Principled Design of Diffusion-based Optimizers for Inverse Problems

Julio Oscanoa[*]
Bioengineering, Radiology
Stanford University
joscanoa@stanford.edu

Irmak Sivgin[*]
Electrical Engineering
Stanford University
isivgin@stanford.edu

Cagan Alkan[*]
Electrical Engineering
Stanford University
calkan@stanford.edu

Daniel Ennis
Radiology
Stanford University
dbe@stanford.edu

John Pauly
Electrical Engineering
Stanford University
pauly@stanford.edu

Mert Pilanci[†]
Electrical Engineering
Stanford University
pilanci@stanford.edu

Shreyas Vasanawala[†]
Radiology
Stanford University
vasanawala@stanford.edu

Abstract

Score-based diffusion models achieve state-of-the-art performance for inverse problems, but their practical deployment is hindered by long inference times and cumbersome hyperparameter tuning. While pretrained diffusion models can be reused across tasks without retraining, inference-time hyperparameters such as the noise schedule and posterior sampling weights typically require ad-hoc adjustment for each problem setup. We propose principled reparameterizations that induce invariances, allowing the same hyperparameters to be reused across multiple problems without re-tuning. In addition, building on the RED-diff framework, which reformulates posterior sampling as an optimization problem, we further develop the OptDiff pipeline. OptDiff provides a simplified tuning framework that facilitates the integration of convex optimization tools to accelerate inference. Experiments on image reconstruction, deblurring, and super-resolution show substantial speedups and improved image quality. Our code is available at https://github.com/MRSRL/OptDiff.

1 Introduction

The flexibility and expressiveness of diffusion models [1, 2] as powerful image priors have made them strong candidates for solving inverse problems, where the goal is to reconstruct data from incomplete and noisy measurements [3–5]. The key idea is to leverage a pretrained diffusion model to sample from the posterior distribution. However, existing diffusion-based inverse solvers face two major challenges: slow convergence and strong dependence on hyperparameter selection. The iterative sampling process often requires hundreds of steps, resulting in high computational cost that limits practical deployment in time-sensitive applications. Moreover, reconstruction quality is highly sensitive to hyperparameter choices, and poor selection can lead to instability or degraded performance [6, 7].

[*]Equal contribution.
[†]Equal senior contribution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

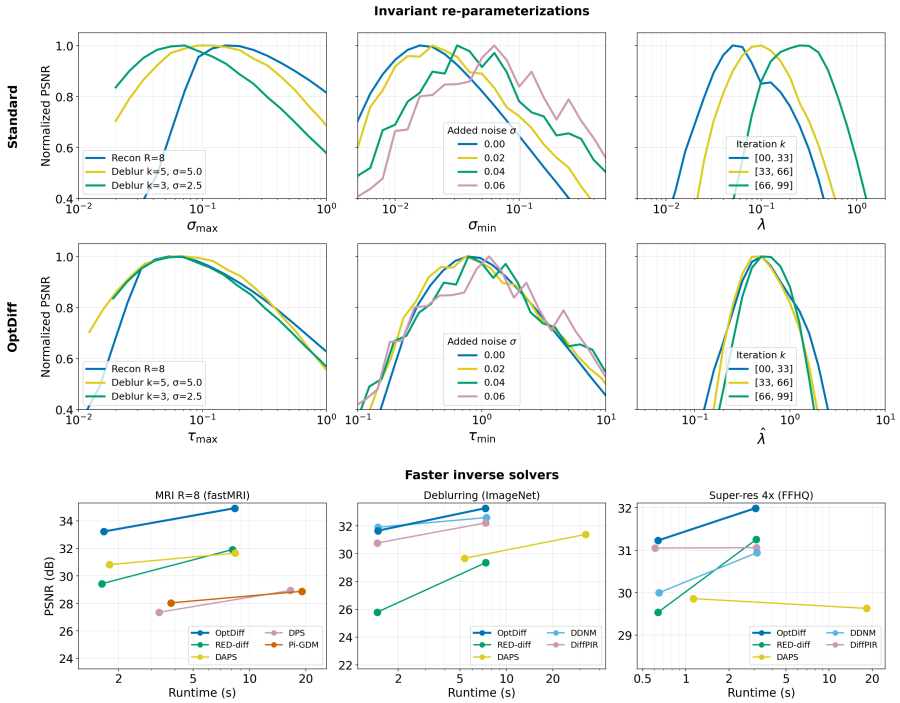

Figure 1: **(Top)** Comparison of standard and proposed hyperparameters for different experimental settings. Our proposed reparameterization $(\tau_{\max}, \tau_{\min})$ for noise schedule hyperparameters $(\sigma_{\max}, \sigma_{\min})$ produces aligned normalized PSNR curves across different inverse problem settings, showcasing invariance of OptDiff parameters. Similarly, our reparameterized sampler hyperparameter $\hat{\lambda}$ yields aligned curves for different iteration intervals, implying that a single $\hat{\lambda}$ value suffices during the entire reverse process in OptDiff. **(Bottom)** Performance comparisons between OptDiff and other diffusion-based inverse problem solvers. Our method OptDiff achieves the best runtime-PSNR tradeoff.

In this work, we address these challenges through two complementary contributions. First, inspired by maximal update parameterization (μP) [8], which stabilizes *training* hyperparameters for large neural networks, we introduce principled reparameterizations that stabilize *inference-time* hyperparameters in diffusion-based inverse solvers. These reparameterizations induce invariances across problem configurations, substantially reducing the need for manual tuning. Second, building on the RED-diff framework [5], which reformulates posterior sampling as a stochastic optimization problem, we propose the *Optimizers with Diffusion-based priors* (OptDiff) pipeline for robust hyperparameter selection. This framework enables the integration of established tools from the optimization literature to design faster inverse solvers. We show that the resulting methods outperform state-of-the-art diffusion-based approaches in both reconstruction quality and inference speed.

1.1 Contributions

- We introduce principled reparameterizations of inference-time hyperparameters for diffusion-based inverse solvers. We organize these into two families: (i) *noise schedule hyperparameters*, such as the start and stop noise levels $\sigma_{\max}$ and $\sigma_{\min}$, and (ii) *sampler hyperparameters*, such as the step size α and regularization weight λ. The proposed reparameterizations induce invariances across problem setups, enabling robust hyperparameter reuse and substantially reducing the need for manual tuning.
- We develop a principled, theory-driven framework for noise schedule design in inverse problems that is solver-agnostic by construction. To the best of our knowledge, this is the first work to study noise schedule design in this context. By analyzing reverse diffusion through the lens of spectral auto-regression [9, 10], we derive tolerance-based criteria for selecting $\sigma_{\max}$ and $\sigma_{\min}$ that adapt naturally to the spectral properties of the data and the

measurement model. We further demonsrrate empirically that the resulting noise schedule transfer across representative diffusion-based inverse solvers.

- For the sampler hyperparameters, we focus on the family of optimization-based solvers in Daras et al. [11]. We present a common-descent analysis of the composite data-consistency and diffusion-prior objective that identifies when a single trade-off scalar can be used throughout the optimization, eliminating step-dependent tuning. We develop and validate this framework through RED-diff, whose explicit objective and update rules enable direct analysis.
- Building on the RED-diff formulation, we propose the OptDiff pipeline for simplified and robust hyperparameter selection. OptDiff combines invariant reparameterizations with a greedy-optimal hyperparameter scheme, allowing the noise schedule and sampler hyperparameters to be decoupled and tuned efficiently using a small tuning set.
- Using the OptDiff pipeline, we demonstrate that incorporating advanced convex optimization techniques leads to significantly faster inference while maintaining or improving reconstruction accuracy. Our results show consistent reductions in both the number of function evaluations and overall runtime.
- We evaluate OptDiff across a range of inverse problems, including MRI reconstruction, deblurring, and super-resolution. Across all tasks, our approach achieves state-of-the-art runtime performance while maintaining competitive or improved reconstruction quality.

2 Background

2.1 Score-based diffusion models

Score-based diffusion models involve forward and reverse diffusion processes that are formulated by stochastic differential equations (SDEs). In particular, the forward process for $t \in [0, T]$ is represented by [12]

$$\mathrm{d}\boldsymbol{x}_t = \boldsymbol{b}\left(\boldsymbol{x}_t, t\right)\,\mathrm{d}t + \sqrt{a(t)}\,\mathrm{d}\boldsymbol{W}_t, \tag{1}$$

where $\boldsymbol{x}_0 \sim p_{\mathrm{data}}$, $\boldsymbol{x}_0 \in \mathbb{R}^n$, and $\boldsymbol{W}_t$ is an n-dimensional Wiener process. For sufficiently large T, the distribution $p_T(\boldsymbol{x}_T)$ approaches an isotropic Gaussian distribution. The reverse diffusion process recovers the data distribution $p(\boldsymbol{x}_0)$ from pure noise via the corresponding reverse SDE [13]

$$\mathrm{d}\boldsymbol{x}_t = \left(\boldsymbol{b}\left(\boldsymbol{x}_t, t\right) - a(t)\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)\right)\mathrm{d}t + \sqrt{a(t)}\,\mathrm{d}\bar{\boldsymbol{W}}_t. \tag{2}$$

where $\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t)$ is the score function. In practice, the score function is approximated by a neural network $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t)$ which is typically trained using denoising score matching [14]:

$$\arg\min_{\boldsymbol{\theta}} \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{x}_t}\left[\left\|\frac{-\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t)}{\sigma_t} - \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{x}_0)\right\|_2^2\right]. \tag{3}$$

where σ_t is the standard deviation of the perturbation kernel $p_t(\boldsymbol{x}_t|\boldsymbol{x}_0)$.

Noise Schedule. The reverse process is typically discretized on a decreasing list of noise levels $\sigma_{\max} = \sigma_{t_N} > \cdots > \sigma_{t_1} = \sigma_{\min}$, where the start level $\sigma_{\max}$ and stop level $\sigma_{\min}$ are inference-time hyperparameters.

2.2 Diffusion models for inverse problems

Inverse problems are described according to a forward model $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{\eta}$, where $\boldsymbol{y} \in \mathbb{R}^m$ represents the measurements, $\boldsymbol{x}_0 \in \mathbb{R}^n$ is the signal to be estimated, $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma_{\boldsymbol{\eta}}^2\mathbf{I})$ is the measurement noise, and $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^m$ is the measurement operator. The goal in inverse problems is to recover the underlying signal $\boldsymbol{x}_0$ from a potentially incomplete set of measurements $\boldsymbol{y}$.

Diffusion-based inverse problem solvers aim to represent and sample from $p(\boldsymbol{x}_0|\boldsymbol{y})$ in a computationally efficient manner. Given the measurements $\boldsymbol{y}$, the conditional score function can be expressed using the Bayes' Theorem as

$$\nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t|\boldsymbol{y}) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) + \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{y}|\boldsymbol{x}_t). \tag{4}$$

Then, the reverse process in (2) can be modified to yield samples from $p(\boldsymbol{x}_0|\boldsymbol{y})$ [15]. The inverse problem scenario additionally requires evaluating the intractable likelihood score function $\nabla_{\boldsymbol{x}_t} \log p_t(\boldsymbol{y}|\boldsymbol{x}_t)$ [4]. To address this challenge, diffusion-based inverse problem solvers seek to approximate the likelihood score function efficiently using various strategies. A comprehensive overview of these methods is provided in the survey by Daras et al. [11].

2.3 Regularization by denoising diffusion (RED-diff)

Mardani et al. [5] propose a variational inference driven formulation called Regularization by Denoising Diffusion (RED-diff) for solving inverse problems, converting diffusion sampling into an optimization problem. RED-diff represents the complex posterior distribution $p(\boldsymbol{x}_0|\boldsymbol{y})$ with a variational distribution $q \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2\mathbf{I})$, and minimizes the KL divergence between them. The final optimization objective is:

$$\min_{\boldsymbol{\mu}} \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{\mu})\|_2^2 + \mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\lambda_t \left\|\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t; t\right) - \boldsymbol{\epsilon}\right\|_2^2\right] \tag{5}$$

where we absorb $\sigma_{\boldsymbol{\eta}}^2$ into λ_t, and λ_t is a time-dependent weighting, for which Mardani et al. [5] propose a signal-to-noise ratio (SNR)-based formulation.

3 Theory

We develop a theoretical framework that yields invariant reparameterizations for diffusion-based inverse solvers.

3.1 Noise schedule hyperparameters

While noise schedule design has been extensively studied for unconditional generation [16, 1, 17], its extension to inverse problems remains largely unexplored. We develop principled guidelines for selecting $\sigma_{\max}$ and $\sigma_{\min}$ by viewing reverse diffusion as spectral auto-regression in the inverse problem setting (Figure 2). Additionally, we derive tolerance-based reparameterizations $(\tau_{\max}, \tau_{\min})$ that enable robust parameter reuse across inverse problems by analyzing the estimation error covariance .

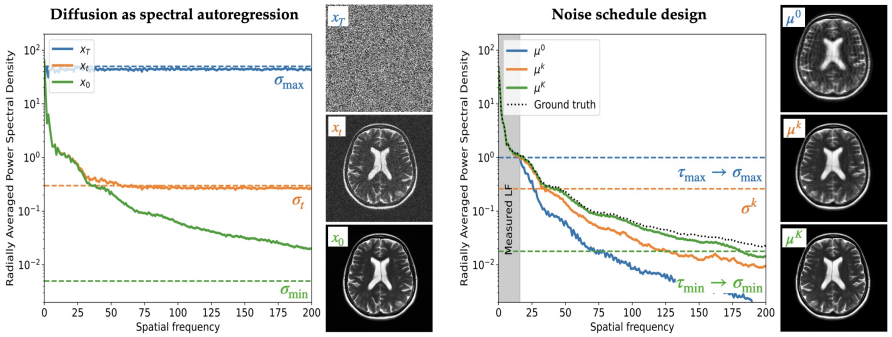

Figure 2: Noise schedule design from a spectral auto-regression perspective. **(Left)** Due to the power-law spectrum of natural images [18], reverse diffusion recovers low-frequency components first and progressively reconstructs higher frequencies conditioned on them [9, 10]. **(Right)** This perspective motivates adapting the start and stop noise levels to measured low-frequency content and the dataset noise floor.

Setup. In Variance Exploding (VE) diffusion, we observe $\boldsymbol{x}_t = \boldsymbol{x}_0 + \sigma_t \boldsymbol{n},\ \boldsymbol{n} \sim \mathcal{N}(0, \boldsymbol{I})$. Let $\boldsymbol{\Sigma} \succeq 0$ be the prior covariance, the LMMSE-optimal estimator yields the covariance posterior map:

$$\Phi(\boldsymbol{\Sigma};\, \sigma_t) \;=\; \boldsymbol{\Sigma} - \boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \sigma_t^2\mathbf{I})^{-1}\boldsymbol{\Sigma}. \tag{6}$$

3.1.1 Start noise level $\sigma_{\max}$

Assume the low-frequency (LF) content of $\boldsymbol{x}_0$ is known exactly, and let the prior $\boldsymbol{\Sigma}_{H|L}$ be the covariance of the high-frequency (HF) components conditioned on LF. Define the covariance residual

$\mathbf{\Delta}_t^{\max} := \mathbf{\Sigma}_{H|L} - \Phi(\mathbf{\Sigma}_{H|L}; \sigma_t) \succeq 0$, and seek the region where this residual is small, meaning that the choice of σ_t has negligible effect on estimation. We therefore start the reverse diffusion process at a noise level where the first step reduces the covariance by at most a fraction $\tau_{\max} \in (0, 1)$ of the prior covariance.

Theorem 3.1 (Start bound). *If $\mathbf{\Delta}_t^{\max} \preceq \tau_{\max} \mathbf{\Sigma}_{H|L}$, then*

$$\boxed{\sigma_{\max}^2 \;\geq\; \frac{1-\tau_{\max}}{\tau_{\max}}\, \nu_{\max}\big(\mathbf{\Sigma}_{H|L}\big),} \tag{7}$$

where $\nu_{\max}(\cdot)$ denotes the maximum eigenvalue.

Corollary 3.2 (WSS/PSD). *Under a WSS model with ideal LF mask, $\mathbf{\Sigma}_{H|L} = \mathrm{diag}(S(\omega) : \omega \in \Omega_H)$, where $S(\omega)$ denotes the power spectral density of the signal. In this case,*

$$\sigma_{\max}^2 \;\geq\; \frac{1-\tau_{\max}}{\tau_{\max}} \max_{\omega \in \Omega_H} S(\omega). \tag{8}$$

The proofs are provided in Appendix C and D. In practice, we choose $\sigma_{\max}$ at the lower bound, since larger values provide little reduction in error covariance (smaller $\Delta_t^{\max}$) but increase the number of reverse diffusion steps. Expressing the bound in terms of the tolerance $\tau_{\max}$ ensures natural scaling with the spectral density and yields a more robust criterion.

3.1.2 Stop noise level $\sigma_{\min}$

Assume the data has an additive white noise floor $\sigma_{\mathbf{s}}^2$, and let $\Sigma_{\mathbf{s}}$ denote the covariance of the clean signal. The best attainable posterior (after removing VE diffusion noise) is $\mathbf{\Sigma}_0 := \Phi(\mathbf{\Sigma}_{\mathbf{s}}; \sigma_{\mathbf{s}})$. Define covariance residual $\mathbf{\Delta}_t^{\min} := \Phi(\mathbf{\Sigma}_{\mathbf{s}}; \sqrt{\sigma_{\mathbf{s}}^2 + \sigma_t^2}) - \mathbf{\Sigma}_0 \succeq 0$, and seek the region where this residual is small, i.e., the current posterior is sufficiently close to the best attainable posterior. We stop the reverse diffusion once this distance is less than a fraction $\tau_{\min} \in (0, 1)$.

Theorem 3.3 (Stop bound). *If $\mathbf{\Delta}_t^{\min} \preceq \tau_{\min} \mathbf{\Sigma}_0$ and $\nu_{\max}(\mathbf{\Sigma}_{\mathbf{s}}) > \tau_{\min}\, \sigma_{\mathbf{s}}^2$, then*

$$\boxed{\sigma_{\min}^2 \;\leq\; \kappa(\sigma_{\mathbf{s}}) := \frac{\tau_{\min}\, \sigma_{\mathbf{s}}^2\, (\nu_{\max}(\mathbf{\Sigma}_{\mathbf{s}}) + \sigma_{\mathbf{s}}^2)}{\nu_{\max}(\mathbf{\Sigma}_{\mathbf{s}}) - \tau_{\min}\, \sigma_{\mathbf{s}}^2},} \tag{9}$$

If $\nu_{\max} \leq \tau_{\min}\, \sigma_{\mathbf{s}}^2$, the bound is vacuous.

Corollary 3.4 (High-SNR regime). *If $\nu_{\max} \gg \sigma_{\mathbf{s}}^2$, then $\kappa(\sigma_{\mathbf{s}}) \approx \tau_{\min}\, \sigma_{\mathbf{s}}^2$, so that*

$$\sigma_{\min}^2 \lesssim \tau_{\min} \sigma_{\mathbf{s}}^2. \tag{10}$$

The proofs are provided in Appendix E and F. In practice, we set $\sigma_{\min}$ at the upper bound, since smaller values already yield a posterior sufficiently close to the best attainable. Expressing the bound in terms of $\tau_{\min}$ ensures natural scaling with the dataset's spectral characteristics, yielding a robust stopping criterion.

3.2 Sampler hyperparameters

Similarly, we propose a reparameterization of the sampler hyperparameters for the RED-diff objective in (5). We show the usefulness of our proposed parameterization from an invariance and common descent direction [19]. Originally, RED-diff solves:

$$\min_{\boldsymbol{\mu}} \; f(\boldsymbol{\mu}) + \mathbb{E}_t[\lambda_t\, g(\boldsymbol{\mu}; t)], \quad f(\boldsymbol{\mu}) = \|\boldsymbol{y} - \mathcal{A}(\boldsymbol{\mu})\|_2^2.$$

At iteration k, let $\nabla f^k = \nabla f(\boldsymbol{\mu}^k)$, $\nabla g^k = \nabla g(\boldsymbol{\mu}^k; t^k)$, $\lambda^k = \lambda_{t_k}$. Then the stochastic gradient descent (SGD) update is

$$\boldsymbol{\mu}^{k+1} \leftarrow \boldsymbol{\mu}^k - \alpha^k \left(\nabla f^k + \lambda^k \nabla g^k\right) \tag{11}$$

Proposed reparameterization. Defining $r^k := \|\nabla f^k\|_2 / \|\nabla g^k\|_2$, we use $\alpha^k = \hat{\alpha}^k / \|\nabla f^k\|_2$, $\lambda^k = \hat{\lambda}\, r^k$ with $\hat{\alpha}^k, \hat{\lambda} > 0$, yielding the *unit-gradient* update

$$\Delta \boldsymbol{\mu}^k = -\hat{\alpha}^k \left(\frac{\nabla f^k}{\|\nabla f^k\|_2} + \hat{\lambda}\, \frac{\nabla g^k}{\|\nabla g^k\|_2} \right). \tag{12}$$

Theorem 3.5 (Re-scaling invariance)**.** *Let $\phi_f, \phi_g : \mathbb{R} \to \mathbb{R}$ be strictly increasing C^1 functions and set $\widetilde{f} = \phi_f \circ f$, $\widetilde{g} = \phi_g \circ g$. Applying* (12) *to $(\widetilde{f}, \widetilde{g})$ yields the same update as with (f, g); the step's direction and length are invariant to monotone reparameterizations. Proof in Appendix G.*

Corollary 3.6 (Practical advantage over RED-diff)**.** *In RED-diff, the prior gradient often can be modeled as $\nabla g(\boldsymbol{\mu}; t) = \phi'(t)\, q(\boldsymbol{\mu}; t)$ with a noise-level factor $\phi'(t)$ (e.g., via σ_t). Normalization cancels this factor:*

$$\frac{\nabla g(\boldsymbol{\mu}; t)}{\|\nabla g(\boldsymbol{\mu}; t)\|_2} = \frac{q(\boldsymbol{\mu}; t)}{\|q(\boldsymbol{\mu}; t)\|_2}.$$

Thus the data–prior balance is the constant $\hat{\lambda}$ across all t, eliminating the need to handcraft/tune an SNR-based schedule λ_t and avoiding searches over functional forms (linear, square-root, etc.).

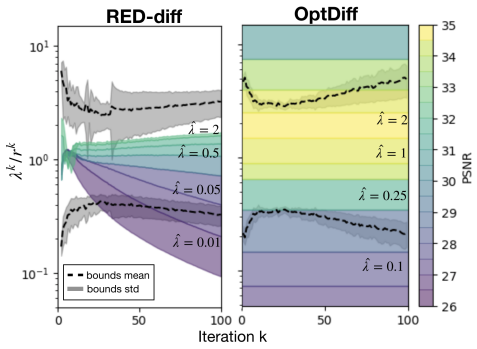

Figure 3: Common descent condition (Theorem 3.8). **(Left)** Standard λ^k SNR-weighting violates condition bounds, resulting in degraded performance. **(Right)** OptDiff yields consistent λ^k / r^k trajectories, simplifying tuning and improving PSNR.

Remark 3.7 (Interpretation). The update depends only on the directions of ∇f^k and ∇g^k, since both gradients are normalized. This cancels out any effect of scaling (e.g., gains, noise levels), so a single $\hat{\lambda}$ generalizes across diffusion levels.

This idea is related to *normalized SGD* methods [20, 21], which rescale updates to remove dependence on gradient magnitudes, and to the *Muon* optimizer [22], which enforces normalization and orthogonalization of updates to stabilize training. Our method brings these normalization principles to the diffusion-based composite objective: instead of normalizing a single gradient, we normalize *both components* (data consistency and diffusion prior) so that their relative contribution is controlled solely by the trade-off parameter $\hat{\lambda}$.

Theorem 3.8 (Common descent condition)**.** *Let $d^k = \nabla f^k + \lambda^k \nabla g^k$ where $\lambda^k > 0$ and $\cos\vartheta^k = \frac{\langle \nabla f^k, \nabla g^k \rangle}{\|\nabla f^k\|\, \|\nabla g^k\|}$. Then common descent ($\langle \nabla f^k, d^k \rangle > 0$ and $\langle \nabla g^k, d^k \rangle > 0$) holds if*

$$\boxed{\begin{cases} \cos\vartheta^k \geq 0: & \lambda^k > 0 \quad (\textit{all positive } \lambda^k \textit{ are valid}), \\ \cos\vartheta^k < 0: & \lambda^k \in \big(r^k(-\cos\vartheta^k),\ r^k(-1/\cos\vartheta^k)\big). \end{cases}}$$

Corollary 3.9 (Simplifications under two-sided normalization)**.** *Under* (12)*, $\lambda^k = \hat{\lambda}\, r^k$ with global $\hat{\lambda} > 0$, and Theorem 3.8 reduces to*

$$\boxed{\hat{\lambda} \in \big(-\cos\vartheta^k,\ -1/\cos\vartheta^k\big) \quad \textit{whenever } \cos\vartheta^k < 0.}$$

The case $\cos\vartheta^k \geq 0$ is equivalent to Theorem 3.8.

The proof is provided in Appendix H. We display averaged common descent bounds and different values for $\hat{\lambda}$ in Figure 3. A single global $\hat{\lambda}$ can guarantee common descent and performance drops whenever $\hat{\lambda}$ falls out of bounds.

4 OptDiff Pipeline

The OptDiff pipeline combines invariant reparameterizations (Section 3) with the RED-diff optimization perspective to enable a simplified and robust strategy for hyperparameter tuning. We demonstrate

its effectiveness by designing a family of efficient optimizers for inverse problems. Below, we outline the pipeline for a given inverse problem setup (e.g., MRI reconstruction with acceleration $R = 8$) and a chosen optimizer (e.g., vanilla gradient descent).

4.1 Hyperparameter tuning

A central component of OptDiff is a greedy-optimal hyperparameter scheme that uses a small tuning set (5 to 10 samples), similar to standard grid search, but additionally decouples noise schedule and sampler hyperparameters, and constrains the search space. This procedure also provides an upper-bound performance reference and enables the integration of convex optimization tools for accelerated posterior sampling (Section 5.2).

Greedy-optimal sampler hyperparameters. At each step with update rule from Eq. 11, define $\boldsymbol{U}^k := \begin{bmatrix} \nabla f^k & \nabla g^k \end{bmatrix}$ and $\boldsymbol{w}^k := \begin{bmatrix} \alpha^k & \alpha^k \lambda^k \end{bmatrix}^\mathsf{T}$, $\boldsymbol{x}^*$ is the corresponding ground-truth image from the small tuning dataset. We estimate the greedy-optimal hyperparameters $\hat{\boldsymbol{w}}^k$ as follows:

$$\hat{\boldsymbol{w}}^k = \underset{\boldsymbol{w} \in \mathbb{R}^2_+}{\arg\min} \|\boldsymbol{x}^* - (\boldsymbol{\mu}^k - \boldsymbol{U}^k \boldsymbol{w})\|_2^2, \tag{13}$$

which can be solved via non-negative least squares (NNLS). This methodology is flexible and can be extended to incorporate additional optimization techniques, e.g. momentum and preconditioning. Details are provided in Appendix I.

Noise schedule tuning. We perform a grid search over the tolerance parameters $\tau_{\min}$ and $\tau_{\max}$, which define the noise schedule bounds $\sigma_{\min}$ and $\sigma_{\max}$ via (8) and (10). For practical implementation, we assume a WSS signal model and compute radially averaged power spectral densities. During this step, the greedy-optimal sampler hyperparameters are used to decouple sampler effects, ensuring a fair and consistent comparison across candidate noise schedules.

Sampler hyperparameters tuning. Once the noise schedule is fixed, we perform a reduced grid search over the sampler hyperparameters $\hat{\alpha}^k$ and $\hat{\lambda}$. Compared to Mardani et al. [5], our invariant reparameterization eliminates the need to select a heuristic SNR-based schedule for λ_t, reducing the dimensionality of the search space. In practice, the optimal hyperparameter curves also serve as reference points, allowing the grid search to be restricted to a small neighborhood around these values. We further demonstrate the robustness of $\hat{\lambda}$ to tuning-set size and image selection through a sensitivity analysis in Appendix I (Table 4).

4.2 Inference

The selected hyperparameters can be directly applied to the test dataset and reused across different setups without further tuning, as shown in Figure 1. In particular, the same hyperparameters generalize across varying calibration regions, noise levels, accelerations, and sampling masks for the MRI reconstruction (Figure 1 and Table 3). In practice, only the step-size hyperparameter required minor adjustment between setups.

5 Experiments

5.1 Invariance tests

We evaluate the invariance property of our proposed reparameterizations using the fastMRI [23] dataset with the MRI reconstruction and deblurring tasks. We conducted the experiments on 10 slices from 10 different patients from the validation subset. The greedy-optimal sampler hyperparameters described in Subsection 4.1 were used to analyze the impact of each hyperparameter in isolation.

Noise schedule hyperparameters.

- **Start noise level ($\sigma_{\max}$).** From (7)-(8), the proposed $\sigma_{\max}$ depends on the low-frequency components through the conditional covariance $\boldsymbol{\Sigma}_{H|L}$ and, under the WSS assumption,

Table 1: Evaluation of noise schedule generalization across baseline solvers. Replacing the standard schedule with the optimized one improves reconstruction quality in most cases. Our method achieves the best overall performance.

Method	MRI Reconstruction R=8 (fastMRI)				Deblurring (ImageNet)					
	Standard schedule		Proposed schedule		Standard schedule			Proposed schedule		
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
DPS [4]	28.94 ± 1.54	0.774 ± 0.066	30.43 ± 1.85	0.803 ± 0.046	27.51 ± 4.92	0.734 ± 0.163	0.296 ± 0.162	38.17 ± 3.96	0.967 ± 0.013	0.007 ± 0.005
DDRM [24]					36.24 ± 3.23	0.953 ± 0.019	0.024 ± 0.013	39.83 ± 3.47	0.976 ± 0.010	0.009 ± 0.007
Pi-GDM [25]	28.88 ± 1.27	0.734 ± 0.078	29.29 ± 1.56	0.808 ± 0.052	33.85 ± 2.84	0.928 ± 0.029	0.027 ± 0.024	32.51 ± 2.42	0.899 ± 0.029	0.045 ± 0.020
RED-diff [5]	31.92 ± 1.44	0.840 ± 0.034	33.38 ± 2.10	0.888 ± 0.042	35.94 ± 4.03	0.955 ± 0.020	0.027 ± 0.016	40.05 ± 3.21	0.977 ± 0.008	$\mathbf{0.003 \pm 0.002}$
DAPS [7]	31.66 ± 1.49	0.870 ± 0.042	31.06 ± 2.44	0.788 ± 0.066	35.03 ± 2.27	0.921 ± 0.017	0.051 ± 0.035	35.96 ± 1.27	0.926 ± 0.024	0.048 ± 0.041
OptDiff (Ours)			$\mathbf{34.93 \pm 1.38}$	$\mathbf{0.921 \pm 0.029}$				$\mathbf{41.04 \pm 3.41}$	$\mathbf{0.982 \pm 0.006}$	0.004 ± 0.002

Table 2: Ablation studies to evaluate performance of (a) our invariant reparameterizations and (b) integration of advanced optimization tools. Our λ-reparameterization outperforms all SNR-based baselines. Momentum and preconditioning improve reconstruction quality.

(a) Invariant reparameterization ablations

λ method	Standard schedule		Proposed schedule	
	PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
Greedy-optimal	33.13 ± 1.22	0.904 ± 0.020	34.78 ± 1.24	0.941 ± 0.021
Constant	30.53 ± 1.31	0.732 ± 0.040	33.56 ± 1.49	0.910 ± 0.030
Linear	30.40 ± 1.35	0.846 ± 0.041	34.39 ± 1.24	0.928 ± 0.021
Square	29.62 ± 1.18	0.834 ± 0.024	32.35 ± 1.29	0.910 ± 0.025
Sqrt.	32.19 ± 1.24	0.835 ± 0.034	34.13 ± 1.31	0.921 ± 0.025
Log	31.98 ± 1.82	0.893 ± 0.025	34.36 ± 1.25	0.928 ± 0.021
Ours	$\mathbf{32.87 \pm 1.31}$	$\mathbf{0.895 \pm 0.026}$	$\mathbf{34.51 \pm 1.37}$	$\mathbf{0.927 \pm 0.025}$

(b) Optimizer ablations

		NFE 20		NFE 100	
		PSNR ↑	SSIM ↑	PSNR ↑	SSIM ↑
RED-diff		29.7 ± 1.2	0.84 ± 0.03	32.2 ± 1.2	0.84 ± 0.03
Greedy-Optimal	Vanilla	30.1 ± 1.7	0.89 ± 0.04	33.8 ± 1.4	0.92 ± 0.02
	Moment	31.8 ± 2.1	0.89 ± 0.04	34.6 ± 1.4	0.93 ± 0.02
	Precond	$\mathbf{32.3 \pm 2.2}$	$\mathbf{0.90 \pm 0.05}$	$\mathbf{34.6 \pm 1.4}$	$\mathbf{0.93 \pm 0.02}$
OptDiff	Vanilla	29.9 ± 1.7	0.88 ± 0.04	33.4 ± 1.6	0.91 ± 0.03
	Moment	31.7 ± 2.1	0.87 ± 0.05	34.4 ± 1.3	0.92 ± 0.02
	Precond	$\mathbf{32.3 \pm 2.1}$	$\mathbf{0.89 \pm 0.05}$	$\mathbf{34.5 \pm 1.4}$	$\mathbf{0.92 \pm 0.02}$

on the power spectral density. Different levels of acceleration and blurring modify $\mathbf{\Sigma}_{H|L}$, thereby shifting the proposed $\sigma_{\max}$. To assess invariance, we vary reconstruction and blurring parameters, and compare the $\sigma_{\max}$ values with their reparameterized counterparts $\tau_{\max}$.

- **Stop noise level** $\sigma_{\min}$**.** Similarly, Eq. 9 shows that the proposed $\sigma_{\min}$ depends on the measurement noise floor. We test invariance by varying the measurement noise level and comparing $\sigma_{\min}$ with the reparameterized $\tau_{\min}$.

Noise schedule transfer across solvers The bounds on $\sigma_{\max}$ and $\sigma_{\min}$ in Section 3.1 are deliberately derived from the diffusion denoising model, independently of any particular inverse solver. This motivates testing the resulting noise schedule beyond OptDiff. We evaluate several diffusion-based inverse solvers using both their standard schedules and our proposed schedule.

Sampler hyperparameters. As discussed in Section 3.2, inverse solving performance depends on the choice of the weighting function λ^k, which compensates for variations in the norm of ∇g. Our reparameterization cancels any positive rescaling of the prior gradient, thereby removing the need for a handcrafted schedule. To test this invariance, we examine whether multiple base values $\hat{\lambda} \in \mathbb{R}$ are required across the reverse diffusion process. We sweep $\hat{\lambda}$ for both the standard update in Eq. 5 with a linear schedule and the normalized update in Eq. 12, evaluating performance over three intervals: $k \in [0, 33]$, $[33, 66]$, and $[66, 99]$. We use the optimal step size α^k as described in subsection 4.1.

Results. As shown in Figure 1, the τ-parameterizations produce well-aligned curves across conditions, whereas the original σ-parameterizations show large variability. This confirms that our reparameterization scales naturally with the spectral properties of the data and the problem setup. Moreover, Table 1 demonstrates that the proposed noise schedule generalizes across solvers. In most cases, replacing the standard schedule with the optimized schedule improves reconstruction quality across baseline methods, while OptDiff achieves the best overall performance under the optimized schedule. Similarly, Figure 1 shows that with the standard formulation, the $\hat{\lambda}$ curves are dispersed across intervals, implying that multiple base values are needed. In contrast, our reparameterization aligns the curves, demonstrating that a single $\hat{\lambda}$ suffices across the entire reverse process by accounting for the dynamics of $\|\nabla g\|_2$.

Table 3: Comparison with baselines using NFE-100. Our method consistently achieves the best or comparable performance across all task–dataset pairs. Best results are shown in bold, and second-best results are underlined.

		fastMRI			
		Acceleration R=4		Acceleration R=8	
Task	**Method**	**PSNR ↑**	**SSIM ↑**	**PSNR ↑**	**SSIM ↑**
MRI recon	DPS [4]	35.82 ± 1.46	0.775 ± 0.069	28.94 ± 1.54	0.774 ± 0.066
	DiffPIR [26]	30.33 ± 0.97	0.691 ± 0.077	27.17 ± 1.08	0.650 ± 0.072
	Pi-GDM [25]	30.69 ± 1.37	0.753 ± 0.086	28.88 ± 1.27	0.734 ± 0.078
	RED-diff [5]	36.79 ± 1.43	0.872 ± 0.035	31.92 ± 1.44	0.840 ± 0.034
	DDS [27]	33.72 ± 1.14	0.930 ± 0.030	28.92 ± 1.28	0.828 ± 0.027
	DAPS [7]	36.50 ± 1.63	0.915 ± 0.033	31.66 ± 1.49	0.870 ± 0.042
	OptDiff (Ours)	$\mathbf{38.45 \pm 1.58}$	$\mathbf{0.948 \pm 0.023}$	$\mathbf{34.93 \pm 1.38}$	$\mathbf{0.921 \pm 0.029}$

		ImageNet			FFHQ		
Task	**Method**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
Deblurring kernel 6	DPS [4]	24.25 ± 3.97	0.602 ± 0.192	0.420 ± 0.187	33.00 ± 2.50	0.898 ± 0.034	0.078 ± 0.028
	DDRM [24]	32.71 ± 3.82	0.908 ± 0.045	0.039 ± 0.017	36.96 ± 1.76	$\mathbf{0.949 \pm 0.013}$	0.030 ± 0.009
	DDNM [28]	32.60 ± 3.89	0.906 ± 0.045	$\mathbf{0.030 \pm 0.013}$	36.95 ± 1.78	$\mathbf{0.949 \pm 0.013}$	$\mathbf{0.027 \pm 0.008}$
	DiffPIR [26]	32.22 ± 3.68	0.892 ± 0.049	0.061 ± 0.032	36.58 ± 1.76	0.942 ± 0.014	0.040 ± 0.012
	RED-diff [5]	29.35 ± 3.92	0.821 ± 0.092	0.175 ± 0.089	35.07 ± 1.77	0.915 ± 0.015	0.063 ± 0.020
	DAPS [7]	31.39 ± 3.13	0.853 ± 0.047	0.107 ± 0.033	34.13 ± 1.39	0.888 ± 0.013	0.088 ± 0.021
	OptDiff (Ours)	$\mathbf{33.28 \pm 3.74}$	$\mathbf{0.914 \pm 0.038}$	0.049 ± 0.024	$\mathbf{37.10 \pm 1.79}$	$\mathbf{0.949 \pm 0.013}$	0.058 ± 0.017
Severe deblur kernel 12	DPS [4]	21.30 ± 3.27	0.484 ± 0.200	0.371 ± 0.159	25.88 ± 6.47	0.749 ± 0.129	0.159 ± 0.115
	DDRM [24]	27.87 ± 3.89	0.785 ± 0.110	0.156 ± 0.074	32.18 ± 2.18	0.889 ± 0.031	0.069 ± 0.021
	DDNM [28]	28.07 ± 4.12	$\mathbf{0.789 \pm 0.110}$	$\mathbf{0.134 \pm 0.059}$	32.54 ± 2.15	$\mathbf{0.893 \pm 0.030}$	0.074 ± 0.022
	DiffPIR [26]	27.41 ± 3.72	0.746 ± 0.114	0.145 ± 0.064	32.05 ± 1.95	0.870 ± 0.030	$\mathbf{0.060 \pm 0.019}$
	RED-diff [5]	24.19 ± 3.49	0.604 ± 0.171	0.428 ± 0.156	29.42 ± 2.23	0.800 ± 0.048	0.183 ± 0.054
	DAPS [7]	26.28 ± 2.80	0.671 ± 0.086	0.247 ± 0.075	28.38 ± 2.19	0.786 ± 0.055	0.199 ± 0.040
	OptDiff (Ours)	$\mathbf{28.27 \pm 3.90}$	0.788 ± 0.100	0.190 ± 0.083	$\mathbf{32.61 \pm 2.22}$	0.891 ± 0.031	0.127 ± 0.034
Super-res 4×	DPS [4]	23.72 ± 4.90	0.618 ± 0.209	0.299 ± 0.207	30.50 ± 2.62	0.866 ± 0.044	$\mathbf{0.071 \pm 0.025}$
	DDRM [24]	25.84 ± 3.86	0.702 ± 0.152	0.331 ± 0.145	31.68 ± 2.50	0.885 ± 0.038	0.125 ± 0.038
	DDNM [28]	25.79 ± 3.93	0.703 ± 0.152	0.286 ± 0.136	30.94 ± 2.45	0.872 ± 0.041	0.085 ± 0.028
	DiffPIR [26]	25.28 ± 3.78	0.682 ± 0.152	0.242 ± 0.117	31.06 ± 2.44	0.873 ± 0.040	0.092 ± 0.031
	RED-diff [5]	25.75 ± 3.82	0.700 ± 0.141	0.311 ± 0.144	31.25 ± 2.37	0.875 ± 0.037	0.108 ± 0.034
	DAPS [7]	$\mathbf{26.50 \pm 3.43}$	0.706 ± 0.132	$\mathbf{0.239 \pm 0.067}$	29.63 ± 1.73	0.763 ± 0.028	0.179 ± 0.028
	OptDiff (Ours)	26.07 ± 3.96	$\mathbf{0.715 \pm 0.148}$	0.350 ± 0.156	$\mathbf{31.99 \pm 2.57}$	$\mathbf{0.891 \pm 0.037}$	0.147 ± 0.043
Super-res 16×	DPS [4]	17.95 ± 3.02	0.365 ± 0.192	$\mathbf{0.388 \pm 0.125}$	22.62 ± 2.28	0.664 ± 0.086	0.286 ± 0.078
	DDRM [24]	20.35 ± 2.77	0.462 ± 0.185	0.683 ± 0.199	23.50 ± 2.24	0.687 ± 0.083	0.316 ± 0.090
	DDNM [28]	20.15 ± 2.68	0.460 ± 0.183	0.633 ± 0.207	22.86 ± 2.12	0.663 ± 0.083	0.249 ± 0.079
	DiffPIR [26]	19.66 ± 2.61	0.439 ± 0.178	0.577 ± 0.194	22.77 ± 2.14	0.658 ± 0.085	$\mathbf{0.246 \pm 0.077}$
	RED-diff [5]	20.34 ± 2.76	0.460 ± 0.182	0.704 ± 0.193	23.01 ± 2.07	0.663 ± 0.082	0.282 ± 0.082
	DAPS [7]	$\mathbf{20.53 \pm 2.49}$	0.418 ± 0.136	0.522 ± 0.081	22.63 ± 1.96	0.572 ± 0.064	0.358 ± 0.046
	OptDiff (Ours)	20.44 ± 2.75	$\mathbf{0.468 \pm 0.184}$	0.704 ± 0.186	$\mathbf{23.67 \pm 2.25}$	$\mathbf{0.689 \pm 0.083}$	0.343 ± 0.094

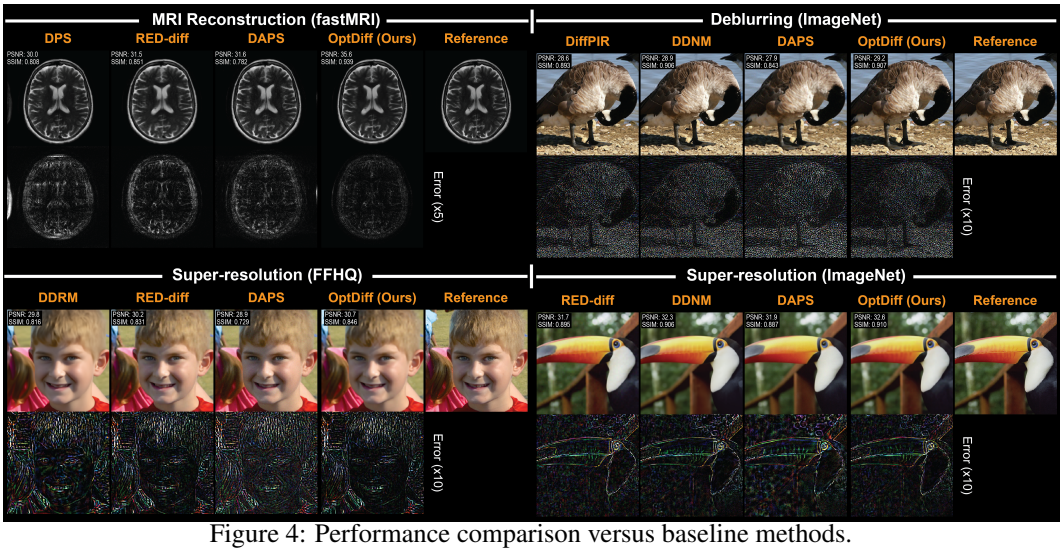

Figure 4: Performance comparison versus baseline methods.

5.2 Ablations

We conduct ablations to evaluate (a) the performance of our invariant reparameterizations, i.e. λ-weighting and noise schedule design , and (b) the effect of incorporating advanced optimization tools to the RED-diff framework. As in Section 5.1, we use MRI reconstruction at acceleration factor $R = 8$ with 100 iterations as the base task. For (a), we compare fixed SNR-based weightings $\lambda_k = h(\mathrm{SNR}_k)$ against our reparameterization under both the default EDM noise schedule ($\sigma_{\min} = 0.002$, $\sigma_{\max} = 20$) and the proposed schedule. For (b), we assess the impact of momentum and polynomial preconditioning at 20 and 100 NFE. In standard diffusion solvers, such methods typically introduce additional hyperparameters; however, our invariant formulation and hyperparameter tuning strategy enables their integration without increasing tuning complexity.

Results. Table 2a shows that our λ-reparameterization consistently outperforms all SNR-based baselines under both standard and optimized noise schedules, with the oracle λ (computed using the optimal hyperparameters in Subsection 4.1) serving as an upper bound. Table 2b further demonstrates that momentum and preconditioning consistently improve reconstruction quality over the vanilla optimizer, with larger gains at 20 steps and diminishing but still positive gains at NFE-100.

5.3 Tests with Baseline Methods

We conduct two main experiments. First, we evaluate whether the proposed noise-schedule design generalizes to state-of-the-art diffusion-based inverse solvers beyond RED-diff and OptDiff. Second, we compare the reconstruction performance of our designed optimizers against these baseline solvers. Experiments are performed on three datasets: fastMRI, ImageNet, and FFHQ, across four inverse problems: MRI reconstruction, deblurring, super-resolution, and inpainting. Details in Appendix J.

Results. Figure 4 and Table 3 compare OptDiff against existing diffusion-based inverse solvers. Across tasks and datasets, OptDiff achieves the best or comparable PSNR, SSIM, and LPIPS values relative to the baselines. In addition, OptDiff provides a favorable speed–quality trade-off. Our method either matches the fastest runtime or remains competitive while achieving higher reconstruction quality.

6 Limitations

We briefly summarize the main limitations here and provide a more detailed discussion with future directions in Appendix K. First, the proposed noise-schedule design is most effective when the forward model preserves sufficient low-frequency information, enabling a reduced $\sigma_{\max}$. This holds for the main tasks considered in this work, but may be less effective for tasks such as inpainting. Second, OptDiff inherits limitations from RED-diff and other optimization-based posterior samplers that produce a single reconstruction, potentially favoring pixelwise fidelity over perceptual quality in highly ill-posed tasks with multiple plausible solutions [11, 29]. This tendency may be further reinforced by our MSE-based hyperparameter tuning. Finally, while the noise-schedule hyperparameters transfer naturally to other inverse solvers, the sampler-hyperparameters are more specific to optimization-based formulations. Extending this component to broader classes of posterior samplers is left for future work.

7 Conclusion

We introduced OptDiff, a framework that improves the robustness and efficiency of diffusion-based inverse solvers through principled reparameterizations and optimization tools. By inducing invariances and reducing the effective hyperparameter search space, OptDiff enables stable performance across inverse problem with minimal retuning. The results also suggest future directions for incorporating more advanced optimization techniques into diffusion-based posterior sampling [30, 31].

8 Acknowledgements

This work was supported in part by the National Science Foundation (NSF) CAREER Award under Grant CCF-2236829, in part by the NSF under Grant CNS-2551282, in part by the National Institutes of Health under Grants R01EB036127 and 1R01AG089509-01A1, in part by the Office of Naval Research under Grant N00014-24-1-2164, and in part by the Defense Advanced Research Projects Agency under Grant HR00112490441.

References

[1] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=St1giarCHLP`.

[2] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851, 2020.

[3] Ajil Jalal, Marius Arvinte, Giannis Daras, Eric Price, Alexandros G. Dimakis, and Jonathan I. Tamir. Robust compressed sensing MRI with deep generative priors. In *Advances in Neural Information Processing Systems*, 2021.

[4] Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong C. Ye. Diffusion posterior sampling for general noisy inverse problems. In *International Conference on Learning Representations*, 2022.

[5] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*, 2023.

[6] Hongkai Zheng, Wenda Chu, Bingliang Zhang, Zihui Wu, Austin Wang, Berthy T. Feng, Caifeng Zou, Yu Sun, Nikola Kovachki, Zachary E. Ross, et al. InverseBench: Benchmarking plug-and-play diffusion priors for inverse problems in physical sciences. *arXiv preprint arXiv:2503.11043*, 2025.

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

[8] Greg Yang and Edward J. Hu. Feature learning in infinite-width neural networks. *arXiv preprint arXiv:2011.14522*, 2020.

[9] Sander Dieleman. Diffusion is spectral autoregression, 2024. URL `https://sander.ai/2024/09/02/spectral-autoregression.html`.

[10] Fabian Falck, Teodora Pandeva, Kiarash Zahirnia, Rachel Lawrence, Richard Turner, Edward Meeds, Javier Zazo, and Sushrut Karmalkar. A Fourier space perspective on diffusion models. *arXiv preprint arXiv:2505.11278*, 2025.

[11] 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.

[12] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020.

[13] Brian D. O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

[14] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011.

[15] Yang Song, Liyue Shen, Lei Xing, and Stefano Ermon. Solving inverse problems in medical imaging with score-based generative models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=vaRCHVj0uGI.

[16] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[17] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Advances in Neural Information Processing Systems*, volume 35, pages 26565–26577, 2022.

[18] A. van der Schaaf and J. H. van Hateren. Modelling the power spectra of natural images: Statistics and information. *Vision Research*, 36(17):2759–2770, 1996.

[19] Jörg Fliege and Benar F. Svaiter. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research*, 51(3):479–494, 2000.

[20] Ryan Murray, Brian Swenson, and Soummya Kar. Revisiting normalized gradient descent: Fast evasion of saddle points. *arXiv preprint arXiv:1711.05224*, 2017. URL https://arxiv.org/abs/1711.05224.

[21] Ashok Cutkosky and Harsh Mehta. Momentum improves normalized SGD. In *International Conference on Machine Learning (ICML)*, 2020. URL https://proceedings.mlr.press/v119/cutkosky20b/cutkosky20b.pdf.

[22] Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024. URL https://kellerjordan.github.io/posts/muon/.

[23] Florian Knoll, Jure Zbontar, Anuroop Sriram, Matthew J. Muckley, Mary Bruno, Aaron Defazio, Marc Parente, Krzysztof J. Geras, Joe Katsnelson, Hersh Chandarana, et al. fastMRI: A publicly available raw k-space and DICOM dataset of knee images for accelerated MR image reconstruction using machine learning. *Radiology: Artificial Intelligence*, 2(1):e190007, 2020.

[24] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems*, volume 35, pages 23593–23606, 2022.

[25] Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

[26] Yuanzhi Zhu, Kai Zhang, Jingyun Liang, Jiezhang Cao, Bihan Wen, Radu Timofte, and Luc Van Gool. Denoising diffusion models for plug-and-play image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1219–1229, June 2023.

[27] Hyungjin Chung, Suhyeon Lee, and Jong Chul Ye. Decomposed diffusion sampler for accelerating large-scale inverse problems. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=DsEhqQtfAG.

[28] Yinhuai Wang, Jiwen Yu, and Jian Zhang. Zero-shot image restoration using denoising diffusion null-space model. *The Eleventh International Conference on Learning Representations*, 2023.

[29] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6228–6237, 2018.

[30] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, 2nd edition, 2006.

[31] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[32] Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Inc., 1993.

[33] Athanasios Papoulis and S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, 4th edition, 2002.

[34] Siddharth Iyer, Frank Ong, Kawin Setsompop, Mariya Doneva, and Michael Lustig. SURE-based automatic parameter selection for ESPIRiT calibration. *Magnetic Resonance in Medicine*, 84(6):3423–3437, 2020.

[35] Daniel S. Weller, Sathish Ramani, Jon-Fredrik Nielsen, and Jeffrey A. Fessler. Monte Carlo SURE-based parameter selection for parallel magnetic resonance imaging reconstruction. *Magnetic Resonance in Medicine*, 71(5):1760–1770, 2014.

[36] Batu Ozturkler, Chao Liu, Benjamin Eckart, Morteza Mardani, Jiaming Song, and Jan Kautz. SMRD: SURE-based robust MRI reconstruction with diffusion models. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 199–209. Springer, 2023.

[37] Olin G Johnson, Charles A Micchelli, and George Paul. Polynomial preconditioners for conjugate gradient calculations. *SIAM Journal on Numerical Analysis*, 20(2):362–376, 1983.

[38] Zulfiquar Ali Bhotto, M. Omair Ahmad, and M. N. S. Swamy. An improved fast iterative shrinkage thresholding algorithm for image deblurring. *SIAM Journal on Imaging Sciences*, 8 (3):1640–1657, 2015.

[39] Siddharth S. Iyer, Frank Ong, Xiaozhi Cao, Congyu Liao, Luca Daniel, Jonathan I. Tamir, and Kawin Setsompop. Polynomial preconditioners for regularized linear inverse problems. *SIAM Journal on Imaging Sciences*, 17(1):116–146, 2024.

[40] Giannis Daras, Kulin Shah, Yuval Dagan, Aravind Gollakota, Alex Dimakis, and Adam Klivans. Ambient diffusion: Learning clean distributions from corrupted data. In *Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=wBJBLy9kBY`.

[41] Martin Uecker, Peng Lai, Mark J. Murphy, Patrick Virtue, Michael Elad, John M. Pauly, Shreyas S. Vasanawala, and Michael Lustig. ESPIRiT—an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA. *Magnetic Resonance in Medicine*, 71(3): 990–1001, 2014.

[42] Santiago Aja-Fernández, Antonio Tristán-Vega, and Carlos Alberola-López. Noise estimation in single- and multiple-coil magnetic resonance data based on statistical models. *Magnetic Resonance Imaging*, 27(10):1397–1409, 2009.

[43] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. ILVR: Conditioning method for denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[44] Ismail Alkhouri, Shijun Liang, Cheng-Han Huang, Jimmy Dai, Qing Qu, Saiprasad Ravishankar, and Rongrong Wang. SITCOM: Step-wise triple-consistent diffusion sampling for inverse problems. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=28Essvtvkw`.

[45] Hengkang Wang, Xu Zhang, Taihui Li, Yuxiang Wan, Tiancong Chen, and Ju Sun. DMPlug: A plug-in method for solving inverse problems with diffusion models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL `https://openreview.net/forum?id=81IFFsfQUj`.

A Notation glossary

This glossary summarizes the principal notation used in the main text and appendices. The diffusion time t and the discrete iteration index k are distinguished throughout; symbols introduced for individual proofs retain their local definitions.

Notation	Meaning
Inverse problems and diffusion	
$\boldsymbol{x}_0, \boldsymbol{x}_t$	Data image and its perturbed state at diffusion time t.
$\boldsymbol{y}, \mathcal{A}$	Measurements and measurement operator, with $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_0) + \boldsymbol{\eta}$.
n, m	Image and measurement dimensions, respectively.
$\boldsymbol{\eta}, \sigma_{\boldsymbol{\eta}}^2$	Gaussian measurement noise and its variance per component, with covariance $\sigma_{\boldsymbol{\eta}}^2 \mathbf{I}$.
t, T	Continuous diffusion time and terminal time.
p_{data}, p_t	Data distribution and marginal distribution at time t.
$\nabla_{\boldsymbol{x}_t} \log p_t$	Score of the perturbed data distribution.
$\boldsymbol{b}(\boldsymbol{x}_t, t), a(t)$	Vector drift and squared diffusion amplitude in the forward SDE; the noise amplitude is $\sqrt{a(t)}$.
$\boldsymbol{W}_t, \bar{\boldsymbol{W}}_t$	Forward and reverse-time Wiener processes.
$\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t)$	Noise-prediction network with learned parameters $\boldsymbol{\theta}$.
$\boldsymbol{n}, \boldsymbol{\epsilon}$	Standard Gaussian noise vectors in the perturbation and sampling formulations.
σ_t	Standard deviation of the diffusion perturbation.
$\sigma_{\max}, \sigma_{\min}$	Starting and stopping noise levels of reverse diffusion.
Spectral quantities and noise-schedule criteria	
$\omega, S(\omega)$	Spatial frequency and signal power spectral density.
Ω_H	Set of unknown high frequencies.
$\boldsymbol{\Sigma}, \boldsymbol{\Sigma}_{H\|L}$	Prior covariance and LMMSE error covariance of HF components after using exactly known LF components.
$\Phi(\boldsymbol{\Sigma}; \sigma)$	LMMSE error covariance for a signal with covariance $\boldsymbol{\Sigma}$ observed with additive white noise of covariance $\sigma^2 \mathbf{I}$, uncorrelated with the signal.
$\boldsymbol{\Delta}_t^{\max}$	Reduction from prior HF uncertainty: $\boldsymbol{\Sigma}_{H\|L} - \Phi(\boldsymbol{\Sigma}_{H\|L}; \sigma_t)$.
$\tau_{\max}, \tau_{\min}$	Relative covariance tolerances, referenced to $\boldsymbol{\Sigma}_{H\|L}$ at the start and $\boldsymbol{\Sigma}_0$ at the stop.
$\boldsymbol{\Sigma}_{\mathbf{s}}, \sigma_{\mathbf{s}}^2$	Clean-signal covariance and effective white noise floor variance in the image domain used in the stopping analysis.
$\boldsymbol{\Sigma}_0$	In the stopping analysis, limiting LMMSE error covariance as $\sigma_t \to 0$: $\Phi(\boldsymbol{\Sigma}_{\mathbf{s}}; \sigma_{\mathbf{s}})$. In Appendix D, this symbol instead denotes $\mathrm{Cov}(\boldsymbol{x}_0)$.
$\boldsymbol{\Delta}_t^{\min}$	Excess LMMSE error covariance above $\boldsymbol{\Sigma}_0$ when the total noise variance is $\sigma_{\mathbf{s}}^2 + \sigma_t^2$.
$\nu_{\max}(\cdot)$	Largest eigenvalue of a covariance matrix.
$\kappa(\sigma_{\mathbf{s}})$	Upper bound on $\sigma_{\min}^2$ when $\nu_{\max}(\boldsymbol{\Sigma}_{\mathbf{s}}) > \tau_{\min}\sigma_{\mathbf{s}}^2$; otherwise the stopping bound is vacuous.
Sampler reparameterization and common descent	
$k, \boldsymbol{\mu}^k$	Iteration index and current reconstruction estimate, represented by the variational mean.
f, g	Data-consistency objective and diffusion-prior regularizer.
$\nabla f^k, \nabla g^k$	Gradients evaluated at the current iterate and, for g, the current diffusion noise level.
α^k, λ^k	Conventional step size and regularization weight.
r^k	Gradient-norm ratio, $\|\nabla f^k\|_2 / \|\nabla g^k\|_2$.
$\hat{\alpha}^k, \hat{\lambda}$	Reparameterized step scale and trade-off weight: $\alpha^k = \hat{\alpha}^k / \|\nabla f^k\|_2$ and $\lambda^k = \hat{\lambda} r^k$.
$\Delta\boldsymbol{\mu}^k$	Reconstruction update, $\boldsymbol{\mu}^{k+1} - \boldsymbol{\mu}^k$.
ϑ^k	Angle between the data-consistency and diffusion-prior gradients.

Continued on next page

Notation	Meaning
d^k	Combined gradient, $\nabla f^k + \lambda^k \nabla g^k$, in the common-descent analysis; the vanilla SGD update is $-\alpha^k d^k$.
Greedy-optimal tuning and solver acceleration	
$\boldsymbol{x}^*$	Ground-truth reference image used for tuning.
$\boldsymbol{U}^k$	Matrix of update directions, including momentum or preconditioned gradients where applicable.
$\boldsymbol{w}^k, \hat{\boldsymbol{w}}^k$	Update coefficients and their greedy-optimal NNLS estimates. For vanilla updates, $\boldsymbol{w}^k = [\alpha^k, \alpha^k \lambda^k]^\mathsf{T}$.
$\boldsymbol{v}^k, \beta^k$	Momentum state and its coefficient; $\boldsymbol{v}^{k+1}$ is subtracted from $\boldsymbol{\mu}^k$, so $\Delta\boldsymbol{\mu}^k = -\boldsymbol{v}^{k+1}$.
$\mathbf{A}$	Matrix representation of a linear measurement operator.
$\mathrm{pol}(\mathbf{A}^\mathsf{T}\mathbf{A})$	Polynomial preconditioner applied to the data-consistency gradient.
N, NFE	Number of diffusion steps and number of neural function evaluations.
LMMSE analysis and matrix notation	
$\boldsymbol{u}, \boldsymbol{v}, \mathbf{B}$	Signal, observation, and linear observation matrix in the appendix LMMSE model.
$\mathbf{D}$	LMMSE linear estimator mapping observations to signal estimates.
$\boldsymbol{u}_L, \boldsymbol{u}_H$	Low- and high-frequency components of a signal.
$\boldsymbol{\Sigma}_{\boldsymbol{u}}, \boldsymbol{\Sigma}_{\boldsymbol{uv}}$	Covariance of $\boldsymbol{u}$ and cross-covariance of $\boldsymbol{u}$ and $\boldsymbol{v}$.
$\boldsymbol{\Sigma}_{\boldsymbol{u}\mid\boldsymbol{v}}$	LMMSE error covariance for estimating $\boldsymbol{u}$ from $\boldsymbol{v}$.
$\mathbf{I}, (\cdot)^\mathsf{T}, (\cdot)^\dagger$	Identity matrix, transpose, and Moore–Penrose pseudoinverse.
$\preceq, \succeq$	Positive-semidefinite matrix ordering.
$\|\cdot\|_2, \langle\cdot,\cdot\rangle$	Euclidean norm and inner product for vectors.

B LMMSE-estimator and the conditional covariance

Consider a linear observation model with Gaussian noise

$$\boldsymbol{v} = \mathbf{B}\boldsymbol{u} + \sigma\,\boldsymbol{n}, \qquad \boldsymbol{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \qquad \mathbb{E}[\boldsymbol{u}] = \mathbb{E}[\boldsymbol{v}] = \mathbf{0}, \tag{14}$$

with $\mathrm{Cov}(\boldsymbol{u}, \boldsymbol{n}) = \mathbf{0}$. For random vectors $\boldsymbol{p}, \boldsymbol{q}$ we write

$$\mathrm{Cov}(\boldsymbol{p}) = \mathbb{E}[\boldsymbol{p}\boldsymbol{p}^\mathsf{T}], \qquad \mathrm{Cov}(\boldsymbol{p}, \boldsymbol{q}) = \mathbb{E}[\boldsymbol{p}\boldsymbol{q}^\mathsf{T}].$$

Let $\boldsymbol{\Sigma}_{\boldsymbol{u}} = \mathrm{Cov}(\boldsymbol{u})$. Linearity of expectations gives

$$\boldsymbol{\Sigma}_{\boldsymbol{v}} = \mathbf{B}\boldsymbol{\Sigma}_{\boldsymbol{u}}\mathbf{B}^\mathsf{T} + \sigma^2\mathbf{I}, \qquad \boldsymbol{\Sigma}_{\boldsymbol{uv}} = \mathrm{Cov}(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{\Sigma}_{\boldsymbol{u}}\mathbf{B}^\mathsf{T}. \tag{15}$$

The LMMSE estimator [32] is the linear map $\mathbf{D}$ that minimizes $\mathbb{E}\big[\|\boldsymbol{u} - \mathbf{D}\boldsymbol{v}\|^2\big]$. By the orthogonality principle,

$$\mathbf{D} = \boldsymbol{\Sigma}_{\boldsymbol{uv}}\,\boldsymbol{\Sigma}_{\boldsymbol{v}}^\dagger. \tag{16}$$

The corresponding error covariance is

$$\boldsymbol{\Sigma}_{\boldsymbol{u}\mid\boldsymbol{v}} = \mathrm{Cov}(\boldsymbol{u} - \hat{\boldsymbol{u}}) = \boldsymbol{\Sigma}_{\boldsymbol{u}} - \boldsymbol{\Sigma}_{\boldsymbol{uv}}\,\boldsymbol{\Sigma}_{\boldsymbol{v}}^\dagger\,\boldsymbol{\Sigma}_{\boldsymbol{vu}}. \tag{17}$$

This matrix quantifies the reduction in uncertainty about $\boldsymbol{u}$ after observing $\boldsymbol{v}$. It is positive semidefinite $\boldsymbol{\Sigma}_{\boldsymbol{u}\mid\boldsymbol{v}} \succeq 0$ and $\boldsymbol{\Sigma}_{\boldsymbol{u}\mid\boldsymbol{v}} \preceq \boldsymbol{\Sigma}_{\boldsymbol{u}}$, making it a natural measure of estimator quality. Smaller values indicate a more informative observation and a more accurate estimation. For an identity observation $\mathbf{B} = \mathbf{I}$, $\boldsymbol{\Sigma}_{\boldsymbol{u}\mid\boldsymbol{v}}$ reduces directly to the posterior map $\Phi(\boldsymbol{\Sigma}_{\boldsymbol{u}}; \sigma)$ used in the main text in (6).

B.1 Conditional covariance $\boldsymbol{\Sigma}_{H\mid L}$

We now apply the LMMSE framework to quantify the uncertainty in the high-frequency content of a signal after its low-frequency content is known. Let $\boldsymbol{u}$ be a zero-mean random vector, and let $\boldsymbol{u}_L$ and $\boldsymbol{u}_H$ denote the components of $\boldsymbol{u}$ associated with the low- and high-frequency subspaces, respectively. Viewing $\boldsymbol{u}_H$ as the unknown quantity and $\boldsymbol{u}_L$ as an available noise-less observation, we may treat

$\boldsymbol{u}_L$ as the vector $\boldsymbol{v}$ in the LMMSE model ((14)) with observation matrix $\mathbf{B}$ equal to the selector that extracts the LF components. The LMMSE error covariance ((17)) directly gives the conditional HF covariance

$$\boldsymbol{\Sigma}_{H|L} = \boldsymbol{\Sigma}_H - \boldsymbol{\Sigma}_{HL}\,\boldsymbol{\Sigma}_L^{\dagger}\,\boldsymbol{\Sigma}_{LH}, \tag{18}$$

where $\boldsymbol{\Sigma}_H = \mathrm{Cov}(\boldsymbol{u}_H)$, $\boldsymbol{\Sigma}_L = \mathrm{Cov}(\boldsymbol{u}_L)$, and $\boldsymbol{\Sigma}_{HL} = \mathrm{Cov}(\boldsymbol{u}_H, \boldsymbol{u}_L)$. This matrix characterizes the uncertainty of the HF component when LF information is known. When the LF and HF components are uncorrelated ($\boldsymbol{\Sigma}_{HL} = 0$), the conditional covariance reduces to $\boldsymbol{\Sigma}_{H|L} = \boldsymbol{\Sigma}_H$, indicating that the LF part carries no predictive information about the HF part.

C Proof of Theorem 3.1 (Start bound)

Recall that

$$\boldsymbol{\Delta}_t^{\max} = \boldsymbol{\Sigma}_{H|L} - \Phi(\boldsymbol{\Sigma}_{H|L}; \sigma_t) = \boldsymbol{\Sigma}_{H|L}(\boldsymbol{\Sigma}_{H|L} + \sigma_t^2\mathbf{I})^{-1}\boldsymbol{\Sigma}_{H|L},$$

where we used the posterior identity

$$\Phi(\boldsymbol{\Sigma}; \sigma) = \boldsymbol{\Sigma} - \boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \sigma^2\mathbf{I})^{-1}\boldsymbol{\Sigma}.$$

For brevity, write

$$\boldsymbol{\Sigma} := \boldsymbol{\Sigma}_{H|L}, \qquad \boldsymbol{\Delta} := \boldsymbol{\Delta}_t^{\max}.$$

Eigen-decomposition and diagonalization. Let

$$\boldsymbol{\Sigma} = \boldsymbol{q}\,\Lambda\,\boldsymbol{q}^{\mathsf{T}}, \qquad \Lambda = \mathrm{diag}(\nu_1, \ldots, \nu_D), \qquad \nu_i \geq 0. \tag{19}$$

Because

$$\boldsymbol{\Sigma} + \sigma_t^2\mathbf{I} = \boldsymbol{q}(\Lambda + \sigma_t^2\mathbf{I})\boldsymbol{q}^{\mathsf{T}}, \qquad (\boldsymbol{\Sigma} + \sigma_t^2\mathbf{I})^{-1} = \boldsymbol{q}(\Lambda + \sigma_t^2\mathbf{I})^{-1}\boldsymbol{q}^{\mathsf{T}},$$

we obtain

$$\boldsymbol{\Delta} = \boldsymbol{\Sigma}(\boldsymbol{\Sigma} + \sigma_t^2\mathbf{I})^{-1}\boldsymbol{\Sigma} \tag{20}$$

$$= \boldsymbol{q}\,\Lambda(\Lambda + \sigma_t^2\mathbf{I})^{-1}\Lambda\,\boldsymbol{q}^{\mathsf{T}}. \tag{21}$$

Thus $\boldsymbol{\Sigma}$ and $\boldsymbol{\Delta}$ are simultaneously diagonalizable with eigenvectors $\boldsymbol{q}$, and the eigenvalues of $\boldsymbol{\Delta}$ are

$$\frac{\nu_i^2}{\nu_i + \sigma_t^2}, \qquad i = 1, \ldots, D. \tag{22}$$

Matrix inequality as scalar inequalities. The condition

$$\boldsymbol{\Delta} \preceq \tau_{\max}\,\boldsymbol{\Sigma}$$

is equivalent, in the common eigenbasis, to

$$\frac{\nu_i^2}{\nu_i + \sigma_t^2} \;\leq\; \tau_{\max}\,\nu_i \qquad \text{for all eigenvalues } \nu_i > 0. \tag{23}$$

Solving the scalar inequality. Fix any $\nu > 0$. Inequality (23) becomes

$$\frac{\nu^2}{\nu + \sigma_t^2} \leq \tau_{\max}\nu.$$

Cancelling the factor $\nu > 0$ gives

$$\frac{\nu}{\nu + \sigma_t^2} \leq \tau_{\max}.$$

Multiply both sides by $\nu + \sigma_t^2$:

$$\nu \;\leq\; \tau_{\max}(\nu + \sigma_t^2) = \tau_{\max}\nu + \tau_{\max}\sigma_t^2.$$

Rearranging yields

$$(1 - \tau_{\max})\,\nu \;\leq\; \tau_{\max}\,\sigma_t^2.$$

Since $\tau_{\max} > 0$,

$$\sigma_t^2 \;\geq\; \frac{1 - \tau_{\max}}{\tau_{\max}}\,\nu.$$

Worst-case mode and conclusion. The bound must hold for every eigenvalue ν_i. Because the right-hand side increases with ν, the most restrictive condition occurs at the largest eigenvalue $\nu_{\max}(\mathbf{\Sigma}_{H|L})$. Thus

$$\sigma_t^2 \;\geq\; \frac{1-\tau_{\max}}{\tau_{\max}}\,\nu_{\max}(\mathbf{\Sigma}_{H|L}).$$

Identifying the minimal feasible starting noise level with $\sigma_{\max}$ proves the claim.

D Proof of Corollary 3.2 (WSS/PSD assumption)

Under the WSS assumption, the covariance of $\boldsymbol{x}_0$ is diagonalized by the discrete Fourier transform (DFT) [33]. Let $\widehat{\boldsymbol{x}}_0 = \mathbf{F}\boldsymbol{x}_0$ and note that $\mathbf{F}$ is unitary. Then

$$\widehat{\mathbf{\Sigma}}_0 := \mathrm{Cov}(\widehat{\boldsymbol{x}}_0) = \mathbf{F}\,\mathbf{\Sigma}_0\,\mathbf{F}^{\mathsf{H}} = \mathrm{diag}\big(S(\omega) : \omega \in \Omega\big), \tag{24}$$

where $S(\omega)$ is the power spectral density (PSD). Because $\mathbf{F}$ is unitary, it preserves eigenvalues and Loewner order:

$$\boldsymbol{A} \preceq \boldsymbol{B} \quad\Longleftrightarrow\quad \mathbf{F}\boldsymbol{A}\mathbf{F}^{\mathsf{H}} \preceq \mathbf{F}\boldsymbol{B}\mathbf{F}^{\mathsf{H}}.$$

Thus transforming to the frequency domain does not change the eigenvalue bounds in Theorem 3.1; it simply diagonalizes the matrices involved.

LF/HF splitting under WSS. An ideal LF/HF mask corresponds to diagonal projection matrices $\mathbf{P}_L$ and $\mathbf{P}_H = \mathbf{I} - \mathbf{P}_L$ selecting disjoint frequency sets Ω_L and Ω_H. Since $\widehat{\mathbf{\Sigma}}_0$ is diagonal, there is no cross-covariance between LF and HF components:

$$\widehat{\mathbf{\Sigma}}_{HL} = \mathbf{P}_H\widehat{\mathbf{\Sigma}}_0\mathbf{P}_L = \mathbf{0}. \tag{25}$$

Therefore the HF covariance is simply

$$\widehat{\mathbf{\Sigma}}_H = \mathbf{P}_H\widehat{\mathbf{\Sigma}}_0\mathbf{P}_H = \mathrm{diag}\big(S(\omega) : \omega \in \Omega_H\big). \tag{26}$$

Conditional HF covariance. The conditional covariance of HF given LF satisfies

$$\mathbf{\Sigma}_{H|L} = \mathbf{\Sigma}_H - \mathbf{\Sigma}_{HL}\mathbf{\Sigma}_L^{\dagger}\mathbf{\Sigma}_{LH}.$$

Since $\mathbf{\Sigma}_{HL} = 0$ under WSS with an ideal LF mask, we have

$$\mathbf{\Sigma}_{H|L} = \widehat{\mathbf{\Sigma}}_{H|L} = \widehat{\mathbf{\Sigma}}_H = \mathrm{diag}\big(S(\omega) : \omega \in \Omega_H\big).$$

Thus the eigenvalues of $\mathbf{\Sigma}_{H|L}$ are precisely the PSD values $S(\omega)$ on the HF set Ω_H. In particular,

$$\nu_{\max}(\mathbf{\Sigma}_{H|L}) = \max_{\omega\in\Omega_H} S(\omega). \tag{27}$$

Substitution into the start bound. Theorem 3.1 states that

$$\sigma_{\max}^2 \;\geq\; \frac{1-\tau_{\max}}{\tau_{\max}}\,\nu_{\max}(\mathbf{\Sigma}_{H|L}).$$

Substituting the PSD expression for $\nu_{\max}(\mathbf{\Sigma}_{H|L})$ yields

$$\sigma_{\max}^2 \;\geq\; \frac{1-\tau_{\max}}{\tau_{\max}}\,\max_{\omega\in\Omega_H} S(\omega),$$

which is exactly the claimed PSD form (8).

E Proof of Theorem 3.3 (Stop bound)

The proof follows the same diagonalization argument used in Theorem 3.1. Since the posterior map $\Phi(\mathbf{\Sigma};\sigma)$ is the same functional form appearing in the start-bound proof, it is diagonalized by the eigenvectors of $\mathbf{\Sigma}$. Thus, as in the previous theorem, all Loewner inequalities may be checked mode-wise in the eigenbasis of $\mathbf{\Sigma}_{\mathbf{s}}$.

Eigenbasis and per-mode posterior variances. Let

$$\mathbf{\Sigma_s} = \boldsymbol{q}\,\Lambda\,\boldsymbol{q}^{\mathsf{T}}, \qquad \Lambda = \mathrm{diag}(\nu_1, \ldots, \nu_D), \qquad \nu_i \geq 0. \tag{28}$$

Since $\Phi(\mathbf{\Sigma}; \sigma)$ is diagonalized by the same eigenvectors as $\mathbf{\Sigma}$ (as in the proof of Theorem 3.1), the matrices

$$\mathbf{\Sigma}_0 = \Phi(\mathbf{\Sigma_s}; \sigma_\mathbf{s}) \quad \text{and} \quad \Phi(\mathbf{\Sigma_s}; \sqrt{\sigma_\mathbf{s}^2 + \sigma_t^2})$$

are diagonal in the same basis $\boldsymbol{q}$. Their diagonal entries are obtained by applying the scalar posterior formula

$$\frac{\nu_i\,\sigma^2}{\nu_i + \sigma^2}$$

to each eigenvalue ν_i of $\mathbf{\Sigma_s}$.

Thus, per mode ν_i,

$$\text{eigenvalue of } \mathbf{\Sigma}_0 : \qquad \frac{\nu_i\,\sigma_\mathbf{s}^2}{\nu_i + \sigma_\mathbf{s}^2}, \tag{29}$$

$$\text{eigenvalue of } \Phi(\mathbf{\Sigma_s}; \sqrt{\sigma_\mathbf{s}^2 + \sigma_t^2}) : \qquad \frac{\nu_i\,(\sigma_\mathbf{s}^2 + \sigma_t^2)}{\nu_i + \sigma_\mathbf{s}^2 + \sigma_t^2}. \tag{30}$$

Residual and tolerance condition. The residual covariance is

$$\mathbf{\Delta}_t^{\min} = \Phi\left(\mathbf{\Sigma_s}; \sqrt{\sigma_\mathbf{s}^2 + \sigma_t^2}\right) - \mathbf{\Sigma}_0 \succeq \mathbf{0}.$$

Since both matrices on the right-hand side share the eigenvectors of $\mathbf{\Sigma_s}$, the Loewner inequality

$$\mathbf{\Delta}_t^{\min} \preceq \tau_{\min}\,\mathbf{\Sigma}_0$$

is equivalent to a set of scalar inequalities on their eigenvalues.

Let $\nu > 0$ be any eigenvalue of $\mathbf{\Sigma_s}$. The corresponding eigenvalues of $\mathbf{\Sigma}_0 = \Phi(\mathbf{\Sigma_s}; \sigma_\mathbf{s})$ and $\Phi(\mathbf{\Sigma_s}; \sqrt{\sigma_\mathbf{s}^2 + \sigma_t^2})$ are, respectively,

$$\frac{\nu\,\sigma_\mathbf{s}^2}{\nu + \sigma_\mathbf{s}^2} \qquad \text{and} \qquad \frac{\nu\,(\sigma_\mathbf{s}^2 + \sigma_t^2)}{\nu + \sigma_\mathbf{s}^2 + \sigma_t^2}.$$

Thus the matrix inequality $\mathbf{\Delta}_t^{\min} \preceq \tau_{\min}\,\mathbf{\Sigma}_0$ is equivalent to requiring, for every $\nu > 0$,

$$\frac{\nu\,(\sigma_\mathbf{s}^2 + \sigma_t^2)}{\nu + \sigma_\mathbf{s}^2 + \sigma_t^2} - \frac{\nu\,\sigma_\mathbf{s}^2}{\nu + \sigma_\mathbf{s}^2} \;\leq\; \tau_{\min}\,\frac{\nu\,\sigma_\mathbf{s}^2}{\nu + \sigma_\mathbf{s}^2}. \tag{31}$$

Canceling the common factor $\nu > 0$ gives

$$\frac{\sigma_\mathbf{s}^2 + \sigma_t^2}{\nu + \sigma_\mathbf{s}^2 + \sigma_t^2} - \frac{\sigma_\mathbf{s}^2}{\nu + \sigma_\mathbf{s}^2} \;\leq\; \tau_{\min}\,\frac{\sigma_\mathbf{s}^2}{\nu + \sigma_\mathbf{s}^2}. \tag{32}$$

Multiplying both sides by $(\nu + \sigma_\mathbf{s}^2)(\nu + \sigma_\mathbf{s}^2 + \sigma_t^2)$ leads to the linear inequality in σ_t^2:

$$(\nu - \tau_{\min}\sigma_\mathbf{s}^2)\,\sigma_t^2 \;\leq\; \tau_{\min}\,\sigma_\mathbf{s}^2\,(\nu + \sigma_\mathbf{s}^2). \tag{33}$$

Active and inactive modes. If $\nu \leq \tau_{\min}\sigma_\mathbf{s}^2$, then the left-hand side of (33) is non-positive and the right-hand side is non-negative; the inequality holds automatically for all $\sigma_t^2 \geq 0$. Such modes impose no constraint.

If $\nu > \tau_{\min}\sigma_\mathbf{s}^2$, we divide by the positive factor $\nu - \tau_{\min}\sigma_\mathbf{s}^2$ to obtain

$$\sigma_t^2 \;\leq\; \frac{\tau_{\min}\,\sigma_\mathbf{s}^2\,(\nu + \sigma_\mathbf{s}^2)}{\nu - \tau_{\min}\sigma_\mathbf{s}^2}. \tag{34}$$

Worst-case eigenvalue and final bound. The right-hand side of (34) is a strictly decreasing function of ν on $(\tau_{\min}\sigma_{\mathbf{s}}^2, \infty)$. Thus the most restrictive constraint is obtained by taking the largest eigenvalue $\nu_{\max} = \nu_{\max}(\mathbf{\Sigma_s})$, giving

$$\sigma_t^2 \;\le\; \frac{\tau_{\min}\,\sigma_{\mathbf{s}}^2\,(\nu_{\max}+\sigma_{\mathbf{s}}^2)}{\nu_{\max}-\tau_{\min}\sigma_{\mathbf{s}}^2}. \tag{35}$$

If $\nu_{\max} \le \tau_{\min}\sigma_{\mathbf{s}}^2$, then all modes are inactive and the constraint is vacuous.

Identifying the maximal allowable reverse-diffusion variance at termination with $\sigma_{\min}^2$ produces

$$\sigma_{\min}^2 \;\le\; \frac{\tau_{\min}\,\sigma_{\mathbf{s}}^2\,(\nu_{\max}+\sigma_{\mathbf{s}}^2)}{\nu_{\max}-\tau_{\min}\sigma_{\mathbf{s}}^2},$$

which establishes the claim.

F Proof of Corollary 3.4 (High-SNR assumption)

From Theorem 3.3, the upper bound on $\sigma_{\min}^2$ is

$$\frac{\tau_{\min}\,\sigma_{\mathbf{s}}^2\,(\nu_{\max}+\sigma_{\mathbf{s}}^2)}{\nu_{\max}-\tau_{\min}\sigma_{\mathbf{s}}^2}, \tag{36}$$

where $\nu_{\max} = \nu_{\max}(\mathbf{\Sigma_s})$ is the largest eigenvalue of the clean-signal covariance. Factor out $\nu_{\max}$ from numerator and denominator:

$$\tau_{\min}\,\sigma_{\mathbf{s}}^2\,\frac{\nu_{\max}+\sigma_{\mathbf{s}}^2}{\nu_{\max}-\tau_{\min}\sigma_{\mathbf{s}}^2} = \tau_{\min}\,\sigma_{\mathbf{s}}^2\,\frac{1+\sigma_{\mathbf{s}}^2/\nu_{\max}}{1-\tau_{\min}\sigma_{\mathbf{s}}^2/\nu_{\max}}. \tag{37}$$

In the high-SNR regime $\nu_{\max} \gg \sigma_{\mathbf{s}}^2$, both ratios $\sigma_{\mathbf{s}}^2/\nu_{\max}$ and $\tau_{\min}\sigma_{\mathbf{s}}^2/\nu_{\max}$ are small, and we obtain the approximation

$$\frac{1+\sigma_{\mathbf{s}}^2/\nu_{\max}}{1-\tau_{\min}\sigma_{\mathbf{s}}^2/\nu_{\max}} \;\approx\; 1, \tag{38}$$

so that

$$\frac{\tau_{\min}\,\sigma_{\mathbf{s}}^2\,(\nu_{\max}+\sigma_{\mathbf{s}}^2)}{\nu_{\max}-\tau_{\min}\sigma_{\mathbf{s}}^2} \;\approx\; \tau_{\min}\,\sigma_{\mathbf{s}}^2. \tag{39}$$

Then, from Theorem 3.3, we conclude that in this regime

$$\sigma_{\min}^2 \;\lesssim\; \tau_{\min}\,\sigma_{\mathbf{s}}^2, \tag{40}$$

which is the claim.

G Proof of Theorem 3.5 (Re-scaling invariance)

We consider the unit-gradient update form used in (12). At iteration k, the update applied to (f, g) is

$$\Delta\boldsymbol{\mu}^k = -\hat{\alpha}^k\left(\frac{\nabla f^k}{\|\nabla f^k\|_2} + \hat{\lambda}\,\frac{\nabla g^k}{\|\nabla g^k\|_2}\right), \tag{41}$$

where $\hat{\alpha}^k > 0$ and $\hat{\lambda} > 0$ is a global sampler parameters.

Let $\phi_f, \phi_g : \mathbb{R} \to \mathbb{R}$ be strictly increasing C^1 functions and define

$$\widetilde{f} = \phi_f \circ f, \qquad \widetilde{g} = \phi_g \circ g.$$

By the chain rule,

$$\nabla\widetilde{f}^k = \phi_f'(f^k)\,\nabla f^k, \qquad \nabla\widetilde{g}^k = \phi_g'(g^k)\,\nabla g^k, \tag{42}$$

with $\phi_f'(f^k) > 0$ and $\phi_g'(g^k) > 0$ because ϕ_f, ϕ_g are strictly increasing. Hence

$$\|\nabla\widetilde{f}^k\|_2 = \phi_f'(f^k)\,\|\nabla f^k\|_2, \qquad \|\nabla\widetilde{g}^k\|_2 = \phi_g'(g^k)\,\|\nabla g^k\|_2. \tag{43}$$

Therefore the normalized gradients are unchanged:

$$\frac{\nabla \widetilde{f}^k}{\|\nabla \widetilde{f}^k\|_2} = \frac{\phi_f'(f^k)\nabla f^k}{\phi_f'(f^k)\|\nabla f^k\|_2} = \frac{\nabla f^k}{\|\nabla f^k\|_2}, \tag{44}$$

$$\frac{\nabla \widetilde{g}^k}{\|\nabla \widetilde{g}^k\|_2} = \frac{\phi_g'(g^k)\nabla g^k}{\phi_g'(g^k)\|\nabla g^k\|_2} = \frac{\nabla g^k}{\|\nabla g^k\|_2}. \tag{45}$$

Applying the same unit-gradient update (41) to $(\widetilde{f}, \widetilde{g})$ with the same parameters $(\hat{\alpha}^k, \hat{\lambda})$ yields

$$\Delta\widetilde{\boldsymbol{\mu}}^k = -\hat{\alpha}^k\left(\frac{\nabla \widetilde{f}^k}{\|\nabla \widetilde{f}^k\|_2} + \hat{\lambda}\,\frac{\nabla \widetilde{g}^k}{\|\nabla \widetilde{g}^k\|_2}\right) = -\hat{\alpha}^k\left(\frac{\nabla f^k}{\|\nabla f^k\|_2} + \hat{\lambda}\,\frac{\nabla g^k}{\|\nabla g^k\|_2}\right) = \Delta\boldsymbol{\mu}^k.$$

Thus both the direction and the magnitude of the update are invariant under any strictly increasing C^1 reparameterizations of f and g (including positive scalings).

H Proof of Theorem 3.8 (Common descent condition)

Let

$$d^k = \nabla f^k + \lambda^k \nabla g^k, \qquad \cos\vartheta^k = \frac{\langle \nabla f^k, \nabla g^k\rangle}{\|\nabla f^k\|_2\,\|\nabla g^k\|_2}, \qquad r^k = \frac{\|\nabla f^k\|_2}{\|\nabla g^k\|_2}.$$

Simultaneous descent [19] requires

$$\langle \nabla f^k, d^k\rangle > 0 \quad \text{and} \quad \langle \nabla g^k, d^k\rangle > 0.$$

We compute both inner products in terms of $\cos\vartheta^k$, r^k, and λ^k. First,

$$\begin{aligned}
\langle \nabla f^k, d^k\rangle &= \|\nabla f^k\|_2^2 + \lambda^k\langle \nabla f^k, \nabla g^k\rangle && (46)\\
&= \|\nabla f^k\|_2^2 + \lambda^k\|\nabla f^k\|_2\|\nabla g^k\|_2\cos\vartheta^k && (47)\\
&= \|\nabla f^k\|_2^2\big(1 + \lambda^k\tfrac{\cos\vartheta^k}{r^k}\big). && (48)
\end{aligned}$$

Similarly,

$$\begin{aligned}
\langle \nabla g^k, d^k\rangle &= \langle \nabla g^k, \nabla f^k\rangle + \lambda^k\|\nabla g^k\|_2^2 && (49)\\
&= \|\nabla g^k\|_2^2\big(r^k\cos\vartheta^k + \lambda^k\big). && (50)
\end{aligned}$$

Since both norm-squared factors are strictly positive, the signs of (48)–(50) are determined by

$$1 + \lambda^k\frac{\cos\vartheta^k}{r^k} \quad \text{and} \quad r^k\cos\vartheta^k + \lambda^k.$$

We now distinguish two geometric regimes.

Case 1: $\cos\vartheta^k \geq 0$**.** The gradients form an acute angle. Since $\lambda^k > 0$ by construction, and noting that $\cos\vartheta^k \geq 0$, $r^k > 0$, we obtain

$$1 + \lambda^k\frac{\cos\vartheta^k}{r^k} \geq 1 > 0, \qquad r^k\cos\vartheta^k + \lambda^k \geq 0,$$

with strict positivity whenever at least one gradient is nonzero (as assumed). Thus every $\lambda^k > 0$ yields simultaneous descent.

Case 2: $\cos\vartheta^k < 0$**.** The gradients form an obtuse angle; the two inequalities now impose nontrivial constraints.

From (48),

$$1 + \lambda^k\frac{\cos\vartheta^k}{r^k} > 0 \quad\Longleftrightarrow\quad \lambda^k < -\frac{r^k}{\cos\vartheta^k}.$$

Here $\cos\vartheta^k < 0$ and $r^k > 0$, so multiplying by $r^k/\cos\vartheta^k$ reverses the inequality.

From (50),

$$r^k \cos\vartheta^k + \lambda^k > 0 \quad \Longleftrightarrow \quad \lambda^k > -\, r^k \cos\vartheta^k.$$

Combining both constraints gives the feasible interval

$$\lambda^k \in \left(r^k(-\cos\vartheta^k),\ r^k(-1/\cos\vartheta^k)\right),$$

which is exactly the claimed condition for $\cos\vartheta^k < 0$. **Note.** In the degenerate edge case where $cos^k = -1$ (perfect anti-alignment), the admissible $\hat{\lambda}$ value would be 1, and there is no update to μ^k. Practically, we never observe $cos^k = -1$ due to numerical computation.

I Greedy-optimal sampler hyperparameters

We detail the update rules and the optimal-hyperparameters scheme for the suite of optimizers tested in this work. The proposed scheme resembles approaches in the compressed sensing and diffusion-based reconstruction literature, where unbiased MSE estimators have been employed for hyperparameter tuning [34–36].

$$\boldsymbol{\mu}^{k+1} \leftarrow \boldsymbol{\mu}^k - \boldsymbol{U}^k\,\hat{\boldsymbol{w}}^k, \qquad \hat{\boldsymbol{w}}^k = \underset{\boldsymbol{w}\in\mathbb{R}^2_+}{\arg\min} \left\|\boldsymbol{x}^* - \left(\boldsymbol{\mu}^k - \boldsymbol{U}^k\boldsymbol{w}\right)\right\|_2^2. \tag{51}$$

Vanilla The first-order vanilla update is

$$\boldsymbol{v}^{k+1} \leftarrow \alpha^k\left(\nabla f^k + \lambda^k \nabla g^k\right), \tag{52}$$

$$\boldsymbol{\mu}^{k+1} \leftarrow \boldsymbol{\mu}^k - \boldsymbol{v}^{k+1}. \tag{53}$$

These updates yield

$$\boldsymbol{U}^k = \begin{bmatrix}\nabla f^k & \nabla g^k\end{bmatrix}, \qquad \boldsymbol{w}^k := \begin{bmatrix}\alpha^k \\ \alpha^k\lambda^k\end{bmatrix}. \tag{54}$$

Momentum The optimal hyperparameters scheme becomes:

$$\hat{\boldsymbol{w}}^k = \underset{\boldsymbol{w}\in\mathbb{R}^3_+}{\arg\min} \left\|\boldsymbol{x}^* - \left(\boldsymbol{\mu}^k - \boldsymbol{U}^k\boldsymbol{w}\right)\right\|_2^2. \tag{55}$$

The momentum update becomes

$$\boldsymbol{v}^{k+1} \leftarrow \alpha^k\left(\nabla f^k + \lambda^k \nabla g^k\right) + \beta^k \boldsymbol{v}^k, \tag{56}$$

$$\boldsymbol{\mu}^{k+1} \leftarrow \boldsymbol{\mu}^k - \boldsymbol{v}^{k+1}. \tag{57}$$

These updates yield

$$\boldsymbol{U}^k = \begin{bmatrix}\nabla f^k & \nabla g^k & \boldsymbol{v}^k\end{bmatrix}, \qquad \boldsymbol{w}^k := \begin{bmatrix}\alpha^k \\ \alpha^k\lambda^k \\ \beta^k\end{bmatrix}. \tag{58}$$

Momentum + Preconditioner Adding a polynomial preconditioner for linear measurement operator $\mathbf{A}$,

$$\boldsymbol{v}^{k+1} \leftarrow \alpha^k\left(\mathrm{pol}(\mathbf{A}^\mathsf{T}\mathbf{A})\nabla f^k + \lambda^k \nabla g^k\right) + \beta^k \boldsymbol{v}^k, \tag{59}$$

$$\boldsymbol{\mu}^{k+1} \leftarrow \boldsymbol{\mu}^k - \boldsymbol{v}^{k+1}. \tag{60}$$

These updates yield

$$\boldsymbol{U}^k = \begin{bmatrix}\mathrm{pol}(\mathbf{A}^\mathsf{T}\mathbf{A})\nabla f^k & \nabla g^k & \boldsymbol{v}^k\end{bmatrix}, \qquad \boldsymbol{w}^k := \begin{bmatrix}\alpha^k \\ \alpha^k\lambda^k \\ \beta^k\end{bmatrix}. \tag{61}$$

Sensitivity to tuning-set size. We evaluate the sensitivity of $\hat{\lambda}$ to tuning-set size and image selection using the ImageNet Gaussian deblurring task. For tuning-set sizes of 2, 3, 5, and 10 images, we randomly sample 20 sets of images for each size and independently tune $\hat{\lambda}$ for each set. Table 4 reports the mean and standard deviation of the resulting values across the 20 sets. The mean remains stable across tuning-set sizes, ranging from 0.939 to 0.947, while the standard deviation decreases from 0.030 to 0.013 as the tuning-set size increases. These results suggest that $\hat{\lambda}$ can be reliably selected using a small tuning set, with limited sensitivity to image selection.

Table 4: Sensitivity analysis of tuning-set size on ImageNet Gaussian deblurring.

Tuning-set size	$\hat{\lambda}$ **values**
2	0.947 ± 0.030
3	0.944 ± 0.023
5	0.945 ± 0.015
10	0.939 ± 0.013

J Datasets and Implementation Details

We implemented the OptDiff pipeline with momentum and preconditioner addons in our GitHub repository. The repository is publicly available. Single-image reconstruction runtimes were obtained on a single NVIDIA A40 GPU with 48 GB memory.

J.1 Preconditioners

We evaluated several polynomial preconditioners from the literature [37–39], with polynomial degrees ranging from 1 to 10. The best performance was achieved by the degree-2 preconditioner proposed by Iyer et al. [39].

J.2 MRI Reconstruction

Datasets: We use the fastMRI T2-weighted dataset [23] with the following splits:

- Tuning set, 10 slices from the validation dataset used for hyperparameter tuning and invariance experiments (Subsection 5.1)
- Ablation set, 275 slices from the validation dataset used for the ablation experiments (Section 5.2)
- Test set, 330 slices from the test dataset used for comparisons against baseline methods (Subsection 5.3)

Score model: Score network from Daras et al. [40]

Implementation: In MRI, the measurement operator $\mathcal{A}$ is a complex valued mapping that incorporates coil sensitivities, Fourier Transform and sub-sampling operations. We implemented the MRI forward model and subsampling operations in our OptDiff repository. T2-weighted brain MR images from the fastMRI dataset were center cropped to 384×384 matrix size similar to Jalal et al. [3]. We applied coil compression to retain 8 virtual channels per slice. Coil sensitivity maps were estimated using ESPIRiT [41] algorithm.

Experiments were conducted with several configurations combining different acceleration factors $R = 4$ and $R = 8$, with calibration region sizes of 32 and 16 respectively, under both uniform and random undersampling masks. To estimate the effective noise floor, we used image-domain references reconstructed from fully sampled multicoil k-space data. After applying the same intensity normalization as in our experiments, we computed the root-mean-square intensity of background pixels outside the anatomy in the tuning set, following standard MRI noise-estimation procedures [42]. The procedure yielded a noise floor of $\sigma_{\mathbf{s}} = 0.021$.

Baseline methods: We adapted the DPS, Pi-GDM, RED-diff and DAPS solvers in our codebase for baseline comparisons. We implemented DiffPIR [26] based on the original repository. For DDS, we used the original codebase [27]. We tuned the hyperparameters of all baseline methods for each setting. DDRM [24] and DDNM [28] require access to an explicit SVD of the measurement operator, which is computationally intractable for the multi-coil MRI forward model; therefore, they are not included as baselines for MRI reconstruction.

J.3 Natural Image Tasks

Datasets: We use a subset of the ImageNet validation dataset (5 images for hyperparameter tuning, 330 images for testing) obtained from Kaggle. We use a subset of the FFHQ validation dataset (5 images for hyperparameter tuning, 330 images for testing).

Score models: We adopt the unconditional pretrained diffusion models from Mardani et al. [5] and Choi et al. [43] for ImageNet and FFHQ, respectively.

J.3.1 Degredations

Deblurring: As pre-processing, the images were cropped to $3 \times 256 \times 256$. The deblurring forward model was adopted from Mardani et al. [5]. We used kernel sizes $6, 12$ both with $\sigma = 3$ and additional measurement noise with variance 0.01 across both datasets.

Super-resolution: As pre-processing, the images were cropped to $3 \times 256 \times 256$. For the experiments, each image is downsampled by a factor of $4, 16$ in both dimensions and corrupted with Gaussian noise. We used noise with variance 0.01 in FFHQ and 0.02 in ImageNet. The super-resolution forward model was adopted from Mardani et al. [5].

Inpainting: As pre-processing, the images were cropped to $3 \times 256 \times 256$. Each image was masked by a 64×64 square mask with randomly selected center coordinates (which was fixed across images). We added Gaussian noise with variance 0.01 in FFHQ and 0.02 in ImageNet. The masking forward model was adopted from Mardani et al. [5].

J.3.2 Baseline methods:

DPS, DDRM, Pi-GDM, RED-diff baselines are based on the repository from Mardani et al. [5]. DDNM and DiffPIR have been adapted into the Mardani et al. [5] repository. For DAPS, SITCOM and DMPlug, we used the original implementations from Zhang et al. [7], Alkhouri et al. [44], Wang et al. [45], respectively. We used the hyperparameters provided by each repository as reference and further tuned them for each setting.

K Extended Limitations

First, our noise-schedule design leverages forward models where the measurements preserve sufficient low-frequency information, enabling a reduced $\sigma_{\max}$. This reduction is available for the main tasks considered herein, including MRI reconstruction, deblurring, and super-resolution, but may be limited in problems such as inpainting, where the effective $\sigma_{\max}$ can match the default unconditional-generation value. Thus, the proposed design remains broadly applicable, but may not always reduce the starting noise level. Future work could extend the design beyond the endpoints ($\sigma_{\max}$ and $\sigma_{\min}$) toward principled rules for the curvature of the noise schedule, which determines how sampling steps are allocated across noise levels and, consequently, frequency regions.

Second, our sampler hyperparameter reparameterization builds upon RED-diff, an optimization-based posterior solver. Consequently, OptDiff inherits some limitations of this class of methods, which produce a single reconstruction and may favor pixelwise fidelity over perceptual quality, particularly in highly ill-posed settings with multiple plausible solutions [11, 29]. This limitation is most evident in $16\times$ super-resolution and inpainting, where sampling-based methods such as DAPS [7] achieve better perceptual results, as shown in Table 6. Moreover, our NNLS-based hyperparameter tuning may reinforce this tendency, since its reference values are obtained by minimizing MSE. Future

work could investigate more general tuning objectives that incorporate perceptual criteria, such as deep-feature losses, to better balance pixelwise distortion and perceptual similarity.

Lastly, the proposed reparameterizations, including the noise-schedule hyperparameters, are readily applicable to other posterior samplers (Subsection 5.3, Table 1). We note that the noise schedule design could be carried over to flow-based solvers through a change of variables, left as future work. Integrating our sampler hyperparameter design is less direct for methods that are not formulated as optimization problems. Extending this component of the framework to other posterior samplers would require additional theoretical justification, which we leave for future work.

L Comparison with baselines: full tables

Table 5: Comparison with baselines using NFE-20. Our method consistently achieves the best or comparable performance across task–dataset pairs. Best results are shown in bold, and second-best results are underlined.

		fastMRI					
		Acceleration R=4			Acceleration R=8		
Task	**Method**	**PSNR ↑**	**SSIM ↑**		**PSNR ↑**	**SSIM ↑**	
MRI recon	DPS [4]	31.93 ± 1.50	0.763 ± 0.059		27.36 ± 1.48	0.682 ± 0.054	
	DiffPIR [26]	27.04 ± 1.01	0.727 ± 0.054		24.96 ± 1.15	0.587 ± 0.075	
	Pi-GDM [25]	29.63 ± 1.27	0.739 ± 0.063		28.04 ± 1.24	0.707 ± 0.062	
	RED-diff [5]	33.24 ± 1.49	0.870 ± 0.033		29.43 ± 1.58	0.825 ± 0.042	
	DDS [27]	30.27 ± 1.27	$\underline{0.901 \pm 0.032}$		25.42 ± 1.38	0.765 ± 0.031	
	DAPS [7]	$\underline{35.64 \pm 1.47}$	0.894 ± 0.031		$\underline{30.82 \pm 1.45}$	$\underline{0.848 \pm 0.036}$	
	OptDiff (Ours)	$\mathbf{37.19 \pm 2.01}$	$\mathbf{0.943 \pm 0.030}$		$\mathbf{33.23 \pm 2.08}$	$\mathbf{0.903 \pm 0.046}$	

		ImageNet			FFHQ		
Task	**Method**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
Deblurring kernel 6	DDNM [28]	$\mathbf{31.91 \pm 3.52}$	$\mathbf{0.894 \pm 0.048}$	$\mathbf{0.051 \pm 0.021}$	$\mathbf{35.87 \pm 1.68}$	$\mathbf{0.939 \pm 0.015}$	$\underline{0.043 \pm 0.012}$
	DiffPIR [26]	30.77 ± 3.40	0.853 ± 0.051	0.088 ± 0.045	$\underline{35.11 \pm 1.72}$	0.918 ± 0.015	0.053 ± 0.016
	RED-diff [5]	25.79 ± 2.68	0.677 ± 0.074	0.247 ± 0.079	29.07 ± 1.15	0.763 ± 0.026	0.147 ± 0.033
	DAPS [7]	29.67 ± 1.97	0.777 ± 0.038	0.148 ± 0.046	32.55 ± 1.90	0.875 ± 0.021	0.110 ± 0.022
	OptDiff (Ours)	$\underline{31.66 \pm 3.81}$	$\underline{0.890 \pm 0.047}$	$\underline{0.082 \pm 0.038}$	$\mathbf{35.87 \pm 1.72}$	$\underline{0.937 \pm 0.014}$	$\mathbf{0.039 \pm 0.011}$
Severe deblur kernel 12	DDNM [28]	$\mathbf{27.89 \pm 3.87}$	$\mathbf{0.782 \pm 0.110}$	$\mathbf{0.136 \pm 0.056}$	$\mathbf{32.33 \pm 2.06}$	$\mathbf{0.890 \pm 0.030}$	$\mathbf{0.080 \pm 0.022}$
	DiffPIR [26]	$\underline{27.27 \pm 3.73}$	$\underline{0.739 \pm 0.122}$	$\underline{0.229 \pm 0.103}$	$\underline{31.87 \pm 2.08}$	$\underline{0.871 \pm 0.033}$	$\underline{0.110 \pm 0.032}$
	RED-diff [5]	23.42 ± 2.68	0.589 ± 0.128	0.438 ± 0.125	26.79 ± 1.66	0.755 ± 0.053	0.320 ± 0.053
	DAPS [7]	25.06 ± 3.00	0.609 ± 0.119	0.326 ± 0.072	28.09 ± 1.90	0.734 ± 0.040	0.217 ± 0.032
	OptDiff (Ours)	26.27 ± 3.24	0.716 ± 0.110	0.243 ± 0.085	30.77 ± 1.75	0.848 ± 0.031	0.111 ± 0.029
Super-res 4×	DDNM [28]	25.23 ± 3.64	0.681 ± 0.154	0.286 ± 0.114	30.00 ± 2.27	0.851 ± 0.044	$\mathbf{0.110 \pm 0.028}$
	DiffPIR [26]	25.38 ± 3.65	0.685 ± 0.144	$\underline{0.281 \pm 0.127}$	$\underline{31.05 \pm 2.38}$	$\underline{0.872 \pm 0.039}$	$\underline{0.112 \pm 0.033}$
	RED-diff [5]	24.74 ± 3.31	0.672 ± 0.147	0.307 ± 0.108	29.54 ± 2.08	0.831 ± 0.041	0.144 ± 0.031
	DAPS [7]	$\mathbf{26.14 \pm 3.24}$	$\underline{0.697 \pm 0.116}$	$\mathbf{0.229 \pm 0.055}$	29.86 ± 2.17	0.831 ± 0.038	0.132 ± 0.023
	OptDiff (Ours)	$\underline{25.82 \pm 3.80}$	$\mathbf{0.705 \pm 0.145}$	0.340 ± 0.146	$\mathbf{31.23 \pm 2.57}$	$\mathbf{0.879 \pm 0.039}$	0.148 ± 0.035
Super-res 16×	DDNM [28]	19.38 ± 2.52	0.422 ± 0.172	$\underline{0.560 \pm 0.159}$	22.25 ± 2.06	$\underline{0.637 \pm 0.084}$	$\mathbf{0.258 \pm 0.069}$
	DiffPIR [26]	19.81 ± 2.58	$\underline{0.444 \pm 0.176}$	0.648 ± 0.181	$\underline{22.30 \pm 2.01}$	$\mathbf{0.640 \pm 0.082}$	$\underline{0.269 \pm 0.073}$
	RED-diff [5]	19.83 ± 2.53	0.439 ± 0.169	0.680 ± 0.145	$\mathbf{22.57 \pm 2.08}$	0.634 ± 0.079	0.348 ± 0.079
	DAPS [7]	$\mathbf{20.23 \pm 2.43}$	0.440 ± 0.150	$\mathbf{0.521 \pm 0.078}$	22.24 ± 1.98	0.598 ± 0.072	0.342 ± 0.047
	OptDiff (Ours)	$\underline{20.19 \pm 2.70}$	$\mathbf{0.453 \pm 0.177}$	0.715 ± 0.173	$\mathbf{22.57 \pm 2.16}$	0.634 ± 0.083	0.421 ± 0.082
Inpainting 64×64	DDNM [28]	26.28 ± 3.03	0.910 ± 0.020	$\underline{0.055 \pm 0.016}$	$\underline{33.26 \pm 2.63}$	0.961 ± 0.009	0.021 ± 0.006
	DiffPIR [26]	$\mathbf{28.35 \pm 3.63}$	$\mathbf{0.935 \pm 0.012}$	$\mathbf{0.049 \pm 0.018}$	$\mathbf{34.65 \pm 2.85}$	$\mathbf{0.972 \pm 0.007}$	$\mathbf{0.010 \pm 0.004}$
	RED-diff [5]	24.45 ± 2.52	0.919 ± 0.011	0.080 ± 0.015	26.32 ± 2.14	0.947 ± 0.007	0.066 ± 0.010
	DAPS [7]	$\underline{27.17 \pm 3.25}$	0.922 ± 0.013	0.066 ± 0.031	33.21 ± 2.70	0.964 ± 0.007	0.023 ± 0.009
	OptDiff (Ours)	26.61 ± 3.02	$\underline{0.926 \pm 0.013}$	0.064 ± 0.015	33.12 ± 3.39	$\underline{0.968 \pm 0.008}$	$\underline{0.017 \pm 0.007}$

Table 6: Comparison with baselines using NFE-100. Our method consistently achieves the best or comparable performance across all task–dataset pairs. Best results are shown in bold, and second-best results are underlined.

		fastMRI					
		Acceleration R=4			Acceleration R=8		
Task	**Method**	**PSNR ↑**	**SSIM ↑**		**PSNR ↑**	**SSIM ↑**	
MRI recon	DPS [4]	35.82 ± 1.46	0.775 ± 0.069		28.94 ± 1.54	0.774 ± 0.066	
	DiffPIR [26]	30.33 ± 0.97	0.691 ± 0.077		27.17 ± 1.08	0.650 ± 0.072	
	Pi-GDM [25]	30.69 ± 1.37	0.753 ± 0.086		28.88 ± 1.27	0.734 ± 0.078	
	RED-diff [5]	36.79 ± 1.43	0.872 ± 0.035		31.92 ± 1.44	0.840 ± 0.034	
	DDS [27]	33.72 ± 1.14	0.930 ± 0.030		28.92 ± 1.28	0.828 ± 0.027	
	DAPS [7]	36.50 ± 1.63	0.915 ± 0.033		31.66 ± 1.49	0.870 ± 0.042	
	OptDiff (Ours)	$\mathbf{38.45 \pm 1.58}$	$\mathbf{0.948 \pm 0.023}$		$\mathbf{34.93 \pm 1.38}$	$\mathbf{0.921 \pm 0.029}$	

		ImageNet			FFHQ		
Task	**Method**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
Deblurring kernel 6	DPS [4]	24.25 ± 3.97	0.602 ± 0.192	0.420 ± 0.187	33.00 ± 2.50	0.898 ± 0.034	0.078 ± 0.028
	SITCOM [44]	28.60 ± 3.27	0.792 ± 0.060	0.099 ± 0.049	32.41 ± 2.17	0.878 ± 0.031	0.081 ± 0.024
	DMPlug [45]	25.83 ± 3.32	0.684 ± 0.141	0.268 ± 0.104	30.19 ± 1.94	0.840 ± 0.046	0.177 ± 0.044
	DDRM [24]	32.71 ± 3.82	0.908 ± 0.045	0.039 ± 0.017	36.96 ± 1.76	$\mathbf{0.949 \pm 0.013}$	0.030 ± 0.009
	DDNM [28]	32.60 ± 3.89	0.906 ± 0.045	$\mathbf{0.030 \pm 0.013}$	36.95 ± 1.78	$\mathbf{0.949 \pm 0.013}$	$\mathbf{0.027 \pm 0.008}$
	DiffPIR [26]	32.22 ± 3.68	0.892 ± 0.049	0.061 ± 0.032	36.58 ± 1.76	0.942 ± 0.014	0.040 ± 0.012
	RED-diff [5]	29.35 ± 3.92	0.821 ± 0.092	0.175 ± 0.089	35.07 ± 1.77	0.915 ± 0.015	0.063 ± 0.020
	DAPS [7]	31.39 ± 3.13	0.853 ± 0.047	0.107 ± 0.033	34.13 ± 1.39	0.888 ± 0.013	0.088 ± 0.021
	OptDiff (Ours)	$\mathbf{33.28 \pm 3.74}$	$\mathbf{0.914 \pm 0.038}$	0.049 ± 0.024	$\mathbf{37.10 \pm 1.79}$	$\mathbf{0.949 \pm 0.013}$	0.058 ± 0.017
Severe deblur kernel 12	DPS [4]	21.30 ± 3.27	0.484 ± 0.200	0.371 ± 0.159	25.88 ± 6.47	0.749 ± 0.129	0.159 ± 0.115
	SITCOM [44]	24.25 ± 3.27	0.596 ± 0.142	0.285 ± 0.118	28.65 ± 2.29	0.797 ± 0.054	0.157 ± 0.046
	DMPlug [45]	22.72 ± 3.01	0.547 ± 0.176	0.390 ± 0.141	27.02 ± 2.21	0.763 ± 0.070	0.235 ± 0.058
	DDRM [24]	27.87 ± 3.89	0.785 ± 0.110	0.156 ± 0.074	32.18 ± 2.18	0.889 ± 0.031	0.069 ± 0.021
	DDNM [28]	28.07 ± 4.12	$\mathbf{0.789 \pm 0.110}$	$\mathbf{0.134 \pm 0.059}$	32.54 ± 2.15	$\mathbf{0.893 \pm 0.030}$	0.074 ± 0.022
	DiffPIR [26]	27.41 ± 3.72	0.746 ± 0.114	0.145 ± 0.064	32.05 ± 1.95	0.870 ± 0.030	$\mathbf{0.060 \pm 0.019}$
	RED-diff [5]	24.19 ± 3.49	0.604 ± 0.171	0.428 ± 0.156	29.42 ± 2.23	0.800 ± 0.048	0.183 ± 0.054
	DAPS [7]	26.28 ± 2.80	0.671 ± 0.086	0.247 ± 0.075	28.38 ± 2.19	0.786 ± 0.055	0.199 ± 0.040
	OptDiff (Ours)	$\mathbf{28.27 \pm 3.90}$	0.788 ± 0.100	0.190 ± 0.083	$\mathbf{32.61 \pm 2.22}$	0.891 ± 0.031	0.127 ± 0.034
Super-res 4×	DPS [4]	23.72 ± 4.90	0.618 ± 0.209	0.299 ± 0.207	30.50 ± 2.62	0.866 ± 0.044	$\mathbf{0.071 \pm 0.025}$
	SITCOM [44]	25.75 ± 3.91	0.696 ± 0.146	0.286 ± 0.141	31.04 ± 2.49	0.863 ± 0.040	0.091 ± 0.029
	DMPlug [45]	25.03 ± 3.78	0.665 ± 0.160	$\mathbf{0.224 \pm 0.084}$	30.36 ± 2.23	0.854 ± 0.042	0.151 ± 0.036
	DDRM [24]	25.84 ± 3.86	0.702 ± 0.152	0.331 ± 0.145	31.68 ± 2.50	0.885 ± 0.038	0.125 ± 0.038
	DDNM [28]	25.79 ± 3.93	0.703 ± 0.152	0.286 ± 0.136	30.94 ± 2.45	0.872 ± 0.041	0.085 ± 0.028
	DiffPIR [26]	25.28 ± 3.78	0.682 ± 0.152	0.242 ± 0.117	31.06 ± 2.44	0.873 ± 0.040	0.092 ± 0.031
	RED-diff [5]	25.75 ± 3.82	0.700 ± 0.141	0.311 ± 0.144	31.25 ± 2.37	0.875 ± 0.037	0.108 ± 0.034
	DAPS [7]	$\mathbf{26.50 \pm 3.43}$	0.706 ± 0.132	0.239 ± 0.067	29.63 ± 1.73	0.763 ± 0.028	0.179 ± 0.028
	OptDiff (Ours)	26.07 ± 3.96	$\mathbf{0.715 \pm 0.148}$	0.350 ± 0.156	$\mathbf{31.99 \pm 2.57}$	$\mathbf{0.891 \pm 0.037}$	0.147 ± 0.043
Super-res 16×	DPS [4]	17.95 ± 3.02	0.365 ± 0.192	$\mathbf{0.388 \pm 0.125}$	22.62 ± 2.28	0.664 ± 0.086	0.286 ± 0.078
	SITCOM [44]	20.25 ± 2.81	0.463 ± 0.185	0.616 ± 0.191	23.49 ± 2.36	$\mathbf{0.689 \pm 0.085}$	0.263 ± 0.073
	DMPlug [45]	19.57 ± 2.63	0.431 ± 0.177	0.517 ± 0.142	22.75 ± 2.47	0.646 ± 0.085	0.267 ± 0.057
	DDRM [24]	20.35 ± 2.77	0.462 ± 0.185	0.683 ± 0.199	23.50 ± 2.24	0.687 ± 0.083	0.316 ± 0.090
	DDNM [28]	20.15 ± 2.68	0.460 ± 0.183	0.633 ± 0.207	22.86 ± 2.12	0.663 ± 0.083	0.249 ± 0.079
	DiffPIR [26]	19.66 ± 2.61	0.439 ± 0.178	0.577 ± 0.194	22.77 ± 2.14	0.658 ± 0.085	$\mathbf{0.246 \pm 0.077}$
	RED-diff [5]	20.34 ± 2.76	0.460 ± 0.182	0.704 ± 0.193	23.01 ± 2.07	0.663 ± 0.082	0.282 ± 0.082
	DAPS [7]	$\mathbf{20.53 \pm 2.49}$	0.418 ± 0.136	0.522 ± 0.081	22.63 ± 1.96	0.572 ± 0.064	0.358 ± 0.046
	OptDiff (Ours)	20.44 ± 2.75	$\mathbf{0.468 \pm 0.184}$	0.704 ± 0.186	$\mathbf{23.67 \pm 2.25}$	$\mathbf{0.689 \pm 0.083}$	0.343 ± 0.094
Inpainting 64×64	DPS [4]	23.41 ± 3.49	0.649 ± 0.186	0.349 ± 0.176	32.09 ± 3.84	0.931 ± 0.030	0.074 ± 0.037
	SITCOM [44]	28.87 ± 3.65	0.929 ± 0.015	0.038 ± 0.015	$\mathbf{36.11 \pm 2.33}$	0.973 ± 0.006	$\mathbf{0.009 \pm 0.004}$
	DMPlug [45]	21.41 ± 3.19	0.765 ± 0.091	0.246 ± 0.098	29.64 ± 3.10	0.881 ± 0.029	0.143 ± 0.039
	DDRM [24]	29.02 ± 3.68	0.935 ± 0.016	0.053 ± 0.018	35.42 ± 2.70	$\mathbf{0.978 \pm 0.006}$	$\mathbf{0.009 \pm 0.004}$
	DDNM [28]	28.07 ± 3.61	0.928 ± 0.013	$\mathbf{0.036 \pm 0.014}$	34.53 ± 2.57	0.970 ± 0.007	0.010 ± 0.004
	DiffPIR [26]	$\mathbf{29.18 \pm 3.89}$	$\mathbf{0.944 \pm 0.012}$	0.047 ± 0.019	35.40 ± 2.62	0.975 ± 0.006	0.010 ± 0.004
	RED-diff [5]	27.70 ± 3.02	0.932 ± 0.010	0.058 ± 0.016	29.49 ± 1.51	0.931 ± 0.009	0.035 ± 0.008
	DAPS [7]	28.80 ± 3.64	0.931 ± 0.011	0.059 ± 0.022	34.53 ± 2.79	0.967 ± 0.007	0.021 ± 0.009
	OptDiff (Ours)	28.36 ± 3.38	0.930 ± 0.014	0.054 ± 0.018	33.40 ± 3.77	0.966 ± 0.019	0.023 ± 0.023

M Runtime comparisons

Table 7: Runtime per reconstruction in seconds at NFE-20 and NFE-100. Mean and standard deviation are calculated on a single NVIDIA A40 GPU with batch size 1, over 20 reconstructions.

		fastMRI			
		Acceleration R=4		Acceleration R=8	
Task	**Method**	**NFE-20**	**NFE-100**	**NFE-20**	**NFE-100**
MRI recon	DPS [4]	3.30 ± 0.00	16.57 ± 0.01	3.30 ± 0.00	16.55 ± 0.01
	DiffPIR [26]	1.63 ± 0.00	8.14 ± 0.01	1.63 ± 0.00	8.15 ± 0.01
	Pi-GDM [25]	3.81 ± 0.00	19.05 ± 0.01	3.81 ± 0.00	19.06 ± 0.01
	RED-diff [5]	1.63 ± 0.00	8.12 ± 0.01	1.63 ± 0.00	8.12 ± 0.01
	DAPS [7]	1.79 ± 0.01	8.38 ± 0.01	1.79 ± 0.00	8.39 ± 0.02
	OptDiff (Ours)	1.67 ± 0.01	8.36 ± 0.02	1.67 ± 0.01	8.36 ± 0.01

		ImageNet		FFHQ	
Task	**Method**	**NFE-20**	**NFE-100**	**NFE-20**	**NFE-100**
Deblurring kernel 6	DPS [4]	5.33 ± 0.11	26.29 ± 0.03	1.68 ± 0.00	8.39 ± 0.04
	DDRM [24]	1.48 ± 0.00	7.56 ± 0.25	0.63 ± 0.00	3.15 ± 0.01
	DDNM [28]	1.48 ± 0.00	7.43 ± 0.01	0.64 ± 0.00	3.14 ± 0.06
	DiffPIR [26]	1.46 ± 0.00	7.34 ± 0.01	0.62 ± 0.00	3.07 ± 0.03
	RED-diff [5]	1.46 ± 0.00	7.34 ± 0.01	0.65 ± 0.00	3.12 ± 0.02
	DAPS [7]	5.36 ± 0.05	32.59 ± 0.73	1.27 ± 0.09	28.25 ± 0.38
	OptDiff (Ours)	1.48 ± 0.00	7.33 ± 0.01	0.64 ± 0.03	3.15 ± 0.04
Severe deblur kernel 12	DPS [4]	5.26 ± 0.01	26.39 ± 0.07	1.68 ± 0.02	8.40 ± 0.02
	DDRM [24]	1.48 ± 0.00	7.42 ± 0.01	0.63 ± 0.00	3.22 ± 0.01
	DDNM [28]	1.48 ± 0.00	7.44 ± 0.01	0.65 ± 0.04	3.14 ± 0.06
	DiffPIR [26]	1.46 ± 0.00	7.36 ± 0.02	0.63 ± 0.04	3.14 ± 0.01
	RED-diff [5]	1.47 ± 0.02	7.34 ± 0.01	0.65 ± 0.00	3.13 ± 0.03
	DAPS [7]	5.44 ± 0.08	24.52 ± 0.75	4.53 ± 0.03	28.08 ± 0.51
	OptDiff (Ours)	1.48 ± 0.01	7.43 ± 0.04	0.64 ± 0.03	3.15 ± 0.01
Super-res $4\times$	DPS [4]	5.21 ± 0.01	26.29 ± 0.02	1.66 ± 0.00	8.35 ± 0.03
	DDRM [24]	1.48 ± 0.00	7.41 ± 0.01	0.63 ± 0.00	3.20 ± 0.02
	DDNM [28]	1.49 ± 0.03	7.42 ± 0.01	0.65 ± 0.00	3.13 ± 0.02
	DiffPIR [26]	1.46 ± 0.00	7.34 ± 0.01	0.61 ± 0.00	3.12 ± 0.02
	RED-diff [5]	1.46 ± 0.00	7.32 ± 0.01	0.64 ± 0.02	3.09 ± 0.00
	DAPS [7]	1.77 ± 0.02	22.82 ± 0.60	1.13 ± 0.03	18.28 ± 0.71
	OptDiff (Ours)	1.47 ± 0.00	7.31 ± 0.02	0.64 ± 0.05	3.06 ± 0.07
Super-res $16\times$	DPS [4]	5.21 ± 0.01	26.21 ± 0.02	1.68 ± 0.00	8.33 ± 0.02
	DDRM [24]	1.47 ± 0.00	7.42 ± 0.02	0.63 ± 0.00	3.11 ± 0.03
	DDNM [28]	1.48 ± 0.02	7.42 ± 0.01	0.65 ± 0.00	3.17 ± 0.10
	DiffPIR [26]	1.47 ± 0.01	7.35 ± 0.01	0.62 ± 0.01	3.04 ± 0.05
	RED-diff [5]	1.46 ± 0.01	7.32 ± 0.01	0.63 ± 0.00	3.11 ± 0.03
	DAPS [7]	1.80 ± 0.08	22.37 ± 1.44	0.97 ± 0.05	18.37 ± 0.23
	OptDiff (Ours)	1.47 ± 0.02	7.33 ± 0.01	0.62 ± 0.03	3.08 ± 0.01
Inpainting 64×64	DPS [4]	5.27 ± 0.03	26.32 ± 0.06	1.67 ± 0.00	8.38 ± 0.02
	DDRM [24]	1.47 ± 0.00	7.40 ± 0.01	0.62 ± 0.01	3.09 ± 0.02
	DDNM [28]	1.47 ± 0.00	7.37 ± 0.04	0.63 ± 0.00	3.05 ± 0.02
	DiffPIR [26]	1.46 ± 0.01	7.35 ± 0.01	0.61 ± 0.00	3.05 ± 0.04
	RED-diff [5]	1.46 ± 0.01	7.33 ± 0.03	0.66 ± 0.05	3.17 ± 0.02
	DAPS [7]	1.68 ± 0.06	13.80 ± 1.30	0.68 ± 0.00	9.81 ± 0.73
	OptDiff (Ours)	1.46 ± 0.00	7.33 ± 0.01	0.62 ± 0.02	3.06 ± 0.01

N Additional Qualitative Results

We provide visual comparisons on additional test set slices for the MRI reconstruction, super-resolution, deblurring, inpainting tasks.

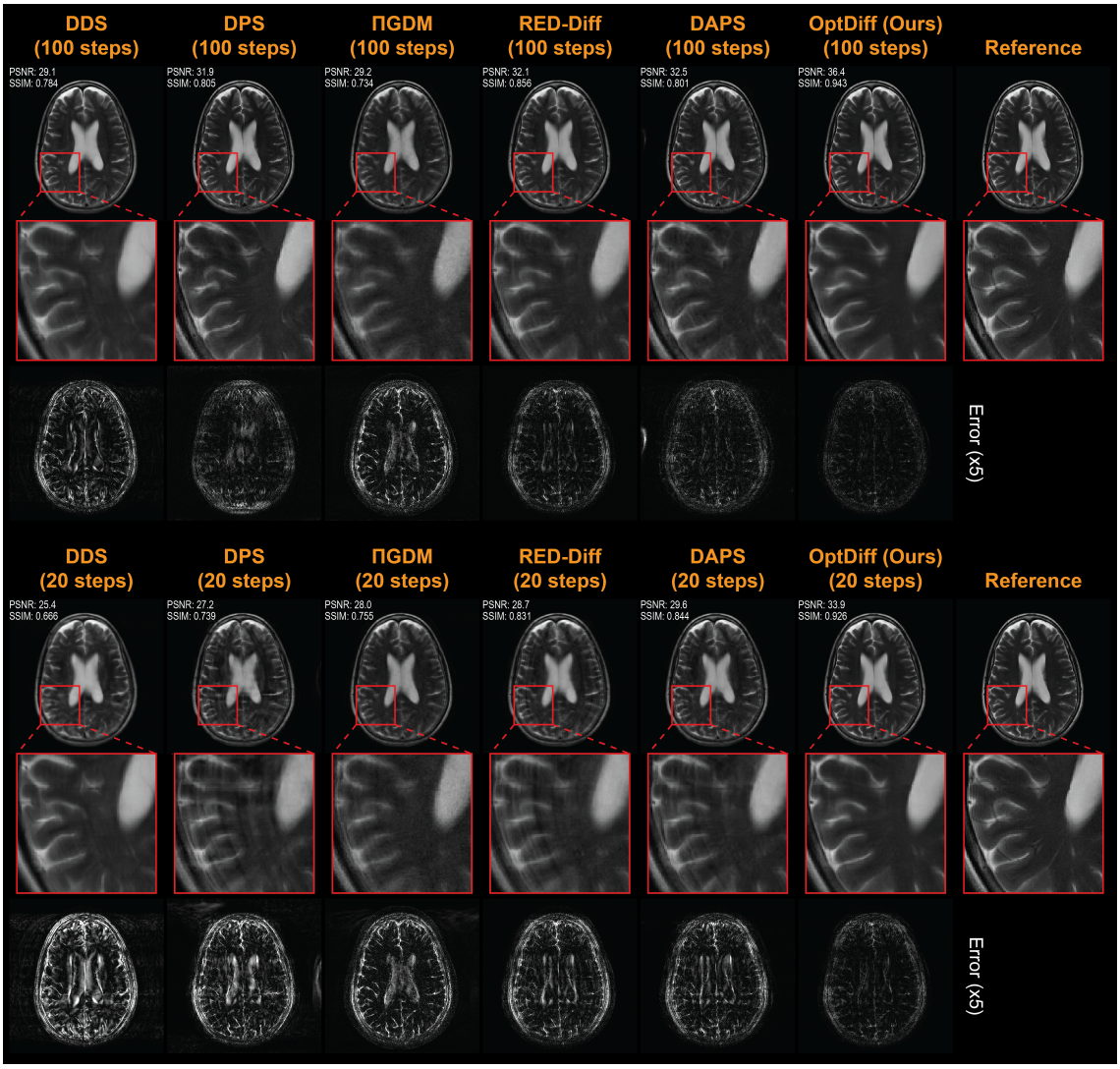

Figure 5: Reconstruction performance comparison on a representative test slice for the MRI Reconstruction task with $R = 8$ random subsampling on the fastMRI dataset.

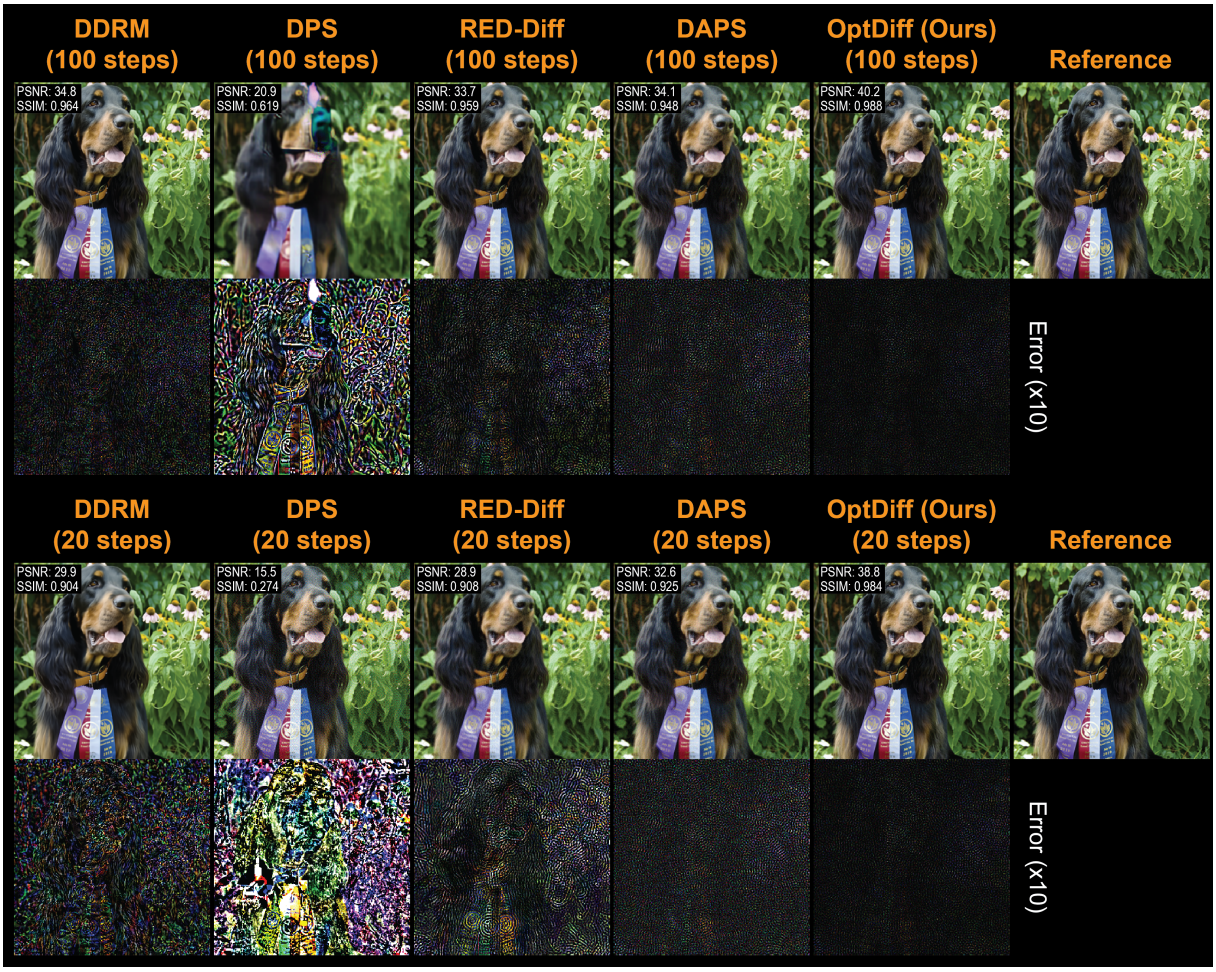

Figure 6: Reconstruction performance comparison on a representative test slice for deblurring task on the ImageNet dataset.

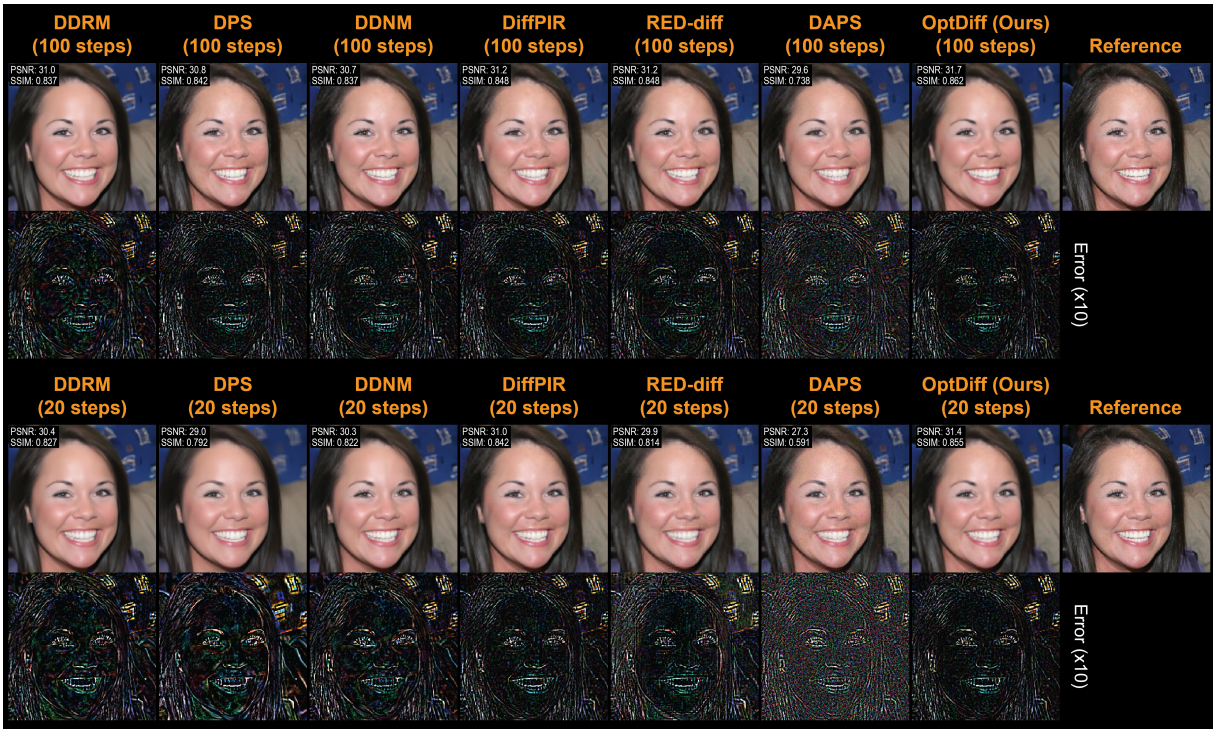

Figure 7: Reconstruction performance comparison on a representative test slice for $4\times$ super-resolution task on the FFHQ dataset.

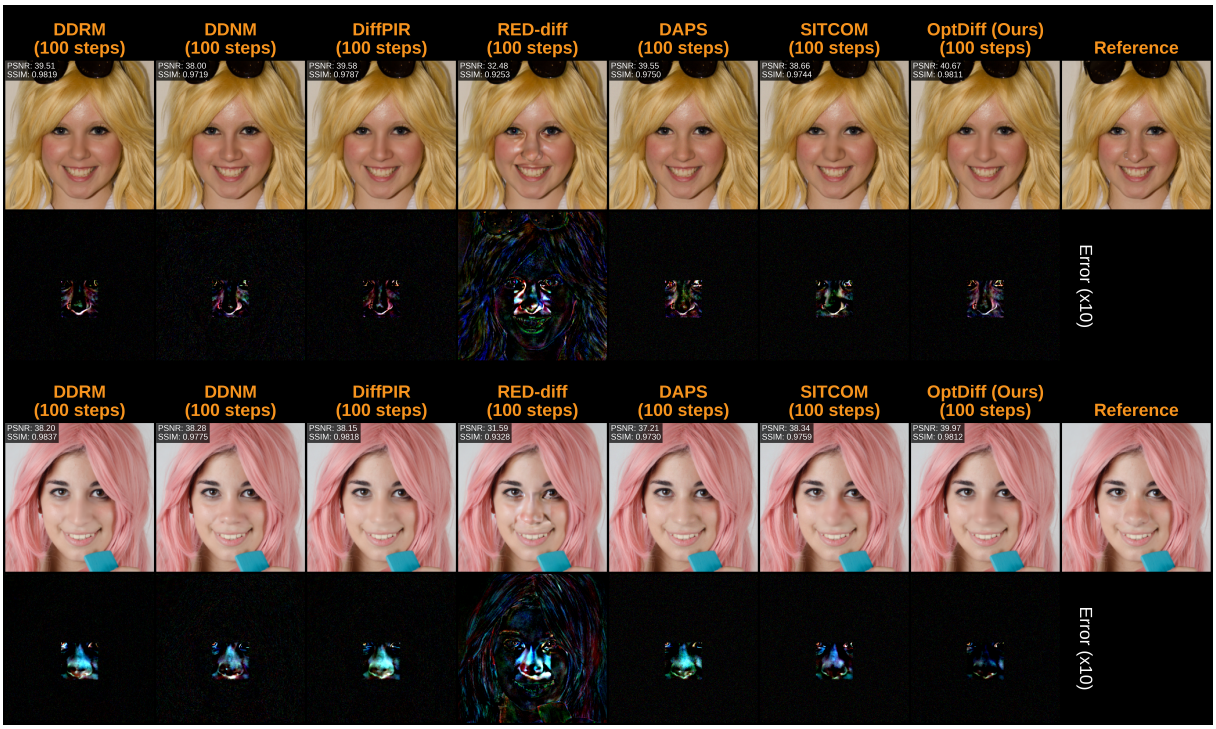

Figure 8: Reconstruction performance comparison on representative test slices for inpainting task on the FFHQ dataset.

O Broader Impacts

The methods developed in this work have the potential to significantly improve MRI image reconstruction quality and efficiency, which could translate into meaningful clinical benefits. Improved reconstructions may lead to more accurate diagnoses and better patient outcomes. Additionally, by enabling faster acquisitions, these methods could enhance scanner throughput and increase access to MRI, particularly in resource-limited settings.

However, there are also potential risks and limitations to consider. This study relies on the publicly available fastMRI dataset, and while it has become a benchmark in the community, it may not fully capture the diversity of patient populations, scanner hardware, or clinical conditions. This raises concerns about potential bias and generalization issues when deploying the method in real-world settings. Moreover, our evaluation has been limited to standard image quality metrics such as PSNR and SSIM, which, while useful, may not fully reflect clinical relevance. A more thorough assessment involving expert radiologists and task-specific evaluations is essential before considering clinical deployment.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We list our contributions under introduction, clearly outlining the claims we make in the paper. These are all summarized in the abstract. We also provide theorems to motivate the rules we use for noise schedule and hyperparameter selections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have addressed the limitations under Section 6.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Please refer to Appendices B–H for all mathematical justifications for our proofs. We state the assumptions and reference the proofs in the main paper as well.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We explain all the dataset and experimental details in Appendix J. The code is publicly available at `https://github.com/MRSRL/OptDiff`.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We use existing publicly available datasets, and our code is publicly available at `https://github.com/MRSRL/OptDiff`.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We detail data splits, how we tuned the hyperparameters for the baselines under Appendix J. For our method, the optimization and hyperparameter selection pipeline is studied in depth under Section 4, OptDiff Pipeline.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: We have extensive results tables that report average metrics with standard deviations for validation and test datasets.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We state the GPUs used under Appendix J and report runtimes on Fig 1 as well as tables under Appendix M.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We conform with the code of ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss broader impacts under Appendix O.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The existing assets in this paper have been properly referenced or cited.

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets`

has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our code is publicly available at `https://github.com/MRSRL/OptDiff`; the method and implementation details are described in Section 4 and Appendix J.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: We work with open source datasets (fastMRI, ImageNet, FFHQ).

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: We work with open source datasets (fastMRI, ImageNet, FFHQ).

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were only used for writing and editing assistance.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.