ENSEMBLE KALMAN SAMPLING AND DIFFUSION PRIOR IN TANDEM: A SPLIT GIBBS FRAMEWORK

Austin Wang[1], **Hongkai Zheng**[1], **Zihui Wu**[1], **Ricardo Baptista**[1], **Daniel Zhengyu Huang**[2], **Yisong Yue**[1]
[1]Caltech, [2]Peking University

ABSTRACT

We study the inverse problems in the derivative-free setting, where the forward model only permits black-box access. Traditional derivative-free methods, such as Markov chain Monte Carlo (MCMC) and Sequential Monte Carlo (SMC), have limited applicability in this problem due to their reliance on the prior density access (up to a normalizing constant), which is intractable in many applications. Recent works leveraging diffusion models (DMs) as flexible priors show promise for high-dimensional inverse problems, but we find they often deviate from the true posterior, even in linear Gaussian case. To address these limitations, we propose SG-EKDP (Split Gibbs with Ensemble Kalman sampling and Diffusion Prior), a novel algorithm that integrates ensemble Kalman sampling and diffusion prior within a split Gibbs sampling framework. Our method provably converges to the posterior in the mean-field limit for general linear inverse problems and remains effective in nonlinear settings via local linearization. We evaluate the effectiveness of SG-EKDP across various inverse problems. Numerical experiments on linear Gaussian and Gaussian mixture show that SG-EKDP can accurately approximate the true posterior. It also achieves strong empirical performance on high-dimensional image restoration tasks including both linear and nonlinear problems.

1 INTRODUCTION

We consider inverse problems of the following form:

$$\boldsymbol{y} = \mathcal{G}(\boldsymbol{x}^*) + \boldsymbol{\epsilon}, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^m$ is the observed data, $\boldsymbol{x}^* \in \mathbb{R}^n$ is the unknown signal, and $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is the measurement noise, typically modeled as Gaussian $\mathcal{N}(0, \sigma_y^2 \mathbf{I})$. The goal of Bayesian inversion is to estimate a posterior distribution over the unknown $\boldsymbol{x}_0$ given the noisy measurement $\boldsymbol{y}$. In this work, we study the **derivative-free scenario** where only black-box access to the forward model $\mathcal{G}$ is assumed, and derivative or adjoint information is either unavailable or computationally infeasible to compute. Such a setting arises in many scientific and engineering applications (Bougeault et al., 2010; Schneider et al., 2017). For example, it occurs when the forward model is the numerical solver of partial differential equations (PDEs) and the numerical methods are not differentiable (e.g. embedded boundary method (Peskin, 1977; Huang et al., 2020) and adaptive mesh refinement (Borker et al., 2019)) or differentiating through the iterative solver's large computation graph is impractical (Kochkov et al., 2021).

Traditional methods for derivative-free Bayesian inverse problems include Markov chain Monte Carlo (MCMC) methods (Geyer, 1992; Gelman et al., 1997; Cotter et al., 2013) and Sequential Monte Carlo (SMC) (Del Moral et al., 2006). These methods enjoy theoretical convergence guarantees but face significant scalability and flexibility challenges in real-world applications, especially in high dimensions. A key limitation is their reliance on knowledge of the prior density (up to a normalizing constant), which may be unavailable or computationally intractable for complex, high-dimensional prior distributions. More recent ensemble methodology (Garbuno-Inigo et al., 2020; Carrillo et al., 2022) are developed with improved efficiency but are restricted to linear Gaussian case.

To address these challenges, recent work has explored the use of diffusion models (Song et al., 2020b; Ho et al., 2020) as plug-and-play priors for inverse problems. Diffusion priors can be easily learned from a set of prior samples and are broadly applicable across different domains. Existing derivative-free algorithms (Zheng et al., 2024; Tang et al., 2024; Huang et al., 2024) with diffusion prior have demonstrated effectiveness in solving inverse problems where traditional methods struggle, across applications such as image restoration, symbolic music generation, and fluid dynamics. However, these approaches often fail to approximate the true posterior distribution, as we demonstrate in simple settings like linear Gaussian and Gaussian mixture cases in Figure 1. Meanwhile, other works (Cardoso et al., 2024; Dou & Song, 2024) have achieved asymptotic convergence to the posterior by integrating diffusion priors into SMC methodologies, but these methods are restricted to linear problems and do not generalize to nonlinear settings.

Our contributions In this work, we propose a new derivative-free algorithm for posterior estimation, which we refer to as SG-EKDP (Split Gibbs with Ensemble Kalman sampling and Diffusion Prior). Our method effectively handles complex prior distributions by leveraging expressive diffusion priors while maintaining provable convergence to the true posterior in general linear inverse problems. Furthermore, for nonlinear problems, SG-EKDP provides an effective approximation by employing a local linearization strategy. Numerical experiments on linear Gaussian and Gaussian mixture show that the proposed SG-EKDP can accurately capture the true posterior, outperforming the competing algorithms. Additionally, we evaluate SG-EKDP on various image restoration tasks, spanning both linear and nonlinear inverse problems, where it exhibits strong empirical performance.

2 BACKGROUND

2.1 DIFFUSION MODELS

We consider diffusion models in the unified EDM framework (Karras et al., 2022). Diffusion models define a forward stochastic process to evolve the original data distribution $p_0(\boldsymbol{x})$ to an approximately Gaussian distribution $p_T(\boldsymbol{x}) = \mathcal{N}(0, s^2(T)\sigma(T)^2\mathbf{I})$, where $\sigma(t)$ is a pre-defined noise schedule function and $s(t)$ is the pre-defined scaling function. Without loss of generality, we set $s(t) = 1$ because every other schedule is equivalent to it up to a simple reparameterization as shown in Karras et al. (2022). The corresponding reverse process sequentially denoises the noisy data into clean data given by

$$\mathrm{d}\boldsymbol{x}_t = -2\dot{\sigma}(t)\sigma(t)\nabla_{\boldsymbol{x}_t} \log p\left(\boldsymbol{x}_t; \sigma(t)\right) \mathrm{d}t + \sqrt{2\dot{\sigma}(t)\sigma(t)}\mathrm{d}\bar{\boldsymbol{w}}_t. \tag{2}$$

Generating new samples from $p_0(\boldsymbol{x})$ amounts to integrating Eq. (2) from a random noise sample from $p_T(\boldsymbol{x}_T)$. Crucially, this requires computation of the time-dependent score function $\nabla \log p(\boldsymbol{x}_t; \sigma(t))$. To this end, training a diffusion model amounts to learning the score function with a neural network parameterized by θ, given by $s_\theta(\boldsymbol{x}_t, t) \approx \nabla \log p(\boldsymbol{x}_t; \sigma(t))$. In our work, we assume that we have access to such a pre-trained score function, which we will simply refer to as the diffusion model.

2.2 SPLIT GIBBS SAMPLING

The Split Gibbs Sampler (SGS) (Vono et al., 2019) is a Markov chain Monte Carlo (MCMC) method that aims to sample the posterior distribution

$$p(\boldsymbol{x} \mid \boldsymbol{y}) \propto p(\boldsymbol{y} \mid \boldsymbol{x})p(\boldsymbol{x}) = \exp(-f(\boldsymbol{x}; \boldsymbol{y}) - g(\boldsymbol{x})), \tag{3}$$

where $f(\boldsymbol{x}; \boldsymbol{y}) = -\log p(\boldsymbol{y} \mid \boldsymbol{x})$ is the negative log-likelihood and $g(\boldsymbol{x}) = -\log p(\boldsymbol{x})$ is the negative log-prior. Instead of directly sampling Eq. (3), SGS aims to sample the auxiliary distribution

$$\pi^{XZ}(\boldsymbol{x}, \boldsymbol{z}) \propto \exp\left(-f(\boldsymbol{z}; \boldsymbol{y}) - g(\boldsymbol{x}) - \frac{1}{2\rho^2}\left\|\boldsymbol{x} - \boldsymbol{z}\right\|_2^2\right), \tag{4}$$

where $\boldsymbol{z} \in \mathbb{R}^n$ is an auxiliary variable and ρ is a parameter that controls the distance between $\boldsymbol{x}$ and $\boldsymbol{z}$. As shown in Vono et al. (2019), as $\rho \to 0$, the marginal distribution $\pi^X(\boldsymbol{x}) = \int \pi(\boldsymbol{x}, \boldsymbol{z})\mathrm{d}\boldsymbol{z}$

converges to the posterior distribution $p(\boldsymbol{x} \mid \boldsymbol{y})$ in total variation distance. This means that sampling the posterior is equivalent to sampling Eq. (4) as ρ approaches 0. Sampling Eq. (4) can be done with Gibbs sampling. Given an initial point $\boldsymbol{x}^{(0)}$ and for iterations $k \in \{0, \cdots, K-1\}$:

1. Likelihood step: sample $\boldsymbol{z}^{(k)} \sim \pi^{Z|X=\boldsymbol{x}^{(k)}}(\boldsymbol{z}) \propto \exp\left(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\rho^2}\left\|\boldsymbol{x}^{(k)} - \boldsymbol{z}\right\|_2^2\right)$
2. Prior step: sample $\boldsymbol{x}^{(k+1)} \sim \pi^{X|Z=\boldsymbol{z}^{(k)}}(\boldsymbol{x}) \propto \exp\left(-g(\boldsymbol{x}) - \frac{1}{2\rho^2}\left\|\boldsymbol{x} - \boldsymbol{z}^{(k)}\right\|_2^2\right)$

3 Related work

3.1 Diffusion-based Split Gibbs Sampling

Recent works (Bouman & Buzzard, 2023; Coeurdoux et al., 2023; Xu & Chi, 2024; Wu et al., 2024) have explored adapting generative model priors into the split Gibbs framework. These methods primarily differ in how they implement the prior step. Bouman & Buzzard (2023) employs a proximal Gaussian approximation by invoking first-order Taylor expansion and a diffusion model denoiser. Coeurdoux et al. (2023) executes a diffusion denoising process with a starting time determined by a wavelet-based estimator. Both Xu & Chi (2024) and Wu et al. (2024) also formulate the prior step as a denoising diffusion process. However, Xu & Chi (2024) focuses on adapting the DDPM and DDIM (Song et al., 2020a) samplers whereas Wu et al. (2024) connects the implementation to the EDM framework Karras et al. (2022), enabling the use of any pre-trained diffusion model as the generative prior (as opposed to relying specifically on DDPM). Our work closely follows the prior step of Wu et al. (2024) but with a fundamental difference in the likelihood step. Instead of the gradient-based Langevin Monte Carlo that Wu et al. (2024) uses, we introduce an interactive particle system to approximate the distribution using only black-box access to the forward model. This distinction allows our approach to be applicable in a wider range of settings where derivative information is unavailable or impractical.

3.2 Ensemble Kalman methods

Ensemble Kalman methodology was first introduced by Evensen (1994) in the context of filtering problems and has since gained popularity in applications such as reservoir modeling (Oliver & Chen, 2011) and weather forecasting (Houtekamer & Zhang, 2016) due to its derivative-free nature and effectiveness in high-dimensional settings. Iglesias et al. (2013) revisits this idea in the context of the inverse problem, proposing Ensemble Kalman Inversion (EKI). Further developments include momentum-based EKI for training neural networks (Kovachki & Stuart, 2018) and various regularization techniques to improve stability and efficiency (Iglesias, 2016; Chada et al., 2020; Iglesias & Yang, 2021). Zheng et al. (2024) extends this idea into the diffusion model guidance for the first time, enabling the use of more flexible and complex priors in inverse problems. More recently, Kim et al. (2024) explores a similar approach in explainable AI. Our method builds upon Ensemble Kalman Sampling (EKS) (Garbuno-Inigo et al., 2020), which provides provable convergence to the posterior in the mean-field limit for linear Gaussian problems. However, EKS is restricted to Gaussian priors. In contrast, our approach extends to a broader class of prior distributions by leveraging diffusion models and split Gibbs framework while retaining the convergence properties towards the true posterior.

4 Method

Our method builds upon the split Gibbs sampling framework studied in the papers (Vono et al., 2019; Yuan et al., 2023; Wu et al., 2024). The proposed algorithm, SG-EKDP, operates by evolving an ensemble of interactive particles that alternate between a likelihood step and a prior step. The likelihood step is built upon ensemble Kalman sampling (Garbuno-Inigo et al., 2020) and the prior step is a denoising diffusion process (Karras et al., 2022). Subsection 4.1 details the likelihood step, Subsection 4.2 describes the prior step, and Subsection 4.3 outlines the full algorithm along with a convergence analysis for general linear inverse problems.

Algorithm 1 Our Method

Require: initial ensemble $\mathbf{X}^{(0)} = \{\mathbf{x}^{(j)} \in \mathbb{R}^n\}_{j=1}^J$, number of iterations K, $\{\rho_k\}_{k=0}^K$, observed data $\mathbf{y} \in \mathbb{R}^m$, pre-trained diffusion model D_θ

for $k \in \{0, \dots, K-1\}$ **do**
  $\mathbf{Z}^{(k)} \leftarrow \textsc{LikelihoodStep}(\mathbf{X}^{(k)}, \rho_k)$ ▷ Algorithm 2
  $\mathbf{X}^{(k+1)} \leftarrow \textsc{PriorStep}(\mathbf{Z}^{(k+1)}, D_\theta, \rho_k)$ ▷ Algorithm 3
end for
return $\mathbf{X}^{(K)}$

Algorithm 2 Likelihood Step

Require: forward model $\mathcal{G}, \mathbf{y}, \rho$, number of steps N, step size scale λ, initial ensemble $\mathbf{X}$

$\mathbf{Z}_0 \leftarrow \mathbf{X}$
for $i \in \{0, \dots, N-1\}$ **do**
  $\epsilon_i \sim \mathcal{N}(0, \mathbf{I})$
  $\mathrm{d}_1^{(j)} \leftarrow -\frac{1}{J}\sum_{g=1}^J \langle \mathcal{G}(\boldsymbol{z}_i^{(g)}) - \bar{\mathcal{G}}, \mathcal{G}(\boldsymbol{z}_i^{(j)}) - \boldsymbol{y}\rangle_\Gamma (\boldsymbol{z}_i^{(g)} - \bar{\boldsymbol{z}}_i), j = 1, \dots, J$
  $\mathrm{d}_2^{(j)} \leftarrow -\frac{1}{\rho^2}\mathbf{C}_i(\boldsymbol{x}^{(j)} - \boldsymbol{z}_i^{(j)}), j = 1, \dots, J$
  $\eta \leftarrow \frac{\lambda}{\|\mathrm{d}_1 + \mathrm{d}_2\|_2^2}$
  $\boldsymbol{z}_{i+1}^{(j)} \leftarrow \boldsymbol{z}_i^{(j)} + (\mathrm{d}_1^{(j)} + \mathrm{d}_2^{(j)})\eta + \sqrt{2\mathbf{C}_i\eta}\epsilon_i, j = 1, \dots, J$
end for
return $\mathbf{Z}_N$

Algorithm 3 Prior Step

Require: Denoiser network D_θ, ρ, number of steps N, initial ensemble $\mathbf{Z}$, $\sigma(t) = t$, $s(t) = 1$, $t_{i \in \{0, \cdots, N\}}$

$\mathbf{X}_0 \leftarrow \mathbf{Z}$
$i^* \leftarrow \min\{i \geq 0 \mid \sigma(t_i) \leq \rho\}$
for $i \in \{i^*, \dots, N-1\}$ **do**
  $\lambda \leftarrow 2$ **if** *SDE* **else** 1
  $\hat{\mathbf{X}}_0 \leftarrow D_\theta(\mathbf{X}_i, \sigma(t_i))$
  $\mathrm{d}_i \leftarrow \lambda \frac{\mathbf{X}_i - \hat{\mathbf{X}}_0}{t_i}$
  $\mathbf{X}_{i+1} \leftarrow \mathbf{X}_i + (t_{i+1} - t_i)\mathrm{d}_i$
  if $i \neq N-1$ **and** *SDE* **then**
    $\epsilon_i \sim \mathcal{N}(0, \mathbf{I})$
    $\mathbf{X}_{i+1} \leftarrow \mathbf{X}_{i+1} + \sqrt{2t_i(t_i - t_{i+1})}\epsilon_i$
  end if
end for
return $\mathbf{X}_N$

4.1 Likelihood Step

Let $\mathbf{X}^{(k)} = \{\boldsymbol{x}^{(j)}\}_{j=1}^J$ denote the ensemble of J particles at k-th alternating iteration of the SGS framework. In the likelihood step, we aim to sample $\boldsymbol{z}^{(j)}$ from $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z}) \propto \exp(-f(\boldsymbol{z}; \boldsymbol{y}) - \frac{1}{2\rho^2}\|\boldsymbol{z} - \boldsymbol{x}^{(j)}\|_2^2)$ for each $j \in \{1, \dots, J\}$. Our starting point is the covariance-preconditioned Langevin dynamics with the large particle limit:

$$\mathrm{d}\boldsymbol{z}_t^{(j)} = -C_t \nabla \left(f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y}) + \frac{1}{2\rho^2}\|\boldsymbol{z}_t^{(j)} - \boldsymbol{x}^{(j)}\|_2^2 \right) \mathrm{d}t + \sqrt{2C_t}\mathrm{d}\boldsymbol{w}_t, \tag{5}$$

where q_t is the particle distribution and

$$\bar{\boldsymbol{z}}_t := \mathbb{E}_{q_t}[\boldsymbol{z}_t], \quad C_t := \mathbb{E}_{q_t}[(\boldsymbol{z}_t - \bar{\boldsymbol{z}}_t)(\boldsymbol{z}_t - \bar{\boldsymbol{z}}_t)^\top].$$

As shown in Lemma 1, Eq. (5) admits $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z})$ as its stationary distribution, under the mild assumption that the particle distribution does not collapse to a Dirac measure. For the inverse problem in Eq. (1), we have $f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y}) = \frac{1}{2\sigma_y^2}\|\mathcal{G}(\boldsymbol{z}_t^{(j)}) - \boldsymbol{y}\|_2^2$. Therefore, running Eq. (5) relies on the derivative of the forward model $\mathcal{G}$, which may not be available. To circumvent this, we approximate $\mathcal{G}$ with a linear surrogate model $\boldsymbol{y} = \hat{A}\boldsymbol{z}_t + \boldsymbol{b}$ with the minimal least square error defined by

$$\min_{A, \boldsymbol{b}} \mathbb{E}_{q_t}\|\mathcal{G}(\boldsymbol{z}_t) - (A\boldsymbol{z}_t + \boldsymbol{b})\|_2^2.$$

Setting the derivatives w.r.t. A and $\boldsymbol{b}$ to zero gives the closed-form solution:

$$\hat{A} = \mathbb{E}_{q_t}[(\mathcal{G}(\boldsymbol{z}_t) - \mathbb{E}_{q_t}\mathcal{G}(\boldsymbol{z}_t))\boldsymbol{z}_t^\top]C_t^{-1}, \boldsymbol{b} = \mathbb{E}_{q_t}\mathcal{G}(\boldsymbol{z}_t) - \mathbb{E}_{q_t}\hat{A}\boldsymbol{z}_t,$$

where C_t^{-1} is the pseudo-inverse of the covariance matrix. This statistical linearization, first introduced in (Booton, 1954), is exact when $\mathcal{G}$ is linear. Let $D\mathcal{G}$ denote the Jacobian of $\mathcal{G}$. Replacing $D\mathcal{G}$ with $D\hat{A} = \hat{A}$ in $\nabla f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y})$ yields

$$\nabla f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y}) = \frac{1}{\sigma_y^2} D^\top \mathcal{G}(\mathcal{G}(\boldsymbol{z}_t^{(j)}) - \boldsymbol{y}) \approx \frac{1}{\sigma_y^2} C_t^{-1} \mathbb{E}_{q_t}[\boldsymbol{z}_t(\mathcal{G}(\boldsymbol{z}_t) - \mathbb{E}_{q_t}\mathcal{G}(\boldsymbol{z}_t))^\top](\mathcal{G}(\boldsymbol{z}_t^{(j)}) - \boldsymbol{y}). \quad (6)$$

Substituting Eq. (6) into Eq. (5) gives

$$\mathrm{d}\boldsymbol{z}_t^{(j)} = -\left[\frac{1}{\sigma_y^2}\mathbb{E}_{q_t}[\boldsymbol{z}_t(\mathcal{G}(\boldsymbol{z}_t) - \mathbb{E}_{q_t}\mathcal{G}(\boldsymbol{z}_t))^\top](\mathcal{G}(\boldsymbol{z}_t^{(j)}) - \boldsymbol{y}) + \frac{1}{\rho^2} C_t(\boldsymbol{z}_t^{(j)} - \boldsymbol{x}^{(j)})\right] \mathrm{d}t + \sqrt{2C_t}\mathrm{d}\boldsymbol{w}_t. \quad (7)$$

Eq. (7) is equivalent to Eq. (5) when $\mathcal{G}$ is linear and serves as an effective approximation when $\mathcal{G}$ is nonlinear. Crucially, it eliminates the need for derivatives of the forward model, allowing us to solve inverse problems with black-box access to $\mathcal{G}$.

Practical implementation For the numerical implementation of Eq. (7), we employ Euler-Maruyama method with a finite ensemble of particles. Let $\boldsymbol{z}_i^{(j)}$ denote the j-th particle at discrete time t_i, and C_i denote the empirical covariance at t_i. To address several practical challenges in high-dimensional settings, we employ the following strategies. First, computing the matrix square root $\sqrt{2C_i}$ can be computationally expensive. Therefore, when Cholesky decomposition becomes inefficient, we use a diagonal approximation for C_i when computing the square root. Second, the empirical covariance matrix C_i may be ill-conditioned in practice. To ensure numerical stability during Cholesky decomposition, we add a small regularization term, replacing $\sqrt{2C_i}$ with $\sqrt{2C_i + \epsilon I}$, where ϵ is a small positive constant. Third, the time step size for the discretization is chosen adaptively to balance accuracy and efficiency. The details are provided in 7.2.2. The resulting likelihood step, summarized in Algorithm 2, is similar to Ensemble Kalman Sampling (EKS) (Garbuno-Inigo et al., 2020). However, a key distinction is that each particle in our likelihood step has its own target distribution, whereas EKS assumes a common target distribution for all particles.

4.2 Prior Step

Let $\mathbf{Z}^{(k)} = \{\boldsymbol{z}^{(j)}\}_{j=1}^J$ denote the ensemble of J particles at k-th alternating iteration of the SGS framework. In the likelihood step, we aim to sample $\boldsymbol{x}^{(j)}$ from $\pi^{X|Z=\boldsymbol{z}^{(j)}}(\boldsymbol{x}) \propto \exp(-g(\boldsymbol{x}) - \frac{1}{2\rho^2}\|\boldsymbol{x} - \boldsymbol{z}^{(j)}\|_2^2)$ for each $j \in \{1, \ldots, J\}$. As shown in Coeurdoux et al. (2023); Wu et al. (2024), the prior step can be formulated as denoising diffusion process. Specifically, recall that the forward process gives $p(\boldsymbol{x}_t \mid \boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{x}_0, \sigma(t)^2\mathbf{I})$ under the EDM (Karras et al., 2022) framework, where $s(t) = 1$. By Bayes' theorem, we have

$$p(\boldsymbol{x}_0 \mid \boldsymbol{x}_t) \propto p(\boldsymbol{x}_t \mid \boldsymbol{x}_0)p(\boldsymbol{x}_0) \propto \exp\left(-g(\boldsymbol{x}_0) - \frac{1}{2\sigma^2(t)}\|\boldsymbol{x}_0 - \boldsymbol{x}_t\|_2^2\right). \quad (8)$$

By comparing Eq. (8) with the target distribution $\pi^{X|Z=\boldsymbol{z}^{(j)}}(\boldsymbol{x})$, we can see that if $\rho = \sigma(t)$ and $\boldsymbol{x}_t = \boldsymbol{z}^{(j)}$, sampling from $\pi^{X|Z=\boldsymbol{z}^{(j)}}(\boldsymbol{x})$ is equivalent to sampling from $p(\boldsymbol{x}_0|\boldsymbol{x}_t = \boldsymbol{z}^{(j)})$. Therefore, the prior step of j-th particle can be implemented as the standard reverse process of the diffusion model given by Eq. (2) starting from $\boldsymbol{z}^{(j)}$ at time t^* where t^* is chosen so that $\sigma(t^*) = \rho$. While many numerical solvers can be used in this setting, we use the Euler solver for the corresponding probability flow (PF) ODE (or SDE). Detailed pseudocode for the prior step can be found in Algorithm 3. Note that in Algorithm 3, $\mathbf{X}_i$ represents the ensemble of particles at t_i step, and the updates for all particles can be computed in parallel.

4.3 Putting it together

We provide pseudocode for the complete sampling algorithm in Algorithm 1. The method operates by iteratively updating an ensemble of particles, alternating between the likelihood and prior steps discussed above. At the same time, the parameter ρ follows an annealing schedule that gradually decreases towards zero. This annealing schedule helps refine the approximation, allowing the samples to converge to the target posterior distribution as shown in Vono et al. (2019). We employ two

strategies for initializing the ensemble in practice. The first samples from a Gaussian $\mathcal{N}(0, \rho_0^2\mathbf{I})$, drawing from the intuition that ρ_k acts as a noise level at each step of the algorithm. The second strategy initializes the ensemble by sampling from the prior distribution, represented by the diffusion model. In principle, any initialization distribution should allow for convergence as long as the KL divergence between the target and initialization distributions is finite.

Theoretical analysis We analyze the convergence of the proposed algorithm through its continuous-time and mean-field limits, leveraging the existing non-asymptotic convergence analysis from Wu et al. (2024) that builds upon the interpolation techniques developed in Vempala & Wibisono (2019); Sun et al. (2024). In this work, we focus on the convergence analysis for linear problems. Extending this analysis to nonlinear problems is theoretically possible, particularly for smooth forward models, but accounting for the approximation error introduced by the linearization remains an open question. We leave this for future work.

Theorem 1 (Informal)**.** *Consider the problems where $\mathcal{G}$ is linear. If the particle distribution is not a Dirac measure, then π^{XZ} is a stationary distribution of SG-EKDP for any given $\rho > 0$. Furthermore, if the particle covariance C_t is positive definite, π^{XZ} is the unique stationary distribution. Suppose ν_τ and π_τ the distributions at time τ of the non-stationary and stationary process. We assume $D_{\mathrm{KL}}(\pi^X||\nu_0^X) < +\infty$. For K iterations of Algorithm 1 or equivalently $T := K(t^* + 1)$, we have the following convergence results:*

$$\frac{1}{T}\int_0^T D_{\mathrm{FI}}(\pi_\tau||\nu_\tau)\mathrm{d}\tau \leq \frac{4D_{\mathrm{KL}}(\pi^X||\nu_0^X)}{T\min(\rho,\delta)^2} + \frac{4\epsilon_{\mathrm{score}}}{(t^*+1)\delta^2}, \tag{9}$$

where D_{FI} is the Fisher divergence, D_{KL} is the Kullback–Leibler (KL) divergence, $\epsilon_{\mathrm{score}}$ is the score approximation error of the pre-trained diffusion model, t^* is the starting time of the denoising process defined in Eq. (2) *such that $\sigma(t^*) = \rho$, and δ is the smallest diffusion coefficient used for the prior step.*

The formal version and complete proof are deferred to Appendix 7.1. Intuitively, Theorem 1 states that $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z})$is a stationary distribution of SG-EKDP for general linear inverse problems. If the covariance matrix is positive definite, $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z})$ is the unique stationary distribution. Further, if the initial distribution has a finite KL divergence from the stationary distribution, then the average Fisher divergence from the target stationary process converges at an algebraic rate. This result is similar to the result in Wu et al. (2024) but introduces additional conditions such as the linear forward model and positive definite covariance matrix. Notably, the linearity condition can be relaxed by smooth perturbations, though a detailed theoretical analysis of this extension is left for future work.

5 Experiments

To evaluate how well our method estimates the posterior, we begin by comparing it against baselines on linear Gaussian and Gaussian mixture problems in Subsection 5.1, where the ground truth posterior distributions are available in closed form. We then extend our evaluation to various image restoration tasks in Subsection 5.2, including both linear and nonlinear inverse problems, to study the effectiveness of our approach in high-dimensional settings with complex image priors. We primarily focus on the comparison against other derivative-free diffusion-based baselines: EnKG (Zheng et al., 2024), SCG (Huang et al., 2024), and DPG (Tang et al., 2024). We use the same pre-trained diffusion model checkpoint for each method across all tasks for fair comparison. All the experiments are conducted on a single A100 GPU.

5.1 Compare with ground truth posterior

Problem Setting We first evaluate our method on a synthetic data inverse problem, given by the following measurement model:

$$\boldsymbol{y} = \mathbf{H}\boldsymbol{x}^* + \epsilon, \tag{10}$$

where $\epsilon \sim \mathcal{N}(0, \sigma_y^2\mathbf{I})$ and $\mathbf{H} \in \mathbb{R}^{m\times n}$ is a linear forward model. We consider the **linear Gaussian** setting, where $\boldsymbol{x}^*$ follows the prior distribution $\mathcal{N}(m_{\boldsymbol{x}}, \sigma_{\boldsymbol{x}}^2\mathbf{I})$, and the **linear Gaussian mixture**

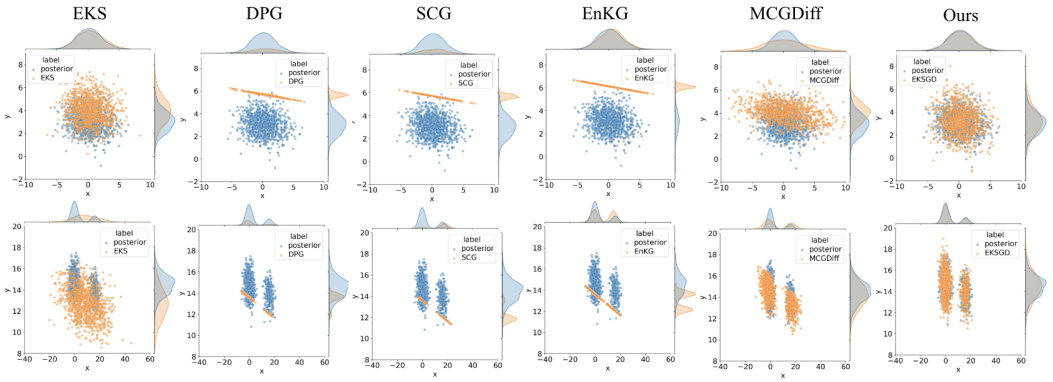

Figure 1: Linear Gaussian and Linear Gaussian mixture inverse problems, $\boldsymbol{x} \in \mathbb{R}^2$. **Top row**: linear Gaussian results; ground truth posterior samples, and generated samples depicted. **Bottom row**: linear Gaussian mixture results; ground truth posterior samples and generated samples depicted.

setting, where $\boldsymbol{x} \sim \sum_{i=1}^{K} \gamma_i \mathcal{N}(m_i, \Sigma_i), \sum_{i=1}^{K} \gamma_i = 1$. For evaluation, the prior mean, standard deviation, and forward model $\mathbf{H}$ are pre-established. As this problem is linear and uses a known prior, we have knowledge of the ground truth posterior distribution $p(\boldsymbol{x}^* \mid \boldsymbol{y})$; the derivation of the Gaussian mixture posterior can be found in Section 7.3. We then generate 512 samples from SG-EKDP, ENKG, DPG, SCG, EKS, MCGDIFF and quantitatively compare against the posterior distribution.

Metrics and Results For the linear Gaussian problem, we use the sliced Wasserstein distance and KL Divergence metrics to measure the quality of posterior sampling. Wasserstein distance is computed directly between generated samples and samples from the posterior distribution. For computation of KL divergence, we use the known posterior mean and covariance, and the empirical mean and covariance of the generated samples. Numerical results can be found in Table 1.

As seen, our method consistently outperforms other derivative-free diffusion-based algorithms in both metrics. Notably, our method is the most robust against measurement noise σ_y, achieving relatively low Wasserstein distance and KL divergence even as uncertainty increases. In the Gaussian mixture case, our method accurately recovers the mean and covariance of each mode of the posterior, whereas other diffusion-based methods fail to approximate the spread, and EKS cannot capture the different modes.

5.2 IMAGE RESTORATION

Problem Setting We evaluate our algorithm on the standard FFHQ256 dataset. Of the three image restoration problems we chose, inpainting and super-resolution are linear, while Fourier phase retrieval is nonlinear. We use a random box mask for inpainting. Bicubic downsampling is used for superresolution ($\times 4$). Finally, for phase retrieval, we compute the magnitude of the Fourier transform of the input image. All problem settings use a measurement noise of $\sigma_y = 0.05$ and follow the same configuration as that of DPS (Chung et al., 2023), unless otherwise stated. The pre-trained model is taken from Chung et al. (2023) (FFHQ256) and converted into an EDM checkpoint with their Variance-Preserving (VP) preconditioning (Karras et al., 2022).

Metrics and Results For image tasks, we measure the Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) (Wang et al., 2004), and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018) between generated samples and the ground truth image. Notably, our method is able to achieve strong PSNR results, even when compared to gradient-based methods, reflecting its ability to generate high-likelihood samples. Results on the phase retrieval task demonstrate robustness to highly ill-posed, nonlinear forward models.

Table 1: Experimental Results for Sliced Wasserstein Distance (SWD) and KL Divergence ($\mathbb{D}_{\text{KL}}$) across different σ_y values and methods for various d_x.

d_x	Method	σ_y							
		0.5		1.5		2.5		3.5	
		SWD	$\mathbb{D}_{\text{KL}}$	SWD	$\mathbb{D}_{\text{KL}}$	SWD	$\mathbb{D}_{\text{KL}}$	SWD	$\mathbb{D}_{\text{KL}}$
2	DPG	4.199	11.83	3.955	122.11	3.969	350.76	4.646	1219.96
	SCG	2.826	85k	2.704	$\geq$ 100k	3.072	$\geq$ 100k	3.814	$\geq$ 100k
	EnKG	1.832	$\geq$ 100k	1.752	$\geq$ 100k	1.972	$\geq$ 100k	3.020	$\geq$ 100k
	EKS	1.651	**0.374**	2.072	0.493	2.061	0.502	2.204	0.549
	MCGdiff	**1.423**	1.002	**1.497**	1.238	**1.511**	0.899	1.760	0.985
	SG-EKDP (Ours)	1.915	0.556	1.725	**0.348**	1.763	**0.423**	**1.678**	**0.381**
80	DPG	6.786	69.66	7.005	67.32	6.905	66.11	7.289	67.13
	SCG	6.022	1708.	6.059	15916.	6.033	37121.	6.013	91466.
	EnKG	4.997	$\geq$ 100k	4.938	$\geq$ 100k	5.180	$\geq$ 100k	5.068	$\geq$ 100k
	EKS	**2.444**	**27.72**	**2.357**	**25.91**	**2.366**	**25.69**	**2.371**	**25.60**
	MCGdiff	33.03	177.3	32.90	177.2	32.87	177.2	32.93	177.5
	SG-EKDP (Ours)	4.367	64.13	4.492	65.99	4.284	67.65	4.579	61.06
400	DPG	6.111	$\geq$ 100k	6.149	$\geq$ 100k	6.259	$\geq$ 100k	6.181	$\geq$ 100k
	SCG	6.199	$\geq$ 100k	6.172	$\geq$ 100k	6.182	$\geq$ 100k	6.276	$\geq$ 100k
	EnKG	8.636	$\geq$ 100k	7.432	$\geq$ 100k	11.513	$\geq$ 100k	11.825	$\geq$ 100k
	EKS	**1.047**	**1817.**	**1.092**	1777.	**1.114**	2156.	**1.130**	2134.
	MCGdiff	33.09	$\geq$ 100k	33.08	$\geq$ 100k	32.98	$\geq$ 100k	33.06	$\geq$ 100k
	SG-EKDP (Ours)	4.414	2478.	4.527	**1747.**	4.074	**1794.**	4.340	**1805.**

Table 2: Qualitative evaluation on FFHQ 256x256 dataset. We report average metrics for image quality and samples consistency on three tasks. Measurement noise level $\sigma = 0.05$ is used if not otherwise stated. (†: PnP-DM results on phase retrieval are significantly worse than those originally reported. Note that we use a different forward model configuration, larger measurement noise, and full-color images, which differs from the original PnP-DM setup.)

	Inpaint (box)			**SR (×4)**			**Phase retrieval**		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
Gradient access									
DiffPIR	22.87	0.653	0.268	26.48	0.744	0.220	25.86	0.824	**0.199**
DPS	21.77	0.767	0.213	24.90	0.710	0.265	16.79	0.589	0.448
PnP-DM†	22.17	**0.832**	**0.136**	25.86	0.808	0.193	18.98	0.650	0.409
Black-box access									
Central-GSG	18.76	0.720	0.229	26.55	0.740	0.169	11.48	0.427	0.571
DPG	20.89	0.752	0.184	28.12	**0.831**	**0.126**	8.76	0.297	0.663
SCG	4.71	0.302	0.763	4.71	0.302	0.760	5.04	0.306	0.733
EnKG	21.70	0.727	0.286	27.17	0.773	0.237	24.02	0.796	0.232
SG-EKDP (Ours)	**23.70**	0.763	0.225	**29.01**	0.826	0.204	**25.99**	**0.839**	0.215

6 Conclusion

We propose a novel particle-based diffusion posterior sampling method for inverse problems. Our method is based off of the split Gibbs sampler, alternating between an EKS step for data fitting and a denoising step for sample quality. Notably, our method is derivative-free, allowing for applicability to various inverse problems that only permit black-box access to the forward model. Experimental results demonstrate that our method is comparable to or outperforms other existing diffusion-based solvers, even those that rely on gradient information.

References

Richard C Booton. Nonlinear control systems with random inputs. *IRE Transactions on Circuit Theory*, 1(1):9–18, 1954.

Raunak Borker, Daniel Huang, Sebastian Grimberg, Charbel Farhat, Philip Avery, and Jason Rabinovitch. Mesh adaptation framework for embedded boundary methods for computational fluid dynamics and fluid-structure interaction. *International Journal for Numerical Methods in Fluids*, 90(8):389–424, 2019.

Philippe Bougeault, Zoltan Toth, Craig Bishop, Barbara Brown, David Burridge, De Hui Chen, Beth Ebert, Manuel Fuentes, Thomas M Hamill, Ken Mylne, et al. The thorpex interactive grand global ensemble. *Bulletin of the American Meteorological Society*, 91(8):1059–1072, 2010.

Charles A Bouman and Gregery T Buzzard. Generative plug and play: Posterior sampling for inverse problems. In *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1–7. IEEE, 2023.

Gabriel Cardoso, Yazid Janati el idrissi, Sylvain Le Corff, and Eric Moulines. Monte carlo guided denoising diffusion models for bayesian linear inverse problems. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=nHESwXvxWK`.

José A Carrillo, Franca Hoffmann, Andrew M Stuart, and Urbain Vaes. Consensus-based sampling. *Studies in Applied Mathematics*, 148(3):1069–1140, 2022.

Neil K Chada, Andrew M Stuart, and Xin T Tong. Tikhonov regularization within ensemble kalman inversion. *SIAM Journal on Numerical Analysis*, 58(2):1263–1294, 2020.

Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In *The Eleventh International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=OnD9zGAGT0k`.

Florentin Coeurdoux, Nicolas Dobigeon, and Pierre Chainais. Plug-and-play split gibbs sampler: embedding deep generative priors in bayesian inference. *arXiv preprint arXiv:2304.11134*, 2023.

Simon L Cotter, Gareth O Roberts, Andrew M Stuart, and David White. Mcmc methods for functions: modifying old algorithms to make them faster. 2013.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential monte carlo samplers. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 68(3):411–436, 2006.

Zehao Dou and Yang Song. Diffusion posterior sampling for linear inverse problem solving: A filtering perspective. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=tplXNcHZs1`.

Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using monte carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5): 10143–10162, 1994.

Alfredo Garbuno-Inigo, Franca Hoffmann, Wuchen Li, and Andrew M Stuart. Interacting langevin diffusions: Gradient structure and ensemble kalman sampler. *SIAM Journal on Applied Dynamical Systems*, 19(1):412–441, 2020.

Andrew Gelman, Walter R Gilks, and Gareth O Roberts. Weak convergence and optimal scaling of random walk metropolis algorithms. *The annals of applied probability*, 7(1):110–120, 1997.

Charles J Geyer. Practical markov chain monte carlo. *Statistical science*, pp. 473–483, 1992.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Peter L Houtekamer and Fuqing Zhang. Review of the ensemble kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 144(12):4489–4532, 2016.

Daniel Z Huang, Philip Avery, Charbel Farhat, Jason Rabinovitch, Armen Derkevorkian, and Lee D Peterson. Modeling, simulation and validation of supersonic parachute inflation dynamics during mars landing. In *AIAA scitech 2020 forum*, pp. 0313, 2020.

Yujia Huang, Adishree Ghatare, Yuanzhe Liu, Ziniu Hu, Qinsheng Zhang, Chandramouli S Sastry, Siddharth Gururani, Sageev Oore, and Yisong Yue. Symbolic music generation with non-differentiable rule guided diffusion. *arXiv preprint arXiv:2402.14285*, 2024.

Marco Iglesias and Yuchen Yang. Adaptive regularisation for ensemble kalman inversion. *Inverse Problems*, 37(2):025008, 2021.

Marco A Iglesias. A regularizing iterative ensemble kalman method for pde-constrained inverse problems. *Inverse Problems*, 32(2):025002, 2016.

Marco A Iglesias, Kody JH Law, and Andrew M Stuart. Ensemble kalman methods for inverse problems. *Inverse Problems*, 29(4):045001, 2013.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022.

Won Jun Kim, Hyungjin Chung, Jaemin Kim, Sangmin Lee, Byeongsu Sim, and Jong Chul Ye. Derivative-free diffusion manifold-constrained gradient for unified xai. *arXiv preprint arXiv:2411.15265*, 2024.

Dmitrii Kochkov, Jamie A Smith, Ayya Alieva, Qing Wang, Michael P Brenner, and Stephan Hoyer. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118(21):e2101784118, 2021.

Nikola Kovachki and Andrew Stuart. Ensemble kalman inversion: A derivative-free technique for machine learning tasks. *arXiv preprint*, 2018. URL `https://arxiv.org/abs/1808.03620`.

Chunyuan Li, Changyou Chen, David Carlson, and Lawrence Carin. Preconditioned stochastic gradient langevin dynamics for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Yi-An Ma, Tianqi Chen, and Emily Fox. A complete recipe for stochastic gradient mcmc. *Advances in neural information processing systems*, 28, 2015.

Dean S Oliver and Yan Chen. Recent progress on reservoir history matching: a review. *Computational Geosciences*, 15:185–221, 2011.

Charles S Peskin. Numerical analysis of blood flow in the heart. *Journal of computational physics*, 25(3):220–252, 1977.

Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. 1996.

Tapio Schneider, Shiwei Lan, Andrew Stuart, and Joao Teixeira. Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters*, 44(24):12–396, 2017.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Yu Sun, Zihui Wu, Yifan Chen, Berthy T Feng, and Katherine L Bouman. Provable probabilistic imaging using score-based generative priors. *IEEE Transactions on Computational Imaging*, 2024.

Haoyue Tang, Tian Xie, Aosong Feng, Hanyu Wang, Chenyang Zhang, and Yang Bai. Solving general noisy inverse problem via posterior sampling: A policy gradient viewpoint. In *International Conference on Artificial Intelligence and Statistics*, pp. 2116–2124. PMLR, 2024.

Santosh Vempala and Andre Wibisono. Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices. *Advances in neural information processing systems*, 32, 2019.

Maxime Vono, Nicolas Dobigeon, and Pierre Chainais. Split-and-augmented gibbs sampler—application to large-scale inference problems. *IEEE Transactions on Signal Processing*, 67(6):1648–1661, 2019.

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

Zihui Wu, Yu Sun, Yifan Chen, Bingliang Zhang, Yisong Yue, and Katherine L Bouman. Principled probabilistic imaging using diffusion models as plug-and-play priors. *arXiv preprint arXiv:2405.18782*, 2024.

Xingyu Xu and Yuejie Chi. Provably robust score-based diffusion posterior sampling for plug-and-play image reconstruction. *arXiv preprint arXiv:2403.17042*, 2024.

Zhewei Yao, Amir Gholami, Sheng Shen, Mustafa Mustafa, Kurt Keutzer, and Michael Mahoney. Adahessian: An adaptive second order optimizer for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 10665–10673, 2021.

Bo Yuan, Jiaojiao Fan, Jiaming Liang, Andre Wibisono, and Yongxin Chen. On a class of gibbs sampling over networks. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 5754–5780. PMLR, 2023.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Hongkai Zheng, Wenda Chu, Austin Wang, Nikola Kovachki, Ricardo Baptista, and Yisong Yue. Ensemble kalman diffusion guidance: A derivative-free method for inverse problems. *arXiv preprint arXiv:2409.20175*, 2024.

7 APPENDIX / SUPPLEMENTAL MATERIAL

This section includes the theoretical analysis of our algorithm for general linear inverse problems. We assume all the target distributions $\pi^{XZ}, \pi^{X|Z}, \pi^{Z|X}$ satisfy the standard regularity conditions defined in Theorem 2.1 of Roberts & Tweedie (1996) throughout the analysis.

7.1 PROOFS

Lemma 1 (Stationary distribution of the likelihood step)**.** *Assume the particle distribution is not a Dirac measure, the dynamics of Eq.* (5) *admits* $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z}) \propto \exp(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\rho^2}\|\boldsymbol{z}-\boldsymbol{x}^{(j)}\|_2^2)$ *as a stationary distribution. Further, if the covariance matrix is positive definite, the stationary distribution is unique.*

Proof. This result has been proved in various forms in the literature (Ma et al., 2015; Garbuno-Inigo et al., 2020), we provide a simple proof of our use case for ease of understanding. Suppose $\nu(t,\boldsymbol{z})$ is the probability density of $\boldsymbol{z}$ at time t. For the ease of notation, we ignore the particle index j in $\boldsymbol{z}_t$. Let $\Phi(\boldsymbol{z}) = f(\boldsymbol{z};\boldsymbol{y}) + \frac{1}{2\rho^2}\|\boldsymbol{z}-\boldsymbol{x}^{(j)}\|_2^2$. The corresponding Fokker-Planck equation for Eq. (5) reads

$$\frac{\partial \nu}{\partial t} = \nabla\cdot(\nu C_t \nabla\Phi(\boldsymbol{z})) + \nabla\cdot(C_t\nabla\nu)\,,$$

which can be rewritten as

$$\frac{\partial \nu}{\partial t} = \nabla\cdot(\nu C_t(\nabla\Phi(\boldsymbol{z}) + \nabla\log\nu))\,. \tag{11}$$

Let ν_∞ denote the stationary distribution of Eq. (11). We have

$$0 = \nabla\cdot(\nu C_t(\nabla\Phi(\boldsymbol{z}) + \nabla\log\nu_\infty))\,.$$

If the particle distribution is not Dirac, $C_t \neq 0$ due to Lemma 2.1 in Garbuno-Inigo et al. (2020). Therefore,

$$\nabla\Phi(\boldsymbol{z}) + \nabla\log\nu_\infty = c,$$

where c is a constant. Integrating both sides gives

$$\nu_\infty \propto \exp(-\Phi(\boldsymbol{z})) = \exp(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\rho^2}\|\boldsymbol{z}-\boldsymbol{x}^{(j)}\|_2^2),$$

showing that $\pi^{Z|X=\boldsymbol{x}^{(j)}}(\boldsymbol{z})$ is a stationary distribution of the dynamics of Eq. (5). Further, if C_t is positive definite, it ensures the irreducibility and strong Feller, and the stationary distribution is unique (Roberts & Tweedie, 1996). □

Proposition 1. *For the k-th alternating iteration and j-th particle, the likelihood step of SG-EKDP is equivalent to running the following SDE from $t=0$ to $t=1$:*

$$\mathrm{d}\boldsymbol{x}_t = \rho^2\nabla\log\phi_t(\boldsymbol{x}_t)\mathrm{d}t + \rho\mathrm{d}\boldsymbol{w}_t, \tag{12}$$

where $\boldsymbol{x}_0 = \boldsymbol{x}^{(j)}$ and $\phi_t(\boldsymbol{x}) = \int \exp(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\rho^2(1-t)}\|\boldsymbol{x}-\boldsymbol{z}\|_2^2)\mathrm{d}\boldsymbol{z}$.

Proof. By Lemma 1, the stationary distribution of the likelihood step of SG-EKDP is $\pi^{Z|X=\boldsymbol{x}^{(j)}} \propto \exp(-f(\boldsymbol{z};\boldsymbol{y}) - \frac{1}{2\rho^2}\|\boldsymbol{z}-\boldsymbol{x}^{(j)}\|_2^2)$. On the other hand, due to Proposition A.2 of Wu et al. (2024) and Lemma 3.4 of Yuan et al. (2023), the distribution $p(\boldsymbol{x}_1|\boldsymbol{x}_0)$ of the dynamics of Eq. (12) is proportional to $\exp(-f(\boldsymbol{x}_1;\boldsymbol{y}) - \frac{1}{2\rho^2}\|\boldsymbol{x}_1-\boldsymbol{x}_0\|_2^2)$. Comparing the two distributions, we conclude that running the SDE (12) from 0 to 1 with the same initialization is equivalent to performing a likelihood step. □

Theorem 2 (Full version of Theorem 1)**.** *Suppose $\mathcal{G}$ is linear. If the particle distribution is not a Dirac measure. Then π^{XZ} is a stationary distribution of Algorithm 1. Furthermore, if the particle covariance C_t is positive definite, π^{XZ} is the unique stationary distribution. Suppose ν_τ and π_τ the distributions at time τ of the non-stationary and stationary process initialized at ν_0 with*

$D_{\mathrm{KL}}(\pi^X||\nu_0^X) < +\infty$ and $\pi_0 = \pi^X$, respectively. For K iterations of Algorithm 1 or equivalently $T := K(t^* + 1)$, we have the following convergence results:

$$\frac{1}{T}\int_0^T D_{\mathrm{FI}}(\pi_\tau||\nu_\tau)\mathrm{d}\tau \leq \frac{4D_{\mathrm{KL}}(\pi^X||\nu_0^X)}{T\min(\rho,\delta)^2} + \frac{4\epsilon_{\mathrm{score}}}{(t^*+1)\delta^2}, \tag{13}$$

where D_{FI} is the Fisher divergence, D_{KL} is the Kullback–Leibler (KL) divergence, t^* is the starting time of the denoising process defined in Eq. (2) such that $\sigma(t^*) = \rho$, δ is the smallest diffusion coefficient used for the prior step, and $\epsilon_{\mathrm{score}}$ is the score approximation error of the pre-trained diffusion model defined as

$$\epsilon_{\mathrm{score}} := \int_0^{t^*} 2\dot{\sigma}(\tau)\sigma(\tau)\mathbb{E}_{\boldsymbol{x}\sim p_\tau}\|s_\theta(\boldsymbol{x},\tau) - \nabla \log p_\tau(\boldsymbol{x})\|_2^2 \mathrm{d}\tau.$$

Proof. For the first part, we prove that π^{XZ} defined in Eq.(4) is a stationary distribution of SG-EKDP. We prove this by directly verifying the invariance property, i.e., if the samples $(\boldsymbol{x}, \boldsymbol{z})$ are from the joint distribution π^{XZ}, then after one iteration of the algorithm, the new samples $(\boldsymbol{x}', \boldsymbol{z}')$ stay in the same distribution π^{XZ}. To begin with, we note that when $\mathcal{G}$ is linear, the dynamics of Eq. (7) is equivalent to that of Eq. (5). By Lemma 1, $\pi^{Z|X=\boldsymbol{x}}$ is a stationary distribution of the likelihood step defined in Eq. (7). Therefore, after the likelihood step, the joint density of $(\boldsymbol{x}, \boldsymbol{z}')$ becomes

$$p(\boldsymbol{x}, \boldsymbol{z}') = \int \pi^{XZ}(\boldsymbol{x}, \boldsymbol{z})\pi^{Z|X=\boldsymbol{x}}(\boldsymbol{z}')\mathrm{d}\boldsymbol{z} = \pi^X(\boldsymbol{x})\pi^{Z|X=\boldsymbol{x}}(\boldsymbol{z}'),$$

where π^X is the marginal distribution. As shown in Eq. (8), after sampling $\boldsymbol{x}'$ given $\boldsymbol{z}'$ according to the prior step, the joint density of $(\boldsymbol{x}', \boldsymbol{z}')$ becomes

$$\begin{aligned}
p(\boldsymbol{x}', \boldsymbol{z}') &= \int p(\boldsymbol{x}, \boldsymbol{z}')\pi^{X|Z=\boldsymbol{z}'}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x} \\
&= \int \pi^X(\boldsymbol{x})\pi^{Z|X=\boldsymbol{x}}(\boldsymbol{z}')\pi^{X|Z=\boldsymbol{z}'}(\boldsymbol{x}')\mathrm{d}\boldsymbol{x} \\
&= \pi^Z(\boldsymbol{z}')\pi^{X|Z=\boldsymbol{z}'}(\boldsymbol{x}') \\
&= \pi^{XZ}(\boldsymbol{x}', \boldsymbol{z}'),
\end{aligned}$$

showing that the distribution of $(\boldsymbol{x}', \boldsymbol{z}')$ remains π^{XZ} after one round of updates. Therefore, π^{XZ} is a stationary distribution. Furthermore, by Lemma 1, if the particle covariance remains positive definite, the stationary distribution of the likelihood step is unique. Consequently, it follows that π^{XZ} is the unique stationary distribution of SG-EKDP.

For the second part, we prove the convergence results in Eq.(13) following the same arguments in the Theorem 3.1 of Wu et al. (2024). Firstly, due to Proposition 1, we can equivalently define the dynamics of the likelihood step as that of Eq. (12). Then the total time of K alternating iterations is $T := K(1 + t^*)$ where each likelihood step takes one unit of time and each prior step takes t^* units. Note that the stationary process π_τ alternates between π^X and π^Z as follows:

$$\pi_\tau = \begin{cases} \pi^X & \tau = k(1+t^*), k \in \{0, \ldots, K\} \\ \pi^Z & \tau = k(1+t^*) + 1, k \in \{0, \ldots, K-1\} \end{cases}$$

Applying Lemma 2 in Yuan et al. (2023) to the likelihood dynamics defined in Eq. (12) for the non-stationary and stationary processes, we have

$$\partial_\tau D_{\mathrm{KL}}(\pi_\tau\|\nu_\tau) = -\frac{\rho^2}{2}D_{\mathrm{FI}}(\pi_\tau||\nu_\tau) \leq -\frac{\rho^2}{4}D_{\mathrm{FI}}(\pi_\tau||\nu_\tau),$$

where $\tau \in [k(1+t^*), k(1+t^*)+1], k = 0, \ldots, K-1$. Integrating over $[k(1+t^*), k(1+t^*)+1]$ gives

$$\int_{k(1+t^*)}^{k(1+t^*)+1} D_{\mathrm{FI}}(\pi_\tau||\nu_\tau)\mathrm{d}\tau \leq \frac{4}{\rho^2}\left(D_{\mathrm{KL}}(\pi^X||\nu_{k(1+t^*)}) - D_{\mathrm{KL}}(\pi^Z||\nu_{k(1+t^*)+1})\right), \tag{14}$$

where $k = 0, \ldots, K-1$. Further, we apply Lemma A.4 in Wu et al. (2024) to the dynamics of the prior step defined in Eq. (2) with

$$\begin{aligned} b(\boldsymbol{x}_t, t) &:= -2\dot{\sigma}(t)\sigma(t)\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t; \sigma(t)) \\ (\boldsymbol{x}_t, t) &:= -2\dot{\sigma}(t)\sigma(t) s_\theta(\boldsymbol{x}_t, t) \\ c(t) &:= \sqrt{2\dot{\sigma}(t)\sigma(t)} \\ \delta &:= \inf_{t\in[0,t^*]} c(t), \end{aligned}$$

and have that

$$\begin{aligned} \partial_t D_{\mathrm{KL}}(\pi_\tau \| \nu_\tau) &\leq -\frac{2\dot{\sigma}(t)\sigma(t)}{4} D_{\mathrm{FI}}(\pi_\tau || \nu_\tau) + 2\dot{\sigma}(t)\sigma(t)\mathbb{E}_{\pi_\tau} \|\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t; \sigma(t)) - s_\theta(\boldsymbol{x}_t, t)\|_2^2 \\ &\leq -\frac{\delta^2}{4} D_{\mathrm{FI}}(\pi_\tau || \nu_\tau) + 2\dot{\sigma}(t)\sigma(t)\mathbb{E}_{\pi_\tau} \|\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t; \sigma(t)) - s_\theta(\boldsymbol{x}_t, t)\|_2^2, \end{aligned}$$

where $\tau \in [k(1+t^*)+1, (k+1)(1+t^*)], k = 0, \ldots, K-1$. Integrating over $[k(1+t^*)+1, (k+1)(1+t^*)]$ gives

$$\int_{k(1+t^*)+1}^{(k+1)(1+t^*)} D_{\mathrm{FI}}(\pi_\tau || \nu_\tau) \mathrm{d}\tau \leq \frac{4}{\delta^2} \left(D_{\mathrm{KL}}(\pi^Z || \nu_{k(1+t^*)+1}) - D_{\mathrm{KL}}(\pi^X || \nu_{(k+1)(1+t^*)} \right) + \frac{4\epsilon_{\mathrm{score}}}{\delta^2}. \tag{15}$$

Adding the both sides of Eq. (14) and Eq. (15) and dividing the both sides by $T = k(1+t^*)$ gives

$$\begin{aligned} \frac{1}{T}\int_0^T D_{\mathrm{FI}}(\pi_\tau || \nu_\tau) &\leq \frac{4}{T\min(\rho^2, \delta^2)} \left(D_{\mathrm{KL}}(\pi^X || \nu_0) - D_{\mathrm{KL}}(\pi^X || \nu_T) \right) + \frac{4K}{\delta^2 T}\epsilon_{\mathrm{score}} \\ &\leq \frac{4}{T\min(\rho^2, \delta^2)} D_{\mathrm{KL}}(\pi^X || \nu_0) + \frac{4}{\delta^2(1+t^*)}\epsilon_{\mathrm{score}}, \end{aligned}$$

concluding the proof.

□

7.2 Implementation Details

7.2.1 Covariance Matrix

The preconditioned Langevin dynamic requires computation of the matrix square root of the empirical covariance $\mathbf{C}_t$. For experiments on lower dimension data (linear Gaussian, linear Gaussian mixture), we use a built-in PyTorch solver. However, for higher dimension datasets such as FFHQ256, direct computation is too expensive, necessitating approximation of the matrix. Following similar practice as in Li et al. (2016); Yao et al. (2021), we approximate $\mathbf{C}_t$ with the diagonal matrix $\mathrm{Diag}(\mathbf{C}_t)$.

7.2.2 Langevin Step Size

The likelihood step involves discretizing a SDE. The step size determines the discretization scheme. We simply choose the adaptive step size below following the practice in Kovachki & Stuart (2018)

$$\eta_t = \frac{\lambda}{\left\| -C_t \nabla \left(f(\boldsymbol{z}_t^{(j)}; \boldsymbol{y}) + \frac{1}{2\rho^2}\|\boldsymbol{z}_t^{(j)} - \boldsymbol{x}^{(j)}\|_2^2 \right) \right\|_2^2}, \tag{16}$$

where λ is a tunable hyperparameter. Empirically, our adaptive step size ensures numerical stability during sampling.

7.2.3 Hyperparameters

Following Wu et al. (2024), we define our ρ annealing schedule as exponential decay:

$$\rho_k = \max(\alpha^k \rho_{\max}, \rho_{\min}) \tag{17}$$

Table 3: Hyperparameter choices for FFHQ256

Hyperparameter	**Inpaint (box)**	**SR**	**Phase retrieval**
N (total steps)	30	30	30
J (ensemble size)	2048	2048	2048
ρ_{min}	0.2	0.2	0.2
ρ_{max}	10	10	10
α	0.9	0.9	0.9
λ	300	300	300
$N_{likelihood}$ (likelihood iterations)	500	500	500
N_{prior} (prior iterations)	25	25	32

7.3 Experimental Details

Linear-Gaussian As previously discussed, the Linear-Gaussian experiments use a linear forward model with Gaussian measurement noise $\boldsymbol{z} \sim \mathcal{N}(0, \sigma_y^2 \mathbf{I})$ and Gaussian prior $\boldsymbol{x}_0 \sim \mathcal{N}(\mathbf{m}_{\boldsymbol{x}}, \sigma_{\boldsymbol{x}^2}\mathbf{I})$. For all experiments, we randomly generate $\mathbf{m}_{\boldsymbol{x}}$ and choose $\sigma_{\boldsymbol{x}} = 5$. We also randomly generate the linear operator $\mathbf{H}$. For each $d_x \in \{2, 40, 80, 400\}$, we set $d_y = 1$.

Linear Gaussian mixture We consider the general linear inverse problem given by

$$\boldsymbol{y} = H\boldsymbol{x} + \epsilon, \tag{18}$$

where $\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{y} \in \mathbb{R}^m, H \in \mathbb{R}^{m\times n}, \epsilon \sim \mathcal{N}(0, \Sigma_\epsilon)$. Given the measurement $\boldsymbol{y}$, we aim to sample from the posterior distribution $p(\boldsymbol{x}|\boldsymbol{y})$. We consider and analyze the case where the prior distribution of $\boldsymbol{x}$ is a mixture of Gaussians given by

$$p(\boldsymbol{x}) = \sum_{i=1}^{K} \gamma_i \mathcal{N}(m_i, \Sigma_i), \sum_{i=1}^{K} \gamma_i = 1, \tag{19}$$

where the mean $m_i \in \mathbb{R}^n$ and the covariance matrix $\Sigma_i \in \mathbb{R}^{n\times n}$. By linearity, the distribution of $\boldsymbol{y}$ is also a Gaussian mixture given by

$$p(\boldsymbol{y}) = \sum_{i=1}^{K} \gamma_i \mathcal{N}(Hm_i, H\Sigma_i H^\top + \Sigma_\epsilon), \sum_{i=1}^{K} \gamma_i = 1. \tag{20}$$

Using Bayes theorem, the posterior distribution is given by

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x})}{p(\boldsymbol{y})}. \tag{21}$$

The likelihood $p(\boldsymbol{y}|\boldsymbol{x})$ reads

$$p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{y}; H\boldsymbol{x}, \Sigma_\epsilon). \tag{22}$$

Therefore,

$$p(\boldsymbol{x}|\boldsymbol{y}) = \frac{\sum_{i=1}^{K} \gamma_i \mathcal{N}(\boldsymbol{x}; m_i, \Sigma_i)\mathcal{N}(\boldsymbol{y}; H\boldsymbol{x}, \Sigma_\epsilon)}{\sum_{i=1}^{K} \gamma_i \mathcal{N}(\boldsymbol{y}; m_i, H\Sigma_i H^\top + \Sigma_\epsilon)}, \tag{23}$$

which can be written as the exponential of a quadratic in $\boldsymbol{x}$. Therefore, the posterior distribution is also a mixture of Gaussians,

$$p(\boldsymbol{x}|\boldsymbol{y}) = \sum_{i=1}^{K} \omega_i \mathcal{N}(\boldsymbol{x}; \hat{m}_i, C_i), \tag{24}$$

where the posterior mean $\hat{m}_i$ and covariance C_i are given by

$$\hat{m}_i = \left(H^\top \Sigma_\epsilon^{-1} H + \Sigma_i^{-1}\right)^{-1} \left(H^\top \Sigma_\epsilon^{-1} \boldsymbol{y} + \Sigma_i^{-1} m_i\right), \tag{25}$$

$$C_i = \left(H^\top \Sigma_\epsilon^{-1} H + \Sigma_i^{-1}\right)^{-1}, \tag{26}$$

and the weight of each mode is given by

$$\omega_j = \frac{\gamma_j \mathcal{N}(\boldsymbol{y}; Hm_j, H\Sigma_j H^\top + \Sigma_\epsilon)}{\sum_{i=1}^{K} \gamma_i \mathcal{N}(\boldsymbol{y}; Hm_i, H\Sigma_i H^\top + \Sigma_\epsilon)}, j = 1, \ldots, K. \tag{27}$$

In our experiments, we set the prior to be a mixture of four Gaussians where the variance of each Gaussian is $2\mathbf{I}$ and the means are $(16i, 16j)$ for $(i, j) \in \{0, 1\}^2$. We set $\Sigma_\epsilon = 1.5\mathbf{I}$. The linear forward model H and observed data $\boldsymbol{y}$ are both randomly generated from Gaussian.

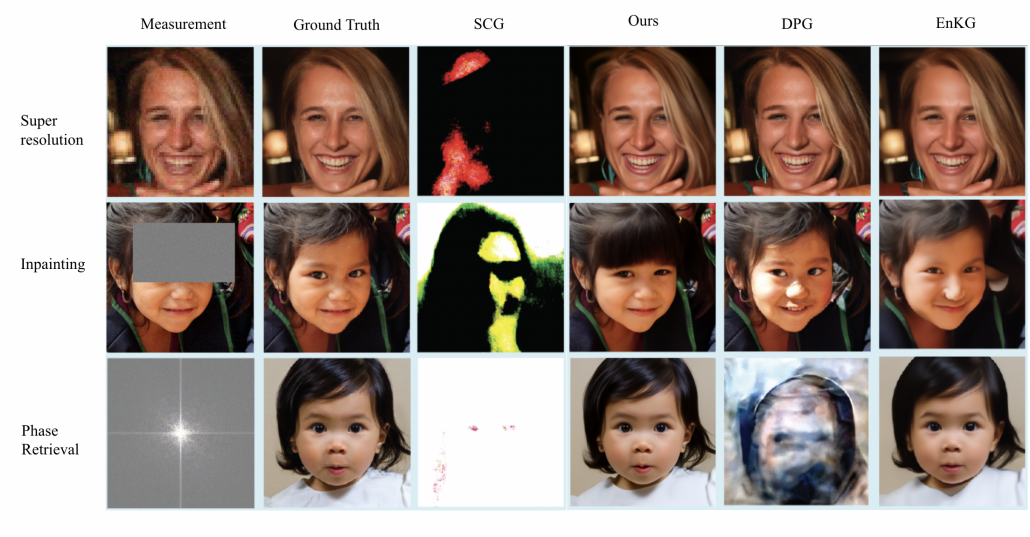

Figure 2: Generated samples on the FFHQ256 inverse problems.

Image Restoration We measure metrics on a small validation subset of FFHQ256, which consists of ten images. The forward model implementations for box inpainting and super-resolution are taken from Chung et al. (2023), and the implementation of phase retrieval is taken from Wu et al. (2024). While the setup remains the same as that of Chung et al. (2023) for inpainting and super-resolution, our phase retrieval setup differs slightly from Wu et al. (2024), as we choose $\sigma_y = 0.05$, use full-color images (as opposed to grayscale), and set the oversampling parameter to 2.0.