images: mean brightness, contrast, saturation, colorfulness, and sharpness. The regression gives $R^2 = 0.043$ and an adjusted gap of 0.282 $[0.098, 0.421]$.

A.3 Does increasing the number of evaluations remove the disagreement?

We repeat the main selection/evaluation protocol at 16 and 24 calls, keeping the model, guidance, solver, prompts, seeds, reference, and bootstrap procedure fixed. Both budgets use 24 schedules defined at arbitrary budgets: powers 0.4–2.5, uniform native time, log-SNR, and Karras ($\rho = 7$). We do not interpolate AYS schedules or re-optimize GITS for these budgets. At 16 calls, refining the reference from 750 to 1000 nodes changes its endpoint by 0.67%, or 1.65% of median candidate error; scheduler replay is exact.

Table 4: Objective conflict against the step budget (Stable Diffusion 1.5, DPM-Solver++, 128 held-out prompts, style-stratified bootstrap with the prompt as the inference unit). Error difference is endpoint search minus PickScore search (negative: closer to the reference). Both score differences are PickScore search minus endpoint search (positive: better images by that score). Brackets are 95% intervals.

Calls	Endpoint selection	Score selection	Error difference	PickScore difference	HPSv2.1 difference
10	power-0.5	power-1.6	−0.130	+0.333 [+0.222, +0.446]	+0.0172
16	power-0.7	power-2.5	−0.193	−0.020 [−0.128, +0.087]	+0.0002
24	power-0.5	power-1.5	−0.126	+0.199 [+0.115, +0.284]	+0.0083

At 16 calls, the PickScore winner lies at the edge of the candidate range, winning calibration by 0.0267 against a total spread of 0.1226. Its small margin makes selection sensitive to

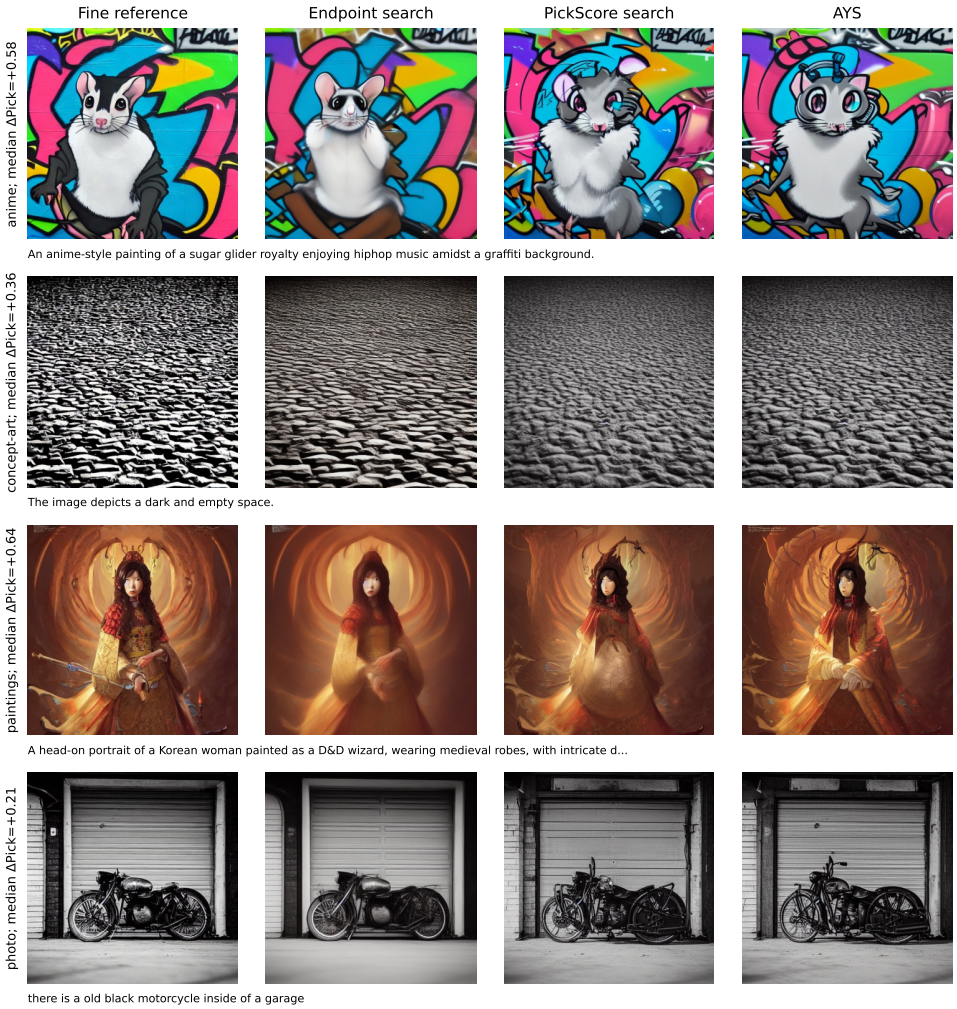

Figure 3: Paired examples from the original held-out image set. Within each HPSv2 style, the prompt is selected by the median-ranked original PickScore difference between the PickScore-selected and endpoint-selected schedules. Columns show the fine reference, endpoint-selected power-0.5, PickScore-selected power-1.6, and AYS, all with the same prompt and initial noise. Labels retain the original scoring run used to select these examples.

prompts. The endpoint-selected power-0.7 ends at 143, violating $B_{16} = 78$ despite no clear held-out PickScore loss. The ten-call plateau in Figure 1C does not establish a quality threshold at sixteen calls. Across candidates, endpoint error versus PickScore has Spearman $\rho = +0.321$ ($p = 0.015$) at ten calls, $+0.190$ ($p = 0.373$) at sixteen, and $+0.337$ ($p = 0.108$) at twenty-four.

ImageNet comparison. Endpoint error and FID need not disagree. For nine DiT-XL/2 DDIM schedules at six calls and guidance 1, median endpoint error over 16 images correlates with FID over 10,000 samples at $\rho = 0.933$; both select power-1.3. At guidance 1.5 they also select the same schedule ($\rho = 0.867$). At guidance 4, they select different schedules, separated by 1.06 FID, with $\rho = 0.10$. No repeated-sampling variance estimate is available for that FID contrast.

A.4 Reference convergence at high guidance

SD1.5 has 1000 distinct native indices, so $750 \to 1000$ is the last distinct refinement. On eight calibration prompts at guidance 7.5, median endpoint movement is 2.3% for $500 \to 750$ and 1.6% for $750 \to 1000$. The latter is 3.5% of median candidate error. Power-0.5 is closer on all eight prompts: mean errors are 0.473 versus 0.638 for power-1.6. SDXL's final refinement moves the endpoint by 2.8%, or 5.6% of median candidate error.

A 1000-node DPM-Solver++ 2M reference changes both the integrator and the output to zero noise. Its median distances from DDIM are 0.031 on calibration and 0.036 on evaluation (maxima 0.117, 0.460), versus median candidate error 0.58. All 57 calibration schedules retain their ranking ($\rho = 1.00$) and the same unconstrained and constrained selections. Power-0.5 is closer than power-1.6 on 119/128 prompts under either reference, by $0.130\,[0.112, 0.149]$ under DDIM and $0.129\,[0.111, 0.148]$ under DPM-Solver++.

A.5 Guidance sweep and the second model

The sweep in Table 5 tests guidance 1, 1.5, 2, 3 and 5, using the same guidance for each candidate and its reference. It follows the selection/evaluation protocol of §3 with a smaller 24-schedule candidate set: powers 0.4–2.5, uniform native index, log-SNR and Karras ($\rho = 7$), excluding the published ten-call AYS and GITS lists.

Guidance windows. The high-noise arm uses $w(t) = 7.5$ for $t > 500$ and 1 otherwise; the low-noise arm reverses these regions. All candidates and their 1000-node references use the same window. Table 5 reports selections and contrasts. In the low-noise arm, refining 750 to 1000 nodes moves the reference by 1.21% of its norm, failing the original 1% absolute threshold but meeting the amended relative criterion at 3.2% of median candidate error (Appendix A.4). We report this arm under that amendment.

Prompt alignment and additional quality measures. On the 128 held-out prompts of the main experiment, CLIP ViT-L/14 text–image similarity, which scores prompt alignment rather than preference, favors power-1.6 over power-0.5 by 0.0066 ($[0.0023, 0.0110]$), positive for every style. ImageReward v1.0 favors power-1.6 by 0.165 ($[0.054, 0.278]$), positive on 59% of prompts, with differences that vary by style (anime +0.23, concept art +0.13, paintings +0.36, photographs −0.05). On the 5000 COCO generations, the Fréchet distance on DINOv2-large's image-level features (the CLS token) is 367.4 for power-0.5 and 285.0 for power-1.6, in the same direction as FID.

The SDXL replication uses SDXL-base-1.0 at 1024×1024 with the same solver, budget, prompts and seeds, the same set of 24 candidates, and a reference over all 1000 native indices. The held-out endpoint-error contrast uses the first 32 prompts, the anime block in the style-ordered list, as specified before the run to limit reference computation. The preference contrast uses all 128 prompts and all four styles.

Table 5: Selection versus guidance and final-call restrictions. Upper/lower schedules minimize endpoint error/maximize PickScore; parentheses give the last one or two calls. Error difference is endpoint minus PickScore selection; score differences reverse that order. Brackets are 95% intervals. SD1.5, DPM-Solver++, ten calls, and 128 prompts unless stated; SDXL errors use the 32-prompt anime subset.

Setting	Selected schedules endpoint / score	Error difference	PickScore difference	HPSv2.1 difference
Varying guidance throughout generation				
	Karras (0)		+0.027	
1.0	p=1.9	−0.063	[−0.022, 0.078]	−0.0020
	Karras (0)		+0.068	
1.5	p=1.4	−0.030	[+0.017, 0.118]	+0.0011
	Karras (0)		+0.081	
2.0	p=1.4	−0.036	[+0.027, 0.134]	+0.0018
	Karras (0)		+0.084	
3.0	p=1.9	−0.079	[+0.012, 0.151]	+0.0010
	p=0.7 (199)		+0.255	
5.0	p=1.5	−0.068	[+0.164, 0.351]	+0.0093
	p=0.5 (316)		+0.333	+0.0172
7.5	p=1.6	−0.130	[+0.222, +0.446]	[+0.0127, +0.0219]
Guidance 7.5 in one region, 1 elsewhere				
	p=0.5 (316)		+0.296	
High noise only	p=0.9	−0.056	[+0.201, 0.394]	+0.0149
	p=0.9 (126)		+0.067	
Low noise only	p=2.3	−0.065	[−0.028, 0.161]	+0.0001
Restricting where the final call can occur (guidance 7.5 unless stated)				
	uniform/Karras (50)		+0.023	+0.0012
7.5, last ≤ 100	p=1.6	−0.064	[−0.078, +0.122]	[−0.0025, +0.0049]
	uniform (100)		+0.106	
5.0, last ≤ 100	p=1.5	−0.040	[+0.027, 0.186]	+0.0027
	p=0.8 (79)		+0.054	
24 calls, last ≤ 100	p=1.5	−0.105	[−0.010, 0.118]	−0.0005
	uniform (100)		−0.068	
SDXL, last ≤ 100	p=1.4	−0.046	[−0.150, 0.012]	−0.0065
	p=0.7 (233→24)		+0.073	
7.5, last fixed at 24	p=1.4	−0.077	[−0.021, 0.164]	+0.0008
	p=0.6 (285→24)		+0.245	
5.0, last fixed at 24	AYS	−0.061	[+0.163, 0.329]	+0.0045
	p=0.7 (233→24)		+0.120	
SDXL, last fixed at 24	p=1.0	−0.046	[+0.029, 0.210]	−0.0022

A.6 Separating the last evaluation from the spacing between evaluations

We separate final-call position from preceding spacing. Selections use the original calibration prompts and are frozen before held-out generation. Table 5 reports final-call controls; Tables 2 and 6 report the spacing bound.

Restricting the last call. The last-call filter retains 51 schedules, excluding powers 0.4–0.9. Endpoint search chooses $(999, 908, 807, 693, 561, 413, 272, 170, 103, 50)$; PickScore search chooses power-1.6. A cutoff of 50 selects the same pair. Table 5 gives the original 128-prompt comparison. On 128 additional prompts unused by earlier experiments, preference-minus-filter is $-0.031\,[-0.126, 0.066]$ PickScore; pooled over 256, it is $-0.004\,[-0.075, 0.067]$, and filter-minus-unconstrained is $+0.343\,[0.267, 0.421]$.

Fixing the last call. The 23 candidates are AYS and $t_i = \text{round}(24 + 975(1 - i/9)^p)$ for $i = 0, \ldots, 9$, $p = 0.4, 0.5, \ldots, 2.5$. Each ends at 24 and is recalibrated in each setting. Table 5 gives all selections and contrasts.

Table 6: Schedule identities for Table 2. Error difference is constrained minus PickScore-selected: negative means closer. Distances use 128 prompts on SD1.5, 32 anime prompts on SDXL, and 32 prompts on PixArt; scores use all 128. The bound is 125 at ten calls and 52 at twenty-four. Free-placement sets have 57 candidates (24 at twenty-four calls).

Setting	Constrained endpoint search (ours)	Max. gap	PickScore search	Error difference
SD1.5, 7.5	uniform–Karras 0.25 (last 74)	124	$p = 1.6$	-0.063
SD1.5, 24 calls	uniform (last 42)	42	$p = 1.5$	-0.078
SD1.5, UniPC	uniform (last 99)	100	AYS	-0.053
PixArt-Σ, 7.5	$p = 1.0$ (last 100)	100	uniform	-0.001
PixArt-Σ, 4.5	uniform–AYS 0.25 (last 80)	103	$p = 1.7$	-0.127
Every candidate's last call is fixed at index 24				
SD1.5, 7.5	uniform	109	$p = 1.4$	-0.049
SD1.5, 5.0	uniform	109	AYS	-0.033
SDXL	uniform	109	same schedule	—

The spacing bound. We fix the 125-index bound after inspecting calibration-only selections under 110, 120, and 150, before generating held-out images for this rule. Equation (2) counts both entry into low noise and the final output step. Each candidate set reuses its calibration prompts; only the selected pair is evaluated on held-out prompts (Table 2).

The DPM scheduler appends zero noise, assigned coordinate 0 for gap accounting. Karras and log-SNR call at index 0, so their final index gap is zero despite a remaining noise change; AYS ends at 24. Uniform conventions differ: Diffusers `anchor_uniform` uses $(999, 899, 799, 699, 599, 499, 399, 299, 199, 99)$, whereas `power_1.00` uses $(999, 899, 799, 699, 599, 500, 400, 300, 200, 100)$. Artifacts store both IDs and node arrays.

Recomputing eligibility for boundaries $b \in \{400, 500, 600\}$ with $B_{10} = 125$ admits 22, 20, and 15 of the 57 candidates, respectively. Endpoint search chooses the same uniform–Karras quarter-mixture $(999, 903, 803, 696, 580, 456, 336, 234, 151, 74)$ in all three cases.

At $\gamma = 1.10, 1.25, 1.50$, the 57-schedule search selects power-1.0, the uniform–Karras mixture ending at 74, and power-0.9, respectively. Their PickScore differences from power-1.6 are $+0.027$ $[-0.067, +0.121]$, $+0.014$ $[-0.078, +0.102]$, and -0.023 $[-0.120, +0.071]$ on the same 128 held-out prompts. All three schedules lie on the PickScore plateau in Fig. 1C.

For the scaled-linear noise schedule shared by SD1.5 and SDXL, the native index has the following noise-level conversion. The last-call threshold near $t = 160$ in Fig. 1C is $\sigma = 0.48$.

native t	500	316	160	125	100	74	25
σ_t	1.62	0.88	0.48	0.40	0.34	0.28	0.15
log-SNR	-0.96	0.26	1.47	1.83	2.13	2.52	3.73

For fixed-last-call rows, gains in Table 2 compare schedules within the same candidate set. For example, at guidance 7.5, the gain is $0.073 - (-0.024) = 0.097$. Gains of 0.366, 0.232, and 0.433 against the original free-placement selections instead combine fixing the last call with adding the bound.

Named schedules and search cost. Table 8 compares eleven schedules on the same 128 prompts. AYS exceeds the schedule from constrained endpoint search by $+0.092$ $[+0.004, +0.176]$ PickScore; the latter is 0.053 $[0.040, 0.066]$ closer to the reference. The filter-selected midpoint has gap 148 but no clear PickScore loss, showing that the spacing bound is conservative. Calibration costs $16(1000 + 57 \times 10) = 25{,}120$ guided model evaluations.

How often does each schedule score higher? Table 7 counts automatic-score wins on the Table 8 images. AYS's mean advantage over constrained search has an interval excluding zero, but its win-rate interval reaches 50%: mean differences measure magnitude, while wins count direction.

Table 7: Automatic PickScore wins on 128 paired prompts, conditional on fixed selections. There are no ties. Brackets give style-stratified 95% bootstrap intervals. Images and scores are those of Table 8, not the regenerated Table 1 images.

First schedule vs. second schedule	Wins / prompts	Win rate (%)
PickScore vs. endpoint search	91/128	71.1 [63.3, 78.9]
Constrained vs. endpoint search	94/128	73.4 [65.6, 80.5]
PickScore vs. constrained search	69/128	53.9 [45.3, 62.5]
AYS vs. constrained search	75/128	58.6 [50.0, 67.2]

Selecting a schedule at guidance 1 and deploying it at 7.5 halves the original PickScore loss but leaves 0.129 [0.021, 0.233]. Thus using weak guidance for selection does not fully remove the strong-guidance loss in this setting.

A bound-optimized schedule. Xue et al. (2024) minimize a summed per-step bound on numerical error that depends only on the noise schedule, requiring no model calls or calibration images. We run their released optimizer on SD1.5 with ten steps and uniform-time initialization, map times by round$(1000t - 1)$, and evaluate with our DPM-Solver++ zero-noise output. The released setting gives $(999, 871, 801, 753, 645, 504, 339, 183, 87, 25)$; its alternative for more than seven steps gives $(999, 936, 819, 711, 623, 515, 355, 197, 95, 28)$. Their entry gaps of 165 and 160 violate the bound without a clear PickScore loss (Table 8). The released setting gains 0.337 [0.231, 0.441] over power-0.5. Among listed schedules with gaps 148–165, only GITS has a loss whose interval excludes zero; all listed gaps of 196 or more do. The bound is therefore conservative.

Table 8: SD1.5 named and selected schedules: guidance 7.5, DPM-Solver++, ten calls, 128 prompts. Gap is the maximum low-noise step; bound marks eligibility at 125. The final column is power-1.6 minus the row in PickScore, with paired 95% intervals. Bound-optimized rows use the released code of Xue et al. (2024).

schedule	last call	gap	bound	distance	PickScore	HPSv2.1	p=1.6 minus schedule
power-0.5 (endpoint-selected)	316	316	✗	0.466	20.32	0.232	+0.333 [+0.222, +0.446]
uniform	99	100	✓	0.538	20.63	0.247	+0.024 [−0.069, +0.118]
uniform–Karras 0.25 (spacing bound)	74	124	✓	0.533	20.64	0.247	+0.014 [−0.078, +0.102]
uniform–Karras midpoint (last call ≤ 100)	50	148	✗	0.532	20.63	0.248	+0.023 [−0.078, +0.122]
bound-optimized, released setting (Xue et al., 2024)	25	165	✗	0.572	20.66	0.249	−0.004 [−0.104, +0.090]
bound-optimized, alternative setting	28	160	✗	0.554	20.62	0.247	+0.032 [−0.087, +0.147]
power-1.6 (PickScore-selected)	25	124	✓	0.597	20.65	0.249	0
AYS	24	112	✓	0.587	20.73	0.252	−0.077 [−0.159, +0.006]
GITS (published nodes)	17	149	✗	0.640	20.54	0.242	+0.107 [+0.006, +0.208]
Karras	0	196	✗	0.543	20.52	0.243	+0.128 [+0.021, +0.233]
log-SNR	0	208	✗	0.594	20.46	0.241	+0.187 [+0.098, +0.279]

Schedules with irregular spacing. To search beyond the named schedule families, we draw random proportions of the total interval from Dirichlet distributions, with concentration $\{0.3, 0.6, 1, 2, 5\}$ and map cumulative lengths to native indices. Adding 512 distinct schedules gives 569 candidates, of which 40 satisfy the spacing bound. The four lowest prompt IDs per style from the original evaluation set provide 16 selection prompts; the other 112 are held out only for this selection. Unconstrained endpoint search chooses power-0.5 and constrained endpoint search selects uniform. On the 112 prompts, the former is closer by 0.0731 [0.0554, 0.0915], while the latter gains 0.3437 [0.2417, 0.4490] PickScore and 0.01646 [0.01235, 0.02072] HPSv2.1.

Searching by the sum of step errors. Following the GITS cost construction, we use 100 candidate nodes (999 and every tenth index down to 9) and the 1000-node DDIM teacher trajectories of the 16 calibration prompts. Each edge costs the mean

relative L_2 distance from a one-step DDIM update at the teacher source state to the teacher target state. Dynamic programming, which builds the lowest-cost sequence from shorter sequences, minimizes the summed cost for ten model calls. With a free last call, it selects $(999, 859, 719, 619, 509, 409, 299, 189, 89, 19)$; fixing that node to 9 gives $(999, 859, 709, 599, 499, 379, 259, 149, 59, 9)$. Their maximum low-noise gaps are 110 and 120, so adding the spacing bound changes neither. On 128 prompts, PickScore-selected minus searched PickScore is $-0.028\,[-0.093, 0.036]$ and $-0.002\,[-0.054, 0.050]$, respectively; both searched schedules are closer to the reference by $0.010\,[0.005, 0.015]$ and $0.005\,[0.001, 0.010]$. This tests GITS's L_2 cost option, rather than its default trajectory-deviation cost.

Replication settings. Table 2 reports the spacing bound across settings. In the freely placed guidance-5 candidate set it selects the same uniform schedule as the last-call filter and retains the same $+0.106\,[0.027, 0.186]$ PickScore residual. Tables 5 and 2 distinguish these freely placed candidate sets from candidate sets with a fixed last call.

Last-call position and distributional metrics. Table 9 reports means on 5000 COCO captions. KID uses all 5000 features; its paired differences below average 100 paired subsamples of 1000, with central 95% intervals, so they need not equal differences of full-set estimates. The filter selects the uniform–Karras midpoint (FID 23.77, KID 0.00479); the bound selects its quarter-mixture. Their KID improves over power-0.5 but remains worse than power-1.6, whereas PickScore exceeds both:

Selection	Compared with	KID difference	PickScore difference
Last-call filter	power-0.5	$-0.00337\,[-0.00382, -0.00283]$	$+0.263\,[0.246, 0.279]$
	power-1.6	$+0.00066\,[0.00027, 0.00118]$	$+0.052\,[0.036, 0.069]$
Spacing bound	power-0.5	$-0.00324\,[-0.00368, -0.00282]$	$+0.283\,[0.267, 0.299]$
	power-1.6	$+0.00079\,[0.00042, 0.00122]$	$+0.072\,[0.057, 0.089]$

Table 9: Last-call position versus quality: SD1.5, guidance 7.5, ten calls, 128 prompts. Distances for the six new schedules use cached FP16 references. The paired PickScore contrast is power-1.6 minus the row. FID/KID use 5000 COCO images; on those images schedules ending at 50–126 gain 0.05–0.08 PickScore over power-1.6. NR means not measured.

schedule	last call	nodes ≤ 500	distance	PickScore	p=1.6 minus schedule	FID / KID
power-0.5	316	2	0.466	20.32	$+0.333\,[+0.222, +0.446]$	32.15 / 0.0082
power-0.6	251	3	0.478	20.52	$+0.133\,[+0.022, +0.243]$	NR
power-0.7	199	3	0.487	20.57	$+0.082\,[-0.010, +0.174]$	27.77 / 0.0065
power-0.8	158	4	0.504	20.64	$+0.012\,[-0.081, +0.105]$	NR
power-0.9	126	4	0.522	20.67	$-0.023\,[-0.120, +0.071]$	25.52 / 0.0056
power-1.0	100	5	0.538	20.63	$+0.027\,[-0.067, +0.121]$	24.82 / 0.0052
uniform–Karras 0.25 (spacing bound)	74	5	0.533	20.64	$+0.014\,[-0.078, +0.102]$	24.27 / 0.0049
uniform–Karras midpoint (last ≤ 100)	50	5	0.532	20.63	$+0.023\,[-0.078, +0.122]$	23.77 / 0.0048
power-1.3	50	5	0.569	20.66	$-0.009\,[-0.099, +0.078]$	NR
power-1.6	25	6	0.597	20.65	0	22.23 / 0.0041

Last-call comparisons across settings. We evaluate the same eight power schedules on 128 prompts. Power-1.6 minus power-0.5 gives PickScore $+0.361\,[0.259, 0.467]$ at guidance 1, $+0.402\,[0.277, 0.530]$ at guidance 5, $+0.179\,[0.045, 0.313]$ with UniPC, and $+0.275\,[0.162, 0.390]$ with DDIM (both at guidance 7.5). The bend in PickScore remains near last call 100–160. On SDXL it peaks at last call 100–126; power-1.6 is below power-1.0 by $0.142\,[0.052, 0.235]$. Guidance-1/5 sweeps here are quality-only; UniPC and DDIM endpoint errors use cached references.

Refining selected intervals. Power-0.5, the uniform–Karras quarter-mixture, and power-1.6 initially use 16 held-out prompts, four per style. High-noise intervals end above 500; the crossing interval belongs to low noise. The baseline is unmodified second-order DPM-Solver++. High-noise refinement substitutes native-step DDIM, evaluating DPM predictions at reference states to advance its history before resuming coarse updates. Low-noise refinement retains early coarse steps and runs native-step DDIM from the crossing interval's starting state to the output. Reference integration uses DDIM throughout. Each interval

Table 10: Refining high- or low-noise intervals at guidance 7.5: 64 prompts for power-0.5 and 16 each for the two covered schedules. Error is relative L_2; e_{full} and e_{repair} are endpoint-error vectors relative to the fine reference. c is their cosine; $\Delta_v = \|e_{\text{repair}} - e_{\text{full}}\|/\|e_{\text{full}}\|$. Score changes are refinement minus the original schedule. Intervals crossing 500 belong to the low block.

Schedule	Arm	Error	c	Δ_v	Δ PickScore [95% CI]
power-0.5	full	0.466	1.00	0.00	0
	refine high	0.259	0.47	0.86	+0.144 [+0.039, +0.252]
	refine low	0.407	0.75	0.63	+0.375 [+0.266, +0.483]
quarter-mixture	full	0.552	1.00	0.00	0
	refine high	0.167	0.13	1.01	+0.145 [−0.172, 0.481]
	refine low	0.547	0.91	0.41	+0.070 [−0.085, 0.207]
power-1.6	full	0.619	1.00	0.00	0
	refine high	0.134	0.04	1.02	+0.098 [−0.283, 0.462]
	refine low	0.616	0.92	0.37	+0.007 [−0.126, 0.153]

appears once; retained coarse updates never restart at first order. Trajectory changes prevent interpreting these measurements as additive shares of the original error.

Power-0.5 is repeated on 48 additional, previously studied prompts, totaling the first 16 per style (64). All 64 images are scored together from lossless PNGs; the other schedules retain their original scoring. Low-minus-high PickScore gain is 0.196 [0.041, 0.354] on the added prompts and 0.231 [0.087, 0.372] pooled. Table 10 gives individual gains. The pooled high/low HPSv2.1 gains are 0.0047 [0.0006, 0.0089] and 0.0186 [0.0149, 0.0225], with paired difference 0.0140 [0.0092, 0.0186].

In the original 16-prompt run, the paired low-minus-high PickScore difference was +0.301 [0.028, 0.587] at guidance 7.5 and +0.303 [0.134, 0.488] at guidance 1. For the quarter-mixture and power-1.6 it is −0.075 [−0.410, 0.243] and −0.091 [−0.466, 0.282] at guidance 7.5. At guidance 1, the power-0.5 schedule's full, high-noise-refined, and low-noise-refined errors are 0.335 [0.318, 0.353], 0.316 [0.301, 0.332], and 0.102 [0.074, 0.132]. Cosines to the full error are 0.96 and 0.26 for high- and low-noise refinement; their PickScore gains are 0.010 [−0.054, 0.072] and 0.313 [0.135, 0.511]. Guidance therefore reverses the relative endpoint-norm sensitivity while low-noise refinement retains the quality gain.

An earlier SD3.5-Medium pilot (1024^2, guidance 4.5, 100 GenEval prompts (Ghosh et al., 2023)) compared six ten-step schedules with a 50-step reference: mean error fell from 0.540 at power-2.0 to 0.432 at power-0.5, without quality evaluation. Appendix A.9 gives the full selection test.

A.7 Expanding the candidate set and redistributing evaluations

This experiment adds one candidate to the endpoint search, then tests whether rearranging its evaluations improves PickScore at the same cost. To obtain the candidate, we adapt Optimal Stepsize for Diffusion Sampling (OSS), the schedule-search method of Pei et al. (2025), to SD1.5 using its official dynamic-programming implementation. Set $c_t = \sqrt{\bar{\alpha}_t} + \sqrt{1 - \bar{\alpha}_t}$. The coordinates $y = x/c_t$ and $s = \sqrt{1 - \bar{\alpha}_t}/c_t$ give velocity $v = \epsilon_\theta - \hat{x}_0$, where $\hat{x}_0$ is the model's clean-state prediction. An Euler update in these coordinates equals a DDIM jump. The teacher uses 200 native-index evaluations and a zero-noise output. We use the official coordinatewise median, taking the median time at each schedule position across the original 16 calibration prompts, without amplitude renormalization. The selected calls are [994, 944, 894, 843, 788, 713, 572, 402, 221, 45]. With this update on the original 128 evaluation prompts, its PickScore difference from power-1.6 is −0.030 [−0.124, 0.063], and from AYS is −0.070 [−0.162, 0.023]. Replacing the first call by 999 changes PickScore by 0.012 [−0.030, 0.058]. These comparisons use a common lossless-image scoring run and concern the SD1.5 adaptation.

We then replace the first call by 999 and transfer these times to DPM-Solver++. Regenerating all 58 candidates on the original 16 calibration prompts with 1000-step DDIM references

reproduces the original 57-candidate selections; the added schedule changes only the endpoint winner. Selections are frozen before evaluation on the previously studied 128 prompts. Outputs are regenerated one prompt at a time and scored together from quality-95 JPEGs with the fixed processor. The original 0.333 PickScore gap becomes 0.350, and power-0.5 scores 20.302 rather than 20.32. Both runs use JPEGs but remain separate. All schedules share cached references; reconstructing old distances from FP16 caches changes them by at most 1.8×10^{-5}.

In Table 1's rerun, adding the schedule reduces the PickScore-selection advantage from $0.350\,[0.233, 0.465]$ to $0.132\,[0.022, 0.238]$. The new endpoint winner is $0.141\,[0.122, 0.161]$ closer to the reference. It passes the last-call filter but violates the gap bound; constrained search scores $0.126\,[0.041, 0.209]$ higher. Thus the DPM-Solver++ conflict extends beyond power-0.5, although the adaptation's own-update contrast above includes zero. Intervals here are paired 95% intervals.

Redistributing evaluations without changing their number. The transferred schedule has three excessive gaps: $572 \to 402$, $402 \to 221$, and $221 \to 45$. We insert their rounded midpoints 487, 312, and 133, then delete three calls from $\{944, 894, 843, 788, 713\}$. All ten combinations retain the first call 999, last call 45, and ten-call budget, with maximum low-noise gap 91. We choose the smallest mean endpoint error on the original 16 calibration prompts, using their cached references and no quality scores. The selected schedule is $[999, 843, 713, 572, 487, 402, 312, 221, 133, 45]$.

On the original 128 prompts, the rearrangement gains $0.143\,[0.044, 0.240]$ PickScore and increases endpoint error by $0.142\,[0.124, 0.162]$. Its PickScore differences from power-1.6, constrained search, and AYS are $0.011\,[-0.044, 0.066]$, $0.017\,[-0.064, 0.098]$, and $-0.045\,[-0.112, 0.020]$. Images are scored together with baselines. This test uses previously studied prompts and one predefined rearrangement family; it establishes a fixed-cost improvement, not a necessary or optimal gap threshold.

Replication on a second prompt set. Without re-selection, we evaluate the same schedules on the second 128-prompt cohort, previously used for the spacing check but not this pair. Cached 1000-step references and common scoring are reused. Compared with power-0.5, the transferred schedule is $0.013\,[0.004, 0.022]$ closer and $0.284\,[0.208, 0.362]$ better in PickScore. Its deficit to PickScore selection is $0.061\,[-0.045, 0.167]$ on this cohort, versus $0.097\,[0.020, 0.173]$ over all 256 prompts. Rearrangement gains $0.104\,[0.002, 0.207]$ on the second cohort and $0.124\,[0.051, 0.197]$ pooled, with $0.160\,[0.140, 0.179]$ more endpoint error; its HPSv2.1 change is $-0.0002\,[-0.0039, 0.0035]$. Constrained search exceeds the transferred schedule by $0.102\,[0.004, 0.203]$ on the second cohort and $0.114\,[0.048, 0.180]$ pooled.

Adding the constraint within OSS search on PixArt. On PixArt-Σ at guidance 4.5, search and generation both use the OSS adaptation's DDIM-equivalent update: 200 teacher evaluations, ten image-generation calls, FP32, and the weights and embeddings of Appendix A.10. The constrained arm rejects steps entering or within $t \leq 500$ with gaps above 125, including the output step. The remaining search, including the coordinatewise median over 16 calibration prompts, is unchanged. Each arm costs 35,200 guided evaluations; schedules are frozen before testing.

The evaluation reuses the prior cohort, excluding one exact-text calibration duplicate with a different index and seed. The remaining 127 prompts are text-disjoint from calibration; all four schedules are scored together from PNGs. Unconstrained search already has gap 96 and last call 90. The bound changes one early call ($778 \to 773$), with no distinguishable preference gain (Table 11). Its advantage over uniform therefore does not establish a constraint benefit. On 32 prompts with screened FP32 RK4 references, endpoint error increases by $0.0078\,[0.0030, 0.0137]$.

A.8 SD1.5 propagation and fixed-budget allocation controls

These exploratory tests reuse 16 previously studied prompts, four per style, with their original seeds. They measure finite perturbation responses and fixed-budget node exchanges.

Table 11: PixArt OSS adaptation, same first-order update in search and sampling. Differences are constrained minus each comparison on 127 paired prompts, with pointwise style-stratified 95% bootstrap intervals. Unconstrained OSS is the primary comparison; uniform and transferred SD AYS are secondary.

Comparison	PickScore difference	HPSv2.1 difference
Unconstrained OSS	$+0.002\,[-0.021, 0.028]$	$+0.0003\,[-0.0003, 0.0009]$
Uniform	$+0.099\,[0.042, 0.156]$	$+0.0044\,[0.0025, 0.0062]$
Transferred SD AYS	$+0.028\,[-0.039, 0.092]$	$+0.0046\,[0.0028, 0.0065]$

Matched perturbations. On the power-0.5 schedule we retain the state y and history h after updates ending at 948, 774, 547, and 316. We form trajectories under guidance 1 and 7.5. For each source trajectory, we continue with either guidance strength, holding y and h fixed. Let $F_w(y, h)$ denote that continuation's endpoint. Two unit directions u come from the earlier weak- and strong-guidance interval corrections. We measure

$$G_\eta = \frac{\|F_w(y + \eta u, h) - F_w(y - \eta u, h)\|}{2\eta}, \qquad \eta \in \{0.01, 0.1\}.$$

The denoising neural network (U-Net) uses 32-bit floating-point arithmetic (FP32), with its original 16-bit (FP16) weights promoted. The reduced-precision TensorFloat-32 mode (TF32) is disabled. Unperturbed replay reproduces the source endpoint when source and continuation guidance agree. Mixed-guidance continuations deliberately change the field after the intervention. Means first average sites and directions within prompt; intervals resample prompts within style. On strong-guidance source states, mean high-noise $G_{0.1}$ rises from 15.4 to 76.3 when continuation guidance changes from 1 to 7.5. At $\eta = 0.01$ it rises from 20.4 to 208.6. For $\eta = 0.1$, the paired high-noise increase is $60.9\,[43.4, 81.7]$. At the late site, the two means are 1.13 and 2.32, with a paired increase of $1.19\,[0.67, 1.77]$. Reducing η to 0.01 leaves these late-site means at 1.14 and 2.33. On weak-guidance source states the corresponding high-noise $G_{0.1}$ values are 19.2 and 112.6. The direction of the guidance effect persists, but high-noise slopes are amplitude-dependent. These are finite directional responses, not spectral norms or converged derivatives.

Moving a call at fixed budget. From power-0.5, each of seven interior nodes above 500 is exchanged for each of 224, 100, and 50, giving 21 ten-call schedules. Their maximum low-noise gaps are 224, 216, and 266, respectively: even exchanges passing the last-call filter violate 125. This geometric distinction does not establish that the bound is necessary for a quality gain. Deletion-only and insertion-only controls use nine and eleven calls. Each arm reruns DPM-Solver++ with its evolving history; references use 1000 DPM-Solver++ nodes with matching zero-noise output, and images are scored together from PNGs. Averaging exchanges within prompt, endpoint error falls by $0.121\,[0.104, 0.136]$ at guidance 1 but rises by $0.047\,[0.024, 0.070]$ at 7.5. PickScore gains are $0.250\,[0.141, 0.361]$ and $0.040\,[-0.092, 0.196]$. The numerical allocation incentive reverses, but the strong-guidance score gain remains uncertain.

A.9 A flow-matching model: SD3.5-Medium

SD3.5-Medium (Esser et al., 2024) uses flow matching. At 1024×1024, we fix model revision b940f670, all three text encoders, ten Euler evaluations, paired initial noise and zero-noise output. Transformer evaluations use FP16; state updates and guidance use FP32. The 24 schedules comprise powers 0.4–2.5 in input noise level, cosine-squared spacing and default spacing on the model's shifted flow schedule. Sixteen calibration prompts, four per style, select by mean distance to an 800-call Heun reference or by mean PickScore. Evaluation uses the first sixteen prompts per style from the original SD1.5 set (64 total), held out from this selection but previously studied on SD1.5. The SD1.5 spacing threshold is not applied.

Reference checks. On four calibration prompts (one per style), the initial screen requires median movement after doubling Heun calls to be at most 1% of endpoint norm and 5% of

median candidate error. Guidance 4.5 passes the 400/800/1600-call check. At 7.5, movement remains 6.1% for 800 → 1600 and 6.7% for 1600 → 3200; two prompts are near 1% and two at 13–16%. Rankings remain identical ($\rho = 1.00$, winner power-0.4). We amended the criterion on this evidence before generating the remaining comparisons; pointwise reference convergence remains unestablished.

A later single-GPU check gives median 800 → 1600 movement of 1.90% in FP16 and 2.20% in FP32. The hardest prompt's successive FP32 changes fall from 5.34% to 2.78% at 3200 calls. All 24 locally rerun schedules still select power-0.4 under both precisions and reference resolutions. Three schedules reproduce exactly on twelve originally local prompts, but differ on four A100-generated prompts for an unknown reason; this check therefore pairs local candidates with local references.

Selection and a change in evaluation time. At guidance 4.5, both criteria select cosine-squared spacing: 0.163 [0.028, 0.303] higher PickScore and 0.093 smaller reference distance than default spacing. At 7.5, the selected schedules differ (Table 12). Power-0.4 is 0.063 [0.046, 0.080] closer but scores 0.220 [0.079, 0.369] lower than power-0.7. The latter wins PickScore on 59% [48%, 70%] of prompts; the HPSv2.1 interval includes zero (+0.003 [−0.003, 0.008]).

Table 12: SD3.5-Medium at guidance 7.5, ten calls, 64 held-out prompts. Endpoint error is the relative distance to the 800-call Heun reference. Contrasts are paired, with style-stratified 95% bootstrap intervals.

schedule	selected by	last call σ	endpoint error ↓	PickScore ↑	HPSv2.1 ↑
power-0.4	endpoint error	0.665	0.633	21.96	0.288
power-0.7	PickScore	0.428	0.696	22.18	0.291
model default	—	0.009	0.742	21.78	0.270

Two initial three-call reallocations enlarge the first step about fivefold and reduce PickScore by 0.775 and 0.594. A follow-up preserves the first three calls and tests replacing one of six interior calls with shifted $\sigma \in \{0.55, 0.40, 0.25\}$. PickScore on the original 16 calibration prompts selects 0.829 → 0.400 from these 18 variants and unchanged power-0.4. The frozen choice is tested on 64 new prompts, 16 per style, excluding recorded SD3.5 and SD1.5 cohorts. Schedules share one noise sample per prompt and PNG scoring.

The change gains 0.228 [0.174, 0.287] PickScore (57/64 wins) and 0.0068 [0.0054, 0.0084] HPSv2.1 over power-0.4. Its PickScore gains over power-0.7 and default spacing are 0.334 [0.157, 0.523] and 0.795 [0.520, 1.089]. The original pair reverses on this cohort (21.87 for power-0.7, 21.97 for power-0.4). The change also reduces calibration reference distance by 0.0115; it establishes no accuracy–quality tradeoff because reference convergence and new held-out endpoint errors remain unverified.

A.10 A text-conditioned diffusion transformer: PixArt-Σ

PixArt-Σ (Chen et al., 2024b) combines a DiT-XL/2 transformer, T5-XXL text encoder, and 1000-step linear-beta noise schedule (Peebles & Xie, 2023). We use its 512^2 checkpoint, ten-call DPM-Solver++ 2M, and the SD1.5 16/128 prompt split and initial noise. The 57-schedule construction is unchanged except that log-SNR and Karras anchors use this noise schedule; 34 node lists match SD1.5. AYS transfers SD times without PixArt optimization. The spacing bound remains 125.

FP32 protocol and reference checks. Both guided settings use FP32 for transformer, guidance, states, and decoding, with TF32 disabled. Original FP16-rounded weights and cached embeddings are cast to FP32, preserving the model. The reference interpolates log signal power between native indices and uses fourth-order Runge–Kutta (RK4) at 1000 and 2000 nodes. Including the same final denoising operation as the candidates costs 3997 and 7997 calls. All candidates are evaluated on 16 calibration prompts before freezing selections,

Table 13: FP32 PixArt-Σ, ten calls. Endpoint errors use 2000-node RK4 references on 32 preselected prompts; both scores use 128. Rows give the fixed calibration selections and baselines. AYS transfers SD times without PixArt-specific optimization.

Schedule	Endpoint error ↓	PickScore ↑	HPSv2.1 ↑
Guidance 7.5			
Endpoint search (power-0.7)	0.468	21.64	0.288
PickScore search (uniform)	0.541	22.04	0.302
Constrained search (power-1.0)	0.540	22.03	0.302
Transferred AYS	0.651	21.83	0.296
Guidance 4.5			
Endpoint search (power-0.8)	0.328	22.12	0.298
PickScore search (power-1.7)	0.452	22.60	0.313
Constrained search (uniform–AYS 0.25)	0.324	22.52	0.311
Uniform	0.322	22.43	0.308
Transferred AYS	0.343	22.42	0.310

which agree at both resolutions. Eight evaluation prompts per style (32 total) are chosen before reference generation.

At guidance 7.5, all 16 calibration prompts pass both checks: reference movement below 1% of endpoint norm and 10% of median candidate distance. On evaluation, 31/32 pass, with median movement 0.034%. A follow-up chosen after the check refines the remaining prompt from 2000 to 4000 nodes, reducing its movement from 1.24% to 0.20% and passing the unchanged criteria. All prompts remain included. Substituting this finer reference changes the mean distance contrast by less than 10^{-6} and preserves all four rankings. The separately pre-registered guidance-4.5 run passes on all $16 + 32$ prompts, with evaluation median movement 0.011% and maximum 0.53%.

Accuracy and image scores. Table 13 gives the FP32 selections. At 7.5, uniform beats endpoint-selected power-0.7 by $0.397\,[0.285, 0.505]$ PickScore (96/128 wins) and $0.0134\,[0.0103, 0.0163]$ HPSv2.1. Constrained search gains $0.391\,[0.282, 0.498]$ PickScore, leaving $0.006\,[-0.011, 0.023]$ to uniform. On the 32 reference prompts, uniform is $0.073\,[0.030, 0.119]$ farther yet gains $0.476\,[0.259, 0.707]$ PickScore. The distance contrast changes by less than 10^{-5} between reference resolutions.

At default guidance 4.5, power-1.7 beats power-0.8 by $0.486\,[0.401, 0.571]$ PickScore (109/128 wins) and $0.0151\,[0.0127, 0.0176]$ HPSv2.1. The constrained uniform–AYS quarter-mixture gains $0.402\,[0.346, 0.460]$ PickScore and $0.0132\,[0.0113, 0.0152]$ HPSv2.1 over power-0.8, leaving a $0.084\,[0.016, 0.153]$ PickScore residual. The score winner is excluded by a $545 \to 419$ gap of 126, just above 125, despite its late last call at 20. Constrained search exceeds uniform by $0.089\,[0.056, 0.123]$ PickScore and transferred AYS by $0.096\,[0.029, 0.168]$. Its endpoint-error difference from unconstrained search is $-0.004\,[-0.018, 0.010]$ on the 32 reference prompts, while power-1.7 is $0.123\,[0.091, 0.157]$ farther. These prompts were previously studied but held out from calibration.

Reference sensitivity in the FP16 sweep. On two calibration prompts, refining DDIM from 750 to 1000 steps changes guided endpoints by 3.7–16.3%, versus less than 1% without guidance. Before calibration images were scored, the protocol was amended to require agreement between DDIM and Heun references. They select the same schedules, although median endpoint disagreement is 0.050 at guidance 7.5 and 0.029 at 4.5. The FP32 validations above resolve the precision check at both guided settings; the FP16 distances below are retained as the initial sweep, and the unguided arm passed the original check in FP16. The near-black AYS image for "a dark and empty space" is retained in scoring.

The earlier FP16 guidance sweep. The transformer and guidance use FP16, with FP32 state updates and decoding. Table 14 gives unguided controls; Figure 4 shows the full sweep's paired PickScore differences. At guidance 7.5, power-0.7 wins 87% of calibration

bootstrap draws; uniform wins PickScore on 100/128 evaluation prompts. At 4.5, power-0.8 wins 44% of draws and the bounded choice 28%. Both guided settings retain the same selections under FP32. At default guidance, all reported PickScore contrasts change by less than 0.03. The FP16 HPSv2.1 advantages of PickScore selection are 0.0128 [0.0094, 0.0159] at 7.5 and 0.0147 [0.0122, 0.0173] at 4.5.

Without guidance (1.0), endpoint search already satisfies the bound and gives the highest mean evaluation PickScore. It exceeds calibration PickScore selection by 0.079 [0.023, 0.136] PickScore and 0.0103 [0.0085, 0.0122] HPSv2.1, while lying 0.024 closer to the reference. Its PickScore gains over AYS and uniform are 0.118 [0.075, 0.165] and 0.144 [0.107, 0.179]. The guided failure is absent in this arm.

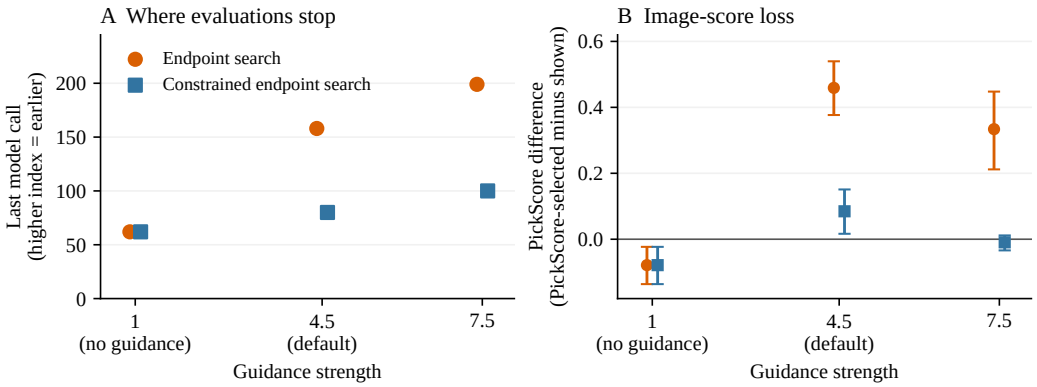

Figure 4: **Guidance changes the selected allocation on PixArt-Σ.** Left: the last model call selected by endpoint distance, with and without the spacing bound. Right: mean PickScore of the PickScore-selected schedule minus that of each shown schedule, on 128 paired prompts; positive values indicate a loss. Bars are 95% bootstrap intervals. This is the initial FP16 guidance sweep. The two reference solvers agree on these selections; Table 13 reports the subsequent FP32 validation at guidance 7.5 and 4.5.

Table 14: Unguided PixArt-Σ controls from the initial FP16 sweep: ten calls, 128 prompts, relative distance d to the 1000-node DDIM reference. Heun distances agree within 0.003. The spacing bound is 125. Guided results are shown in Figure 4, with screened FP32 validation in Table 13.

guidance	schedule (selected by)	last call	low gap	$d \downarrow$	PickScore ↑	HPSv2.1 ↑
1.0	unif.–AYS 0.50 (endpoint error, inside bound)	62	106	0.180	21.35	0.257
1.0	unif.–Karras 0.50 (calibration PickScore)	50	155	0.204	21.27	0.247
1.0	uniform	99	100	0.197	21.20	0.250
1.0	AYS	24	112	0.186	21.23	0.258

A.11 Human preference evaluation: setup and analysis

We compared human preferences among three schedules using SDXL with DPM-Solver++ 2M, ten model evaluations, and classifier-free guidance scale 7.5. Each comparison used 100 prompts randomly sampled from the Pick-a-Pic v2 test set (Kirstain et al., 2023). None of these prompts appeared in the calibration set used to select the schedules.

Compared schedules. Uniform spacing is the baseline. Constrained endpoint search selects a schedule by minimizing endpoint error subject to the low-noise spacing constraint. The transferred OSS schedule uses evaluation times found with OSS's first-order, DDIM-equivalent error objective, then applies those times with DPM-Solver++ 2M. This comparison evaluates the transferred times, not OSS with its original update. The constrained schedule is selected by search rather than by manually moving three evaluations, as in the separate redistribution experiment.

Uniform spacing fixes equal intervals throughout generation. The constraint only caps low-noise intervals, permitting but not requiring a non-uniform schedule.

Rating procedure. Raters recruited through Prolific compared images in a blind A/B test. Method labels were hidden, and image positions were randomized. Each pair was assessed by three to five different raters for visual quality, text alignment, and overall preference, with A, Tie, and B as response options.

Aggregation and analysis. We aggregate ratings into one A/Tie/B outcome per image pair. A or B is assigned when it receives the unique largest number of votes; otherwise, the outcome is Tie. Each comparison includes 100 prompt pairs. Win rates and hypothesis tests exclude ties, leaving 82, 78, and 81 pairs for the three overall-preference comparisons, respectively. We use two-sided exact binomial tests against a win probability of 0.5 and apply Holm correction across these three comparisons. Visual quality and text alignment are reported descriptively.

Table 15: Overall human preference on 100 prompt pairs per comparison. Counts give one outcome per pair. Win rates exclude ties, with the numerator and denominator shown explicitly. The final column applies Holm correction across the three tests. Images use SDXL with DPM-Solver++ 2M, ten model calls, and guidance 7.5.

Comparison (A vs. B)	A / Tie / B	Win rate excluding ties	Raw p	Holm p
Constrained vs. transferred OSS	67/18/15	81.71% (67/82)	5.26×10^{-9}	1.58×10^{-8}
Constrained vs. uniform	57/22/21	73.08% (57/78)	5.57×10^{-5}	1.11×10^{-4}
Transferred OSS vs. uniform	31/19/50	38.27% (31/81)	0.0448	0.0448

Raters preferred constrained endpoint search to both alternatives, and uniform spacing to the transferred OSS schedule (Table 15). These judgments compare image quality; they do not by themselves establish the endpoint-error ordering on the human-study prompts. For these three comparisons, visual-quality counts (A/Tie/B) were 68/17/15, 58/21/21, and 31/19/50, respectively. Text-alignment counts were 53/32/15, 51/34/15, and 42/36/22. The transferred OSS schedule therefore received more favorable judgments than uniform spacing on text alignment, but fewer on visual quality and overall preference.

A.12 A stochastic sampler

A stochastic sampler adds fresh noise during generation. Comparing individual endpoints across step schedules would require coupling the Brownian paths that supply this noise. Here we compare output distributions instead: fidelity is the kernel Inception distance to the same sampler run at 200 steps.

The sampler is `sde-dpmsolver++` on Stable Diffusion 1.5 at guidance 7.5 and ten evaluations, with the noise generator seeded per image so that schedules are paired. Prompts are the first 1000 COCO captions used elsewhere. Across the nine power and log-SNR schedules, distributional fidelity has Spearman correlation -0.633 with mean PickScore ($p = 0.067$). Smaller distance to the converged distribution accompanies higher PickScore. FID against the same 5000 real images is also lowest at p=1.3, the most faithful schedule, and highest at the two extremes. With nine schedules, the correlation interval includes zero.

A.13 When sums of local errors distinguish schedules

Summing step errors can preserve low-noise evaluations. Our GITS-style search uses $M = \sum_{k=1}^{K} \|\tau_k\|$, where τ_k is a coarse update's error starting from the fine reference state and K counts solver steps, each of which may require multiple calls. It avoids the large gaps and quality loss (Appendix A.6), but propagation and cancellation can distort its endpoint-error ranking. The tests below measure this ranking on the DiT image-generation transformer. Appendix B gives the propagation analysis for solvers without stored prediction history.

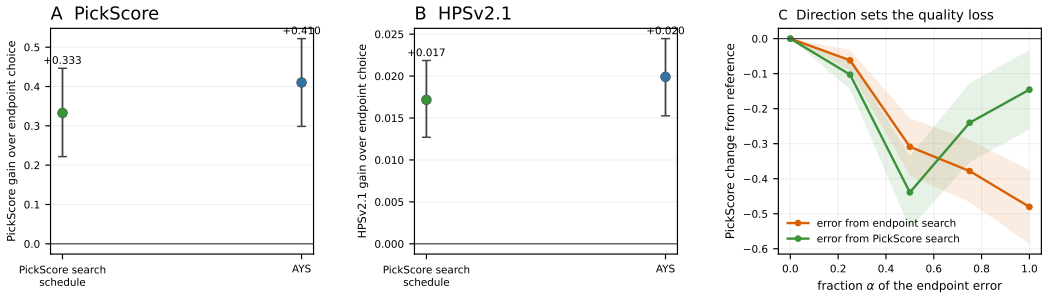

Figure 5: Additional views of the main comparison. A–B: paired preference-score gains over the endpoint-selected schedule with 95% bootstrap intervals. C: decoding points between each coarse endpoint and the fine reference changes the quality ordering of their error directions. Shading is 1.96 standard errors of the prompt means.

On a separate sixteen-schedule DiT test, M ranks widely separated candidates well: median within-image correlations with endpoint error are 0.951, 0.944, and 0.926 at four, eight, and twelve calls (10 images). We next ask whether M can distinguish schedules with similar endpoint errors, and what happens when the search minimizes M directly. We use DiT-XL/2, deterministic DDIM, guidance 1.0, a 250-interval reference schedule augmented with the coarse boundary nodes, and NFE in $\{4, 6, 8, 12\}$. Both searches choose timesteps from the same 126-node set, obtained by taking every second node of the fine schedule. Eight images (500–507) are used for calibration and twelve (508–519) are held out. No quantity is fitted on the held-out images.

The map $\Psi^{\text{cheap}}_{i\to j}$ takes one coarse step between candidate times i, j. The edge cost $c(i,j) = \|\Psi^{\text{cheap}}_{i\to j}(x^{\text{ref}}_i) - x^{\text{ref}}_j\|$ is averaged over calibration images. A K-step path through this graph costs exactly M, so its shortest path S_{DP} is the exact $\arg\min_S M(S)$. For comparison, S_{SEARCH} minimizes mean relative endpoint discrepancy over the same nodes. It uses coordinate descent from three starts: the best parametric schedule and two fixed-seed random restarts. Each coordinate takes its best improving move; search stops after a sweep with no improvement.

Each budget has 61–65 candidates: 14 power schedules, log-SNR, four Karras schedules, 36 Dirichlet placements, the two search outputs, and up to eight search-trace schedules. Before evaluation, the eight lowest-calibration-error schedules define the competitive subset. "Resolution" is the smallest endpoint-error spread (largest divided by smallest calibration-mean error, minus one) at which the correlation interval excludes zero and continues to do so for all wider candidate sets. Table 16 reports ranking and the held-out cost of minimizing M.

Table 16: Sum of local-error norms M on the searched schedules (DiT-XL/2, DDIM, 12 held-out images; image-clustered 95% CIs). "Competitive ρ" ranks the eight schedules with the lowest calibration error. "min-M vs random" is the paired regret advantage over uniform random selection, so positive favors M. "Limit" is the candidate-set spread from which the ordering remains resolved. "$\arg\min M$ cost" compares the exact M-minimizer with direct endpoint search on held-out images.

	NFE 4	NFE 6	NFE 8	NFE 12
competitive ρ	0.55 [−0.12, 0.94]	0.23 [−0.24, 0.62]	0.15 [−0.61, 0.43]	0.51 [−0.05, 0.69]
min-M vs random	+0.012 [−0.005, 0.028]	−0.0005 [−0.0082, 0.0060]	−0.006 [−0.010, −0.001]	+0.004 [−0.002, 0.009]
limit (spread)	0.09	0.20	0.19	0.20
$\arg\min M$ cost	+0.2% [−4.2, 5.1]	+5.5% [−0.7, 12.7]	+5.4% [2.7, 8.2]	+16.2% [4.1, 28.8]

At NFE 8, the minimum-M schedule in the competitive set has higher regret than a random choice. At NFE 12, the best power schedule has held-out discrepancy 0.259, slightly below

S_{SEARCH} at 0.263; S_{DP} reaches 0.300. The small reversal between the power schedule and direct search is consistent with mild overfitting to the eight calibration images.

Per-step amplification. The measured amplification of one local error is $g_k = \|s_k\|/\|\tau_k\|$, with s_k the measured single-error response of Appendix B.3. We use DiT-XL/2 with DDIM at NFE 6 (10 images, 160 image–schedule pairs). Taking medians over schedules within each image and then over images gives g_k values of 11.8, 16.2, 12.3, 2.5, 1.1, and 1.0 from the first to the last step, and the within-trajectory ratio of the largest to the smallest gain is 19.9 $[14.2, 32.9]$. A final-step error passes to the endpoint unchanged, whereas an early error of the same size is amplified by an order of magnitude. The sum M weights every step equally.

A separate DDIM check. An earlier SD1.5 test compares seven ten-call schedules at guidance 7.5 against a 500-node reference augmented with all candidate nodes. Four calibration prompts select the four most accurate schedules; twelve disjoint prompts evaluate their rankings. Within this subset, $\rho(M, d) = -0.50\,[-0.80, -0.30]$. Minimum-$M$ selection chooses power-1.3 on nine prompts, whereas the endpoint oracle chooses power-0.7 on nine. A separate 256-prompt preference check of the previously selected schedules favors the numerically less accurate power-1.3 by $0.137\,[0.068, 0.204]$ PickScore and $0.00594\,[0.00364, 0.00823]$ HPSv2.1. These prompts are excluded from the main DPM-Solver++ study.

B Allocation model and error propagation

We first prove the allocation result from Section 4. The remaining sections examine how errors made at individual steps affect the endpoint, including the assumptions needed to reconstruct that effect from local measurements.

B.1 An exactly solvable allocation model

This section proves Proposition 1 and records the numerical example behind it. The model is a two-dimensional, state-independent ODE with one early and one late interval. It isolates one mechanism, a larger early local error, and has propagation gain one, so it does not by itself establish the amplification measured in Appendix A.8. It is not a theorem about SD1.5, SDXL or PixArt-Σ.

Setting. Sampling time $u \in [0, 2]$ increases toward the output. The early interval is $[0, 1]$ and the late interval $[1, 2]$. The field is $z'(u) = (\frac{1}{2} + a(u - \frac{1}{2}), \frac{1}{2})$ on $[0, 1)$ and $z'(u) = (\frac{1}{2}, \frac{1}{2} + b(u - \frac{3}{2}))$ on $[1, 2]$, with the right-hand field at $u = 1$. The centered terms integrate to zero and the constant drifts to one, so $z^* = (1, 1)$ for every $a, b > 0$. Explicit Euler uses $N \geq 2$ calls, includes $u = 1$, and places $n \in \{1, \dots, N-1\}$ steps in the early interval.

Proof of the error formula and of uniform optimality. On an early Euler step of width h starting at u, the exact first-coordinate increment is $h/2 + a[(u - \frac{1}{2})h + h^2/2]$ and Euler produces $h/2 + a(u - \frac{1}{2})h$; constant drifts are integrated exactly. The field does not depend on the state, so step errors add exactly: for early widths h_i and late widths ℓ_j, $z_N - z^* = \big(-\frac{a}{2}\sum_i h_i^2, -\frac{b}{2}\sum_j \ell_j^2\big)$. Each set of widths sums to one, and Cauchy–Schwarz gives $\sum_i h_i^2 \geq 1/n$ and $\sum_j \ell_j^2 \geq 1/(N-n)$, with equality exactly at uniform spacing. Both $E = \|z_N - z^*\|^2$ and $-Q(z_N)$ increase with the magnitude of either coordinate, so uniform spacing within each interval minimizes both for every n, and substitution gives $E(n)$ and $-Q(z_N)$ as stated.

Proof of the allocation formula. E and $L(n) = -Q(z_N)$ are strictly convex on $(0, N)$ and diverge at its ends. Setting $E'(n) = 0$ gives $(n/(N-n))^3 = a^2/b^2$; L is E with a^2 and b^2 replaced by $\lambda_H a^2$ and $\lambda_L b^2$, which gives the second ratio, smaller by the factor $(\lambda_H/\lambda_L)^{1/3} < 1$. For integer schedules, if n_1 minimizes E at $A_1 = (a/b)^2$ and n_2 at $A_2 > A_1$, the two optimality inequalities imply $(A_2 - A_1)(1/n_2^2 - 1/n_1^2) \leq 0$, so $n_2 \geq n_1$:

the endpoint-optimal early count is nondecreasing in a/b, and the score-optimal count is no larger than the endpoint-optimal one. Integer ties can hide the difference.

Proof of the gap statement. A late maximum gap $h \le 1/m$ requires at least m late steps, and uniform spacing makes $N - n \ge m$ sufficient. Uniformity still minimizes E for each feasible n. In the continuous relaxation, E decreases strictly up to n_E and L increases strictly beyond n_Q. If $n_Q \le N - m < n_E$, the constrained endpoint optimum is $n = N - m$, which has larger E and smaller L than the continuous endpoint optimum n_E. Integer optima are checked separately in the following example.

Guidance example. With $b = 1$ and $a(w) = w - \frac{19}{20}$, the early field equals $v_{\text{uncond}} + w(v_{\text{cond}} - v_{\text{uncond}})$ for fields whose early centered coefficients are $-\frac{19}{20}$ and $\frac{1}{20}$ and whose late fields coincide. Guidance therefore changes only the early interval; the initial state, exact endpoint, score, solver and cost stay fixed. For $N = 10$ and $\lambda_H = 1/100$, $\lambda_L = 1$, exact rational arithmetic over all nine allocations gives Table 17. The selected allocations are strict minimizers, so small perturbations of the model preserve them. The values are a mathematical illustration, not a measurement.

Table 17: The two-interval model at $N = 10$, $\lambda_H/\lambda_L = 1/100$. Early calls lie in the guided interval. Squared endpoint error E and score loss $-Q$ are exact rational values, rounded.

guidance w	selection	early / late calls	E	$-Q$
1	both criteria	1 / 9	0.0037	0.0031
7.5	minimum endpoint error	8 / 2	0.230	0.0642
7.5	maximum score	4 / 6	0.677	0.0136
7.5	endpoint error, late gap $\le 1/4$	6 / 4	0.314	0.0186

Would weighting the distance by quality help? Near a smooth score maximum x^*, the score loss is $Q(x^*) - Q(x^* + e) = \frac{1}{2} e^\top H e + o(\|e\|^2)$, where $H = -\nabla^2 Q(x^*)$ measures sensitivity to different error directions. This weighted distance approximates score loss locally. At a general reference, the expansion also contains the first-order term $-\langle \nabla Q(x^*), e \rangle$, and $-\nabla^2 Q(x^*)$ need not be positive semidefinite. Thus a score Hessian does not automatically define a distance. At a score maximum, weighting by the gradient instead gives zero in every direction. Either construction requires information from the score; its ranking accuracy away from the reference needs testing. The argument below shows that unequal directional sensitivities can make Euclidean distance misrank even nearby endpoints.

Distance can misrank arbitrarily close endpoints. Let Q be twice continuously differentiable near x^* with $\nabla Q(x^*) = 0$ and $H = -\nabla^2 Q(x^*)$ positive definite with eigenvalues $0 < \lambda_{\min} < \lambda_{\max}$, so x^* is a strict local maximizer. For unit eigenvectors $v_{\max}, v_{\min}$ and $1 < c < \sqrt{\lambda_{\max}/\lambda_{\min}}$, set $x_A = x^* + \epsilon v_{\max}$ and $x_B = x^* + c\epsilon v_{\min}$. Then $\|x_A - x^*\| < \|x_B - x^*\|$ while $Q(x_A) - Q(x_B) = -\frac{1}{2}(\lambda_{\max} - c^2 \lambda_{\min})\epsilon^2 + o(\epsilon^2) < 0$ for all small $\epsilon > 0$: the closer endpoint scores lower. If the Hessian is L-Lipschitz on the relevant ball, the ordering holds for $\epsilon < 3(\lambda_{\max} - c^2 \lambda_{\min})/[L(1 + c^3)]$. Euclidean distance treats directions equally and the score need not; this is why a best-scoring reference does not make distance to it a reliable ranking, as the interpolation test of Section 4 shows empirically.

Testing a first-order score prediction. An exploratory SD1.5 check uses the original 16 calibration prompts and 57 schedules. For each prompt, we predict a candidate's score by $Q(x^*) + \langle \nabla Q(x^*), x_S - x^* \rangle$, using PickScore with decoding included. This replaces distance by the score's sensitivity at the reference. The correlation between predicted and measured schedule-mean scores is 0.13, and the prediction selects power-0.6, whose last call is at 251. Its correlation advantage over negative endpoint error has a paired bootstrap interval containing zero. Scores use FP32 decoding and differentiable image resizing. A directional derivative on one prompt is checked by finite differences in 64-bit floating-point arithmetic (FP64).

We also evaluate $Q(x^*+\alpha(x_S-x^*))$ for power-p with $p \in \{0.5, 0.6, 0.8, 0.9, 1.0, 1.6\}$, uniform–Karras 0.25, and AYS, at $\alpha \in \{0, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1\}$. Several mean score curves dip and recover rather than follow the predicted line. The mean absolute prediction error grows with displacement, while all eight signed mean-error intervals at $\alpha = 1$ include zero. These calibration results show limited accuracy for this first-order predictor. They do not test a quadratic predictor or rule out other score-informed objectives. No schedule from this check was evaluated on held-out prompts.

B.2 How a local correction changes the endpoint

Classical numerical analysis describes how later steps propagate and correct an earlier error (Hairer et al., 1993; Skeel, 1986; Zadunaisky, 1976; Stetter, 1978; Shampine & Watts, 1976). Sampling analyses bound distribution error (Li & van der Schaar, 2024), how small state changes are amplified (Tao & Choi, 2026), and accumulated error sizes (Chen et al., 2023). The following analysis examines these effects along the generation path.

From a local correction to an endpoint gain. An error introduced at one step can grow or shrink as later steps act on it. Its effect on endpoint error therefore depends on three things: the size of the local correction, how much later steps amplify it, and the direction of the final change. Let δ_k correct the state after step k, and let r_k be the original endpoint minus the endpoint obtained by continuing from this corrected state. Incoming solver history is held fixed at the intervention; subsequent predictions and history evolve normally. For nonzero e, δ_k, and r_k, define the correction size $a_k = \|\delta_k\|$, finite propagation gain $g_k = \|r_k\|/a_k$, and alignment (the cosine between directions) $c_k = \langle e, r_k\rangle/(\|e\|\|r_k\|)$. Equation (1) gives

$$\|e\|^2 - \|e - r_k\|^2 = 2\|e\|g_k a_k c_k - (g_k a_k)^2. \tag{3}$$

Thus a large local error or propagation gain alone does not imply a large reduction in endpoint error. The correction reduces endpoint error exactly when $c_k > g_k a_k/(2\|e\|)$. In words, the propagated correction must point sufficiently toward the reference, without overshooting it. Whether that same move improves the image score is a separate condition in Eq. (1).

Guidance and finite corrections. For the affine guided field of Section 4, $J_w = D_x v_{\text{uncond}} + wD_x(v_{\text{cond}} - v_{\text{uncond}})$ is the Jacobian, the derivative of the velocity field with respect to the state (D_x). An infinitesimal perturbation η obeys $\dot\eta = J_w\eta$ along the guided trajectory. This equation describes propagation in the continuous ODE. A multistep numerical solver instead propagates an augmented state that includes its prediction history. The finite gain g_k in Eq. (3) holds incoming history fixed at injection and lets future history evolve; it is a directional response in the tested direction, not the largest amplification over all directions (the Jacobian spectral norm). Substituting $\|r_k\| = g_k a_k$ and $\langle e, r_k\rangle = \|e\|g_k a_k c_k$ into Eq. (1) gives the stated gain and its sign condition without a linearization assumption.

B.3 First-order reconstruction: definitions and algorithm

We compare two ways to predict the final error. The first measures what each step's error does to the endpoint and adds those effects. The second replaces each measured effect with a linear approximation. We then compare these predictions with the error of the actual coarse solve. The experiments use one-step solvers without stored history.

Definitions. Let $t_1, \ldots, t_{K+1} = T$ be the boundaries of K solver steps in sampling order, and $x_k^{\text{ref}} = x^{\text{ref}}(t_k)$ the reference states. Write N for model calls (NFE): $N = K$ for these DDIM/Euler runs and $N = 2K$ for these Heun runs. The update Ψ_k^{cheap} maps from t_k to t_{k+1}, for $k = 1, \ldots, K$. The difference after one step is the signed local truncation error

$$\tau_k = \Psi_k^{\text{cheap}}(x_k^{\text{ref}}) - x_{k+1}^{\text{ref}}. \tag{4}$$

This is a vector in the model state space. To determine what that error alone does to the final sample, we inject it at the end of step k and integrate accurately from there. With

$\mathcal{P}_{t_{k+1}\to T}$ denoting the fine integration map to terminal time T, the measured response is

$$s_k = \mathcal{P}_{t_{k+1}\to T}\big(x^{\mathrm{ref}}_{k+1} + \tau_k\big) - \mathcal{P}_{t_{k+1}\to T}\big(x^{\mathrm{ref}}_{k+1}\big). \tag{5}$$

To measure several errors together, we inject their reference-based vectors at their assigned steps and propagate the perturbed state accurately between injections. Let F_{fr} return the resulting endpoint change. The subscript denotes a frozen-error experiment: the injection times and vectors are fixed, even though the evolving state is perturbed. Then $s_k = F_{\mathrm{fr}}(0,\dots,\tau_k,\dots,0)$ and $e_{\mathrm{fr}} = F_{\mathrm{fr}}(\tau_1,\dots,\tau_K)$. The actual coarse-solver error is $e_{\mathrm{coarse}} = x^{\mathrm{coarse}}_T - x^{\mathrm{ref}}_T$: in a coarse solve, an early error changes later states, model evaluations, truncation errors, and, for a multistep method, solver history. If the frozen responses were exactly additive, e_{fr} would equal $\sum_k s_k$. For $e_{\mathrm{fr}} \neq 0$, we report the relative remainder

$$\chi = \frac{\|e_{\mathrm{fr}} - \sum_k s_k\|}{\|e_{\mathrm{fr}}\|}, \qquad \chi = 0 \text{ is perfect additivity.} \tag{6}$$

Three approximations can therefore fail separately. Linearization can misestimate a single response s_k. Adding the measured responses can miss interactions in e_{fr}. Finally, the frozen experiment can differ from the actual solver error e_{coarse} because later errors are held fixed. The bound below addresses interactions in the frozen system.

The linear reconstruction is $\hat{e} = \sum_k \Phi(T, t_{k+1})\,\tau_k$, where $\Phi(T,t) = D_x\mathcal{P}_{t\to T}(x^{\mathrm{ref}}(t))$ is the state Jacobian of the fine integration map. It describes how a small change at time t changes the endpoint. We abbreviate $\Phi(T, t_{k+1})$ by Φ_k.

Reference trajectory and residuals. The SiT reference uses approximately 200 fine Heun steps. For Euler with field $u(x,t)$, $H_k = t_{k+1} - t_k$, the local error is $\tau_k = x^{\mathrm{ref}}_k + H_k u(x^{\mathrm{ref}}_k, t_k) - x^{\mathrm{ref}}_{k+1} = -\frac{1}{2}H_k^2\ddot{x}(t_k) + O(H_k^3)$. It is an endpoint displacement, distinct from the velocity-space defect of a continuous interpolant. Its sign agrees with e_{coarse}.

Tangent propagation. The transported vector $\Phi(T, t_{k+1})\tau_k$ is the directional derivative of the near-exact endpoint map $\mathcal{P}_{t_{k+1}\to T}$ along τ_k. We estimate it by a central finite difference at the reference state, with dimensionless perturbation scale ε:

$$\ell_k \;=\; \frac{\mathcal{P}_{t_{k+1}\to T}(x^{\mathrm{ref}}_{k+1} + \varepsilon\tau_k) - \mathcal{P}_{t_{k+1}\to T}(x^{\mathrm{ref}}_{k+1} - \varepsilon\tau_k)}{2\varepsilon}, \qquad \varepsilon = 0.05, \tag{7}$$

We sum the signed responses before taking norms: $\hat{e} = \sum_k \ell_k$.

Reconstruction evidence. On DiT-XL/2 with DDIM at six calls, we compare the linear reconstruction $\hat{e} = \sum_k \ell_k$ with the measured single-error sum $\hat{e}_{\mathrm{s}} = \sum_k s_k$. Ten fresh images and sixteen schedules (eight power schedules, log-SNR, and seven Dirichlet schedules) give median $\|\hat{e} - \hat{e}_{\mathrm{s}}\|/\|\hat{e}_{\mathrm{s}}\| = 0.73$ (interquartile range 0.53–0.96). On SiT-XL/2 with six-call Euler and 32 images, the cosines of the linear and measured single-error sums with the actual coarse error are 0.51 and 0.91. The frozen response itself has cosine 0.96 and norm ratio 1.20 against that target. Thus linearization and freezing later errors introduce distinct discrepancies. For a smooth frozen map, $F_{\mathrm{fr}}(\varepsilon\tau) = \varepsilon\sum_k \Phi_k\tau_k + O(\varepsilon^2)$; this expansion excludes feedback through changing local errors in the actual solver.

B.4 Interaction bound and counterexample

Adding individual effects can miss the way errors change one another's propagation. The first proposition bounds this interaction. The second shows why a small interaction alone does not imply a small endpoint error: individual effects may add exactly but still add to a large value.

Proposition 2 (bound on interactions across steps)**.** For a subset $S \subseteq \{1,\dots,K\}$, let I_S denote the interaction that requires exactly the steps in S, obtained by inclusion–exclusion over their exact endpoint responses. Then $e_{\mathrm{fr}} - \sum_k s_k = \sum_{|S|\geq 2} I_S$. If the q-th mixed derivative of the endpoint map is bounded in operator norm by L_q on the injection box $\{(a_1\tau_1,\dots,a_K\tau_K) : a_k \in [0,1]\}$, for $q = 2,\dots,K$, then $\|e_{\mathrm{fr}} - \sum_k s_k\| \leq \sum_{q\geq 2} L_q e_q(m)$,

where $m = (m_1, \ldots, m_K)$, $m_k = \|\tau_k\|$, and $e_q(m) = \sum_{1 \le i_1 < \cdots < i_q \le K} m_{i_1} \cdots m_{i_q}$ is the q-th elementary symmetric polynomial. For fixed $M = \sum_k m_k > 0$, each e_q decreases as the vector of error magnitudes becomes more concentrated. The second-order term is $\frac{1}{2}LM^2(1-H)$, with $L = L_2$ and $H = \sum_k (m_k/M)^2$.

Proof of Proposition 2. Let $F = F_{\mathrm{fr}}$, with $F(0) = 0$, and let τ_S retain only the injections indexed by S. Inclusion–exclusion defines $I_S = \sum_{T \subseteq S} (-1)^{|S|-|T|} F(\tau_T)$. Its inverse identity gives $F(\tau) = \sum_{S \neq \varnothing} I_S$. Since $I_{\{k\}} = s_k$,

$$R_{\mathrm{int}} = e_{\mathrm{fr}} - \sum_k s_k = \sum_{|S| \ge 2} I_S. \tag{8}$$

For a $C^{|S|}$ map on the injection box, repeated application of the fundamental theorem of calculus expresses I_S as an integral of a mixed derivative. The operator-norm bound therefore gives $\|I_S\| \le L_{|S|} \prod_{i \in S} m_i$, and hence

$$\|R_{\mathrm{int}}\| \le \sum_{q=2}^{K} L_q e_q(m). \tag{9}$$

For completeness, concentration means majorization: at equal total M, every leading partial sum of the decreasingly sorted m^A exceeds or equals that of m^B. For $q \ge 2$, $(m_i - m_j)(\partial_i e_q - \partial_j e_q) = -(m_i - m_j)^2 e_{q-2}(m_{\setminus\{i,j\}}) \le 0$, where $e_0 = 1$. Thus e_q is Schur-concave and $m^A \succ m^B$ implies $e_q(m^A) \le e_q(m^B)$. Finally, $e_2 = \frac{1}{2}(M^2 - \sum_k m_k^2) = \frac{1}{2}M^2(1-H)$. This comparison holds the propagation map and constants L_q fixed. Because the same constant is used for all step subsets of a given size, the bound does not distinguish early from late errors. Their actual interactions can differ. Changing the schedule also changes the map and constants, and cancellation can reduce the measured interaction. The concentration H determines the second-order term, not the full bound.

Small-error limit. For a twice continuously differentiable F, write $J_k = D_{\delta_k} F(0)$. If $\sum_k J_k \tau_k \neq 0$, then $F(\varepsilon\tau) = \varepsilon \sum_k J_k \tau_k + O(\varepsilon^2)$ has norm $\Theta(\varepsilon)$, whereas $R_{\mathrm{int}}(\varepsilon\tau) = O(\varepsilon^2)$. Consequently $\chi = O(\varepsilon)$. This conclusion need not hold if the first-order responses cancel.

Proposition 3 (additivity does not bound accuracy)**.** A smooth three-step injection system admits equal-total-size allocations A, B with $\chi_A < \chi_B$ but $\|F_A\| > \|F_B\|$.

Construction and proof. Start at scalar state zero. Inject τ_1 after step one, propagate by $P_1(y) = ay$, inject τ_2 after step two, then propagate by $P_2(y) = gy + \lambda y^2$. Set the final injection $\tau_3 = 0$, with $a > 1$, $g > 0$, and $\lambda > 0$. Both nonzero injections precede nonlinear propagation, so this construction respects the end-of-step injection convention. Its response is $F(\tau_1, \tau_2, 0) = g(a\tau_1 + \tau_2) + \lambda(a\tau_1 + \tau_2)^2$. The single-error responses are $s_1 = ga\tau_1 + \lambda a^2 \tau_1^2$ and $s_2 = g\tau_2 + \lambda \tau_2^2$, giving $R_{\mathrm{int}} = 2\lambda a \tau_1 \tau_2$. For $A = (M, 0, 0)$ and $B = (M/2, M/2, 0)$, with $M > 0$,

$$F_A = gaM + \lambda a^2 M^2, \qquad \chi_A = 0,$$

$$F_B = \tfrac{1}{2} g(a+1)M + \tfrac{1}{4}\lambda(a+1)^2 M^2, \qquad \chi_B = \frac{\lambda a M^2}{2F_B} > 0.$$

Both endpoints are positive, and $F_A - F_B = \frac{1}{2}g(a-1)M + \frac{1}{4}\lambda(a-1)(3a+1)M^2 > 0$. Thus perfect additivity can accompany the larger endpoint error.