Improving Diffusion Inverse Problem Solving with Decoupled Noise Annealing

Bingliang Zhang* | Wenda Chu* | Julius Berner | Chenlin Meng | Anima Anandkumar | Yang Song
arXiv 2024

Weekly Meeting - 2024-07-19
KAIST Geometric AI Lab - Jaihoon Kim

Inverse Problem

Solving an inverse problem involves finding the underlying signal $\mathbf{x}_0$ from its partial, noisy measurement $\mathbf{y}$, where A is the forward model.

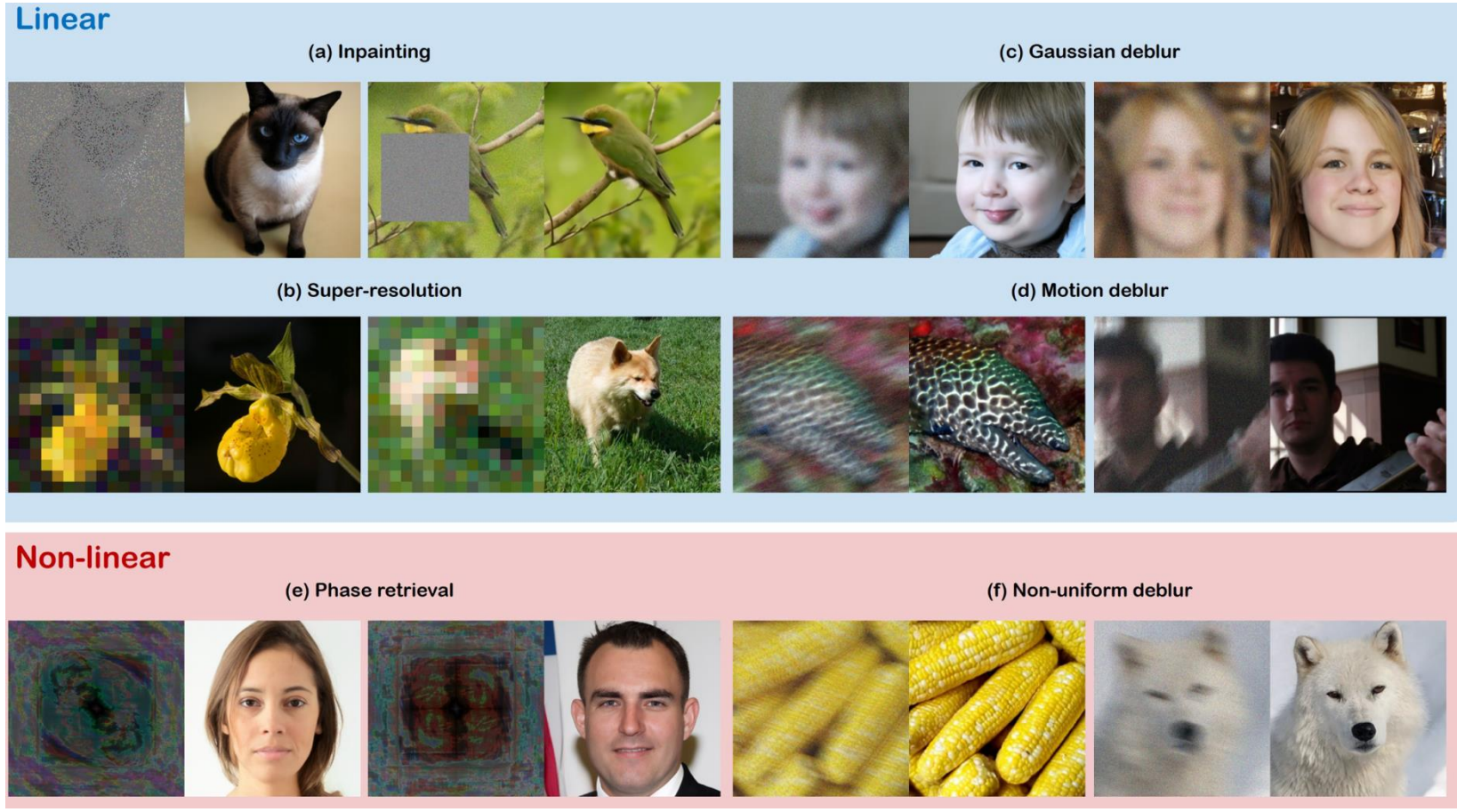
Measurement y

Signal $\mathbf{x}_0$

$$\boldsymbol{y = A(x_0) + n}$$

Inverse Problems are Ubiquitous

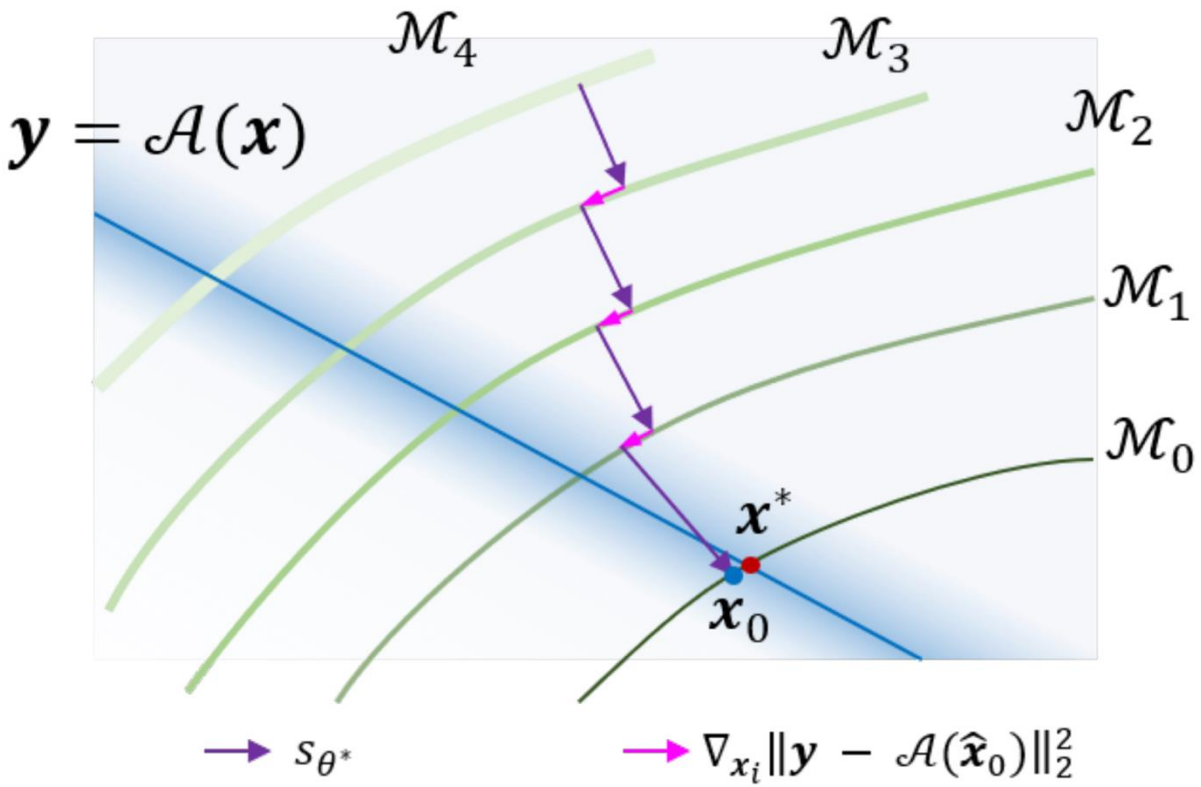
Inverse problems can be found in many areas including image restoration and medical imaging.



DPS
[Chung *et al.*, 2022]

Bayesian Inverse Problems with Diffusion Priors

Previous works utilized diffusion models to sample $\mathbf{x}_t$ conditioned on the measurement $\boldsymbol{y}$.



DPS
[Chung *et al.*, 2022]

Bayesian Inverse Problems with Diffusion Priors

Previous works utilized diffusion models to sample $\mathbf{x}_t$ conditioned on the measurement $\boldsymbol{y}$.

Algorithm 1 DPS - Gaussian

Require: $N, \boldsymbol{y}, \{\zeta_i\}_{i=1}^N, \{\tilde{\sigma}_i\}_{i=1}^N$

$\boldsymbol{x}_N \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

for $i = N-1$ **to** 0 **do**

  $\hat{\boldsymbol{s}} \leftarrow \boldsymbol{s}_\theta(\boldsymbol{x}_i, i)$

  $\hat{\boldsymbol{x}}_0 \leftarrow \frac{1}{\sqrt{\bar{\alpha}_i}}(\boldsymbol{x}_i + (1-\bar{\alpha}_i)\hat{\boldsymbol{s}})$

  $\boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

  $\boldsymbol{x}'_{i-1} \leftarrow \frac{\sqrt{\alpha_i}(1-\bar{\alpha}_{i-1})}{1-\bar{\alpha}_i}\boldsymbol{x}_i + \frac{\sqrt{\bar{\alpha}_{i-1}}\beta_i}{1-\bar{\alpha}_i}\hat{\boldsymbol{x}}_0 + \tilde{\sigma}_i \boldsymbol{z}$

  $\boldsymbol{x}_{i-1} \leftarrow \boldsymbol{x}'_{i-1} - \zeta_i \nabla_{\boldsymbol{x}_i} \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|_2^2$

end for

return $\hat{\mathbf{x}}_0$

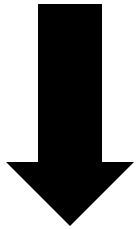
DPS
[Chung *et al.*, 2022]

Diffusion Model in Bayesian Inverse Problem

However, previous methods struggle to fix the global errors as $\mathbf{x}_t$ and $\mathbf{x}_{t+\Delta}$ are closely related.

$$\mathbf{x}_t \sim p(\mathbf{x}_t | \mathbf{x}_{t+\Delta}, \boldsymbol{y})$$

Decouple $\mathbf{x}_t$ and $\mathbf{x}_{t+\Delta}$



$$\mathbf{x}_t \sim p(\mathbf{x}_t | \boldsymbol{y})$$

Decoupled Annealing Posterior Sampling (DAPS)

Background

Forward and reverse process of a diffusion model is formulated as follow:

$$\mathrm{d}\mathbf{x}_t = \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}\mathbf{w}_t \qquad \mathrm{d}\mathbf{x}_t = -2\dot{\sigma}_t\sigma_t\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t;\sigma_t)\mathrm{d}t + \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}\mathbf{w}_t$$

Utilizing diffusion model in Bayesian inverse problem:

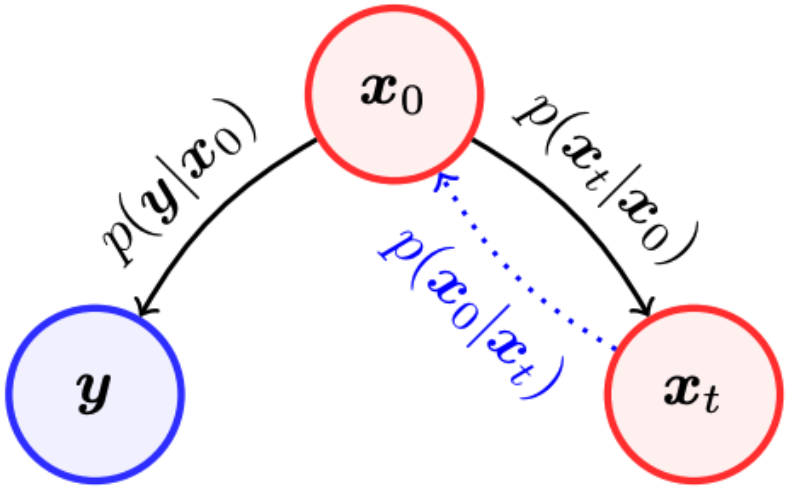
Sample $x_0 \sim p(\mathbf{x}_0|\boldsymbol{y}) \propto p(\mathbf{x}_0)p(\boldsymbol{y}|\mathbf{x}_0) \approx p(\mathbf{x}_0)p(\boldsymbol{y}|\phi(\mathbf{x}_t, \boldsymbol{\epsilon}_\theta(\mathbf{x}_t,)))$.

$$\mathrm{d}\mathbf{x}_t = -2\dot{\sigma}_t\sigma_t\left(\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t;\sigma_t) + \boxed{\nabla_{\mathbf{x}_t}\log p(\mathbf{y}\mid\mathbf{x}_t)}\right)\mathrm{d}t + \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}\mathbf{w}_t$$

Main source of error !

Related Work – DPS

$$p(\boldsymbol{y}|\boldsymbol{x}_t) = \mathbb{E}_{\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0|\boldsymbol{x}_t)}\left[p(\boldsymbol{y}|\boldsymbol{x}_0)\right]$$



Approximate $p(\mathbf{x}_0|\mathbf{x}_t)$ using Tweedie's formula

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\boldsymbol{x}_t) \simeq \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{y}|\hat{\boldsymbol{x}}_0)$$

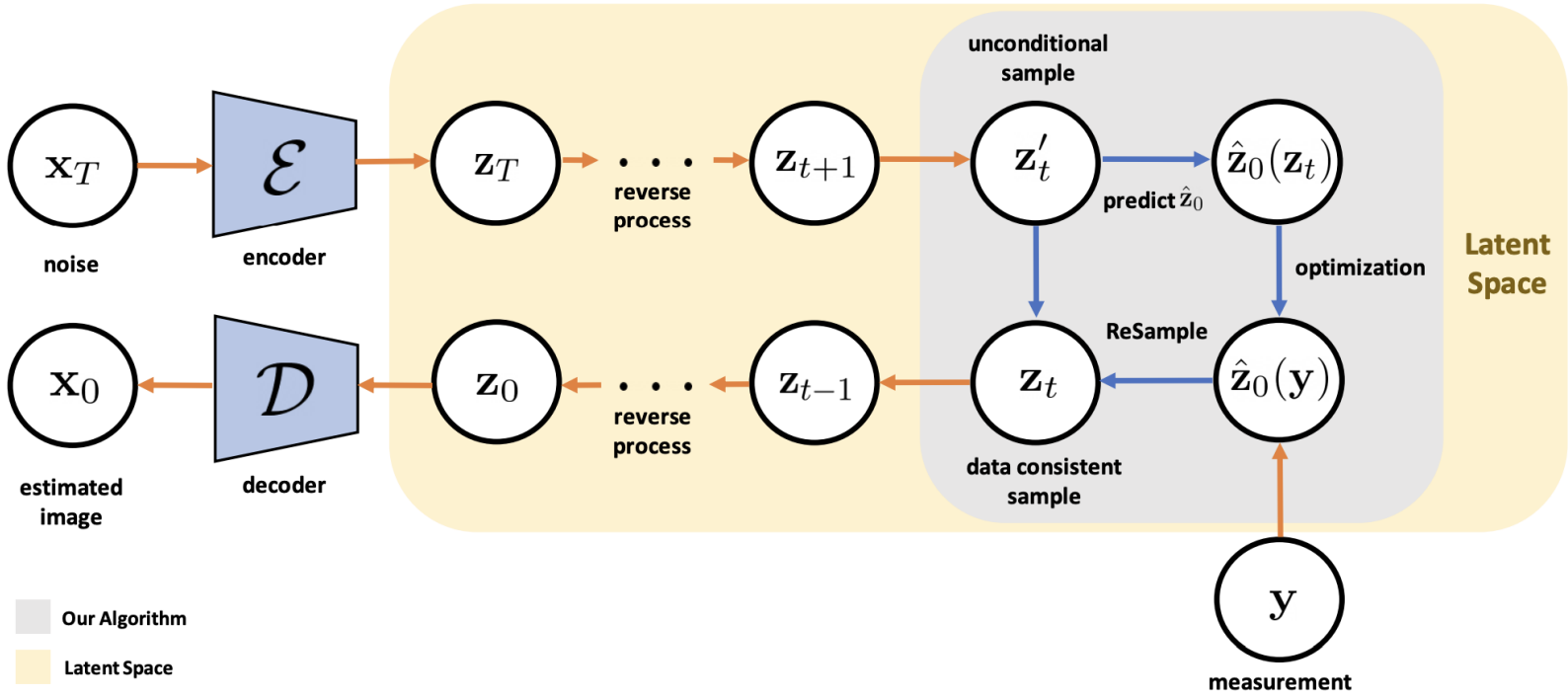
The error (Jensen's gap) is **not negligible** when σ is relatively small.

$$\mathcal{J} \leq \frac{d}{\sqrt{2\pi\sigma^2}} e^{-1/2\sigma^2} \|\nabla_{\boldsymbol{x}} \mathcal{A}(\boldsymbol{x})\| m_1$$

DPS
[Chung *et al.*, 2022]

Related Work – Interleaved Optimization

Interleave optimization during the reverse process and start the optimization process within local proximity of the global solution.



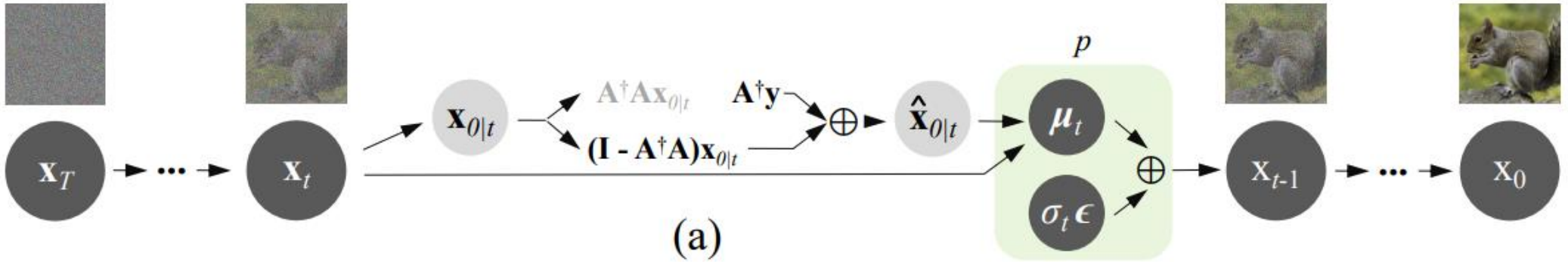
ReSample
[Song *et al.*, 2023]

Related Work – Spectral Denoising

Decompose the forward model $\mathbf{A}$ into its range-space and null-space ($\mathbf{A}^\dagger$: pseudo-inverse).

$$\mathbf{x} \equiv \mathbf{A}^\dagger \mathbf{A}\mathbf{x} + (\mathbf{I} - \mathbf{A}^\dagger \mathbf{A})\mathbf{x}$$

Replace the range-space to $\mathbf{A}^\dagger \mathbf{y}$ and leverage diffusion model to generate the null-space.



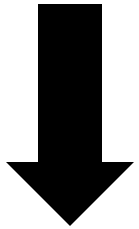
DDNM
[Wang *et al.*, 2022]

Related Work

Previous works sample $\mathbf{x}_t$ from $\mathbf{x}_{t+\Delta t}$ which at most correct local errors.

→ Struggles to correct global errors that require significant modifications to the previous sample.

$$\mathbf{x}_t \sim p(\mathbf{x}_t | \mathbf{x}_{t+\Delta}, \boldsymbol{y})$$



$$\mathbf{x}_t \sim p(\mathbf{x}_t | \boldsymbol{y})$$

How to sample from $p(\mathbf{x}_t \mid \mathbf{y})$?

Factorize $p(\mathbf{x}_t \mid \mathbf{y})$ into conditional distributions (decoupling) and sample from them in turn.

Method

$$\mathrm{d}\mathbf{x}_t = -2\dot{\sigma}_t\sigma_t\Big(\nabla_{\mathbf{x}_t}\log p(\mathbf{x}_t;\sigma_t) + \nabla_{\mathbf{x}_t}\log p(\mathbf{y}\mid\mathbf{x}_t)\Big)\mathrm{d}t + \sqrt{2\dot{\sigma}_t\sigma_t}\mathrm{d}\mathbf{w}_t$$

Ensure $\mathbf{x}_t$ and $\mathbf{x}_{t+\Delta t}$ are conditionally independent given $\mathbf{x}_0$.

Two-step procedure—Decoupled Noise Annealing:

(1) Langevin Dynamics Sampling: $\mathbf{x}_{0|\mathbf{y}} \sim p(\mathbf{x}_0|\mathbf{x}_t, \mathbf{y})$

(2) Forward: $\mathbf{x}_t \sim \mathcal{N}(\mathbf{x}_{0|\mathbf{y}}, \sigma_t^2 \boldsymbol{I})$

Method - Justification

Suppose $\mathbf{x}_{t_1}$ is sampled from $p(\mathbf{x}_{t_1}|\mathbf{y})$, then $\mathbf{x}_{t_2} \sim \mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0|\mathbf{x}_{t_1},\mathbf{y})}\left[\mathcal{N}(\mathbf{x}_0, \sigma_{t_2}^2 \boldsymbol{I})\right]$ satisfies $p(\mathbf{x}_{t_2}|\mathbf{y})$.

$$
\begin{aligned}
p(\mathbf{x}_{t_2} \mid \mathbf{y}) &= \iint p(\mathbf{x}_{t_2}, \mathbf{x}_0, \mathbf{x}_{t_1} \mid \mathbf{y}) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_{t_1} \\
&= \iint p(\mathbf{x}_{t_1} \mid \mathbf{y}) p(\mathbf{x}_0 \mid \mathbf{x}_{t_1}, \mathbf{y}) p(\mathbf{x}_{t_2} \mid \mathbf{x}_0, \mathbf{x}_{t_1}, \mathbf{y}) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_{t_1}. \\
&= \mathbb{E}_{\mathbf{x}_{t_1} \sim p(\mathbf{x}_{t_1}|\mathbf{y})} \mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0|\mathbf{x}_{t_1},\mathbf{y})} p(\mathbf{x}_{t_2} \mid \mathbf{x}_0) \\
&= \mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0|\mathbf{x}_{t_1},\mathbf{y})} \mathcal{N}(\mathbf{x}_{t_2}; \mathbf{x}_0, \sigma_{t_2}^2 \boldsymbol{I}),
\end{aligned}
$$

$\mathbf{x}_{t_2}$ is independent of $\mathbf{x}_{t_1}$ and $\boldsymbol{y}$.

For a sufficiently large σ_T, one can assume $p(\mathbf{x}_T|\mathbf{y}) \approx p(\mathbf{x}_T; \sigma_T) \approx \mathcal{N}(\mathbf{0}, \sigma_T^2 \boldsymbol{I})$.

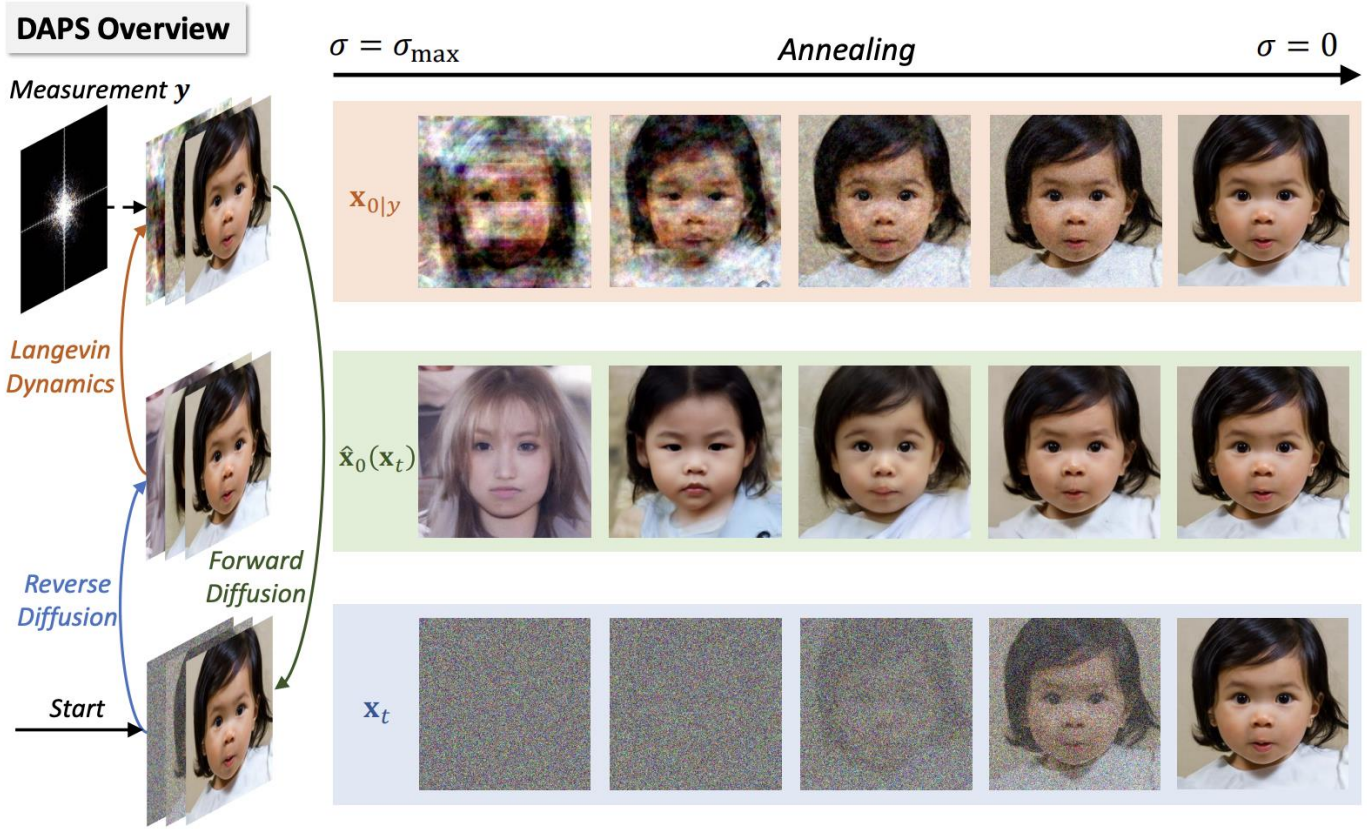
Using this, we can sample from $p(\mathbf{x}_{t_2}|\mathbf{y})$ for any noise level $\sigma_{t_2}^2$ given any sample $\mathbf{x}_{t_1}$.

Method

How to sample $\mathbf{x}_{t_2} \sim \mathbb{E}_{\mathbf{x}_0 \sim p(\mathbf{x}_0|\mathbf{x}_{t_1}, \mathbf{y})}\left[\mathcal{N}(\mathbf{x}_0, \sigma_{t_2}^2 \boldsymbol{I})\right]$?

(1) Langevin Dynamics Sampling: $\mathbf{x}_{0|\mathbf{y}} \sim p(\mathbf{x}_0|\mathbf{x}_t, \mathbf{y})$

(2) Forward: $\mathbf{x}_t \sim \mathcal{N}(\mathbf{x}_{0|\mathbf{y}}, \sigma_t^2 \boldsymbol{I})$



Method – Langevin Dynamics

From Bayes' rule we have

$$p(\mathbf{x}_0 \mid \mathbf{x}_t, \mathbf{y}) = \frac{p(\mathbf{x}_0 \mid \mathbf{x}_t)p(\mathbf{y} \mid \mathbf{x}_0, \mathbf{x}_t)}{p(\mathbf{y} \mid \mathbf{x}_t)} \propto p(\mathbf{x}_0 \mid \mathbf{x}_t)p(\mathbf{y} \mid \mathbf{x}_0)$$

We run Langevin dynamics to sample from the distribution.

$$\mathbf{x}_0^{(j+1)} = \mathbf{x}_0^{(j)} + \eta \cdot \left(\nabla_{\mathbf{x}_0^{(j)}} \log p(\mathbf{x}_0^{(j)} \mid \mathbf{x}_t) + \nabla_{\mathbf{x}_0^{(j)}} \log p(\mathbf{y} \mid \mathbf{x}_0^{(j)}) \right) + \sqrt{2\eta}\boldsymbol{\epsilon}_j$$

Method – Langevin Dynamics and Forward Sampling

Assume $p(\mathbf{x}_0|\mathbf{x_t}) \approx \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0(\mathbf{x}_t), \gamma_t^2 \boldsymbol{I})$, where $\hat{\mathbf{x}}_0(\mathbf{x}_t)$ is an ODE solver.

$$\mathbf{x}_0^{(j+1)} = \mathbf{x}_0^{(j)} - \eta \cdot \nabla_{\mathbf{x}_0^{(j)}} \left(\frac{\|\mathbf{x}_0^{(j)} - \hat{\mathbf{x}}_0(\mathbf{x}_t)\|^2}{2r_t^2} + \frac{\|\mathcal{A}(\mathbf{x}_0^{(j)}) - \mathbf{y}\|^2}{2\beta_{\mathbf{y}}^2} \right) + \sqrt{2\eta}\boldsymbol{\epsilon}_j$$

Lastly, sample $\mathbf{x}_{t-1} \sim p(\mathbf{x}_{t-1}|\mathbf{x}_0,)$ using the predefined forward schedule.

$$\mathbf{x}_{t-1} = \sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\epsilon, \text{ where } \epsilon \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Algorithm

Algorithm 1 Decoupled Annealing Posterior Sampling

Require: Score model $\boldsymbol{s_\theta}$, measurement $\mathbf{y}$, noise schedule σ_t, $(t_i)_{i\in\{0,\dots,N_A\}}$.

Sample $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \sigma_T^2 \boldsymbol{I})$.

for $i = N_A, N_A - 1, \dots, 1$ **do**

Compute $\hat{\mathbf{x}}_0^{(0)} = \hat{\mathbf{x}}_0(\mathbf{x}_{t_i})$ by solving the probability flow ODE in Eq. (39) with $\boldsymbol{s_\theta}$

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

$$\hat{\mathbf{x}}_0^{(j+1)} \leftarrow \hat{\mathbf{x}}_0^{(j)} + \eta_t \left(\nabla_{\hat{\mathbf{x}}_0} \log p(\hat{\mathbf{x}}_0^{(j)} | \mathbf{x}_{t_i}) + \nabla_{\hat{\mathbf{x}}_0} \log p(\mathbf{y} | \hat{\mathbf{x}}_0^{(j)}) \right) + \sqrt{2\eta_t} \boldsymbol{\epsilon}_j, \ \boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}).$$

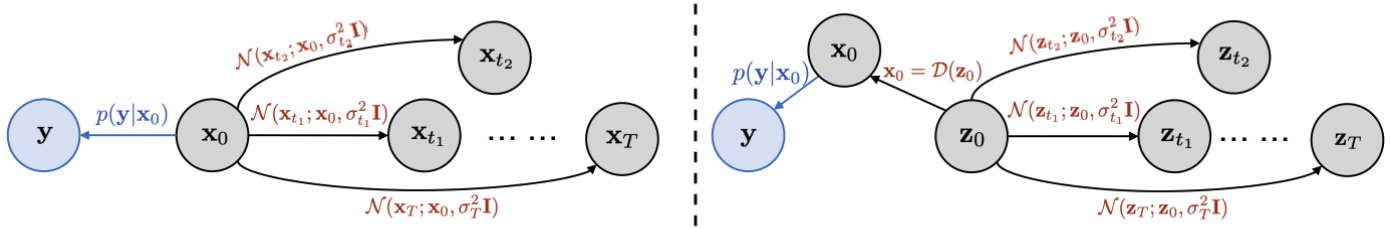
end for

Sample $\mathbf{x}_{t_{i-1}} \sim \mathcal{N}(\hat{\mathbf{x}}_0^{(N)}, \sigma_{t_{i-1}}^2 \boldsymbol{I})$.

end for

Return $\mathbf{x}_0$

Latent Diffusion Model



(a) Diffusion models in pixel spaces.

(b) Latent diffusion models.

$$\mathbf{x}_0^{(j+1)} = \mathbf{x}_0^{(j)} - \eta \cdot \nabla_{\mathbf{x}_0^{(j)}} \left(\frac{\|\mathbf{x}_0^{(j)} - \mathcal{D}(\hat{\mathbf{z}}_0(\mathbf{z}_{t_1}))\|^2}{2r_{t_1}^2} + \frac{\|\mathcal{A}(\mathbf{x}) - \mathbf{y}\|^2}{2\beta_{\mathbf{y}}^2} \right) + \sqrt{2\eta}\boldsymbol{\epsilon}_j$$

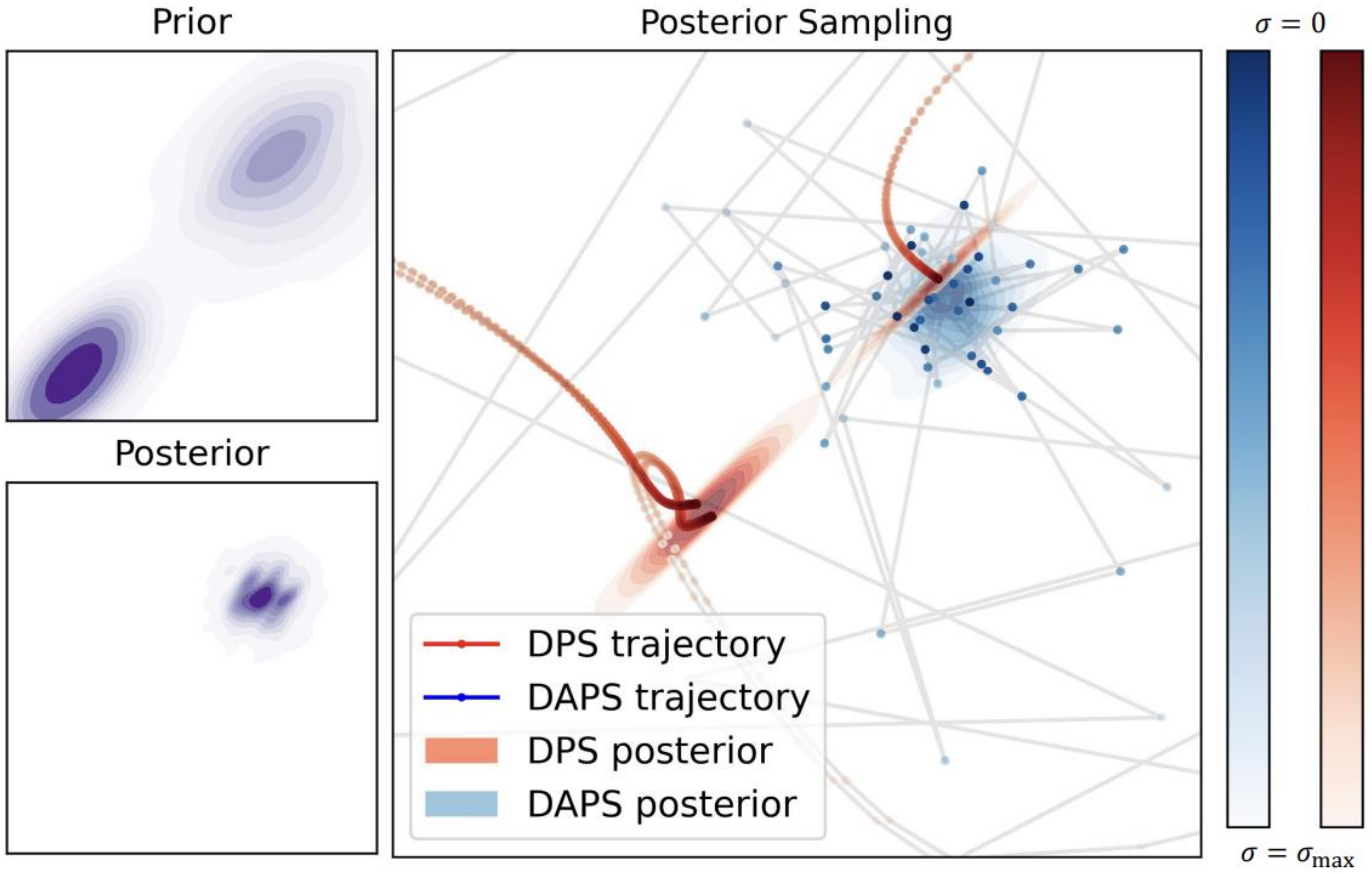
or

$$\mathbf{z}_0^{(j+1)} = \mathbf{z}_0^{(j)} - \eta \cdot \nabla_{\mathbf{z}_0^{(j)}} \left(\frac{\|\mathbf{z}_0^{(j)} - \hat{\mathbf{z}}_0(\mathbf{z}_{t_1})\|^2}{2r_{t_1}^2} + \frac{\|\mathcal{A}(\mathcal{D}(\mathbf{z}_0^{(j)})) - \mathbf{y}\|^2}{2\beta_{\mathbf{y}}^2} \right) + \sqrt{2\eta}\boldsymbol{\epsilon}_j$$

Connection to Previous Methods

DAPS points on the trajectory have significantly larger variations compared to those of DPS.

→ DPS converges to wrong solutions, but DAPS captures the true posterior distribution.



Experimental Setup

Inverse Problem:

- Linear: Super-resolution, Gaussian deblurring, motion blurring, inpainting.
- Non-Linear: Phase retrieval, high dynamic range (HDR) reconstruction, nonlinear deblurring.

Baseline:

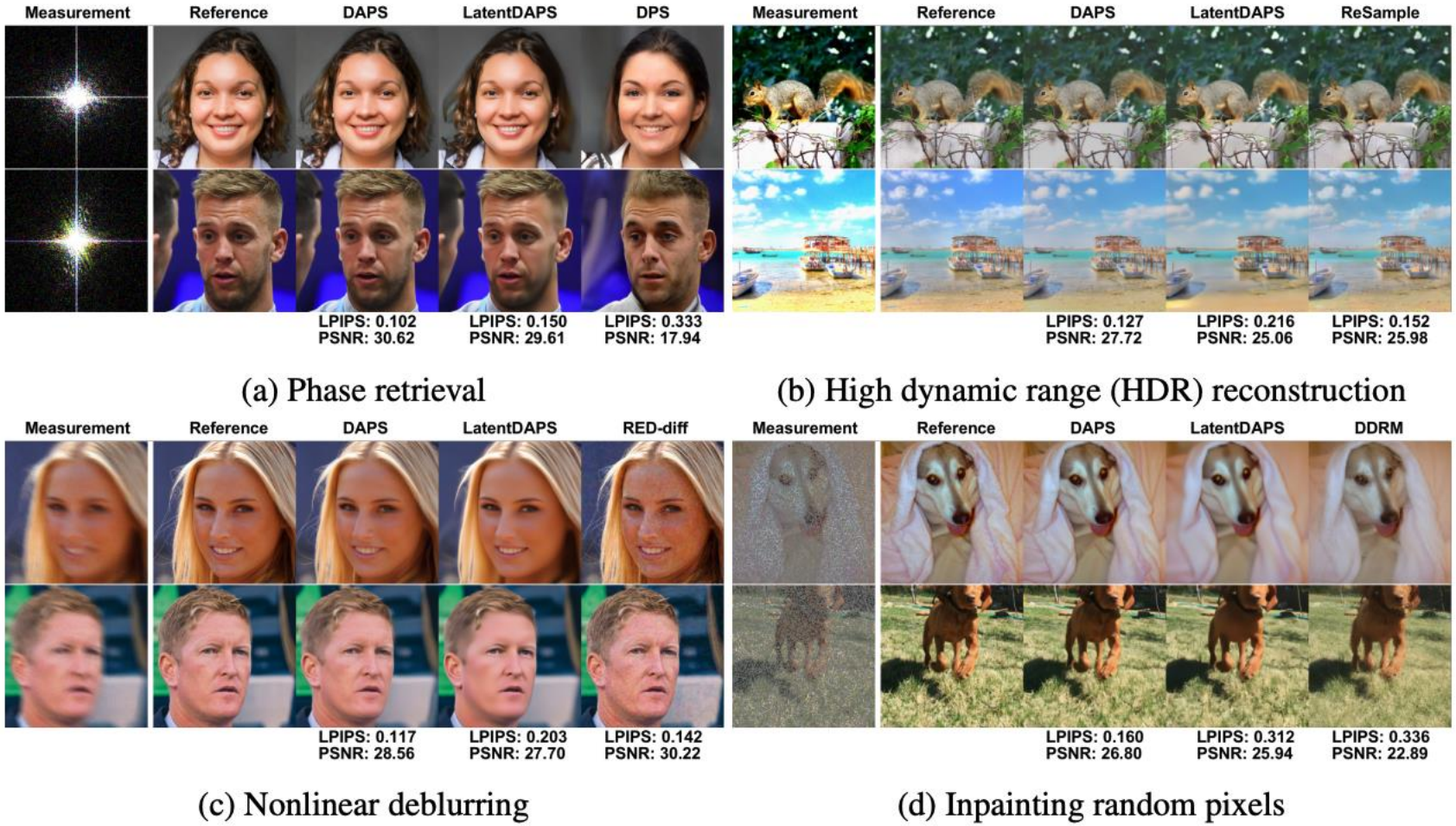
- Optimization: DPS, Resample, PSLD
- Spectral Denoising: DDNM, DDRM

Evaluation Metric:

- LPIPS, PSNR

Qualitative Evaluation

DAPS and LatentDAPS (DAPS counterpart in latent space) achieve superior perceptual quality.



Quantitative Evaluation

DPS suffers from an extremely high failure rate, as seen in their low PSNR and high LPIPS.

Table 3: **Quantitative evaluation on ImageNet 256×256**. Performance comparison of different methods on various linear tasks in image domain.

Method	SR (×4)		Inpaint (Box)		Inpaint (Random)		Gaussian deblurring		Motion deblurring	
	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑	LPIPS↓	PSNR↑
DAPS(ours)	**0.276**	**25.89**	**0.214**	21.43	**0.135**	28.44	**0.253**	26.15	**0.196**	**27.86**
DPS	0.354	23.92	0.309	19.78	0.326	24.43	0.360	21.86	0.357	21.46
DDRM	0.284	25.21	0.229	19.45	0.325	23.23	0.341	23.86	-	-
DDNM	0.475	23.96	0.319	**21.64**	0.191	31.16	0.278	**28.06**	-	-
PnP-ADMM	0.724	22.18	0.702	12.61	0.680	20.03	0.729	20.47	0.684	24.23
LatentDAPS(ours)	0.343	25.06	0.340	17.19	0.219	27.59	0.349	25.05	0.296	26.83
PSLD	0.360	25.42	0.465	20.10	0.337	**31.30**	0.390	25.86	0.511	20.85
ReSample	0.370	22.61	0.262	18.29	0.143	27.50	0.254	25.97	0.227	26.94

Table 4: **Quantitative evaluation on ImageNet 256×256**. Performance comparison of different methods on various **nonlinear** tasks in the image domain. The mean and standard deviation are computed over 100 images.

Method	Phase retrieval		Nonlinear deblurring		High dynamic range	
	LPIPS↓	PSNR↑	LPIPS↓	PSNR ↑	LPIPS↓	PSNR↑
DAPS(ours)	**0.254** (0.125)	**25.78** (6.92)	**0.169** (0.056)	27.73 (3.23)	**0.175** (0.107)	**26.30** (4.15)
DPS	0.447 (0.099)	16.81 (3.61)	0.306 (0.081)	22.49 (3.20)	0.503 (0.106)	19.23 (2.52)
RED-diff	0.536 (0.129)	14.98 (3.75)	0.211 (0.083)	**30.07** (1.41)	0.274 (0.198)	22.03 (5.90)
LatentDAPS (ours)	0.361 (0.150)	20.54 (6.41)	0.314 (0.080)	25.34 (3.44)	0.269 (0.099)	23.64 (4.10)
ReSample	0.403 (0.174)	19.24 (4.21)	0.206 (0.057)	26.20 (3.71)	0.198 (0.089)	25.11 (4.21)

Ablation Study

Effective NFE (Number of Function Evaluations) of DAPS = ODE steps x timesteps.

Outperforms other baselines even with small NFEs.

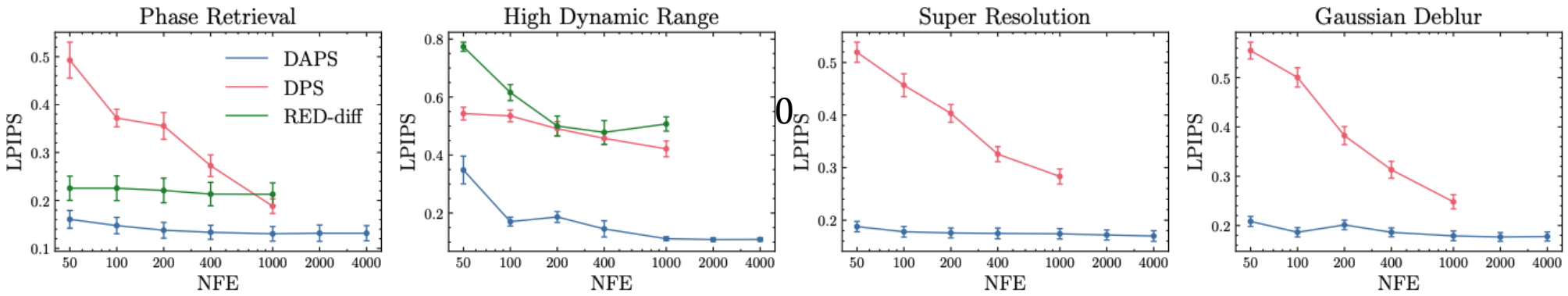


Figure 7: **Quantitative evaluations of image quality for different number of function evaluations (NFE).** Experiments are conducted on the FFHQ 256 dataset for four different tasks.

Limitation

- Only tested using FFHQ and ImageNet. (How scalable is this ?)
- $\mathbf{x}_{0|\mathbf{y}}$ obtained after Langevin dynamics may not lie in the manifold of clean images.
- Needs to evaluate the forward model at every Langevin dynamics step.
- Additional NFE from running the ODE every step.

Conclusion

- Previous diffusion-based inverse problem solvers cannot recover from bad samples.
- DAPS decouples sampling process by running Langevin Dynamics on $\mathbf{x}_0$.
- DAPS is particularly effective in solving inverse problems with complex nonlinear measurement processes such as phase retrieval.