MEAN FLOWS AS CONTROLLABLE PRIORS FOR SOLVING INVERSE PROBLEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Training-free inverse solvers built on flow priors repeatedly need to predict the clean sample reached from an intermediate state along the generative trajectory. Standard flow-matching models expose only an instantaneous velocity, so such predictions need numerical integration or coarse extrapolation. MeanFlow models learn a finite-interval transport map, which lets us use them as *controllable priors* that move a state across any interval, to the next stage or clean endpoint—all in one evaluation. This interface carries the structure of any flow-based solver onto a MeanFlow prior and makes a new control policy natural: *Receding-Horizon State Optimization* (RHSO), a model-predictive controller whose decision variable is the generative state itself. A first-order analysis for general data distributions shows how distributing optimization relaxes but retains the prior's implicit regularization. Capacity-matched comparisons in pixel and latent space measure what MeanFlow priors contribute: they surpass flow-matching priors at every stage count tested, most widely with few stages and in latent space. Across five ImageNet restoration tasks, four solvers and four priors, RHSO is best in 35 of 60 task–metric cells and wins 200 of 240 pairwise comparisons. A depth-controlled study attributes most of its gain to distributing optimization over stages, not deeper numerical planning.

1 INTRODUCTION

Inverse problems seek to recover an unknown signal $x^\star$ from an indirect, incomplete, or noisy observation

$$y = A(x^\star) + \eta, \tag{1}$$

where A is a known forward operator and $\eta \sim \mathcal{N}(0, \sigma^2 I)$ is measurement noise. Because A can discard information, measurement consistency alone need not identify a unique or perceptually plausible solution. Maximum-a-posteriori recovery maximizes $p(x \mid y) \propto p_{Y|X=x}(y \mid x)\, p_X(x)$, but the prior over natural signals is intractable, so one minimizes a proxy summing a data-fidelity term with a regularizer standing in for $-\log p_X(x)$; deep generative models are effective instantiations of that regularizer (Bora et al., 2017; Asim et al., 2020; Altekrüger et al., 2023; Wei et al., 2022; Daras et al., 2024).

Continuous normalizing flows (Chen et al., 2018) trained with Flow Matching (FM) objectives (Lipman et al., 2023; Tong et al., 2024; Liu et al., 2023) scale to high-resolution synthesis in pixel and latent space (Esser et al., 2024; Ma et al., 2024; Rombach et al., 2022; Karras et al., 2022), and their learned velocity field $v_\theta(x, t)$ has repeatedly been used to solve inverse problems without retraining (Kim et al., 2025; Pokle et al., 2024; Zhang et al., 2024; Ben-Hamu et al., 2024; Martin et al., 2025; Pourya et al., 2026; Webber et al., 2026). Every such solver repeatedly asks the prior the same question: *what will the current partially generated state become?* Because v_θ represents only instantaneous motion, answering it for a distant clean state requires either composing many solver steps and differentiating through them, or a coarse one-step extrapolation that degrades over long horizons. MeanFlow changes the interface: it learns the average motion over a finite interval, so a single learned map transports the state between widely separated times (Geng et al., 2025; 2026; Lu et al., 2026). Such maps have been used for test-time reward guidance, including measurement-consistency rewards (Sabour et al., 2025; Huang et al., 2026), and trained for one-step conditional reconstruction (Mammadov et al., 2026; Shi et al., 2026); we ask a complementary question: *how does a pretrained finite-interval map change training-free inverse-problem control?*

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.

We answer it in three steps. First, we generalize the interface of training-free flow solvers: any solver that transports a state along the flow numerically—above all one that repeatedly predicts distant states—can have that transport replaced by a single evaluation of a pretrained MeanFlow map, leaving its outer logic untouched. We test this on three recent baseline families (four solvers): D-Flow, PnP-Flow and the MPC-RHC/MPC-Δt controllers of MPC-Flow, a representative set chosen in Section 2.2. Second, we introduce *Receding-Horizon State Optimization* (RHSO), a control policy that fits naturally with this interface. The policy applies model-predictive control, but with the generative state itself as the decision variable rather than an external control signal. Because a MeanFlow prior predicts any horizon at the same cost, optimization can be distributed along the trajectory without any planning cost growing with the look-ahead. Third, we compare MeanFlow against FM priors in pixel and latent space, including capacity-matched pairs in both spaces, across all of these solvers and five restoration tasks: denoising, deblurring, $2\times$ super-resolution, random inpainting and box inpainting. In short, MeanFlow priors run existing solvers unchanged and surpass FM priors at every stage count tested, and RHSO is the strongest solver on every prior.

Our contributions are: **(i) A finite-interval control interface.** We formulate pretrained MeanFlow maps as controllable priors for training-free inverse problems, enabling direct clean-endpoint prediction over arbitrary horizons without numerical suffix integration, in pixel and latent space, and instantiate four representative solver structures on it without retraining (Sections 3.1 and 4.2). **(ii) RHSO.** We introduce a pre-transport receding-horizon state-optimization policy that repeatedly optimizes the current generative state through a terminal prediction before executing the next transport interval, give a first-order control analysis, exact for Gaussian data, showing that replanning at late stages reconditions the prior-induced control geometry, and isolate, with a depth-controlled standard-flow experiment, the benefit of distributed intervention from that of endpoint-prediction depth (Sections 3.2, 3.3 and 4.4). **(iii) A broad and capacity-controlled evaluation.** A four-prior, five-task benchmark shows that existing solver structures transfer to MeanFlow priors, that the interface lets them take coarse steps—one-terminal-prediction D-Flow is far more competitive on MeanFlow priors than on FM priors—and that RHSO is the strongest solver structure on MeanFlow and FM priors alike (Section 4.2); capacity-matched stage-count sweeps in pixel and latent space, under two optimizer-budget controls, then isolate what the finite-interval interface itself contributes (Section 4.3).

2 Background

2.1 Flow Matching and finite-interval transport

Let $v_\theta : \mathbb{R}^d \times [0,1] \to \mathbb{R}^d$ be a learned velocity field inducing a probability path from a source distribution at $t = 0$ to the data distribution at $t = 1$. For $x_0 \sim p_{\text{source}}$ and $x_1 \sim p_{\text{data}}$, paired independently or by minibatch optimal transport (Tong et al., 2024), the linear conditional path $x_t = (1-t)x_0 + tx_1$ has conditional velocity $x_1 - x_0$, and Flow Matching regresses $\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t,x_0,x_1}\|v_\theta(x_t,t) - (x_1 - x_0)\|_2^2$ without simulating the generative ODE during training (Lipman et al., 2023). Generation then solves

$$\frac{dx_t}{dt} = v_\theta(x_t, t), \qquad x_0 \sim p_{\text{source}}, \tag{2}$$

with a numerical solver; we write $\Psi_{t\to t'}$ for exact transport, which a standard FM model approximates by composing solver steps. Denoising parameterizations may instead predict clean data, as in the pixel-space x-prediction of Li and He (2026), which on the linear path induces $v_\theta(x_t,t) = [D_\theta(x_t,t) - x_t]/(1-t)$; latent-space models (Rombach et al., 2022; Peebles and Xie, 2023; Esser et al., 2024), among them SiT (Ma et al., 2024), apply the same dynamics to the codes of a pretrained autoencoder. Either way, conventional flow inference exposes *local* dynamics whose finite-time transport is obtained numerically.

MeanFlow instead learns the average velocity over a finite interval, parameterized directly by a network $u_\theta(x_t; t, t')$ (Geng et al., 2025). Along a trajectory of Equation (2),

$$u(x_t; t, t') = \frac{1}{t'-t}\int_t^{t'} v(x_\tau, \tau)\, d\tau, \qquad x_{t'} = x_t + (t'-t)\, u(x_t; t, t'). \tag{3}$$

Differentiating Equation (3) in t at fixed t' yields the *MeanFlow identity* $u(x_t;t,t') - (t'-t)\dot{u}(x_t;t,t') = v(x_t,t)$, which links the average and instantaneous fields through one Jacobian-

vector product and makes the objective trainable without simulation (Geng et al., 2026). The learned field yields the finite-interval map

$$T_\theta(x_t; t \to t') = x_t + (t' - t)\, u_\theta(x_t; t, t'), \qquad \lim_{t' \to t^+} u(x_t; t, t') = v(x_t, t). \tag{4}$$

The distinction is simple but consequential: $v_\theta(x_t, t)$ answers *where the state should move locally*, whereas $T_\theta(x_t; t \to t')$ answers *where it should be at time* t'; in particular $T_\theta(x_t; t \to 1)$ predicts the clean endpoint directly. Pixel-space (pMF) and latent-space (iMF) MeanFlow models exist at ImageNet scale (Lu et al., 2026; Geng et al., 2026; Deng et al., 2009).

2.2 Training-free flow solvers for inverse problems

For Equation (1), let

$$\Phi(x; y) = \frac{1}{2m} \|A(x) - y\|_2^2 \tag{5}$$

denote a representative fidelity objective, where m is the number of measured scalars, i.e. the dimension of y (exact normalizations in Appendix C). D-Flow optimizes the generative starting state and simulates the full trajectory from that state through a terminal solve (Ben-Hamu et al., 2024): on a discretization $t = t_0 < \cdots < t_K = 1$ with numerical updates F_j,

$$x_{t_0}^\star = \arg\min_{x_{t_0}} \Phi(F_{K-1} \circ \cdots \circ F_0(x_{t_0}); y)\,, \qquad \nabla_{x_{t_0}} \Phi = J_{F_0}^\top \cdots J_{F_{K-1}}^\top \nabla_{x_1} \Phi, \tag{6}$$

so every optimizer step backpropagates through the complete numerical trajectory. MPC-Flow instead casts guidance as control, repeatedly planning over a finite horizon, executing one interval and replanning (Webber et al., 2026; Mayne et al., 2000); its best instantiation MPC-Δt, like PnP-Flow (Martin et al., 2025), relies on the one-step clean-state estimate

$$D_t(x_t) = x_t + (1 - t) v_\theta(x_t, t), \tag{7}$$

which, for the linear path, is the optimal conditional-mean predictor $\mathbb{E}[x_1 \mid x_t]$ (Liu et al., 2023). PnP-Flow treats D_t as a denoiser inside an alternating fidelity-gradient, reprojection and denoising procedure without backpropagating through the generative model (Venkatakrishnan et al., 2013; Martin et al., 2025), while MPC-Δt uses it to define a terminal value $\widehat{V}(t, x_t) = \Phi(D_t(x_t); y)$. Although optimal as a one-shot predictor, D_t need not match the clean state reached by integrating the marginal flow; as a single Euler extrapolation it can incur substantial error over long jumps when the trajectory is curved (Frans et al., 2025), which both methods tolerate by correcting repeatedly.

Scope of comparison. Other flow solvers include OT-ODE, Flow-Priors/ICTM, FlowDPS, Flower and P-Flow (Pokle et al., 2024; Zhang et al., 2024; Kim et al., 2025; Pourya et al., 2026; Jiang et al., 2026), and diffusion-prior methods use related projection, guidance and posterior-optimization ideas (Chung et al., 2023; Kawar et al., 2022; Song et al., 2023; Mardani et al., 2024). We compare against a representative set: D-Flow, the $N = 1$ case of RHSO (Section 3.2); both MPC-Flow controllers, which share its receding-horizon structure but optimize added controls; and PnP-Flow, a fast solver that never differentiates the prior; MPC-Δt and PnP-Flow are also the two strongest solvers on the linear tasks of the broadest recent comparison, which includes OT-ODE, Flow-Priors and FlowDPS (Webber et al., 2026). The benchmark is thus a controlled cross-interface study, not a solver leaderboard.

3 MeanFlow control for inverse problems

3.1 A finite-interval solver interface

Training-free flow solvers interact with the prior through one primitive: the transport of the current state to a later time, written generically as $\mathcal{G}_\theta(x_t; t \to t')$, which is differentiated whenever the solver differentiates the prior. For $t' = 1$ it is a prediction of the clean reconstruction. With a standard FM prior the primitive is realized numerically, $\mathcal{G}_\theta(x_t; t \to t') = F_{K-1} \circ \cdots \circ F_0(x_t)$ with K solver steps of Equation (2): its cost and the depth of its gradient grow with K, while the cheapest choice $K = 1$, the extrapolation Equation (7), loses accuracy with the length of the interval. A MeanFlow prior supplies the same primitive in closed form,

$$\mathcal{G}_\theta(x_t; t \to t') = T_\theta(x_t; t \to t'), \tag{8}$$

at one evaluation for every pair $t < t'$, and by the limit in Equation (4) it contains the FM primitive as the special case in which t' approaches t. The substitution therefore applies to any solver that consumes $\mathcal{G}_\theta$, not to a particular family, and matters most where long-horizon transport is consumed repeatedly: a composition of numerical steps collapses into one evaluation, a single extrapolation becomes a learned map, and a solver that prefers short steps can still take them as $T_\theta(\cdot; t \to t+\Delta t)$. We instantiate it on four representative structures, outer logic preserved and the model never fine-tuned: PnP-Flow uses $T_\theta(\cdot; t \to 1)$ as the prior projection in place of Equation (7), D-Flow differentiates that single transport rather than the composition in Equation (6), and the MPC variants use T_θ for rollouts and terminal values (Appendix A). For a latent prior the map acts on autoencoder codes z_t and the fidelity is evaluated after decoding, $\Phi(\mathcal{D}(\mathcal{G}_\theta(z_t; t \to 1)); y)$, so a terminal prediction costs one map evaluation plus one decode and its gradient includes the decoder Jacobian. Two properties of the interface matter below: a terminal prediction costs one evaluation of the map (plus one decode for a latent prior) at any horizon, and its gradient is one backpropagation through that map (and decoder) whose depth does not grow with the remaining trajectory.

3.2 Receding-horizon state optimization

Those two properties make a specific controller natural. Model-predictive control plans over a horizon, executes the first step, and replans from the state reached (Mayne et al., 2000); RHSO applies this receding-horizon principle with the generative state itself as the decision variable (Algorithm 1). Unlike MPC-RHC, which optimizes explicit controls over a future rollout, RHSO optimizes the current state; unlike D-Flow it repeats this at every stage; and unlike PnP-Flow it differentiates the prior (Webber et al., 2026; Ben-Hamu et al., 2024; Martin et al., 2025). Differentiating the prior is expensive, but it routes the fidelity gradient through the Jacobian of the remaining generative map, so corrections are expressed in directions the prior can realize—an implicit regularization hypothesized for source-state optimization by Ben-Hamu et al. (2024) and made precise in Section 3.3—and a finite-interval map keeps that gradient to one shallow backpropagation.

Let N be the number of receding-horizon stages, and let

$$0 = t_0 < t_1 < \cdots < t_N = 1, \qquad t_k = \left(\frac{k}{N}\right)^\beta, \qquad k = 0, \ldots, N, \tag{9}$$

be the stage grid for the pure-noise initialization used throughout, so N is also the number of committed generative intervals. The uniform schedule $\beta = 1$ is used in every headline run; $\beta < 1$ concentrates stages toward the data endpoint and $\beta > 1$ toward the noise endpoint (Appendix H.3). Write P_k for the terminal predictor available at t_k and E_k for the executed interval; both are supplied by the prior:

$$\text{MeanFlow:} \quad P_k(x) = T_\theta(x; t_k \to 1), \qquad E_k(x) = T_\theta(x; t_k \to t_{k+1}), \tag{10}$$

$$\text{FM, direct endpoint:} \quad P_k(x) = D_{t_k}(x), \qquad E_k(x) = \text{one numerical interval}, \tag{11}$$

$$\text{FM, integrated suffix:} \quad P_k(x) = \text{numerical solve } t_k \to 1, \qquad E_k(x) = \text{one numerical interval}. \tag{12}$$

Given a differentiable predictor P_k, the stage optimization is prior-agnostic. The direct FM planner is deliberately shallow and so cost-matched to MeanFlow; the integrated suffix is more faithful to the standard-flow trajectory but differentiates through every remaining interval as in Equation (6), and Section 4.4 uses the contrast as a depth-matched control. Saving the pre-optimization state $s = x_{t_k} \in \mathbb{R}^d$, the stage objective is

$$x^\star_{t_k} \approx \arg\min_x \left[\Phi(P_k(x); y) + \frac{\mu}{2d} \|x - s\|_2^2\right], \tag{13}$$

solved only approximately with M optimizer steps, after which we commit $x_{t_{k+1}} = E_k(x^\star_{t_k})$ and repeat. The anchor weight μ penalizes departure from the state actually reached, s (the state dimension d makes it a per-coordinate mean square); $B = NM$ is the total optimizer budget.

Relation to one-shot state optimization. With $N = 1$, RHSO reduces to source-state optimization through a terminal predictor, the structure of D-Flow (Ben-Hamu et al., 2024). Interleaving optimization with execution is shared with recent methods such as MPC-Flow and FMRG (Webber et al., 2026; Huang et al., 2026); RHSO differs in what it optimizes and when: the generative state itself rather than added control variables, and before each transport interval, as a state-optimization

Algorithm 1 Receding-Horizon State Optimization (RHSO)

Require: observation y, fidelity Φ, stages N, inner steps M, schedule exponent β, anchor weight μ, state dimension d; terminal predictor P_k and executed interval E_k supplied by the prior

$t_k \leftarrow (k/N)^\beta$ for $k = 0, \dots, N$ ▷ stage grid of Equation (9); $t_0 = 0$, $t_N = 1$

$x_{t_0} \leftarrow$ sample from the source distribution in $\mathbb{R}^d$

for $k = 0, \dots, N-1$ **do**

  $s \leftarrow x_{t_k}$ ▷ save the state the prior actually reached

  $x^\star_{t_k} \approx \arg\min_x \left[\Phi(P_k(x); y) + \frac{\mu}{2d} \|x - s\|_2^2\right]$ with M steps of a freshly initialized optimizer ▷ optimize via the predicted endpoint

  $x_{t_{k+1}} \leftarrow E_k(x^\star_{t_k})$ ▷ commit a single generative interval

end for

return x_{t_N} (decoded, for a latent prior)

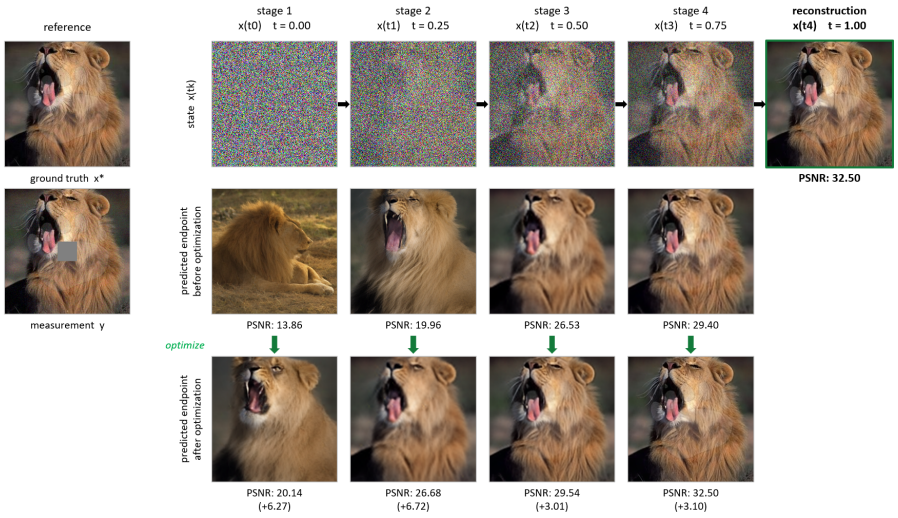

Figure 1: **What RHSO does over one trajectory** (pMF-L/16, box inpainting, $N = 4$, $\beta = 1$, so $t_k \in \{0, .25, .5, .75\}$; fixed structural default of Appendix H). *Top:* the state after each committed interval. *Below:* the terminal prediction entering each stage, $P_k(x_{t_k})$, and the one leaving it, $P_k(x^\star_{t_k})$. Every stage improves its own readout, by 6.27 and 6.72 dB at the two noisiest states, so each committed interval poses a new problem rather than refining one plan; early predictions are poor only because the state is nearly pure noise.

subproblem, rather than as a feedback correction applied after it. MeanFlow does not remove the optimization; it removes the growth of each terminal gradient's cost with the look-ahead, without the accuracy loss a coarse numerical step incurs (Section 4; counts in Appendix A.2).

3.3 Why intervention placement matters

RHSO optimizes the state at several points along the trajectory; placement matters because optimizing a state is not unconstrained optimization of the image: the objective is evaluated on the endpoint, and the two are joined by the remaining generative map. Along the trajectory x_t that the prior alone would follow, let $J_t = D_x \Psi_{t\to 1}(x_t)$, so $\delta x_1 \approx J_t\, \delta x_t$ and the fidelity gradient reaches the state as the pullback $J_t^\top \nabla_{x_1} \Phi$: this is the implicit regularization of Section 3.2 made precise, $-\log p_X$ is never evaluated, and the directions the remaining transport barely amplifies are the costliest for a correction. Since J_t changes along the trajectory, *where* the state is optimized decides that geometry. Corrections $\Delta^{(k)}$ at stages $\mathcal{T} = \{t_k\}$ compose to first order, $\delta x_1 = \sum_k J_{t_k} \Delta^{(k)}$, so the least squared state displacement realizing an endpoint correction δ is

$$\mathcal{C}^\star_{\mathcal{T}}(\delta) = \delta^\top W_{\mathcal{T}}^{-1} \delta, \qquad W_{\mathcal{T}} = \sum_k J_{t_k} J_{t_k}^\top, \tag{14}$$

with $W_{\mathcal{T}}$ the finite-stage controllability Gramian (invertible under the condition below; in general $W_{\mathcal{T}}^{+}$ replaces $W_{\mathcal{T}}^{-1}$ for $\delta \in \mathrm{range}(W_{\mathcal{T}})$, Appendix B) and $\chi(\mathcal{T}) = \mathrm{cond}(W_{\mathcal{T}})$ its *spectral contrast*. A single source stage prices corrections through $J_0 J_0^{\top}$; distributing stages changes the *shape* of $W_{\mathcal{T}}$.

Theorem 1 (Contrast reduction). *Let $W, G \succ 0$ with $\mathrm{cond}(G) < \mathrm{cond}(W)$. Then $\mathrm{cond}(W+G) < \mathrm{cond}(W)$; hence adding a stage t' with $\mathrm{cond}(J_{t'} J_{t'}^{\top}) < \chi(\mathcal{T})$ strictly reduces the contrast.*

Which stages qualify? For the flow-matching transport, the linearized dynamics generating J_t are a scaling plus the Hessian of the log-density of the data smoothed at scale $(1-t)/t$, weighted by $(1-t)/t^3$, which vanishes at the data endpoint (Appendix B). If that curvature is bounded near the endpoint, every J_t is invertible and $\mathrm{cond}(J_t J_t^{\top}) \to 1$ as $t \uparrow 1$, so late enough stages always satisfy the hypothesis of Theorem 1. For Gaussian data, $J_t = \Sigma^{1/2}(t^2\Sigma + (1-t)^2 I)^{-1/2}$, the per-stage contrast falls strictly from $\kappa(\Sigma)$ to 1, every later stage reduces χ, and on a dense grid $\chi \to \sqrt{\kappa(\Sigma)}$.

RHSO thus moves along a regularization trade-off. J_0 is the implicit regularization of source-state optimization, pricing corrections by how much the prior resists them; a late stage ($J_t \approx I$) is nearly unregularized; distributing stages relaxes this regularization without removing it, since the source stage stays in $W_{\mathcal{T}}$. A source-to-endpoint predictor, however deep, exposes only J_0, so depth and placement are distinct, and where the trade-off is best is empirical, depending on the noise level and on whether a decoder regularizes instead (Section 4.4, Appendix H.1). The analysis is first order along the nominal trajectory, "every later stage helps" is Gaussian-only (Appendix B), and nothing here compares MeanFlow with FM, which realize the same transport.

4 Experiments

4.1 Setup

We evaluate Gaussian denoising ($\sigma = .2$), Gaussian deblurring, $2\times$ super-resolution, random inpainting (70% missing) and box inpainting on ImageNet-1k validation images from distinct classes (Deng et al., 2009), conditioning on the known source class and starting every reconstruction from pure noise ($\tau_{\mathrm{init}} = 0$). We report PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018) and measurement RMSE. The priors are pretrained checkpoints of a MeanFlow and a Flow-Matching model per space—pMF (Lu et al., 2026) and JiT (Li and He, 2026) in pixel space, iMF (Geng et al., 2026) and SiT (Ma et al., 2024) in latent space—and each experiment names its checkpoint. We run PnP-Flow, D-Flow (1-step), MPC-RHC and MPC-Δt on every prior through the interface of Section 3.1, alongside RHSO. Every method is tuned per task on a separate tuning set and then frozen; the mechanism studies (Sections 4.3 and 4.4) instead fix the structural settings and freeze the per-task learning rate and anchor (Appendices F and H). Equation (13) uses Adam (Kingma and Ba, 2015) warm-started at s. Operators, normalizations, image-level pairing, statistics, checkpoints and selected configurations: Appendices C and D.

Table 1: **Solver benchmark on the pixel-space MeanFlow prior pMF-L/16.** Each cell is PSNR↑/SSIM↑/LPIPS↓ averaged over 100 ImageNet images, the evaluation scale of prior flow-based solvers (Zhang et al., 2024; Martin et al., 2025; Pourya et al., 2026; Webber et al., 2026); bold marks the best method per task and metric. D-Flow (1-step) is the one-terminal-prediction adaptation of the D-Flow objective, not the original multi-step solver.

Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
Degraded input	20.51/.393/.486	25.91/.712/.273	22.81/.649/.226	12.31/.137/1.044	26.20/.810/.120
PnP-Flow	28.29/**.839**/**.081**	28.20/.819/.165	26.34/.788/.120	26.40/.789/.117	**29.24**/.894/.057
D-Flow (1-step)	26.26/.688/.258	25.58/.690/.301	25.89/.732/.211	26.22/.732/.204	24.58/.777/.189
MPC-RHC	28.89/.805/.123	28.95/.804/.159	25.09/.707/.226	25.01/.693/.236	25.88/.846/.096
MPC-Δt	28.29/.744/.166	29.26/.807/.154	27.35/.806/**.087**	28.18/.843/**.077**	29.16/.893/.059
RHSO (ours)	**29.45**/.826/.093	**30.24**/**.872**/**.104**	**27.50**/**.821**/.095	**28.24**/**.848**/.078	29.00/**.915**/**.046**

Table 2: **Five-task means on all four benchmark priors** (PSNR↑/SSIM↑/LPIPS↓ averaged over the same 100 images; per-task tables in Appendix E). Each block is a separately tuned pretrained system, read within the block; bold marks the best method per prior and metric.

	pMF-L/16 MeanFlow, pixel			JiT-B/16 Flow Matching, pixel			iMF-B/2 MeanFlow, latent			SiT-XL/2 Flow Matching, latent		
Method	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
PnP-Flow	27.69	.826	.108	28.89	.841	.096	25.68	.680	.327	24.94	.673	.434
D-Flow (1-step)	25.71	.724	.233	22.88	.558	.452	25.38	.687	.336	24.75	.644	.290
MPC-RHC	26.76	.771	.168	27.81	.796	.142	26.94	.755	.223	26.86	.749	.230
MPC-Δt	28.45	.819	.109	28.77	.819	.111	**27.13**	**.764**	.194	26.89	.753	.193
RHSO (ours)	**28.89**	**.856**	**.083**	**28.97**	**.856**	**.090**	26.97	.762	**.187**	**26.92**	**.759**	**.190**

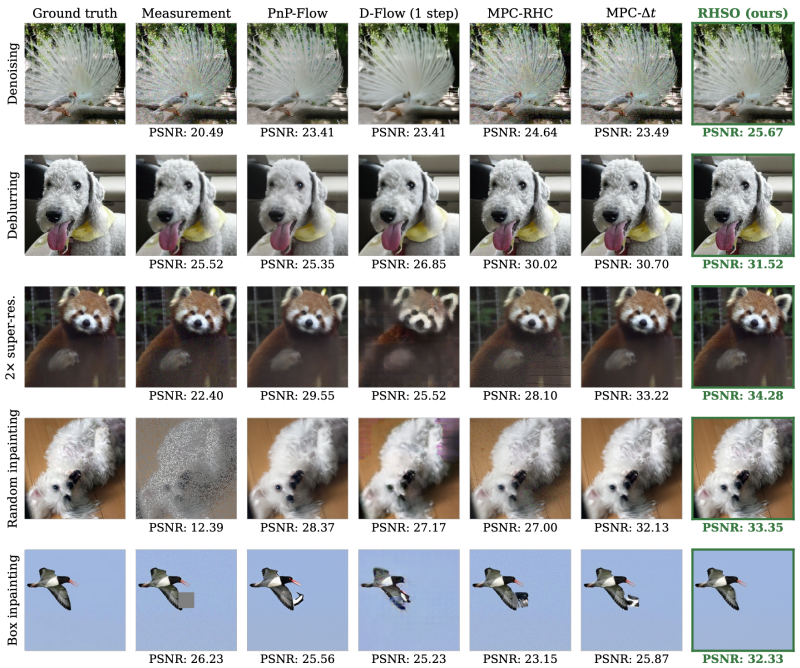

Figure 2: **Qualitative examples on pMF-L/16** (per-image PSNR below each panel; fixed structural default of Appendix H). PnP-Flow and D-Flow (1-step) over-smooth texture; MPC variants leave residual denoising noise; on the box-inpainting example RHSO reconstructs a coherent missing wing. Illustrative fixed examples (aggregates: Table 1); more in Appendix J.

4.2 Does RHSO generalize across priors and tasks?

Table 1 reports the full benchmark on the pixel-space MeanFlow prior pMF-L/16, and Table 2 the five-task means on all four priors (per-task tables: Appendix E). On pMF-L/16, RHSO has the best value in 10 of 15 task–metric cells and the best mean of every metric; across all four priors it is the best solver in 35 of 60 cells, wins 200 of 240 pairwise comparisons against the four baselines, and has the best five-task mean of every metric on every prior except iMF-B/2, where the closest baseline, MPC-Δt (38 of 60), is marginally better in PSNR and SSIM. RHSO buys this quality with compute: PnP-Flow and $K = 1$ MPC-RHC need no prior backpropagation and stay within a few tenths of a dB on the FM and latent priors (Appendix D); it also spends $2.3\times$ MPC-Δt's backpropagations on iMF-B/2 and 2–48$\times$ on the FM priors (Table 6). The interface also changes what solvers can do: one-terminal-prediction D-Flow collapses on the pixel FM prior (22.88 dB), where one Euler extrapolation from noise predicts the endpoint poorly, yet remains functional on the pixel MeanFlow prior (25.71 dB) at the same one-evaluation cost: the finite-interval map makes coarse steps usable (its multi-step form at RHSO's budget is the one-shot suffix control of Table 3). Figure 2 shows the behaviours behind the numbers.

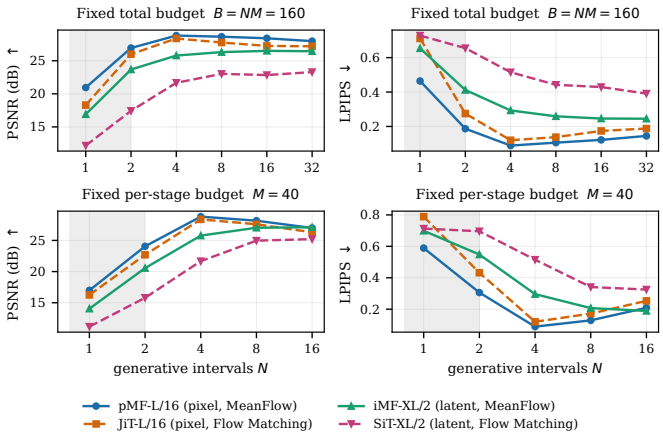

Figure 3: **Capacity- and cost-matched direct-terminal control.** Five-task means of direct-terminal RHSO on the two matched pairs at fixed total budget $B = NM = 160$ (top) and fixed per-stage budget $M = 40$ (bottom); shaded: $N \leq 2$.

4.3 Does the interface matter once capacity and cost are matched?

We now compare the two families at matched capacity: per space, two checkpoints of matching scale—pMF-L/16 (411M parameters) against JiT-L/16 (459M), iMF-XL/2 (610M) against SiT-XL/2 (675M)—run the same direct-terminal RHSO (Equations (10) and (11)), sweeping N under fixed optimizer work $B = NM = 160$ and fixed per-stage work $M = 40$. Cost is matched too (164/160 vs. 167/160 evaluations/backpropagations at $N = 4$, $M = 40$; an integrated suffix would need 647/640, Appendix A.2). Figure 3 gives the result; Table 12 (Appendix F) lists per-N differences.

The MeanFlow prior is better at every stage count in both spaces. Under both budgets and at every N, all three mean metrics favour the MeanFlow prior; per task–metric cell, pMF-L/16 is better in 153 of 165 (exceptions: scattered SSIM and box-inpainting PSNR cells, and denoising at $N \geq 16$ under fixed B, where it over-fits the noise) and iMF-XL/2 in all 165. The FM partner is the larger model in both pairs and still loses every mean comparison: the ordering is not a capacity effect.

Pixel space: the margin concentrates at small trajectory budgets. At fixed B it is $+2.65$ dB/$-.248$ LPIPS at $N = 1$ and $+0.92$/$-.089$ at $N = 2$ (98% and 93% per-pair LPIPS wins), narrowing from $N = 4$ on to $+0.4$–1.2 dB and $-.03$–$.05$ LPIPS (78% LPIPS, 56% PSNR wins at $N = 4$). Both pixel priors peak at $N = 4$ under both budgets and decline beyond it: the few-step advantage survives capacity matching at most of its size, but the ordering at every N is the stronger statement.

Latent space: the margin is per-stage optimizer efficiency, not asymptotic quality. Latent gaps are large everywhere ($+3.2$ to $+6.2$ dB at fixed B) but reflect optimizer efficiency, not a quality ceiling: at $N = 8$, $M = 40$, iMF-XL/2 reaches 27.03 dB, near the decoder-limited level (Table 2), whereas SiT-XL/2 reaches 24.96, and 26.92 in its benchmark configuration ($M = 80$, integrated-suffix planner; Appendix D), the level of the smaller iMF-B/2 at $M = 80$ (26.97). The finite-interval gradient thus reaches that regime with several times fewer updates per stage than the FM gradient; whether the matched latent priors share the same asymptotic quality is untested (no matched iMF-XL/2 run uses $M = 80$).

4.4 Why does receding-horizon optimization help?

Moving from $N = 1$ to $N = 4$ changes where optimization is applied and how deep the predicted trajectory is, so Table 3 adds a one-shot control that keeps all 160 updates at the source but differentiates each through the same four-interval Heun suffix: RHSO's depth without its intervention points.

Table 3: **Separating optimization placement from numerical suffix depth** (JiT-B/16, five-task means over 100 images, Eval./BP per image; default of Appendix H). Matching depth recovers little; redistributing the same 160 updates recovers most.

Variant	PSNR↑	SSIM↑	LPIPS↓	Eval./BP↓
One-shot, direct endpoint	18.18	.375	.713	161/160
One-shot, 4-step integrated suffix	20.84	.500	.551	1127/1120
RHSO, direct endpoint	28.47	.824	.122	167/160
RHSO, integrated suffix	28.79	.828	.121	647/640

Placement, not depth. Deepening the one-shot predictor adds $+2.67$ dB (paired intervals: Appendix G), so coarse endpoint prediction partly explains the weak source-only baseline. Redistributing the same 160 updates across four stages adds a further $+7.63$ dB; with replanning, a full suffix adds only $+0.32$ dB at four times the model-side cost. Re-optimizing intermediate states, not endpoint accuracy, dominates. This matches Section 3.3: one-shot prediction, however deep, exposes only J_0, whereas distributed optimization adds independent controls through different suffix geometries (Equation (14)). It is also why RHSO tolerates the shallow direct planner of Section 4.3: each execution replans from the reached state (Figure 1).

Control strength and measurement consistency. With $\mu = 0$, more stages relax the regularization of Section 3.3 but need not improve the image; the optimal stage count is model- and task-dependent. For both pixel priors at fixed $M = 40$, going from $N = 4$ to 16 lowers the denoising residual while PSNR falls by 1–2 dB (pMF: 29.45→27.49): the extra authority fits the noise in y. For both latent priors the same move lowers the residual *and* raises PSNR (iMF: 26.20→27.72), because the decoder cannot express pixel-level noise and filters what the fidelity gradient would inject (Figure 4, Appendix F). The trade-off under excessive control is thus a pixel-space phenomenon; it is why denoising is RHSO's weakest task (Tables 1 and 2) and why we report quality and measurement consistency separately: a lower residual need not mean a better image. Anchoring, the schedule β, initialization time and noise level move this balance as expected (Appendices H.1 to H.4).

5 Conclusion

Pretrained MeanFlow models are controllable priors for training-free inverse problems: a learned finite-interval map makes any jump along the trajectory, to the next stage or to the clean endpoint, a single shallow operation, so existing solver structures transfer to it with outer logic preserved, in pixel and latent space, and receding-horizon state optimization becomes natural. The evidence separates the contributions: RHSO is the strongest control policy evaluated (best in 35 of 60 task–metric cells and 200 of 240 pairwise comparisons), and a depth-controlled experiment attributes most of its gain to re-optimizing intermediate states; MeanFlow is the better interface for it, leading at every stage count in both spaces at matched capacity and cost, most in latent space and at small budgets.

Limitations. Large-scale MeanFlow generators are still few, so the comparison is limited to the ImageNet checkpoints that exist, on one dataset at one scale; it also covers only linear operators with Gaussian noise and the representative solver set of Section 2.2. Those priors all had to be conditioned on the true class, and class-agnostic reconstruction is untested. RHSO still needs backpropagations through the prior, at least NM per image, so it is slow next to solvers such as PnP-Flow that need none and are sometimes close. And because no MeanFlow prior shares an architecture, training recipe or guidance with a Flow-Matching one, the matched comparison is evidence about the control interface rather than an isolation of the training objective.

Outlook. Nothing in Equation (13) is specific to restoration: any differentiable constraint can replace the fidelity of Equation (5), making RHSO a general controller for pretrained generative models with semantic, counterfactual or safety objectives (Chung et al., 2024; Goyal et al., 2019; Augustin et al., 2022; Ahmed et al., 2026; Zhang et al., 2025). What the approach needs most is more of its priors: the results here are a case for training MeanFlow generators at larger scale and with text conditioning, which would also replace the oracle class label with a description of the target.

REPRODUCIBILITY STATEMENT

All code and configurations are provided as supplementary material with this submission: the implementation of every solver on all four priors, the resolved configuration of every reported run (the four-prior benchmark, the capacity-matched sweeps, the placement-depth control and the ablations), the identifiers of the evaluation images and of the tuning set, and the scripts that regenerate every table and figure of this paper from the logged results. Appendix C specifies the forward operators, fidelity normalizations, pairing and statistics; Appendix D lists the selected configuration of every benchmark cell and the pre-registered grid; Appendix F lists the matched-sweep settings; Appendix A gives the solver adaptations and the evaluation/backpropagation accounting; and Appendix B states all assumptions and proofs. ImageNet images and third-party checkpoints are not redistributed; their sources and licenses are recorded in Appendix C.

ETHICS STATEMENT

This work uses standard public restoration benchmarks and pretrained generative models and involves no human subjects. Generative priors can synthesize plausible content that is not uniquely supported by the observation, particularly in severely underdetermined settings; reconstructions from methods such as RHSO should not be treated as verified ground truth in high-stakes imaging without task-specific validation, uncertainty analysis and safeguards appropriate to the deployment context.

AI USE STATEMENT

We used generative AI tools (large language models) in three ways, in line with the ICLR 2027 policy on AI use. (i) Writing: AI tools helped write and edit the manuscript, including the presentation of the analysis in Section 3.3 and Appendix B, and polish the prose throughout. (ii) Mathematical claims: AI tools helped devise and check the proofs of Section 3.3 and Appendix B, including Theorem 1; every theorem, proposition, lemma and proof was subsequently verified independently by the authors, who are responsible for their correctness. (iii) Code: AI tools helped write and review the accompanying code used to run the experiments. The research questions, the RHSO method and the experimental design were conceived by the authors without AI assistance; all experiments and all reported numbers are the authors' own, no AI tool generated data, results or figures, and AI tools were not used to search for or select related work. The authors reviewed and verified all AI-assisted text, code and derivations and take full responsibility for the final content of this work.

REFERENCES

Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G. Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546, 2017.

Muhammad Asim, Mara Daniels, Oscar Leong, Ali Ahmed, and Paul Hand. Invertible generative models for inverse problems: Mitigating representation error and dataset bias. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 399–409, 2020.

Fabian Altekrüger, Alexander Denker, Paul Hagemann, Johannes Hertrich, Peter Maass, and Gabriele Steidl. PatchNR: Learning from very few images by patch normalizing flow regularization. *Inverse Problems*, 39(6):064006, 2023.

Xinyi Wei, Hans van Gorp, Lizeth Gonzalez-Carabarin, Daniel Freedman, Yonina C. Eldar, and Ruud J. G. van Sloun. Deep unfolding with normalizing flow priors for inverse problems. *IEEE Transactions on Signal Processing*, 70:2962–2971, 2022.

Giannis Daras, Hyungjin Chung, Chieh-Hsin Lai, Yuki Mitsufuji, Jong Chul Ye, Peyman Milanfar, Alexandros G. Dimakis, and Mauricio Delbracio. A survey on diffusion models for inverse problems. *arXiv preprint arXiv:2410.00083*, 2024.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K. Duvenaud. Neural ordinary differential equations. *Advances in Neural Information Processing Systems*, 31, 2018.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

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

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, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 12606–12633, 2024.

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*, 2024.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

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

Jeongsol Kim, Bryan Sangwoo Kim, and Jong Chul Ye. FlowDPS: Flow-driven posterior sampling for inverse problems. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12328–12337, 2025.

Ashwini Pokle, Matthew J. Muckley, Ricky T. Q. Chen, and Brian Karrer. Training-free linear image inverses via flows. *Transactions on Machine Learning Research*, 2024.

Yasi Zhang, Peiyu Yu, Yaxuan Zhu, Yingshan Chang, Feng Gao, Ying Nian Wu, and Oscar Leong. Flow priors for linear inverse problems via iterative corrupted trajectory matching. *Advances in Neural Information Processing Systems*, 37, 2024.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-Flow: Differentiating through flows for controlled generation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 3462–3483, 2024.

Ségolène Tiffany Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. PnP-Flow: Plug-and-play image restoration with flow matching. In *International Conference on Learning Representations*, 2025.

Mehrsa Pourya, Bassam El Rawas, and Michael Unser. Flower: A flow-matching solver for inverse problems. In *International Conference on Learning Representations*, 2026.

George Webber, Alexander Denker, Riccardo Barbano, and Andrew J. Reader. Solving inverse problems with flow-based models via model predictive control. In *International Conference on Machine Learning*, 2026.

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

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J. Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 30467–30476, 2026.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows. In *International Conference on Machine Learning*, 2026.

Amirmojtaba Sabour, Michael S. Albergo, Carles Domingo-Enrich, Nicholas M. Boffi, Sanja Fidler, Karsten Kreis, and Eric Vanden-Eijnden. Test-time scaling of diffusions with flow maps. *arXiv preprint arXiv:2511.22688*, 2025.

Jerry Y. Huang, Justin Lin, Sheel Shah, Kartik Nair, and Nicholas M. Boffi. How to guide your flow: Few-step alignment via flow map reward guidance. In *International Conference on Machine Learning*, 2026.

Abbas Mammadov, So Takao, Bohan Chen, Ricardo Baptista, Morteza Mardani, Yee Whye Teh, and Julius Berner. Variational flow maps: Make some noise for one-step conditional generation. *arXiv preprint arXiv:2603.07276*, 2026.

Xiao Shi, Edward P. Chandler, Chicago Y. Park, Shirin Shoushtari, and Ulugbek S. Kamilov. NullFlow: One-step generative reconstruction. *arXiv preprint arXiv:2606.22696*, 2026.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 36115–36125, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

David Q. Mayne, James B. Rawlings, Christopher V. Rao, and Pierre O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

S. V. Venkatakrishnan, Charles A. Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948, 2013.

Kevin Frans, Danijar Hafner, Sergey Levine, and Pieter Abbeel. One step diffusion via shortcut models. In *International Conference on Learning Representations*, 2025.

Zehua Jiang, Fenghao Zhu, Xinquan Wang, Chongwen Huang, and Zhaoyang Zhang. P-Flow: Proxy-gradient flows for linear inverse problems. *arXiv preprint arXiv:2605.08328*, 2026.

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

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, 2022.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023.

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

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Jiwoo Chung, Sangeek Hyun, and Jae-Pil Heo. Style injection in diffusion: A training-free approach for adapting large-scale diffusion models for style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8795–8805, 2024.

Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2376–2384, 2019.

Maximilian Augustin, Valentyn Boreiko, Francesco Croce, and Matthias Hein. Diffusion visual counterfactual explanations. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Hashir Ahmed, Zahra Khotanlou, Chenghao Tan, Ahmed Abdelaal, and Amir-Hossein Karimi. RecourseBench: A modular framework for reproducible algorithmic recourse evaluation. *arXiv preprint arXiv:2606.16113*, 2026.

Jichen Zhang, Liqun Zhao, Antonis Papachristodoulou, and Jack Umenberger. Constrained diffusers for safe planning and control. In *Advances in Neural Information Processing Systems*, 2025.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022.

Philip Hartman. *Ordinary Differential Equations*, volume 38 of *Classics in Applied Mathematics*. SIAM, Philadelphia, 2 edition, 2002.

Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2013.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Herm Jan Brascamp and Elliott H. Lieb. On extensions of the Brunn–Minkowski and Prékopa–Leindler theorems, including inequalities for log concave functions, and with an application to the diffusion equation. *Journal of Functional Analysis*, 22(4):366–389, 1976.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006.

Roger W. Brockett. *Finite Dimensional Linear Systems*. Wiley, New York, 1970.

A Additional methodological details

A.1 Solver adaptations

The main paper writes the generative transition available to a solver as $\mathcal{G}_\theta(x_t; t \to t')$. Standard FM obtains this transition by numerically integrating the instantaneous velocity field; MeanFlow evaluates the learned map in Equation (4). This substitution leaves the outer inverse-solver logic unchanged. For the latent priors every solver operates on the autoencoder code and evaluates the fidelity in pixel space after decoding, so the prior projection, terminal value or terminal prediction of each solver is composed with the decoder $\mathcal{D}$.

D-Flow (1-step) source-state optimization. Following the source-state objective of Ben-Hamu et al. (2024), our D-Flow (1-step) variant optimizes a single source state through one terminal predictor,

$$x^\star_{t_0} = \arg\min_x \Phi(\mathcal{G}_\theta(x; t_0 \to 1); y), \tag{15}$$

with M Adam updates at learning rate lr starting from the shared source state x_{t_0}, and returns $\mathcal{G}_\theta(x^\star_{t_0}; t_0 \to 1)$. For a MeanFlow prior the predictor is $T_\theta(\cdot; t_0 \to 1)$; for a standard-flow prior it is the single Euler extrapolation D_{t_0} of Equation (7). If one terminal prediction costs E_D generative-model evaluations, the model-side cost is $E_D(M+1)$ evaluations and $E_D M$ backpropagations, with $E_D = 1$ in both cases. This is a one-terminal-prediction adaptation of the D-Flow objective, not a reproduction of the original multi-step D-Flow solver or its reported hyperparameters; the depth-matched suffix control of Section 4.4 integrates four Heun intervals before evaluating the terminal loss.

PnP-Flow. Following Martin et al. (2025), PnP-Flow alternates a fidelity gradient step, a stochastic reprojection onto the flow path and a prior projection, and never backpropagates through the generative model. With $\tau_{\mathrm{init}} = 0$ the N correction times are

$$t_k = \Big(\frac{k}{N+1}\Big)^\beta, \qquad k = 1, \dots, N, \tag{16}$$

strictly inside $(0,1)$ ($\beta = 1$ throughout), with step sizes $\gamma_k = \gamma_0(1-t_k)^\alpha$. The iterate is initialized by one prior projection of the shared source state, $x^{(0)} = \mathcal{G}_\theta(x_{t_0}; t_0 \to 1)$, and iteration k performs

$$z_k = x^{(k-1)} - \gamma_k \nabla_x \Phi\big(x^{(k-1)}; y\big), \qquad q_k = t_k\, z_k + (1-t_k)\,\varepsilon_k,\ \ \varepsilon_k \sim \mathcal{N}(0, I), \qquad x^{(k)} = \mathcal{G}_\theta(q_k; t_k \to 1), \tag{17}$$

so the prior projection is D_{t_k} for a standard-flow prior and $T_\theta(\cdot; t_k \to 1)$ for a MeanFlow prior, with one noise realization per iteration (the original averages several). The reconstruction is $x^{(N)}$; the method uses $N+1$ prior evaluations and no backpropagation through the prior. On the latent priors the fidelity update of PnP-Flow, like the objective of D-Flow (1-step) and of every other solver, acts in code space: the current code is decoded by the frozen autoencoder, the forward operator is applied to the decoded image, the residual against the observation is formed in pixel space, and its gradient is propagated through the decoder back to the code; z_k, ε_k and q_k live in code space. No pixel-space update or re-encoding takes place, so each fidelity step costs one decoder evaluation and one decoder backpropagation, which the Eval./BP counts omit (Appendix A.2).

MPC-Flow. MPC-RHC and MPC-Δt optimize explicit controls rather than the generative state (Webber et al., 2026), following the receding-horizon principle standard in model-predictive control (Mayne et al., 2000). Both run on the outer grid $t_k = k/N$ with $\Delta = 1/N$, initialize the controls at zero at every stage and optimize them with C Adam updates at learning rate lr, and use the inherited sum-normalized fidelity $\tilde{\Phi}(x; y) = \frac{1}{2}\,\|A(x) - y\|_2^2$ and control cost $\|u\|_2^2$ (Appendix C).

MPC-RHC plans over the remaining horizon $r = 1 - t_k$ in K equal intervals of length $h = r/K$. For a standard-flow prior the nominal velocity $v_0 = v_\theta(x, t_k)$ is held fixed and the rollout is

$$x^{(1)} = x + h\,(v_0 + u_0), \qquad x^{(j+1)} = x^{(j)} + h\big[v_\theta\big(x^{(j)}, t_k + jh\big) + u_j\big], \quad j = 1, \dots, K-1; \tag{18}$$

for a MeanFlow prior every Euler step is replaced by the learned transition, $x^{(1)} = T_\theta(x; t_k \to t_k + h) + h\,u_0$ and $x^{(j+1)} = T_\theta\big(x^{(j)}; t_k + jh \to t_k + (j+1)h\big) + h\,u_j$. The stage objective is

$$J(u_0, \dots, u_{K-1}) = h \sum_{j<K} \|u_j\|_2^2 + \lambda\, \tilde{\Phi}\big(x^{(K)}; y\big), \tag{19}$$

and only the first control is executed, $x \leftarrow x + \Delta\,(v_0 + u_0^\star)$ or $x \leftarrow T_\theta(x; t_k \to t_{k+1}) + \Delta\, u_0^\star$. For $K = 1$ the terminal state is a fixed nominal endpoint, $x + r\, v_0$ or $T_\theta(x; t_k \to 1)$, plus $r\, u_0$, so the prior is never differentiated.

MPC-Δt plans one interval and prices its outcome by the terminal value of the state reached: with $x'(u) = x + \Delta\,(v_0 + u)$ for a standard-flow prior, or $x'(u) = \bar{x} + \Delta\, u$ with the fixed nominal transition $\bar{x} = T_\theta(x; t_k \to t_{k+1})$ for a MeanFlow prior,

$$J(u) = \|u\|_2^2 + \lambda\, \hat{V}\big(t_{k+1}, x'(u)\big), \qquad \hat{V}(t, x) = \tilde{\Phi}\big(\mathcal{G}_\theta(x; t \to 1); y\big), \tag{20}$$

i.e. $\hat{V}$ uses $D_{t_{k+1}}$ for a standard-flow prior and $T_\theta(\cdot; t_{k+1} \to 1)$ for a MeanFlow prior, and the stage executes $x \leftarrow x'(u^\star)$; λ enters verbatim, without a $1/\Delta$ factor. At the last stage $t_{k+1} = 1$, $\hat{V} = \tilde{\Phi}(x')$ and the prior is not differentiated.

Per stage, MPC-RHC costs $1 + C(K-1)$ prior evaluations and $C(K-1)$ backpropagations for a standard-flow prior, and one evaluation more for a MeanFlow prior, whose executed transition $T_\theta(x; t_k \to t_{k+1})$ is not one of the planning transitions; MPC-Δt costs $1 + C$ evaluations and C backpropagations per stage, and 1 and 0 at the last stage. These counts reproduce those of Table 6. Selected horizons and control budgets are listed in Appendix D.

Prior sample (no measurement). The Prior sample rows of the per-task tables are a class-conditional sample from pure prior noise with the selected number of sampling steps and no measurement information; they are identical across tasks by construction and serve as the no-measurement floor. A partial-noising SDEdit pass (Meng et al., 2022) would inject measurement content through its initialization; with $\tau_{\text{init}} = 0$ it degenerates to this floor.

A.2 RHSO IMPLEMENTATION AND COSTS

At every outer stage, the optimized state is initialized from the state actually reached and a fresh Adam state is created. Optimizer momentum and computation graphs are not propagated between stages. The anchor in Equation (13) uses a stop-gradient copy of the reached state and remains fixed during all M inner updates.

For a MeanFlow prior, N stages and M inner updates require

$$C_{\text{eval}}^{\text{MF}} = N(M+1), \qquad C_{\text{bp}}^{\text{MF}} = NM. \tag{21}$$

For the direct standard-flow planner, there are NM one-step terminal predictions plus the executed Heun intervals with an Euler final step ($2N - 1$ evaluations; $2N$ for the SiT sampler, which keeps a Heun step on the final interval). At $N = 4, M = 40$, this gives 167/160 Eval./BP on JiT. For the full-suffix planner, a suffix of j intervals uses $2j - 1$ evaluations, so

$$C_{\text{plan}}^{\text{suffix}} = M \sum_{j=1}^{N} (2j - 1) = MN^2, \qquad C_{\text{exec}}^{\text{suffix}} = 2N - 1. \tag{22}$$

The $N = 4, M = 40$ configuration therefore uses 647/640 Eval./BP on JiT; on SiT every suffix and the executed path cost one evaluation more, so a suffix of j intervals costs $2j$ evaluations and the benchmark's $N = 8$, $M = 80$ configuration on SiT-XL/2 uses 5776/5760 (Table 6). The four-step one-shot suffix uses the same seven-evaluation terminal graph for every one of 160 source-state updates, giving 1127/1120 Eval./BP after the final generation pass.

Collecting these at $N = 4, M = 40$: every prior performs the same $NM = 160$ optimizer updates, and only the *depth* of each terminal gradient differs. The MeanFlow prior uses 164 evaluations and 160 backpropagations per image and the direct standard-flow planner 167/160, whereas the full-suffix planner differentiates the remaining $4, 3, 2, 1$ intervals and uses 647/640. This is the concrete sense in which a finite-interval map makes each terminal gradient shallow and independent of the remaining horizon: it does not remove the optimization, only the growth of its cost with the look-ahead.

For a latent prior, every terminal prediction additionally costs one decoder evaluation (and, wherever the prior is differentiated, one decoder backpropagation), i.e. $NM + 1$ decodes for RHSO; the decoder is shared by both priors of a matched pair, so this term is identical across the matched comparison and is omitted from the Eval./BP counts.

B THEORY DETAILS AND PROOFS

This appendix states and proves the first-order theory summarized in Section 3.3. Throughout, $X_t = (1-t)X_0 + tX_1$ is the random interpolant with $X_0 \sim \mathcal{N}(0, I)$ independent of $X_1 \sim p_{\mathrm{data}}$, $v_t(x) = \mathbb{E}[X_1 - X_0 \mid X_t = x]$ is the exact marginal velocity that v_θ approximates, $\Psi_{t\to s}$ its flow, and $x_t = \Psi_{0\to t}(x_0)$ the nominal trajectory of a fixed source sample. For $M \succ 0$, $\mathrm{cond}(M) = \lambda_{\max}(M)/\lambda_{\min}(M)$; $\varsigma_{\max}, \varsigma_{\min}$ denote extreme singular values. A MeanFlow model and an FM model that fit the marginal dynamics exactly realize the same $\Psi_{t\to s}$, so nothing below distinguishes the two families.

B.1 SETTING

Assumption 1 (Regular transport)**.** (i) $\mathbb{E}\,\|X_1\|^2 < \infty$. (ii) v is continuous on $[0,1) \times \mathbb{R}^d$ and C^1 in x with $D_x v$ continuous there, and $\Psi_{t\to s}$ exists for $0 \le t \le s < 1$. (iii) For every $t \in [0,1)$, $\Psi_{t\to s}$ and $D_x\Psi_{t\to s}$ converge as $s \uparrow 1$, locally uniformly in x; the limit $\Psi_{t\to 1}$ is then C^1, and $\Psi_{1\to 1} := \mathrm{id}$.

Nothing in (iii) is assumed jointly in the two times at the endpoint. Under (ii), for $0 \le t \le s < 1$ the map $(t, s, x) \mapsto (\Psi_{t\to s}(x), D_x\Psi_{t\to s}(x))$ is jointly continuous and the Jacobian solves the variational equation (Hartman, 2002, Ch. V); along the nominal trajectory, with $J_{t\to s} := D_x\Psi_{t\to s}(x_t)$ and $J_t := J_{t\to 1}$,

$$\partial_s J_{t\to s} = A_s J_{t\to s}, \qquad J_{t\to t} = I, \qquad A_s := D_x v_s(x_s) \quad (0 \le s < 1), \tag{23}$$

and the chain rule gives $J_{t\to s'} = J_{s\to s'}\, J_{t\to s}$ for $t \le s \le s' \le 1$, the case $s' = 1$ by Assumption 1(iii). Liouville's formula, $\det J_{t\to s} = \exp\int_t^s \mathrm{tr}\, A_r\, dr$, makes $J_{t\to s}$ invertible for $s < 1$; invertibility of J_t itself, and any continuity of $t \mapsto J_t$ at $t = 1$, are *not* implied by Assumption 1 (Remark 5) and are proved under (26) below. At the source, $v_0(x) = \mathbb{E}[X_1] - x$ and $A_0 = -I$ exactly.

B.2 SUPERPOSITION, ALLOCATION AND CONTRAST REDUCTION

Proposition 2 (Superposition and least-effort correction)**.** *Under Assumption 1, let corrections $\Delta^{(0)}, \dots, \Delta^{(N-1)}$ be applied to the state at stages $0 \le t_0 < \cdots < t_{N-1} < 1$, each before the transport interval that follows it, and let z_1 be the endpoint reached. Then $z_1 = x_1 + \sum_k J_{t_k}\Delta^{(k)} + o(\|\Delta\|)$. With $W_{\mathcal{T}} = \sum_k J_{t_k} J_{t_k}^\top$ and $\mathcal{R}_{\mathcal{T}} = \mathrm{range}(W_{\mathcal{T}}) = \sum_k \mathrm{range}(J_{t_k})$, for $\delta \in \mathcal{R}_{\mathcal{T}}$ the least-norm corrections with $\sum_k J_{t_k}\Delta^{(k)} = \delta$ are $\Delta^{(k)} = J_{t_k}^\top W_{\mathcal{T}}^+ \delta$, with cost $\mathcal{C}_{\mathcal{T}}^\star(\delta) = \delta^\top W_{\mathcal{T}}^+ \delta$; for $\delta \notin \mathcal{R}_{\mathcal{T}}$ no first-order combination reaches δ. Under* (26)*, $W_{\mathcal{T}} \succ 0$ and $W^+ = W^{-1}$, which is* (14)*.*

Proof. Define $z_{t_0} = x_{t_0} + \Delta^{(0)}$, $z_{t_{k+1}} = \Psi_{t_k\to t_{k+1}}(z_{t_k}) + \Delta^{(k+1)}$ and $z_1 = \Psi_{t_{N-1}\to 1}(z_{t_{N-1}})$. The map $F(\Delta) := z_1$ is a composition of C^1 maps (Assumption 1(ii),(iii)), so it is C^1 with $F(0) = x_1$, and its partial differential in $\Delta^{(k)}$ at 0 is the differential of $\Delta^{(k)} \mapsto \Psi_{t_k\to 1}(x_{t_k} + \Delta^{(k)})$, namely J_{t_k}, by the composition property of flows. For the allocation, stack $\mathcal{J} = [J_{t_0} \cdots J_{t_{N-1}}]$, so $W_{\mathcal{T}} = \mathcal{J}\mathcal{J}^\top$ and $\mathrm{range}(\mathcal{J}) = \mathcal{R}_{\mathcal{T}}$; $\Delta^\star = \mathcal{J}^\top W^+\delta$ satisfies $\mathcal{J}\Delta^\star = WW^+\delta = \delta$, every feasible Δ is $\Delta^\star + n$ with $n \perp \mathrm{range}(\mathcal{J}^\top) \ni \Delta^\star$, so $\|\Delta\|^2 = \|\Delta^\star\|^2 + \|n\|^2$, and $\|\Delta^\star\|^2 = \delta^\top W^+ W W^+ \delta = \delta^\top W^+ \delta$. □

Proof of Theorem 1. Weyl's inequalities (Horn and Johnson, 2013, Sec. 4.3) give $\lambda_{\max}(W+G) \le \lambda_{\max}(W) + \lambda_{\max}(G)$ and $\lambda_{\min}(W+G) \ge \lambda_{\min}(W) + \lambda_{\min}(G) > 0$, so $\mathrm{cond}(W+G) \le (a+a')/(b+b')$ with $a = \lambda_{\max}(W)$, $b = \lambda_{\min}(W)$, $a' = \lambda_{\max}(G)$, $b' = \lambda_{\min}(G)$; and for positive numbers, $(a+a')/(b+b') < a/b \iff a'b < ab' \iff a'/b' < a/b$, which is the hypothesis. The same bound with several $G_j \succ 0$, each with $\mathrm{cond}(G_j) < \mathrm{cond}(W)$, is a weighted average with positive weights of $\mathrm{cond}(W)$ and the $\mathrm{cond}(G_j)$, hence strictly below $\mathrm{cond}(W)$. The second statement is the case $W = W_{\mathcal{T}}$, $G = J_{t'}J_{t'}^\top$, both positive definite under (26). If W is singular, the argument applies to the compressions to $\mathrm{range}(W)$ whenever $\mathrm{range}(G) \subseteq \mathrm{range}(W)$; if not, adding G enlarges $\mathcal{R}_{\mathcal{T}}$, an improvement the condition number does not measure. □

B.3 THE FLOW-MATCHING TRANSPORT AND LATE-STAGE ISOTROPY

Lemma 3 (Tweedie form of the velocity, cf. Efron, 2011)**.** *For every probability measure p_{data} and $\sigma > 0$, the density p_σ of $X_1 + \sigma Z$, $Z \sim \mathcal{N}(0, I)$, is C^∞ and positive, and with $Y = X_1 + \sigma Z$,*

$$\nabla \log p_\sigma(y) = \frac{\mathbb{E}[X_1 \mid Y = y] - y}{\sigma^2}, \qquad \nabla^2 \log p_\sigma(y) = \frac{\mathrm{Cov}(X_1 \mid Y = y)}{\sigma^4} - \frac{I}{\sigma^2}. \tag{24}$$

Consequently, for $t \in (0, 1)$ and $\sigma_t := (1 - t)/t$,

$$v_t(x) = \frac{x}{t} + \frac{1-t}{t^2} \nabla \log p_{\sigma_t}(x/t), \qquad A_s = \frac{1}{s} I + \frac{1-s}{s^3} \mathsf{H}_s, \qquad \mathsf{H}_s := \nabla^2 \log p_{\sigma_s}(x_s/s), \tag{25}$$

and equivalently $A_t = -\frac{1}{1-t} I + \frac{t}{(1-t)^3} \mathrm{Cov}(X_1 \mid X_t = x_t)$, which has no $1/t$ and extends the value $A_0 = -I$.

Proof. With φ_σ the Gaussian density, $p_\sigma(y) = \int \varphi_\sigma(y - x) p_{\mathrm{data}}(dx)$ is C^∞, positive, and differentiable under the integral because the Gaussian factor and its derivatives are dominated by integrable functions locally uniformly in y; $\nabla_y \varphi_\sigma(y - x) = \varphi_\sigma(y - x)(x - y)/\sigma^2$ gives the first identity, and differentiating $m(y) = \mathbb{E}[X_1 \mid Y = y]$ the same way gives $D_y m = \mathrm{Cov}(X_1 \mid Y = y)/\sigma^2$, hence the second (the posterior has a Gaussian factor, so its moments exist without any moment assumption on X_1). Conditioning on $X_t = x$, $X_1 - X_0 = (X_1 - x)/(1 - t)$, so $v_t(x) = (\mathbb{E}[X_1 \mid X_t = x] - x)/(1 - t)$; since $X_t/t = X_1 + \sigma_t X_0$, the first identity at $y = x/t$ gives the velocity formula, and $D_x[\nabla \log p_\sigma(x/t)] = \frac{1}{t} \nabla^2 \log p_\sigma(x/t)$ gives the generator. Inserting the second identity and $(1-t)/(t^3 \sigma_t^2) = 1/(t(1-t))$, $(1-t)/(t^3 \sigma_t^4) = t/(1-t)^3$ yields the covariance form. □

For $p_{\mathrm{data}} = \mathcal{N}(0, \Sigma)$, $\nabla \log p_\sigma(y) = -(\Sigma + \sigma^2 I)^{-1} y$ and (25) reduces to $v_t(x) = (t\Sigma - (1 - t)I) C_t^{-1} x =: M_t x$ with $C_t = t^2 \Sigma + (1 - t)^2 I$, the exact Gaussian velocity. The scalar part of A_s does not affect conditioning: for $0 < t < 1$ and any measurable h with $\int_t^1 \frac{1-s}{s^3} |h| < \infty$, writing $A_s = a(s) I + \widetilde{A}_s$ with $a(s) = \frac{1}{s} + \frac{1-s}{s^3} h(s)$ and $\widetilde{A}_s = \frac{1-s}{s^3} (\mathsf{H}_s - h(s) I)$, the solution of the reduced equation $\partial_s \widetilde{J}_{t \to s} = \widetilde{A}_s \widetilde{J}_{t \to s}$ satisfies $J_{t \to s} = e^{\int_t^s a} \widetilde{J}_{t \to s}$, because $a(s) I$ commutes with every matrix; the midpoint $h(s) = \frac{1}{2}(\lambda_{\max}(\mathsf{H}_s) + \lambda_{\min}(\mathsf{H}_s))$ minimizes $\|\mathsf{H}_s - hI\|$ and gives $\left\| \widetilde{A}_s \right\| = \frac{1-s}{s^3} \cdot \frac{1}{2}(\lambda_{\max}(\mathsf{H}_s) - \lambda_{\min}(\mathsf{H}_s))$.

Proposition 4 (Late-stage isotropy)**.** *Suppose Assumption 1 holds and, for some $t_0 \in (0, 1)$,*

$$\mathcal{L} := \sup_{s \in [t_0, 1)} \|\mathsf{H}_s\| < \infty, \qquad \mathcal{H} := \sup_{s \in [t_0, 1)} \tfrac{1}{2} \big(\lambda_{\max}(\mathsf{H}_s) - \lambda_{\min}(\mathsf{H}_s) \big) \le \mathcal{L}. \tag{26}$$

Then every J_t, $t \in [0, 1]$, is invertible; for $t \in [t_0, 1)$, $J_t = \alpha_t (I + E_t)$ with $\alpha_t > 0$ and $\|E_t\| \le \varepsilon_t := \exp\big(\mathcal{H}(1 - t)^2/(2t^2)\big) - 1$, so that, whenever $\varepsilon_t < 1$,

$$\mathrm{cond}\left(J_t J_t^\top\right) \le \left(\frac{1 + \varepsilon_t}{1 - \varepsilon_t} \right)^2 \xrightarrow[t \uparrow 1]{} 1; \tag{27}$$

moreover $\|J_s - I\| \le e^{\kappa_0 (1 - s)} - 1$ with $\kappa_0 = 1/t_0 + \mathcal{L}(1 - t_0)/t_0^3$, so $J_s \to I$ and $s \mapsto J_s$ is continuous on $[0, 1]$. Consequently, for every stage set with $\chi(\mathcal{T}) > 1$ and $\rho = (\sqrt{\chi(\mathcal{T})} - 1)/(\sqrt{\chi(\mathcal{T})} + 1)$, every $t' \in [t_0, 1)$ with $\varepsilon_{t'} < \rho$ satisfies the hypothesis of Theorem 1; such t' fill an interval $(t^\star, 1)$.

Proof. On $[t_0, 1)$, $\|A_s\| \le 1/s + \frac{1-s}{s^3} \mathcal{L} \le \kappa_0$, so $\int_{t_0}^1 \mathrm{tr}\, A_s \, ds$ is finite and, by Liouville's formula and Assumption 1(iii), $\det J_t = \lim_{s \uparrow 1} \det J_{t \to s} > 0$ for $t \in [t_0, 1)$; for $t < t_0$, $J_t = J_{t_0} J_{t \to t_0}$ is a product of invertible matrices, and $J_1 = I$. Fix $t \in [t_0, 1)$ and use the midpoint reduction above, whose integrability conditions hold because $|h| \le \mathcal{L}$: with $\beta(r) = \left\| \widetilde{A}_r \right\| \le \frac{1-r}{r^3} \mathcal{H}$, the integral form $\widetilde{J}_{t \to s} = I + \int_t^s \widetilde{A}_r \widetilde{J}_{t \to r} \, dr$ and Grönwall's inequality give $\left\| \widetilde{J}_{t \to s} \right\| \le e^{\int_t^s \beta}$ and then $\left\| \widetilde{J}_{t \to 1} - I \right\| \le \int_t^1 \beta(r) e^{\int_t^r \beta} dr = e^{\int_t^1 \beta} - 1$; since $\int_t^1 \frac{1-s}{s^3} ds = \frac{(1-t)^2}{2t^2}$ for $t > 0$, this is ε_t. Take

$\alpha_t = e^{\int_t^1 a}$ and $E_t = \widetilde{J}_{t\to 1} - I$; if $\varepsilon_t < 1$ the singular values of $I + E_t$ lie in $[1-\varepsilon_t, 1+\varepsilon_t]$, and α_t cancels in the condition number. The same Grönwall argument on the full equation gives $\|J_{s\to r} - I\| \le e^{\kappa_0(r-s)} - 1$ for $t_0 \le s \le r < 1$, hence $\|J_s - I\| \le e^{\kappa_0(1-s)} - 1$ as $r \uparrow 1$; continuity of $s \mapsto J_s$ follows from $J_s - J_{s'} = J_{s'}(J_{s\to s'} - I)$ on $[t_0, 1)$, from $J_s = J_{t_0} J_{s\to t_0}$ and joint continuity on $[0, t_0]$, and from $J_s \to I = J_1$. Finally $(\frac{1+\varepsilon}{1-\varepsilon})^2 < \chi \iff \varepsilon < \rho$; if $\mathcal{H} = 0$ then $\varepsilon_{t'} = 0$ for all $t' \in [t_0, 1)$, and if $\mathcal{H} > 0$ then ε_t decreases strictly to 0, so the qualifying t' are exactly $\{t' \in [t_0, 1) : (1-t')^2/t'^2 < 2\log(1+\rho)/\mathcal{H}\}$. □

Remark 5 (Assumption 1 alone does not control the endpoint)**.** Let $X_1 = 0$. Then $v_t(x) = -x/(1-t)$ and $\Psi_{t\to s}(x) = \frac{1-s}{1-t}x$, so Assumption 1 holds, yet $J_t = 0$ for every $t < 1$ while $J_1 = I$: $W_{\mathcal{T}} = 0$ and $J_t \not\to I$. Here $\mathsf{H}_s = -\sigma_s^{-2} I$ has norm $s^2/(1-s)^2 \to \infty$, so (26) fails, as it must. This is the mechanism by which distributions supported on sufficiently regular lower-dimensional sets, for which Gaussian smoothing typically produces normal-direction curvature of order σ^{-2}, are excluded.

Lemma 6 (Sufficient conditions for (26))**.** *(a) If p_{data} has a strictly positive C^2 density p with $\|\nabla^2 \log p\| \le L$ everywhere, then $\|\nabla^2 \log p_\sigma\| \le 2L$ for $\sigma^2 \le 1/(2L)$, so* (26) *holds along every nominal trajectory with $\mathcal{L} \le 2L$ and any $t_0 \ge (1 + (2L)^{-1/2})^{-1}$. (b) For $p_{\mathrm{data}} = \mathcal{N}(0, \Sigma)$ with eigen-standard-deviations $s_{\min} \le s_i \le s_{\max}$, $\mathsf{H}_s = -(\Sigma + \sigma_s^2 I)^{-1}$, $\|\mathsf{H}_s\| \le s_{\min}^{-2}$ and $\mathcal{H} = \frac{1}{2}(s_{\min}^{-2} - s_{\max}^{-2})$ for every $t_0 \in (0,1)$. (c) For the mixture $\frac{1}{2}\mathcal{N}(me_1, \varrho^2 I) + \frac{1}{2}\mathcal{N}(-me_1, \varrho^2 I)$ in $\mathbb{R}^d$, $d \ge 2$, with $\nu = \varrho^2 + \sigma^2$, $\nabla^2 \log p_\sigma(y) = -\nu^{-1} I + m^2 \nu^{-2} \operatorname{sech}^2(my_1/\nu) e_1 e_1^\top$, so* (26) *holds with $\mathcal{H} \le m^2/(2\varrho^4)$, and the Hessians share a fixed eigenbasis with a fixed order.*

Proof. (a) The posterior $q_y(x) \propto p(x)\varphi_\sigma(y - x)$ has $(\sigma^{-2} - L)I \preceq -\nabla^2 \log q_y \preceq (\sigma^{-2} + L)I$, Gaussian tails and a Lipschitz score. For $\sigma^2 < 1/L$ the Brascamp–Lieb inequality (Brascamp and Lieb, 1976) gives $\mathrm{Cov}(q_y) \preceq (\sigma^{-2} - L)^{-1} I$, and the Cramér–Rao inequality for the location family generated by q_y (Cover and Thomas, 2006, Ch. 11) gives $\mathrm{Cov}(q_y) \succeq (\sigma^{-2} + L)^{-1} I$; inserting both into (24) yields $-\frac{L}{1+L\sigma^2} I \preceq \nabla^2 \log p_\sigma(y) \preceq \frac{L}{1-L\sigma^2} I$, and $L\sigma^2 \le \frac{1}{2}$ bounds the right side by $2L$. Since $\sigma_t^2 = (1-t)^2/t^2$ decreases in t, $\sigma_t^2 \le 1/(2L)$ for $t \ge (1 + (2L)^{-1/2})^{-1}$. (b) $p_\sigma = \mathcal{N}(0, \Sigma + \sigma^2 I)$; the spread $\frac{1}{s_{\min}^2 + \sigma^2} - \frac{1}{s_{\max}^2 + \sigma^2}$ decreases in σ^2, so its supremum is its $\sigma \downarrow 0$ limit. (c) $\log p_\sigma(y) = \mathrm{const} - \frac{\|y\|^2 + m^2}{2\nu} + \log\cosh\frac{my_1}{\nu}$; differentiate twice and use $\nu \ge \varrho^2$, $\operatorname{sech}^2 \le 1$. □

B.4 Commuting curvature and the Gaussian case

Proposition 7 (Commuting generators with a time-consistent order)**.** *Suppose Assumption 1 and* (26) *hold and $\mathsf{H}_s = Q\,\mathrm{diag}(\eta_1(s), \dots, \eta_d(s))Q^\top$ for all $s \in (0,1)$, with Q orthogonal, η_i continuous, and $\eta_1 \ge \eta_i \ge \eta_d$ pointwise. With $a_i(s) = \frac{1}{s} + \frac{1-s}{s^3}\eta_i(s)$, extended by $a_i(0) = -1$, and $\gamma_i(t) = \exp\int_t^1 a_i$, one has $J_t = Q\,\mathrm{diag}(\gamma_i(t))Q^\top$, $\gamma_1 \ge \gamma_i \ge \gamma_d > 0$, and $\chi(\{t\}) = (\gamma_1(t)/\gamma_d(t))^2 = \exp\big(2\int_t^1 (a_1 - a_d)\big)$ is nonincreasing in t, strictly decreasing if $\eta_1 > \eta_d$ almost everywhere; in that case $\chi(\mathcal{T} \cup \{t'\}) < \chi(\mathcal{T})$ for every $\mathcal{T} \subset [0,1)$ and $t' > \max \mathcal{T}$.*

Proof. $A_s = Q\,\mathrm{diag}(a_i(s))Q^\top$ for $s \in (0,1)$ and $A_0 = -I$, so the variational equation decouples in the basis Q and $J_{t\to s} = Q\,\mathrm{diag}(e^{\int_t^s a_i})Q^\top$; each a_i is continuous on $[0, t_0]$ and bounded on $[t_0, 1)$, so the limit $s \uparrow 1$ exists. The ordering of the γ_i follows from that of the a_i, and $a_1 - a_d = \frac{1-s}{s^3}(\eta_1 - \eta_d) \ge 0$ makes $\int_t^1 (a_1 - a_d)$ nonincreasing in t, strictly decreasing if the integrand is positive almost everywhere. Then $W_{\mathcal{T}} = Q\,\mathrm{diag}(\sum_k \gamma_i(t_k)^2)Q^\top$ and $\chi(\mathcal{T}) = \sum_k \gamma_1(t_k)^2 / \sum_k \gamma_d(t_k)^2$ is a weighted average of the $\chi(\{t_k\})$ with positive weights, hence at least $\chi(\{\max \mathcal{T}\}) > \chi(\{t'\}) = \mathrm{cond}(J_{t'} J_{t'}^\top)$, and Theorem 1 applies. □

Corollary 8 (Gaussian data)**.** *Let $p_{\mathrm{data}} = \mathcal{N}(0, \Sigma)$, $\Sigma = Q\,\mathrm{diag}(s_i^2)Q^\top \succ 0$, $s_{\max} > s_{\min}$. Then $J_t = Q\,\mathrm{diag}(g_i(t))Q^\top$ with $g_i(t) = s_i/\sqrt{t^2 s_i^2 + (1-t)^2}$, $W_{\mathcal{T}} = Q\,\mathrm{diag}(\sum_k g_i(t_k)^2)Q^\top$, $\mathcal{C}_{\mathcal{T}}^\star(\delta) = \sum_i \tilde{\delta}_i^2 / \sum_k g_i(t_k)^2$ for $\tilde{\delta} = Q^\top \delta$, and $\chi(\{t\}) = (g_{\max}(t)/g_{\min}(t))^2$ decreases strictly from $\chi(\{0\}) = \kappa(\Sigma)$ to 1, so every later stage reduces the contrast. On the grids $\mathcal{T}_N = \{k/N\}_{k<N}$, $\chi(\mathcal{T}_N) \to \sqrt{\kappa(\Sigma)}$, and* (27) *is sharp at leading order: both $\chi(\{t\})$ and the bound equal $1 + (1-t)^2(s_{\min}^{-2} - s_{\max}^{-2}) + O((1-t)^3)$.*

Proof. $v_t(x) = M_t x$ is linear with M_t continuous on $[0,1)$ and $\Psi_{t\to s} = C_s^{1/2} C_t^{-1/2} \to \Sigma^{1/2} C_t^{-1/2}$, so Assumption 1 holds, and (26) is Lemma 6(b). Proposition 7 applies with $\eta_i(s) = -s^2/c_i(s)$, $c_i(s) = s^2 s_i^2 + (1-s)^2$, ordered by s_i at every s; then $a_i(s) = (s s_i^2 - (1-s))/c_i(s) = \dot{c}_i/(2c_i)$, so $\gamma_i(t) = s_i/\sqrt{c_i(t)} = g_i(t)$, and $\eta_1 - \eta_d = (s_{\min}^2 + \sigma_s^2)^{-1} - (s_{\max}^2 + \sigma_s^2)^{-1} > 0$ gives strict monotonicity; the endpoint values are $g_i(0) = s_i$ and $g_i(1) = 1$. For the grids, completing the square in $(1+s_i^2)t^2 - 2t + 1$ gives $\int_0^1 g_i^2\, dt = s_i[\arctan s_i + \arctan(1/s_i)] = \frac{\pi}{2} s_i$, so the Riemann sums give $\frac{1}{N} W_{\mathcal{T}_N} \to \frac{\pi}{2}\Sigma^{1/2}$ and, by continuity of cond, $\chi(\mathcal{T}_N) \to s_{\max}/s_{\min}$. For sharpness put $\epsilon = 1 - t$: $\chi(\{t\}) = \frac{1 - 2\epsilon + \epsilon^2(1 + s_{\min}^{-2})}{1 - 2\epsilon + \epsilon^2(1 + s_{\max}^{-2})} = 1 + \epsilon^2(s_{\min}^{-2} - s_{\max}^{-2}) + O(\epsilon^3)$, while $\varepsilon_t = \frac{1}{2}\mathcal{H}\epsilon^2 + O(\epsilon^3)$ and $(\frac{1+\varepsilon}{1-\varepsilon})^2 = 1 + 4\varepsilon + O(\varepsilon^2)$ give the same leading term. □

B.5 Monotonicity in time is not general

The ordering "later is better" uses the common eigenbasis. Without it, anisotropy generated at different times in different directions can cancel over a long horizon and survive over a short one. The following example is at the level of the linearized dynamics: B_s has the form $\frac{1-s}{s^3}\widehat{\mathsf{H}}_s$ with $\widehat{\mathsf{H}}_s$ symmetric of norm and spread α, i.e. exactly the endpoint decay that (26) imposes on the reduced generator, with $\mathcal{L} = \alpha$ and $\mathcal{H} = \alpha/2$; we do not claim that a particular target produces it.

Proposition 9 (Smoothly rotating anisotropy with endpoint decay)**.** *Let $d = 2$, $\alpha > 0$, $t_1 \in (0,1)$, $\beta(s) = \alpha(1-s)/s^3$ and $\Phi(t) = \int_t^1 \beta = \alpha(1-t)^2/(2t^2)$; let $t_2 \in (t_1, 1)$ solve $\Phi(t_2) = \frac{1}{2}\Phi(t_1)$ and $\delta \in (0, \min\{t_2 - t_1, 1 - t_2\})$. Let $\theta : [t_1, 1] \to [0, \pi/2]$ be C^∞ and nondecreasing with $\theta = 0$ on $[t_1, t_2 - \delta]$ and $\theta = \pi/2$ on $[t_2 + \delta, 1]$, $u(s) = (\cos\theta(s), \sin\theta(s))^\top$, $B_s = \beta(s) u(s) u(s)^\top$, and $K_{t\to s}$ the transition matrix of $\partial_s K = B_s K$. Put $c = e^{\Phi(t_1)/2}$ and $\eta_\delta = 4\delta\,\beta(t_2 - \delta) e^{\Phi(t_1)}$. Then (a) for $t \in [t_2 + \delta, 1]$, $K_{t\to 1} = \mathrm{diag}(1, e^{\Phi(t)})$, so $\mathrm{cond}(K_{t\to 1} K_{t\to 1}^\top) = e^{2\Phi(t)}$ decreases to 1; (b) $\|K_{t_1\to 1} - cI\| \le \eta_\delta$, so $\mathrm{cond}(K_{t_1\to 1} K_{t_1\to 1}^\top) \le (\frac{c+\eta_\delta}{c-\eta_\delta})^2$ when $\eta_\delta < c$. Hence for all sufficiently small δ the contrast is strictly smaller at $t_a = t_1$ than at the later time $t_b = t_2 + \delta$, while it still tends to 1 as $t \uparrow 1$; for $\alpha = 2$, $t_1 = \frac{1}{2}$, $\delta = \frac{1}{500}$ it is at most 1.25 at t_a and ≈ 2.67 at t_b.*

Proof. t_2 exists uniquely because Φ decreases strictly from $\Phi(t_1) > 0$ to 0. (a) On $[t_2 + \delta, 1]$, $B_s = \beta(s) e_2 e_2^\top$ is a commuting family, so $K_{t\to 1} = \exp(\Phi(t) e_2 e_2^\top)$. (b) Let $B_s^0 = \beta(s) e_1 e_1^\top$ on $[t_1, t_2)$ and $\beta(s) e_2 e_2^\top$ on $[t_2, 1]$, with transition K^0; then $K_{t_1\to 1}^0 = \mathrm{diag}(1, e^{\Phi(t_2)})\mathrm{diag}(e^{\Phi(t_1) - \Phi(t_2)}, 1) = cI$ by the choice of t_2. Variation of constants for $D_s = K_{t_1\to s} - K_{t_1\to s}^0$ gives $K_{t_1\to 1} - cI = \int_{t_1}^1 K_{r\to 1}(B_r - B_r^0) K_{t_1\to r}^0\, dr$; Grönwall bounds $\|K_{r\to 1}\| \left\|K_{t_1\to r}^0\right\| \le e^{\Phi(r)} e^{\Phi(t_1) - \Phi(r)} = e^{\Phi(t_1)}$, and $B_r - B_r^0$ vanishes outside $(t_2 - \delta, t_2 + \delta)$ with norm at most $2\beta(t_2 - \delta)$ there since β decreases, which gives η_δ. The consequence combines (a) at $t_2 + \delta$ with (b), because $\eta_\delta \to 0$ while $e^{2\Phi(t_2 + \delta)} \to c^2 > 1$ as $\delta \downarrow 0$. □

For the flow-matching transport, $\mathsf{H}_s - h(s)I$ is, up to positive scalars, $\mathrm{Cov}(X_1 \mid X_s = x_s) - c(s)I$ (Lemma 3), so whether a target falls under Proposition 7 is a question about the posterior covariances along the nominal trajectory; Theorem 1 and Proposition 4 do not need the answer.

B.6 Continuous intervention

Under Assumption 1 and (26), $s \mapsto J_s$ is continuous on $[0,1]$ (Proposition 4), so the Riemann sums $\frac{1}{N} W_{\mathcal{T}_N}$ converge to $W_c = \int_0^1 J_s J_s^\top\, ds$, which is positive definite because every J_s is invertible; W_c is the controllability Gramian of the linear time-varying system $\dot{\delta x}_s = A_s \delta x_s + w(s)$ with $\delta x_0 = 0$, whose endpoint is $\delta x_1 = \int_0^1 J_s w(s)\, ds$, and the least-norm argument of Proposition 2 applied to this map shows that the minimum energy $\int_0^1 \|w\|^2$ reaching δ is $\delta^\top W_c^{-1} \delta$, the minimum-energy control of Brockett (1970). Intervening everywhere along the trajectory thus replaces the source geometry $J_0 J_0^\top$ by W_c; in the Gaussian case $W_c = \frac{\pi}{2}\Sigma^{1/2}$, the dense-grid limit of Corollary 8, and $N\mathcal{C}_{\mathcal{T}_N}^\star(\delta) \to \delta^\top W_c^{-1} \delta$, the factor $1/N$ being generic to splitting a displacement across quadratically penalized inputs.

B.7 A CLOSED-LOOP CORRECTION BOUND

The allocation argument explains possible geometric benefits of multiple locations but does not model prediction error. A complementary model-agnostic bound captures the value of replanning from the reached state. Let P_k be the terminal predictor, E_k the executed short transition, and

$$\epsilon_k = \left\| P_k(x^\star_{t_k}) - P_{k+1}(E_k(x^\star_{t_k})) \right\|. \tag{28}$$

If Φ is L_Φ-Lipschitz in terminal image space on the bounded set the iterates visit (the quadratic fidelity is only locally Lipschitz) and stage k lowers its own predicted terminal objective by $\Delta_k \geq 0$, then

$$V_{k+1} \leq V_k - \Delta_k + L_\Phi \epsilon_k. \tag{29}$$

If the optimized value obeys $V_k^+ \leq (1-\theta_k)V_k$, iteration gives

$$V_N \leq \left[\prod_k (1-\theta_k)\right] V_0 + L_\Phi \sum_k \left[\prod_{j>k} (1-\theta_j)\right] \epsilon_k. \tag{30}$$

The bound formalizes only that later replanning reconditions the objective on the state actually reached. It does not assert that ϵ_k is small or that later optimization is intrinsically faster.

C EXPERIMENTAL PROTOCOL AND REPRODUCIBILITY

C.1 DATASET, PRIORS, AND PAIRING

We use 100 ImageNet-1k validation images at 256×256, one from each of 100 distinct classes, and condition every reconstruction on the known source class. The exact image identifiers are provided in the supplementary material as `data/image_ids.txt`, alongside the implementation and configurations needed to reproduce the reported experiments.

Table 4 lists the six pretrained checkpoints. The solver benchmark of Section 4.2 uses one MeanFlow and one Flow-Matching prior per space; the capacity-matched sweep of Section 4.3 replaces the two smaller checkpoints of the benchmark by the sibling of matching scale. All priors are used as released for inference only and are never fine-tuned. The two latent priors share one pretrained autoencoder, so decoder cost and decoder-limited reconstruction quality are common to both sides of the latent comparison.

Table 4: **Pretrained checkpoints.** B: solver benchmark (Section 4.2); M: capacity-matched sweep (Section 4.3). All are the released EMA weights, used for inference only. Guidance intervals are given in the paper's time convention ($t = 0$ noise, $t = 1$ data); the MeanFlow checkpoints take the guidance scale and interval as conditioning inputs, JiT gates its guidance by the interval, and SiT applies its official rule (guidance on the first three latent channels) over the whole trajectory. "bf16/fp32" denotes bfloat16 network arithmetic with float32 solver states.

Checkpoint	Family	Space	Used in	Params	Training	Guidance	Precision
pMF-L/16	MeanFlow	pixel	B, M	411M	320 epochs	7 on $[.30, .80]$	fp32
JiT-B/16	Flow Matching	pixel	B	131M	600 epochs	3 on $[.10, 1]$	bf16/fp32
JiT-L/16	Flow Matching	pixel	M	459M	600 epochs	2.4 on $[.10, 1]$	bf16/fp32
iMF-B/2	MeanFlow	latent	B	89M	640 epochs	8 on $[.35, .60]$	fp32
iMF-XL/2	MeanFlow	latent	M	610M	800 epochs	8 on $[.38, .58]$	fp32
SiT-XL/2	Flow Matching	latent	B, M	675M	7M steps	4, whole trajectory	bf16/fp32

The checkpoints are the official releases of pMF (Lu et al., 2026), JiT (Li and He, 2026), improved MeanFlow (Geng et al., 2026) and SiT (Ma et al., 2024). Both latent priors use the same autoencoder weights, the Stable Diffusion KL-f8 autoencoder with the MSE-fine-tuned decoder (`sd-vae-ft-mse`; Rombach et al., 2022): $8\times$ spatial downsampling and 4 latent channels, so a $256 \times 256 \times 3$ image is a $32 \times 32 \times 4$ code; iMF loads the Flax port of these weights and SiT the PyTorch weights. pMF and iMF run through their official JAX implementations in float32; JiT and SiT run in PyTorch with bfloat16 network arithmetic and float32 solver states.

For a fixed source image and model, method identity does not enter the random seeds used to construct the inverse problem or initial generative noise. Paired comparisons therefore share source image, class, forward operator, measurement realization, inpainting mask, canonical start time, schedule, and initial Gaussian noise; the matched sweeps additionally share the measurement across the two priors of a pair. Result archives record measurement IDs, guide IDs and initial-state fingerprints, which were checked for parity before paired statistics were computed.

C.2 Forward operators

The five tasks use the following fixed degradation parameters:

- **Denoising:** $y = x^\star + \eta$, $\eta \sim \mathcal{N}(0, .2^2 I)$.
- **Deblurring:** a 7×7 Gaussian kernel with blur standard deviation 1.0 and reflect padding, followed by Gaussian noise with standard deviation .05.
- **Super-resolution:** true $2\times$ spatial downsampling with Gaussian measurement noise of standard deviation .05. Bicubic upsampling is used only as the optional image-space guide.
- **Random inpainting:** 70% missing pixels with Gaussian noise of standard deviation .01 on observed entries.
- **Box inpainting:** a 40×40 missing box with Gaussian noise of standard deviation .05 on observed entries.

The denoising guide is the observation, deblurring uses the observed blurred image, super-resolution uses bicubic upsampling, and inpainting uses zero filling. In every reported experiment the canonical initialization time is $\tau_{\text{init}} = 0$, so the state starts from pure prior noise and the guide contributes no image content to the initial state; Appendix H.4 ablates this choice.

C.3 Fidelity normalization and metrics

D-Flow (1-step), PnP-Flow, and RHSO use the half mean-squared residual per measured scalar,

$$\Phi(x; y) = \frac{1}{2m} \|A(x) - y\|_2^2 , \tag{31}$$

with a batch sum so each image gradient is invariant to computational batch size; for the latent priors $x = \mathcal{D}(z)$. The inherited MPC implementations use their selected sum-normalized objective and control cost. We report PSNR, SSIM, LPIPS, and

$$\text{RMSE}_{\text{meas}} = \frac{\|A(\hat{x}) - y\|_2}{\sqrt{m}}. \tag{32}$$

C.4 Hyperparameter selection

Every benchmark cell of Section 4.2, and the RHSO cell of every checkpoint of Section 4.3, was selected from a pre-registered per-method grid, recorded in the supplementary material and identical across checkpoints for every guided solver, so that no checkpoint received a larger or deeper search. Selection ran as a loop: the grid was screened on 4 tuning images, the best candidates of each cell (and one step beyond any grid edge a winner sat on) were re-run on 8 tuning images, and the lowest-LPIPS configuration inside the pre-registered grid was frozen before the 100-image evaluation. The tuning images are a separate draw of ImageNet validation images, shared by all four priors. The 100-image benchmark size follows the natural-image protocols of prior flow-based solvers (Zhang et al., 2024; Martin et al., 2025; Pourya et al., 2026; Webber et al., 2026) and is set by the cost of solvers that backpropagate through composed network evaluations, D-Flow above all (Table 6); we keep one common paired set for every solver and prior rather than changing the evaluation population with method cost. Three winners sit at the upper edge of their grids (D-Flow (1-step) $M = 640$, PnP-Flow $N = 200$, latent RHSO $M = 80$): these solvers keep improving slowly with more steps beyond the grid, a compute-budget effect, and the edge values are kept because the grids fix a common budget across solvers and priors. For the mechanism studies of Sections 4.3 and 4.4 and the ablations of Appendices H.1 to H.3, no further selection was performed: the structural parameters are fixed or swept as stated, and the per-task learning rate and anchor are frozen at their selected values.

C.5 Statistical analysis

All principal reconstruction comparisons preserve image-level pairing. Paired 95% confidence intervals for the placement-depth control and the pixel-space ablations are nonparametric bootstrap intervals over the 100 paired image differences, using seed 20260903. The matched sweeps report paired mean differences and per-pair win counts over the 500 image–task pairs of each cell (Table 12); because these pairs share images across tasks, we read them as effect sizes and win counts rather than as significance levels.

C.6 Compute environment and third-party assets

All experiments were run on NVIDIA A100 GPUs (A100-SXM4-40GB for the pixel-space ablations; the schedule sweep on one H100 80GB HBM3). The four-prior benchmark and the matched sweeps were run with several shards sharing each GPU, so their wall-clock times are not reported; Eval./BP counts are the clean statement of algorithmic cost. Result archives record accelerator identity, model provenance, seeds and elapsed run metadata in addition to the resolved configurations, and the supplementary material records every experimental detail.

All assets used in this work are pre-existing and are credited to their original sources. The evaluation images are ImageNet-1k validation images (Deng et al., 2009), accessed under the ImageNet terms of use, which grant researchers a non-exclusive right to use the images for non-commercial research and educational purposes. The generative priors are the pretrained checkpoints released with the official JiT (Li and He, 2026), pMF (Lu et al., 2026), improved MeanFlow (Geng et al., 2026) and SiT (Ma et al., 2024) implementations; we comply with their licenses and with any terms accompanying the released checkpoint files themselves. We use these assets only for inference-time evaluation and do not fine-tune or otherwise modify the released weights. The supplementary material redistributes no ImageNet images and no third-party pretrained checkpoints, and releases no new dataset or pretrained generative model.

D Selected benchmark configurations and cost

Table 5 lists the configuration of every cell of the four-prior benchmark, and Table 6 the resulting generative-model evaluations and backpropagations per image. RHSO's budget $B = NM$ counts optimizer updates, not model evaluations: it is $B = 160$ on the pixel priors (as $N = 4, M = 40$ or $N = 8, M = 20$) and $B = 640$ on the latent priors, and every prior performs exactly these updates. What one update costs differs between the two families. On the MeanFlow priors each update is one terminal-map evaluation and one backpropagation, 164–168/160 on pMF-L/16 and 648/640 on iMF-B/2. On the Flow-Matching priors the benchmark RHSO rows use the integrated-suffix planner of Equation (12), the code's default for standard-flow priors, which differentiates through every remaining Heun interval (Appendix A.2): 647–1295/640–1280 on JiT-B/16 and 5776/5760 on SiT-XL/2. On pMF-L/16 RHSO's cost is therefore comparable to MPC-Δt's, on iMF-B/2 it is $2.3\times$ MPC-Δt's, and on the Flow-Matching priors it is the most expensive solver of its block, so the benchmark does not favour the finite-interval interface in compute. The capacity-matched sweeps of Appendix F use the direct planner on both families and are cost-matched (Section 4.3).

E Full benchmark results

Tables 7 to 10 give the per-task PSNR/SSIM/LPIPS behind Table 2 (Table 7 repeats Table 1 with the prior-sample floor), and Table 11 collects the per-task LPIPS of all four priors. Counting task–metric cells in which RHSO is best (ties included), it leads in 10/15 on pMF-L/16, 10/15 on JiT-B/16, 9/15 on SiT-XL/2 and 6/15 on iMF-B/2, 35/60 in total; in pairwise task–metric comparisons it beats PnP-Flow in 52/60, D-Flow (1-step) in 60/60, MPC-RHC in 50/60 and MPC-Δt in 38/60, i.e. 200/240; and it has the best LPIPS in 13 of the 20 prior–task cells. Its systematic exception is denoising, where PnP-Flow (pixel) or MPC-Δt (latent) has the better LPIPS on every prior.

Decoder ceiling. On both latent priors the optimization-based solvers (MPC-RHC, MPC-Δt, RHSO) fall within roughly one dB of one another, with SSIM in .73–.80 and near-identical PSNR across denoising, deblurring and box inpainting, whereas the pixel priors span 27.5–30.3 dB across

Table 5: **Selected benchmark configurations for all four priors.** Every cell was selected from a pre-registered grid on the 8-image tuning subset, considering all four metrics (Appendix C), and frozen before the 100-image evaluation. MPC-RHC keeps the $N = 4$, $C = 40$ rollout budget; MPC-Δt lists its stage count N and control steps C; RHSO lists stages N, inner steps M, learning rate and anchor weight μ; on the standard-flow priors it uses the integrated-suffix planner (Appendix D).

Prior	Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
pMF-L/16	Prior sample	steps=2	steps=2	steps=2	steps=2	steps=2
	PnP-Flow	N=10, γ_0=2×10^5, α=0.1	N=200, γ_0=5×10^6, α=1	N=20, γ_0=10^5, α=0.1	N=20, γ_0=10^5, α=0.1	N=50, γ_0=2×10^5, α=0.1
	D-Flow (1-step)	M=320, lr=0.1	M=640, lr=0.03	M=640, lr=0.03	M=640, lr=0.03	M=640, lr=0.1
	MPC-RHC	K=2, λ=30, lr=0.1	K=1, λ=60, lr=0.2	K=3, λ=180, lr=0.05	K=3, λ=180, lr=0.05	K=1, λ=30, lr=0.05
	MPC-Δt	N=8, C=20, λ=30, lr=0.15	N=8, C=40, λ=1000, lr=0.1	N=8, C=40, λ=360, lr=0.15	N=8, C=40, λ=360, lr=0.15	N=8, C=40, λ=180, lr=0.3
	RHSO (ours)	N=4, M=40, lr=0.03, μ=1	N=8, M=20, lr=0.01, μ=0	N=4, M=40, lr=0.01, μ=0.2	N=4, M=40, lr=0.01, μ=0	N=8, M=20, lr=0.01, μ=0
JiT-B/16	Prior sample	steps=25	steps=25	steps=25	steps=25	steps=25
	PnP-Flow	N=50, γ_0=2×10^5, α=0.5	N=200, γ_0=1.2×10^6, α=0.5	N=20, γ_0=2×10^5, α=0.25	N=50, γ_0=10^5, α=0.1	N=50, γ_0=2×10^5, α=0.1
	D-Flow (1-step)	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3
	MPC-RHC	K=2, λ=15, lr=0.05	K=1, λ=60, lr=0.05	K=2, λ=720, lr=0.05	K=2, λ=240, lr=0.02	K=1, λ=30, lr=0.05
	MPC-Δt	N=8, C=20, λ=30, lr=0.3	N=8, C=40, λ=1000, lr=0.1	N=8, C=40, λ=540, lr=0.05	N=8, C=20, λ=1000, lr=0.05	N=8, C=20, λ=180, lr=0.1
	RHSO (ours)	N=4, M=40, lr=0.005, μ=0.2	N=8, M=20, lr=0.01, μ=0	N=4, M=40, lr=0.01, μ=0.2	N=4, M=40, lr=0.01, μ=0	N=8, M=20, lr=0.01, μ=0
iMF-B/2	Prior sample	steps=8	steps=8	steps=8	steps=8	steps=8
	PnP-Flow	N=200, γ_0=10^5, α=1	N=200, γ_0=10^5, α=1	N=200, γ_0=10^5, α=1	N=200, γ_0=10^5, α=1	N=200, γ_0=10^5, α=1
	D-Flow (1-step)	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3
	MPC-RHC	K=1, λ=15, lr=0.2	K=1, λ=180, lr=0.2	K=1, λ=240, lr=0.2	K=1, λ=180, lr=0.2	K=1, λ=60, lr=0.2
	MPC-Δt	N=8, C=40, λ=360, lr=0.3	N=8, C=40, λ=1000, lr=0.3	N=8, C=40, λ=1000, lr=0.3	N=8, C=40, λ=1000, lr=0.3	N=8, C=40, λ=540, lr=0.3
	RHSO (ours)	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0
SiT-XL/2	Prior sample	steps=1	steps=1	steps=1	steps=1	steps=1
	PnP-Flow	N=200, γ_0=10^5, α=0.5	N=200, γ_0=10^5, α=0.5	N=200, γ_0=2×10^5, α=1	N=200, γ_0=2×10^5, α=1	N=200, γ_0=2×10^5, α=0.75
	D-Flow (1-step)	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3	M=640, lr=0.3
	MPC-RHC	K=1, λ=5, lr=0.2	K=1, λ=720, lr=0.2	K=1, λ=720, lr=0.2	K=1, λ=60, lr=0.2	K=1, λ=15, lr=0.2
	MPC-Δt	N=8, C=40, λ=180, lr=0.3	N=8, C=40, λ=360, lr=0.3	N=4, C=40, λ=540, lr=0.3	N=8, C=40, λ=1000, lr=0.3	N=8, C=40, λ=360, lr=0.3
	RHSO (ours)	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0	N=8, M=80, lr=0.03, μ=0

the same tasks. This flatness is consistent with a common ceiling set by the shared autoencoder rather than by the solvers. The latent PnP-Flow and D-Flow (1-step) rows are weak in absolute terms and near-constant across tasks; both take their fidelity gradient through the decoder (Appendix A), the same gradient every latent solver uses, RHSO included, so the adaptation is common to the block rather than specific to these two.

Table 6: **Generative-model evaluations/backpropagations per image for the selected benchmark configurations**, derived from Table 5 with the accounting of Appendix A.2. For the latent priors every terminal prediction additionally costs one decoder evaluation (and one decoder backpropagation where the prior is differentiated), which the counts omit. The standard-flow RHSO rows use the integrated-suffix planner (Appendix A.2); JiT-B/16's sampler takes an Euler step on the final interval, SiT-XL/2's a Heun step.

Prior	Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
pMF-L/16	PnP-Flow	11/0	201/0	21/0	21/0	51/0
	D-Flow (1-step)	321/320	641/640	641/640	641/640	641/640
	MPC-RHC	168/160	8/0	328/320	328/320	8/0
	MPC-Δt	148/140	288/280	288/280	288/280	288/280
	RHSO (ours)	164/160	168/160	164/160	164/160	168/160
JiT-B/16	PnP-Flow	51/0	201/0	21/0	51/0	51/0
	D-Flow (1-step)	641/640	641/640	641/640	641/640	641/640
	MPC-RHC	164/160	4/0	164/160	164/160	4/0
	MPC-Δt	148/140	288/280	288/280	148/140	148/140
	RHSO (ours)	647/640	1295/1280	647/640	647/640	1295/1280
iMF-B/2	PnP-Flow	201/0	201/0	201/0	201/0	201/0
	D-Flow (1-step)	641/640	641/640	641/640	641/640	641/640
	MPC-RHC	8/0	8/0	8/0	8/0	8/0
	MPC-Δt	288/280	288/280	288/280	288/280	288/280
	RHSO (ours)	648/640	648/640	648/640	648/640	648/640
SiT-XL/2	PnP-Flow	201/0	201/0	201/0	201/0	201/0
	D-Flow (1-step)	641/640	641/640	641/640	641/640	641/640
	MPC-RHC	4/0	4/0	4/0	4/0	4/0
	MPC-Δt	288/280	288/280	124/120	288/280	288/280
	RHSO (ours)	5776/5760	5776/5760	5776/5760	5776/5760	5776/5760

Table 7: **Per-task benchmark on pMF-L/16 (pixel, MeanFlow).** Each cell is PSNR↑/SSIM↑/LPIPS↓ over the 100 images; bold marks the best guided method per task and metric, ties bolded jointly.

Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
Degraded input	20.51/.393/.486	25.91/.712/.273	22.81/.649/.226	12.31/.137/1.044	26.20/.810/.120
Prior sample	9.11/.019/.719	9.11/.019/.719	9.11/.019/.719	9.11/.019/.719	9.11/.019/.719
PnP-Flow	28.29/**.839/.081**	28.20/.819/.165	26.34/.788/.120	26.40/.789/.117	**29.24**/.894/.057
D-Flow (1-step)	26.26/.688/.258	25.58/.690/.301	25.89/.732/.211	26.22/.732/.204	24.58/.777/.189
MPC-RHC	28.89/.805/.123	28.95/.804/.159	25.09/.707/.226	25.01/.693/.236	25.88/.846/.096
MPC-Δt	28.29/.744/.166	29.26/.807/.154	27.35/.806/**.087**	28.18/.843/**.077**	29.16/.893/.059
RHSO (ours)	**29.45**/.826/.093	**30.24/.872/.104**	**27.50/.821**/.095	**28.24/.848**/.078	29.00/**.915/.046**

Table 8: **Per-task benchmark on JiT-B/16 (pixel, Flow Matching).** Same layout as Table 7.

Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
Degraded input	20.51/.393/.486	25.91/.712/.273	22.81/.649/.226	12.31/.137/1.044	26.20/.810/.120
Prior sample	9.17/.014/.720	9.17/.014/.720	9.17/.014/.720	9.17/.014/.720	9.17/.014/.720
PnP-Flow	28.99/.802/**.107**	29.00/.853/.152	27.51/.823/**.085**	27.65/.827/.087	**31.28**/.901/.051
D-Flow (1-step)	23.63/.569/.441	22.89/.530/.480	23.30/.565/.436	23.13/.556/.450	21.45/.572/.451
MPC-RHC	**29.14/.807**/.125	29.15/.810/.182	26.76/.765/.150	26.56/.753/.165	27.44/.847/.087
MPC-Δt	27.98/.726/.193	29.50/.815/.143	27.35/.803/.090	28.27/**.853/.075**	30.76/.898/.052
RHSO (ours)	28.30/.804/.155	**30.32/.874/.087**	**27.80/.829**/.093	**28.36/.853/.075**	30.07/**.921/.042**

Table 9: **Per-task benchmark on iMF-B/2 (latent, MeanFlow).** Same layout as Table 7. All optimization-based solvers sit within roughly one dB of one another and vary little across tasks; see the decoder-ceiling discussion in Appendix E.

Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
Degraded input	20.51/.393/.486	25.91/.712/.273	22.81/.649/.226	12.31/.137/1.044	26.20/.810/.120
Prior sample	8.30/-0.005/.731	8.30/-0.005/.731	8.30/-0.005/.731	8.30/-0.005/.731	8.30/-0.005/.731
PnP-Flow	26.20/.690/.312	25.41/.659/.365	25.86/.687/.311	25.60/.677/.325	25.32/.686/.320
D-Flow (1-step)	25.97/.688/.333	25.39/.682/.376	25.30/.683/.323	25.09/.678/.328	25.13/.705/.322
MPC-RHC	27.44/.747/.220	27.14/.757/.265	**26.67**/**.749**/.209	**26.55**/.747/.213	26.89/.776/.210
MPC-Δt	**27.77**/**.757**/**.196**	**27.71**/**.778**/.209	26.20/.739/.196	26.50/.751/.189	**27.45**/.797/.182
RHSO (ours)	27.69/.747/.200	27.54/.775/**.201**	25.83/.731/**.189**	26.42/**.752**/**.173**	27.36/**.804**/**.173**

Table 10: **Per-task benchmark on SiT-XL/2 (latent, Flow Matching).** Same layout as Table 7.

Method	Denoising	Deblurring	2× SR	Random inpaint.	Box inpaint.
Degraded input	20.51/.393/.486	25.91/.712/.273	22.81/.649/.226	12.31/.137/1.044	26.20/.810/.120
Prior sample	8.97/.018/.735	8.97/.018/.735	8.97/.018/.735	8.97/.018/.735	8.97/.018/.735
PnP-Flow	25.18/.680/.429	24.83/.666/.457	25.31/.676/.418	25.06/.668/.426	24.30/.675/.439
D-Flow (1-step)	25.42/.653/.283	24.50/.614/.324	24.66/.641/.284	24.65/.643/.282	24.51/.668/.277
MPC-RHC	27.15/.736/.228	27.05/.754/.265	**26.72**/**.749**/.214	**26.61**/.742/.219	26.76/.764/.223
MPC-Δt	27.58/**.750**/**.191**	26.98/.747/.217	26.43/.741/**.193**	26.32/.739/.184	**27.16**/.787/.179
RHSO (ours)	**27.64**/.746/.201	**27.61**/**.775**/**.203**	25.96/.727/**.193**	26.35/**.746**/**.179**	27.05/**.800**/**.172**

Table 11: **Per-task LPIPS↓ of every solver on all four benchmark priors.** Bold marks the best guided method per prior and task (ties bolded jointly). Prior sample is a class-conditional sample from pure noise that never sees the measurement and is identical across tasks; it is the no-measurement floor.

Prior	Task	Prior sample	PnP-Flow	D-Flow (1-step)	MPC-RHC	MPC-Δt	RHSO (ours)
pMF-L/16	Denoising	.719	**.081**	.258	.123	.166	.093
	Deblurring	.719	.165	.301	.159	.154	**.104**
	2× SR	.719	.120	.211	.226	**.087**	.095
	Random inpaint.	.719	.117	.204	.236	**.077**	.078
	Box inpaint.	.719	.057	.189	.096	.059	**.046**
JiT-B/16	Denoising	.720	**.107**	.441	.125	.193	.155
	Deblurring	.720	.152	.480	.182	.143	**.087**
	2× SR	.720	**.085**	.436	.150	.090	.093
	Random inpaint.	.720	.087	.450	.165	**.075**	**.075**
	Box inpaint.	.720	.051	.451	.087	.052	**.042**
iMF-B/2	Denoising	.731	.312	.333	.220	**.196**	.200
	Deblurring	.731	.365	.376	.265	.209	**.201**
	2× SR	.731	.311	.323	.209	.196	**.189**
	Random inpaint.	.731	.325	.328	.213	.189	**.173**
	Box inpaint.	.731	.320	.322	.210	.182	**.173**
SiT-XL/2	Denoising	.735	.429	.283	.228	**.191**	.201
	Deblurring	.735	.457	.324	.265	.217	**.203**
	2× SR	.735	.418	.284	.214	**.193**	**.193**
	Random inpaint.	.735	.426	.282	.219	.184	**.179**
	Box inpaint.	.735	.439	.277	.223	.179	**.172**

F CAPACITY-MATCHED STAGE-COUNT SWEEPS

Settings. Every cell is RHSO with direct terminal planning, $\tau_{\text{init}} = 0$, $\beta = 1$, on the 100 benchmark images with a single seed. The per-task learning rate and anchor weight are the values selected for each checkpoint by the protocol of Appendix C.4, frozen across N. The benchmark RHSO cells of pMF-L/16 are cells of the fixed-B sweep ($N = 4, M = 40$ or $N = 8, M = 20$) and agree exactly with Table 7; the benchmark cells of the Flow-Matching priors are not sweep cells, because they use the integrated-suffix planner (Appendix D).

Results. Table 12 gives the MeanFlow-minus-FM differences per stage count, Tables 13 and 14 give five-task means and Tables 15 and 16 the per-task values with measurement RMSE. Counting

task–metric cells, the MeanFlow prior has the better value in 83/90 (pixel, fixed B), 70/75 (pixel, fixed M), 90/90 (latent, fixed B) and 75/75 (latent, fixed M). The MeanFlow prior also has the lower measurement residual in every latent cell and in every pixel cell except denoising at $N \geq 8$ under fixed M, where JiT-L/16 fits the noisy measurement slightly harder.

Per-stage convergence in latent space. SiT-XL/2 at $N = 8$ reaches 24.96 dB (mean LPIPS .341) with the direct planner at $M = 40$ and 25.21 dB when the same $B = 640$ is spent as $N = 16$, $M = 40$; the decoder-limited 26.92 dB (.190) of Table 2 is its benchmark cell, which spends $M = 80$ steps per stage through the integrated-suffix planner at 5776/5760 Eval./BP (Appendix D). iMF-XL/2 is at 27.03 dB (.207) with the direct planner at $M = 40$ (328/320), and iMF-B/2 at 26.97 (.187) with $M = 80$. The latent margin in Figure 3 is therefore a statement about the per-stage optimization and model-side cost at which each interface reaches the decoder-limited quality, not about that quality itself.

Reconstruction versus measurement fit. Figure 4 plots denoising PSNR and measurement RMSE against N at fixed $M = 40$ for the four matched priors. For both pixel priors the residual falls while PSNR falls; for both latent priors the residual falls while PSNR rises. The same pattern holds on the other tasks (e.g. deblurring: pMF-L/16 30.05→26.93 dB with RMSE .0514→.0472; iMF-XL/2 25.76→27.86 with .0819→.0615).

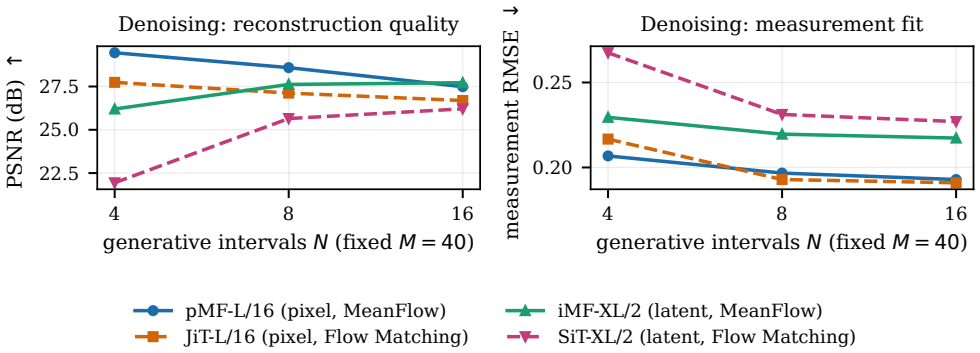

Figure 4: **The excessive-control trade-off is a pixel-space phenomenon.** Denoising at fixed $M = 40$: from $N = 4$ to $N = 16$ the pixel priors fit the measurement better and reconstruct worse, while the latent priors improve on both, because the decoder cannot express pixel-level measurement noise.

Table 12: **Capacity- and cost-matched differences, MeanFlow minus Flow Matching**, five tasks pooled and paired per image (500 image–task pairs per cell). Positive ΔPSNR and negative ΔLPIPS favour the MeanFlow prior; *wins* is the number of the 500 pairs on which the MeanFlow prior has the better LPIPS. Both priors run the same direct-terminal RHSO with per-task learning rates and anchors frozen across N (Appendix F).

		Pixel: pMF-L/16 − JiT-L/16			Latent: iMF-XL/2 − SiT-XL/2		
Budget	N	ΔPSNR (dB)	ΔLPIPS	wins	ΔPSNR (dB)	ΔLPIPS	wins
$B{=}NM{=}160$	1	+2.65	-0.247	491	+4.73	-0.073	350
	2	+0.92	-0.088	466	+6.24	-0.243	489
	4	+0.44	-0.030	388	+4.12	-0.222	490
	8	+0.88	-0.032	260	+3.29	-0.184	473
	16	+1.15	-0.052	296	+3.66	-0.183	467
	32	+0.78	-0.043	338	+3.16	-0.146	449
$M{=}40$	1	+0.72	-0.200	479	+2.87	-0.014	261
	2	+1.36	-0.126	456	+4.76	-0.148	465
	4	+0.44	-0.030	388	+4.10	-0.219	490
	8	+0.57	-0.044	335	+2.07	-0.134	474
	16	+0.69	-0.045	358	+1.91	-0.135	426

Table 13: **Capacity-matched RHSO stage-count sweep at fixed total budget $B = NM = 160$.** Five-task means of PSNR↑/SSIM↑/LPIPS↓ over 100 images; bold marks the better prior of each matched pair per metric.

		pMF-L/16			JiT-L/16			iMF-XL/2			SiT-XL/2		
N	M	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
1	160	**20.95**	**.488**	**.464**	18.30	.381	.711	**16.91**	**.327**	**.654**	12.18	.156	.728
2	80	**26.93**	**.774**	**.187**	26.00	.734	.275	**23.68**	**.603**	**.412**	17.43	.299	.655
4	40	**28.81**	**.854**	**.089**	28.37	.843	.120	**25.79**	**.707**	**.293**	21.67	.500	.515
8	20	**28.64**	**.845**	**.106**	27.76	.810	.138	**26.31**	**.731**	**.259**	23.02	.575	.442
16	10	**28.40**	**.834**	**.122**	27.25	.780	.174	**26.50**	**.737**	**.246**	22.84	.578	.429
32	5	**27.97**	**.809**	**.145**	27.19	.772	.188	**26.44**	**.733**	**.245**	23.29	.590	.391

Table 14: **Capacity-matched RHSO stage-count sweep at fixed per-stage budget $M = 40$** ($B = 40N$). Same layout as Table 13.

		pMF-L/16			JiT-L/16			iMF-XL/2			SiT-XL/2		
N	M	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
1	40	**16.99**	**.341**	**.589**	16.26	.330	.789	**14.05**	**.227**	**.699**	11.17	.109	.713
2	40	**24.05**	**.659**	**.306**	22.69	.594	.432	**20.56**	**.469**	**.548**	15.80	.253	.696
4	40	**28.81**	**.854**	**.089**	28.37	.843	.120	**25.77**	**.706**	**.296**	21.67	.500	.515
8	40	**28.16**	**.828**	**.129**	27.59	.792	.172	**27.03**	**.761**	**.207**	24.96	.659	.341
16	40	**26.99**	**.781**	**.209**	26.30	.735	.253	**27.11**	**.768**	**.189**	25.21	.662	.325

Table 15: **Per-task results of the capacity-matched sweep at fixed total budget $B = 160$.** Each cell is PSNR/SSIM/LPIPS/measurement RMSE.

Task	N	pMF-L/16	JiT-L/16	iMF-XL/2	SiT-XL/2
Denoising	1	22.22/.537/.415/.2715	18.05/.377/.728/.3369	16.89/.328/.656/.3870	12.24/.158/.726/.5432
	2	26.76/.733/.187/.2226	24.55/.665/.344/.2419	23.89/.609/.406/.2507	17.53/.303/.654/.3472
	4	29.45/.826/.093/.2068	27.74/.799/.170/.2167	26.24/.710/.292/.2309	21.94/.510/.507/.2675
	8	28.71/.778/.190/.1983	27.40/.756/.199/.2080	26.81/.730/.256/.2261	23.30/.586/.436/.2553
	16	27.84/.738/.253/.1964	27.34/.746/.228/.2026	27.05/.736/.245/.2235	23.43/.597/.407/.2542
	32	26.77/.671/.327/.1703	27.35/.730/.247/.1996	27.07/.736/.241/.2232	23.84/.606/.378/.2490
Deblurring	1	20.69/.472/.457/.1665	18.37/.382/.707/.2318	16.91/.320/.646/.3002	12.11/.152/.731/.4832
	2	27.18/.775/.238/.0656	26.23/.733/.310/.0802	23.55/.585/.438/.1114	17.28/.287/.660/.2500
	4	30.05/.871/.116/.0514	29.29/.851/.159/.0561	25.87/.705/.309/.0811	21.45/.478/.535/.1409
	8	30.24/.872/.104/.0501	28.90/.817/.162/.0680	26.52/.732/.283/.0729	22.77/.549/.473/.1172
	16	30.41/.878/.096/.0490	28.13/.774/.205/.0760	26.67/.736/.274/.0694	22.74/.559/.456/.1220
	32	30.28/.867/.104/.0481	28.14/.778/.207/.0821	26.66/.733/.269/.0680	23.01/.566/.423/.1102
2× SR	1	20.60/.475/.477/.2135	18.37/.383/.708/.2676	17.00/.331/.658/.3288	12.18/.155/.726/.5116
	2	26.75/.770/.164/.0841	26.34/.746/.235/.1063	23.79/.605/.402/.1560	17.48/.301/.655/.2898
	4	27.50/.821/.095/.0520	27.38/.823/.105/.0575	25.67/.701/.290/.1240	21.77/.502/.512/.1857
	8	27.37/.815/.114/.0440	26.34/.776/.153/.0648	25.99/.719/.258/.1151	23.02/.571/.446/.1636
	16	27.11/.800/.134/.0424	25.79/.739/.193/.0748	26.15/.723/.245/.1090	22.57/.562/.444/.1737
	32	26.78/.780/.146/.0312	25.51/.721/.217/.0859	26.03/.718/.247/.1088	23.18/.576/.400/.1563
Random inpaint.	1	20.85/.483/.476/.1106	18.39/.383/.706/.1440	16.86/.329/.655/.1811	12.14/.156/.729/.2800
	2	27.19/.786/.163/.0380	26.60/.757/.236/.0511	23.63/.605/.401/.0796	17.44/.300/.654/.1561
	4	28.24/.848/.078/.0171	28.13/.841/.093/.0220	25.48/.697/.291/.0616	21.57/.498/.518/.0976
	8	27.88/.844/.076/.0140	27.27/.817/.101/.0308	25.90/.720/.250/.0553	22.83/.568/.445/.0846
	16	27.62/.839/.078/.0133	26.52/.785/.133/.0382	26.10/.728/.240/.0533	22.60/.567/.435/.0900
	32	27.36/.824/.089/.0123	26.33/.778/.138/.0423	26.01/.721/.240/.0534	23.09/.577/.395/.0816
Box inpaint.	1	20.40/.472/.494/.2127	18.32/.381/.708/.2639	16.88/.327/.655/.3272	12.21/.158/.726/.5004
	2	26.74/.806/.183/.0987	26.30/.768/.252/.1132	23.52/.608/.412/.1574	17.44/.305/.653/.2809
	4	28.83/.905/.064/.0718	29.32/.898/.071/.0744	25.71/.724/.285/.1253	21.61/.514/.504/.1806
	8	29.00/.915/.046/.0690	28.89/.886/.074/.0850	26.35/.755/.245/.1177	23.18/.604/.412/.1573
	16	29.01/.915/.051/.0678	28.44/.857/.112/.0959	26.54/.762/.228/.1115	22.85/.608/.402/.1678
	32	28.66/.906/.059/.0595	28.61/.852/.131/.0953	26.45/.758/.228/.1123	23.31/.624/.358/.1557

Table 16: **Per-task results of the capacity-matched sweep at fixed per-stage budget $M = 40$.** Each cell is PSNR/SSIM/LPIPS/measurement RMSE.

Task	N	pMF-L/16	JiT-L/16	iMF-XL/2	SiT-XL/2
Denoising	1	17.35/.372/.589/.3727	16.15/.328/.797/.3914	13.95/.223/.695/.4973	11.18/.109/.713/.6068
	2	25.04/.678/.236/.2349	20.81/.510/.511/.2895	20.70/.476/.543/.2975	15.89/.255/.698/.3978
	4	29.45/.826/.093/.2068	27.74/.799/.170/.2167	26.20/.707/.294/.2295	21.94/.510/.507/.2675
	8	28.59/.766/.212/.1967	27.12/.701/.305/.1929	27.62/.753/.212/.2196	25.65/.673/.322/.2312
	16	27.49/.708/.304/.1929	26.69/.671/.340/.1910	27.72/.750/.204/.2173	26.21/.680/.299/.2270
Deblurring	1	16.80/.324/.579/.2868	16.29/.331/.788/.3057	14.16/.226/.685/.4209	11.16/.109/.714/.5467
	2	23.64/.633/.363/.1124	23.06/.600/.438/.1224	20.37/.452/.558/.1853	15.66/.247/.697/.3197
	4	30.05/.871/.116/.0514	29.29/.851/.159/.0561	25.76/.700/.320/.0819	21.45/.478/.535/.1409
	8	29.28/.841/.137/.0481	28.24/.775/.228/.0514	27.46/.768/.230/.0635	24.52/.620/.408/.0797
	16	26.93/.734/.350/.0472	25.39/.638/.449/.0534	27.86/.785/.205/.0615	24.31/.600/.424/.0738
2× SR	1	16.97/.338/.594/.3141	16.30/.331/.788/.3343	14.19/.235/.707/.4456	11.16/.109/.712/.5766
	2	23.94/.658/.307/.1470	23.35/.621/.399/.1618	20.77/.478/.538/.2234	15.85/.254/.697/.3492
	4	27.50/.821/.095/.0520	27.38/.823/.105/.0575	25.66/.701/.291/.1227	21.77/.502/.512/.1857
	8	26.97/.788/.150/.0419	26.35/.765/.161/.0491	26.36/.743/.199/.0982	24.80/.648/.343/.1288
	16	26.49/.762/.180/.0401	25.38/.719/.213/.0481	26.19/.745/.185/.0933	24.90/.645/.327/.1169
Random inpaint.	1	16.96/.336/.590/.1723	16.30/.331/.787/.1815	14.05/.221/.702/.2486	11.17/.109/.712/.3146
	2	24.07/.663/.302/.0747	23.26/.619/.402/.0845	20.46/.469/.547/.1194	15.85/.254/.692/.1889
	4	28.24/.848/.078/.0171	28.13/.841/.093/.0220	25.49/.700/.289/.0608	21.57/.498/.518/.0976
	8	26.94/.826/.095/.0094	26.85/.813/.100/.0167	26.52/.751/.198/.0470	24.85/.656/.331/.0637
	16	25.72/.790/.139/.0082	25.30/.756/.169/.0187	26.51/.757/.180/.0447	25.02/.655/.315/.0575
Box inpaint.	1	16.86/.333/.594/.3127	16.27/.331/.787/.3302	13.88/.228/.705/.4577	11.18/.109/.713/.5673
	2	23.56/.664/.324/.1506	22.98/.619/.412/.1627	20.51/.471/.554/.2210	15.76/.255/.697/.3440
	4	28.83/.905/.064/.0718	29.32/.898/.071/.0744	25.75/.724/.285/.1248	21.61/.514/.504/.1806
	8	29.03/.919/.049/.0660	29.38/.904/.068/.0684	27.17/.791/.192/.1047	24.96/.697/.300/.1265
	16	28.32/.912/.070/.0654	28.75/.889/.094/.0707	27.28/.804/.173/.1016	25.58/.730/.257/.1168

G FULL PLACEMENT-VERSUS-DEPTH RESULTS

Table 17 gives the complete placement and terminal-depth grid on the standard-flow prior JiT-B/16 at the fixed structural default of Appendix H, and Table 18 the paired depth-controlled effect of deepening the one-shot planner.

Table 17: **Full JiT control-placement and terminal-depth comparison.** Each cell is PSNR/SSIM/LPIPS. One-shot direct and one-shot suffix use 160 source-state optimizer updates. The two RHSO variants use $N = 4$, $M = 40$. Best PSNR, best SSIM and lowest LPIPS within each task column are shown in bold, marked independently.

Variant	Denoise	Deblur	Super-res.	Rand.	Box	Eval./BP
One-shot direct	18.19/.376/.713	18.17/.374/.713	18.19/.376/.713	18.18/.375/.713	18.15/.376/.714	161/160
One-shot 4-step suffix	20.91/.502/.553	20.75/.494/.549	20.87/.501/.552	20.92/.502/.548	20.76/.500/.555	1127/1120
RHSO direct	**27.79/.734/.198**	29.26/.845/.144	27.71/.816/**.102**	28.20/.835/.094	29.39/.891/.074	167/160
RHSO full suffix	27.43/.698/.267	**30.15/.870/.107**	**27.93/.817**/.104	**28.58/.847/.075**	**29.87/.906/.051**	647/640

Table 18: **Depth-controlled one-shot JiT comparison.** Differences are 4-step suffix one-shot minus direct one-shot at the same 160 source-state optimizer updates, with paired bootstrap confidence intervals.

Task	ΔPSNR (dB)	ΔLPIPS
Denoise	+2.72 [+2.40,+3.05]	-0.160 [-0.179,-0.140]
Deblur	+2.57 [+2.28,+2.88]	-0.164 [-0.184,-0.145]
Super-res.	+2.69 [+2.36,+3.03]	-0.161 [-0.181,-0.141]
Rand.	+2.75 [+2.44,+3.07]	-0.165 [-0.185,-0.146]
Box	+2.61 [+2.32,+2.93]	-0.159 [-0.179,-0.140]

The suffix control closes the main depth confound in the $N = 1$ versus $N = 4$ comparison. Numerical depth contributes materially to one-shot quality, but repeated intermediate-state optimization remains much stronger. The full-suffix and direct-planner RHSO rows should not be interpreted as a controlled comparison of endpoint-prediction accuracy alone, because the two planners induce different state updates throughout optimization. Their small average quality difference is nevertheless useful evidence that receding-horizon correction makes RHSO comparatively robust to terminal-planner

fidelity, which is what licenses the direct planner as the FM side of the matched comparison in Section 4.3.

H PIXEL-SPACE ABLATIONS AT THE FIXED STRUCTURAL DEFAULT

The studies in this appendix, the placement-depth control of Appendix G, and Figures 1, 2 and 7 use RHSO on pMF-L/16 and JiT-B/16 at a fixed structural default that predates the benchmark's per-task selection: $N = 4$, $M = 40$, $\beta = 1$, $\mu = 0$, $\tau_{\text{init}} = 0$, direct terminal planning, with Adam learning rates .03/.01/.01/.01/.02 (denoising/deblurring/super-resolution/random/box) on pMF-L/16 and .01 on every task for JiT-B/16. The † marks in the tables denote this default. The absolute values therefore differ slightly from the benchmark rows of Appendix E; the within-prior comparisons are exactly paired.

H.1 CONTROLLED MEASUREMENT-NOISE EXPERIMENT

The experiment uses denoising only and varies $\sigma \in \{0, .1, .2, .4\}$ at fixed $B = 160$. Measurements are exactly paired across N within each noise level and shared across the two priors for a given image and σ. The clean source images and generative seeds are also shared across σ. A protocol nuance is that the measurement seed includes the problem-parameter key, so the standardized Gaussian noise realization is not the same vector rescaled across different σ values; cross-σ comparisons are therefore distribution-level rather than perfectly pathwise scaled-noise comparisons.

Table 19: **Controlled denoising-noise sweep at $B = 160$.** Each cell is PSNR/SSIM/LPIPS. Best PSNR, best SSIM and lowest LPIPS within each row are shown in bold, marked independently; † marks the frozen default.

Prior	σ	$N = 1$	$N = 2$	$N = 4^\dagger$	$N = 8$	$N = 16$
JiT	0	18.20/0.376/0.713	27.42/0.772/0.238	**32.49/0.908/0.069**	32.47/0.886/0.077	32.00/0.873/0.090
JiT	0.1	18.20/0.376/0.713	27.37/0.771/0.235	**31.36/0.883/0.059**	29.87/0.820/0.138	28.93/0.784/0.195
JiT	0.2	18.20/0.376/0.713	27.22/**0.765**/0.226	**27.76**/0.734/**0.202**	26.15/0.651/0.334	25.69/0.637/0.360
JiT	0.4	18.19/0.376/0.713	**25.88/0.695/0.248**	22.00/0.448/0.591	21.45/0.425/0.651	21.28/0.426/0.648
pMF	0	23.00/0.589/0.372	31.34/0.874/0.094	32.57/0.901/0.064	32.71/0.905/0.056	**32.92/0.912/0.048**
pMF	0.1	22.76/0.584/0.378	30.81/0.863/0.088	**31.40/0.879**/0.064	31.30/0.876/**0.061**	30.95/0.863/0.088
pMF	0.2	23.01/0.581/0.377	**29.22/0.818/0.107**	28.54/0.788/0.147	27.81/0.746/0.205	26.97/0.689/0.295
pMF	0.4	22.68/0.575/0.382	**23.99/0.600/0.340**	22.12/0.476/0.582	21.81/0.450/0.631	21.62/0.435/0.659

Table 20: **Measurement RMSE for the controlled denoising-noise sweep.** Lower is better. Lowest measurement RMSE within each row is shown in bold; † marks the frozen default.

Prior	σ	$N = 1$	$N = 2$	$N = 4^\dagger$	$N = 8$	$N = 16$
JiT	0	.2690	.1019	**.0597**	.0757	.0916
JiT	0.1	.2870	.1420	**.1124**	.1204	.1306
JiT	0.2	.3352	.2224	**.1978**	.2002	.2070
JiT	0.4	.4822	.4075	**.3761**	.3773	.3811
pMF	0	.1732	.0625	**.0547**	.0550	**.0547**
pMF	0.1	.2061	.1158	.1105	.1102	**.1093**
pMF	0.2	.2635	.2035	.1982	.1966	**.1946**
pMF	0.4	.4366	.3863	.3751	.3737	**.3724**

For pMF-L/16, the reconstruction-optimal stage count shifts from $N = 16$ at $\sigma = 0$ to approximately $N = 4$ to 8 at .1, then to $N = 2$ at .2 and .4. JiT-B/16 remains strongest near $N = 4$ at low and moderate noise but also moves to $N = 2$ at $\sigma = .4$. The sign reversal in Table 21 is robust at the image level and is the most direct evidence that aggressive distributed control can fit noise rather than the clean target; the latent priors of Appendix F, whose decoder filters that noise, do not show it. The two priors here are not capacity matched, so only within-prior trends should be read.

Table 21: **Noise-dependent value of additional stages.** Differences are $N = 4 - N = 2$ with 95% paired bootstrap confidence intervals.

Prior	σ	ΔPSNR (dB)	ΔLPIPS
JiT	0	+5.07 [+4.71,+5.42]	-0.169 [-0.186,-0.152]
JiT	0.1	+3.99 [+3.67,+4.32]	-0.176 [-0.193,-0.158]
JiT	0.2	+0.54 [+0.07,+1.01]	-0.024 [-0.054,+0.005]
JiT	0.4	-3.89 [-4.40,-3.38]	+0.342 [+0.303,+0.381]
pMF	0	+1.24 [+0.94,+1.55]	-0.030 [-0.038,-0.022]
pMF	0.1	+0.59 [+0.38,+0.80]	-0.024 [-0.032,-0.017]
pMF	0.2	-0.68 [-0.89,-0.48]	+0.040 [+0.027,+0.053]
pMF	0.4	-1.87 [-2.22,-1.54]	+0.242 [+0.208,+0.277]

H.2 State-anchor experiment

Tables 22 to 24 sweep the anchor weight μ of Equation (13) for denoising at $\sigma = .2$, holding the rest of the default configuration fixed.

Table 22: **State-anchor sweep for denoising at $\sigma = .2$.** Each cell is PSNR/SSIM/LPIPS. Best PSNR, best SSIM and lowest LPIPS within each row are shown in bold, marked independently; † marks the frozen default.

Prior	N	$\mu = 0^\dagger$	$\mu = 0.01$	$\mu = 0.03$	$\mu = 0.1$	$\mu = 0.3$	$\mu = 1$
JiT	4	27.76/0.734/0.201	27.83/0.735/0.199	27.88/0.738/0.194	28.27/0.757/0.172	**29.15/0.808/0.105**	28.65/0.802/0.145
JiT	8	26.17/0.652/0.332	26.13/0.650/0.333	26.21/0.653/0.328	26.35/0.662/0.315	26.79/0.688/0.275	**28.29/0.776/0.147**
pMF	4	28.41/0.781/0.155	28.44/0.778/0.162	28.32/0.774/0.168	28.11/0.758/0.198	28.52/0.773/0.174	**29.47/0.826/0.083**
pMF	8	27.74/0.739/0.212	27.61/0.728/0.222	27.63/0.734/0.223	27.59/0.724/0.238	27.63/0.721/0.249	**28.71/0.780/0.164**

Table 23: **Measurement RMSE for the state-anchor sweep.** Lower is better. Lowest measurement RMSE within each row is shown in bold; † marks the frozen default.

Prior	N	$\mu = 0^\dagger$	$\mu = 0.01$	$\mu = 0.03$	$\mu = 0.1$	$\mu = 0.3$	$\mu = 1$
JiT	4	.1978	**.1976**	.1978	.1985	.2025	.2128
JiT	8	.2005	.2008	.2006	**.2004**	.2014	.2080
pMF	4	.1981	.1976	.1974	**.1964**	.1973	.2065
pMF	8	.1965	.1960	.1956	.1948	**.1945**	.1989

Table 24: **Paired effect of the strongest anchor setting in the μ ablation.** Differences are anchored minus $\mu = 0$ with 95% paired bootstrap confidence intervals.

Prior	N	anchor	ΔPSNR (dB)	ΔLPIPS
JiT	4	$\mu = 0.3$	+1.38 [+1.19,+1.58]	-0.096 [-0.110,-0.082]
JiT	8	$\mu = 1$	+2.12 [+1.80,+2.44]	-0.185 [-0.209,-0.162]
pMF	4	$\mu = 1$	+1.05 [+0.81,+1.31]	-0.072 [-0.087,-0.058]
pMF	8	$\mu = 1$	+0.96 [+0.73,+1.21]	-0.048 [-0.066,-0.030]

Strong anchoring improves reconstruction while relaxing measurement fit, but the response is not monotone in μ. In particular, pMF-L/16 at low and intermediate anchor values can reduce measurement RMSE while worsening LPIPS before the stronger $\mu = 1$ condition reverses the perceptual trend. The pMF-L/16 $N = 4, \mu = 1$ denoising condition reaches 29.47/.826/.083 here; because this sweep was evaluated on the 100 evaluation images it is a mechanistic ablation, not a selection. The benchmark's pre-registered selection on the disjoint tuning subset independently chose $\mu = 1$ for pMF-L/16 denoising and reached 29.45/.826/.093 (Table 7), confirming the effect without test-set selection.

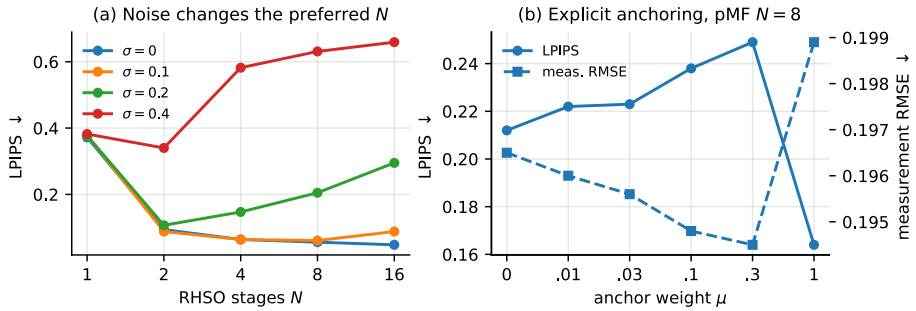

Figure 5: **Distributed control exposes a data-fit versus reconstruction trade-off in pixel space.** (a) In pMF-L/16 denoising, many stages help when the measurement is clean and hurt increasingly as noise grows. (b) At pMF-L/16 $N = 8$, strong anchoring reverses the LPIPS degradation while giving back measurement fit. Values are those of Tables 19, 22 and 23.

H.3 Schedule exponent

Tables 25 and 26 sweep the schedule exponent β of Equation (9) at the default $N = 4, M = 40$ budget, where $\beta = 1$ is the uniform grid used in every headline run.

Table 25: **Nonuniform RHSO schedule sweep.** $N = 4, M = 40$. Each cell is PSNR/SSIM/LPIPS. Best PSNR, best SSIM and lowest LPIPS within each row are shown in bold, marked independently; † marks the frozen default.

Prior	Task	$\beta = 0.5$	$\beta = 0.75$	$\beta = 1^\dagger$	$\beta = 1.5$	$\beta = 2$
JiT	Denoise	26.85/0.665/0.329	27.10/0.687/0.291	27.79/0.734/0.198	**28.36/0.795/0.143**	27.35/0.772/0.201
JiT	Deblur	29.37/0.829/0.128	**29.65/0.856/0.115**	29.26/0.845/0.144	27.74/0.790/0.222	26.49/0.741/0.275
JiT	Super-res.	27.17/0.784/0.139	27.57/0.807/0.114	**27.71/0.816/0.102**	27.32/0.797/0.146	26.38/0.762/0.200
JiT	Rand.	28.09/0.837/**0.071**	**28.32/0.843**/0.076	28.20/0.835/0.094	27.40/0.799/0.153	26.49/0.759/0.210
JiT	Box	29.68/**0.907/0.044**	**29.89**/0.905/0.047	29.39/0.891/0.074	28.15/0.837/0.154	26.72/0.780/0.221
pMF	Denoise	26.94/0.682/0.307	27.74/0.737/0.219	28.56/0.786/0.150	**28.98/0.810/0.118**	28.93/0.809/0.122
pMF	Deblur	29.53/0.845/0.107	**29.87/0.856/0.098**	29.82/0.854/0.113	29.23/0.841/0.147	28.52/0.818/0.174
pMF	Super-res.	26.55/0.778/0.142	27.37/0.808/0.109	27.73/0.820/**0.096**	**27.93/0.823**/0.102	27.72/0.810/0.123
pMF	Rand.	27.51/0.828/0.086	28.15/0.839/**0.075**	**28.30/0.841**/0.078	**28.30**/0.834/0.096	27.92/0.814/0.125
pMF	Box	27.91/**0.904/0.054**	27.91/0.900/0.059	**27.96**/0.892/0.072	27.59/0.863/0.106	27.37/0.849/0.122

Table 26: **Measurement RMSE for the nonuniform schedule sweep.** Lower is better. Lowest measurement RMSE within each row is shown in bold; † marks the frozen default.

Prior	Task	$\beta = 0.5$	$\beta = 0.75$	$\beta = 1^\dagger$	$\beta = 1.5$	$\beta = 2$
JiT	Denoise	**.1904**	.1929	.1981	.2094	.2206
JiT	Deblur	**.0499**	.0526	.0563	.0634	.0783
JiT	Super-res.	**.0472**	.0515	.0587	.0753	.0989
JiT	Rand.	**.0173**	.0194	.0245	.0357	.0486
JiT	Box	**.0662**	.0690	.0751	.0887	.1087
pMF	Denoise	**.1920**	.1948	.1983	.2024	.2055
pMF	Deblur	**.0494**	.0503	.0516	.0536	.0559
pMF	Super-res.	**.0464**	.0486	.0520	.0600	.0714
pMF	Rand.	**.0152**	.0168	.0191	.0242	.0311
pMF	Box	**.0670**	.0682	.0698	.0757	.0792

Measurement RMSE increases strictly with β in all ten model-task pairs. The best LPIPS schedule is $\beta = 1.5$ for denoising in both models, .75 for deblurring, 1 for super-resolution, .5 for box inpainting, and .5 (JiT-B/16) / .75 (pMF-L/16) for random inpainting. This agreement across the two priors supports a task-dependent temporal-allocation interpretation, while the lack of one universal direction motivates keeping $\beta = 1$ as the task-agnostic default.

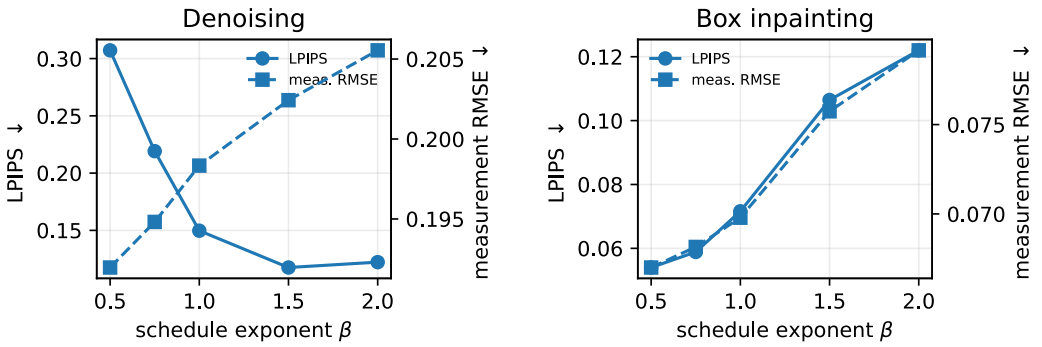

Figure 6: **Task-dependent temporal allocation for pMF-L/16.** Denoising prefers earlier/noisy-side control with $\beta > 1$, even though measurement fit worsens, while box inpainting prefers later/data-side control with $\beta < 1$.

H.4 Initialization time

The canonical time $\tau_{\text{init}} = 0$ denotes pure prior noise. Increasing τ_{init} injects more measurement-guide content into the starting state.

Table 27: **Canonical initialization-time ablation for RHSO.** Each cell is PSNR/SSIM/LPIPS; $\tau_{\text{init}} = 0$ is pure prior noise and larger values inject more measurement-guide content. Best PSNR, best SSIM and lowest LPIPS within each row are shown in bold, marked independently; † marks the frozen default.

Prior	Task	$\tau = 0^\dagger$	.2	.4	.6
JiT	Denoise	**27.79/.734/.198**	26.82/.654/.337	26.62/.643/.352	26.44/.632/.356
	Deblur	29.26/.845/.144	**30.57/.873/.070**	29.62/.827/.120	27.44/.712/.309
	Super-res.	**27.71/.816/.102**	27.46/.795/.121	26.70/.759/.159	25.47/.694/.238
	Box	**29.39**/.891/.074	27.95/**.909/.045**	26.62/.902/.062	26.41/.892/.088
	Rand.	28.20/.835/.094	**28.31/.842/.068**	26.08/.792/.139	18.03/.497/.485
pMF	Denoise	**28.48/.783/.152**	28.15/.762/.186	27.50/.719/.244	26.87/.674/.321
	Deblur	29.95/.861/.103	**30.50/.873/.061**	29.89/.857/.077	28.72/.812/.159
	Super-res.	**27.78/.820/.098**	27.52/.804/.109	26.72/.775/.143	25.70/.732/.195
	Box	**28.10**/.892/.072	27.38/.900/**.063**	26.23/**.901/.063**	26.33/.899/**.063**
	Rand.	**28.29/.840**/.079	28.15/.838/**.074**	27.40/.825/.091	23.90/.752/.205

Table 28: **Measurement RMSE for the initialization-time ablation.** These are aggregate diagnostics. Lowest measurement RMSE within each row is shown in bold; † marks the frozen default.

Prior	Task	$\tau = 0^\dagger$	.2	.4	.6
JiT	Denoise	.1981	.1894	.1884	**.1878**
JiT	Deblur	.0563	.0499	.0488	**.0477**
JiT	Super-res.	.0587	.0468	.0439	**.0419**
JiT	Box	.0751	.0656	.0643	**.0637**
JiT	Rand.	.0245	.0160	**.0144**	.0146
pMF	Denoise	.1977	.1961	.1937	**.1913**
pMF	Deblur	.0508	.0495	.0488	**.0482**
pMF	Super-res.	.0523	.0471	.0446	**.0426**
pMF	Box	.0700	.0680	.0669	**.0663**
pMF	Rand.	.0194	.0159	.0148	**.0141**

Measurement fit generally improves as the start moves toward the data end, but reconstruction often degrades; deblurring is a clear exception that benefits from a moderate measurement-informed start. We retain pure-noise initialization throughout because it is a simple task-agnostic operating point and demonstrates that the reconstruction quality is produced by the control procedure rather than by favorable image content already present in the starting state.

I RUNTIME

Table 29 reports wall-clock time per image at batch size one on an A100-SXM4-40GB for the fixed-structural-default configurations of Appendix H on the two pixel priors. The four-prior benchmark and the matched sweeps ran with several shards per GPU and are not timed. These measurements should not be interpreted as architecture-controlled hardware comparisons because the priors differ in model size, framework and implementation; the Eval./BP counts of Table 6 are the cleaner statement about algorithmic terminal-planning depth. A plain sampling pass performs no state optimization and costs a handful of model evaluations with no backpropagation; every solver that differentiates the prior is orders of magnitude more expensive in backpropagations, which is the cost RHSO trades for reconstruction quality.

Table 29: **Runtime per image at the fixed structural default** (seconds/image, batch size one, A100-SXM4-40GB; pixel priors only). The standard-flow RHSO entry is the integrated-suffix planner.

Prior	Method	Denoising	Deblurring	Super-res.	Rand. inpaint.	Box inpaint.
pMF-L/16	PnP-Flow	0.81	4.00	0.35	0.34	1.69
	D-Flow (1-step)	26.97	35.01	25.49	25.37	27.35
	MPC-RHC	14.33	4.51	26.52	26.39	1.17
	MPC-Δt	21.71	29.83	4.60	4.51	22.20
	RHSO	12.97	17.98	13.23	13.19	14.01
JiT-B/16	PnP-Flow	1.85	3.83	0.76	0.77	3.66
	D-Flow (1-step)	27.98	28.31	27.85	27.93	28.21
	MPC-RHC	14.13	0.49	28.01	28.03	0.24
	MPC-Δt	24.72	25.17	5.38	5.39	24.73
	RHSO suffix	56.68	56.91	56.56	56.75	56.37

J ADDITIONAL QUALITATIVE RESULTS

Figure 2 shows one set of solver outputs on pMF-L/16; Figure 7 shows a second, independently drawn set with the same column layout. Both are illustrative and do not by themselves establish perceptual superiority; the aggregate evidence is Table 2.

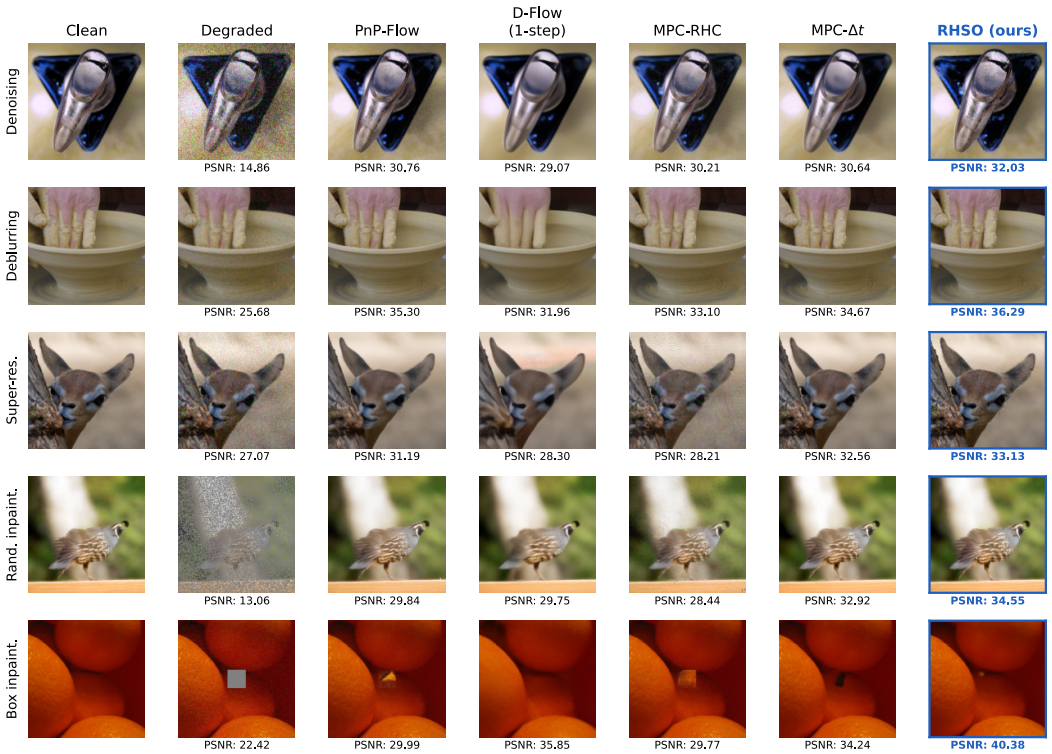

Figure 7: **A second qualitative comparison for the five inverse problems on pMF-L/16.** Columns show the clean target, degraded observation, and solver outputs at the fixed structural default of Appendix H, on images distinct from those of Figure 2.