Quantifying Epistemic Uncertainty in Diffusion Models

Aditi Gupta
Berkeley Lab & ICSI

Raphael A. Meyer
UC Berkeley & ICSI

Yotam Yaniv
Berkeley Lab

Elynn Chen
New York University

N. Benjamin Erichson
Berkeley Lab & ICSI

Abstract

To ensure high quality outputs, it is important to quantify the epistemic uncertainty of diffusion models. Existing methods are often unreliable because they mix epistemic and aleatoric uncertainty. We introduce a method based on Fisher information that explicitly isolates epistemic variance, producing more reliable plausibility scores for generated data. To make this approach scalable, we propose FLARE (Fisher-Laplace Randomized Estimator), which approximates the Fisher information using a uniformly random subset of model parameters. Empirically, FLARE improves uncertainty estimation in synthetic time-series generation tasks, achieving more accurate and reliable filtering than other methods. Theoretically, we bound the convergence rate of our randomized approximation and provide analytic and empirical evidence that last-layer Laplace approximations are insufficient for this task.

1 INTRODUCTION

Diffusion models have become a dominant paradigm for generative modeling, with applications ranging from image synthesis to time-series forecasting (Esser et al., 2024; Liu et al., 2023; Lipman et al., 2023; Lu et al., 2022). The stochasticity of the reverse diffusion process accounts for aleatoric variability, suggesting that diffusion models could also serve as a foundation for uncertainty quantification (UQ) (Ho et al., 2020; Song et al., 2021; Kendall and Gal, 2017; Hüllermeier and Waegeman, 2021). However, while aleatoric randomness is intrinsic to sampling, capturing and interpreting *epistemic* uncertainty (arising from uncertainty in model parameters) remains a challenge (Depeweg et al., 2018; Hüllermeier and Waegeman, 2021). Exact Bayesian inference over the high-dimensional parameter spaces of diffusion networks is computationally infeasible, and common proxies such as sample variance conflate epistemic and aleatoric effects (Shu and Farimani, 2024; De Vita and Belagiannis, 2025).

Several approaches have been proposed to address uncertainty quantification in diffusion models. One line of work uses last-layer Laplace approximations (LLLA) (Daxberger et al., 2021; Kou et al., 2024; Jazbec et al., 2025), which are computationally efficient but restrict uncertainty to a small subset of model parameters. Other methods perturb model parameters directly (Berry et al., 2024; Chan et al., 2024; Shu and Farimani, 2024), capturing richer epistemic structure at the cost of multiple forward passes during inference. Ensemble and hypernetwork-based approaches approximate the weight posterior more explicitly (Krueger et al., 2017; Lakshminarayanan et al., 2017; Maddox et al., 2019; Gal and Ghahramani, 2016), and can in principle separate epistemic and aleatoric uncertainty. In practice, however, these methods are sensitive to architectural choices, hyperpriors, and the mechanism by which randomness is injected, often trading predictive fidelity for diversity if not carefully regularized.

A key source of confusion in this literature is the distinction between *posterior predictive* uncertainty and *epistemic* uncertainty. For example, BayesDiff (Kou et al., 2024; Jazbec et al., 2025) estimates posterior predictive uncertainty in diffusion models using Tweedie-style recursions that propagate data-space variance through the reverse process. This approach aggregates multiple sources of variability, in-

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300. Copyright 2026 by the author(s).

cluding diffusion noise and model output uncertainty, and is well suited for measuring overall sample variability. However, predictive variance alone does not indicate whether a model is uncertain due to lack of knowledge or merely due to stochastic sampling.

In this work, we focus specifically on *epistemic* uncertainty in diffusion models. Rather than asking how variable the generated samples are, we ask when the model itself is uncertain due to its parameters. Diffusion models already introduce randomness through the reverse process, which accounts for aleatoric variability and is present even when the model is well trained. As a result, measures based on sample variance reflect a mixture of stochastic sampling noise and actual model uncertainty. Interpreting this mixture as a single uncertainty signal makes it difficult to tell whether variability reflects a lack of knowledge or merely randomness in generation. To address this, we isolate uncertainty arising from the model parameters and track how it propagates through the reverse diffusion process. We project parameter uncertainty through the Jacobian of the denoiser along a realized reverse trajectory, while excluding diffusion noise from the propagated quantity. This yields a trajectory-level view of epistemic uncertainty and clarifies when and where the model is uncertain.

Specifically, we develop a Fisher–Laplace formulation

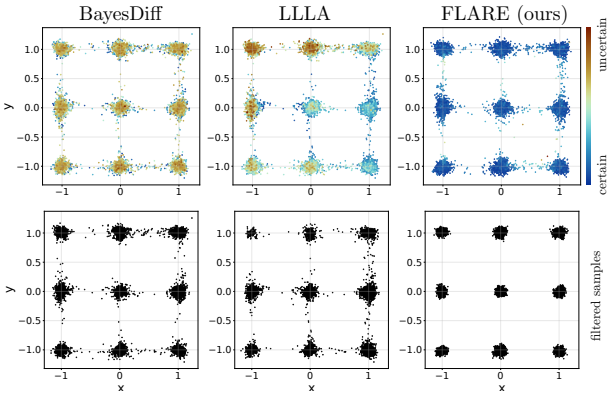


Figure 1: Mode interpolation in a 2D Gaussian mixture adapted from Aithal et al. (2024); Jazbec et al. (2025). The dataset consists of nine Gaussian modes arranged on a square grid. **Top:** uncertainty scores assigned to generated samples by BayesDiff (left), last-layer Laplace (LLLA; middle), and our method (right). **Bottom:** the same samples after filtering using a fixed uncertainty threshold. BayesDiff assigns low uncertainty to samples between modes, while LLLA further suppresses uncertainty due to its restriction to the final layer. In contrast, our method assigns high epistemic uncertainty in low-density regions between modes, enabling reliable removal of low-confidence samples.

that propagates parameter uncertainty through the reverse diffusion process by using the Jacobian of the denoiser at each step. While computing the full Fisher information is prohibitively expensive in large models, the common simplification of restricting focus to the last layer discards sensitivity information from earlier representations and leads to systematically miscalibrated epistemic uncertainty. We therefore introduce FLARE (Fisher–Laplace Randomized Estimator), a scalable algorithm that approximates Fisher–Laplace uncertainty by sampling parameters uniformly at random across all layers of the network. This *random subnetwork approximation* preserves network-wide sensitivity structure while reducing computational cost. We show that the resulting estimator is theoretically justified, converges rapidly as the number of sampled parameters increases, and closely tracks the behavior of the full Fisher–Laplace projection. Figure 1 illustrates the performance of FLARE on a toy example.

We evaluate FLARE on synthetic time-series generation tasks designed to probe multimodality, extrapolation, and low-density regions. We compare against BayesDiff (Kou et al., 2024), parameter-perturbation methods such as HyperDM (Chan et al., 2024), and a version of FLARE that uses the last-layer approximation instead of a random subnetwork. Across all tasks, our method recovers structured epistemic uncertainty in regions where models switch modes, traverse sparse areas, or move off distribution. In contrast, BayesDiff conflates aleatoric and epistemic effects, LLLA suppresses uncertainty arising from earlier layers, and parameter-perturbation methods incur additional computational overhead while mixing parameter variability with diffusion noise. Overall, FLARE provides a simple and effective alternative, yielding faithful, sample-level uncertainty diagnostics with a tunable trade-off between accuracy and runtime.

Contributions. Our main contributions are:

- **Fisher–Laplace projection.** We derive a closed-form projection of parameter uncertainty into data space via the Jacobian of the denoiser, yielding an interpretable epistemic uncertainty map that separates parameter uncertainty from diffusion noise.

- **FLARE.** We introduce a scalable randomized approximation that subsamples parameters uniformly across the network and prove that it preserves epistemic structure with rapidly decaying relative error.

- **Experiments.** We demonstrate improved uncertainty-aware sample filtering on synthetic time-series tasks, achieving up to 100% gap closure and consistently outperforming BayesDiff and last-layer Laplace baselines.

1.1 Preliminaries on Diffusion Models

We consider a discrete-time score-based diffusion model (Nichol and Dhariwal, 2021; Ho et al., 2020) with time steps $t \in \{1, \ldots, T\}$ and data space $\mathbb{R}^d$. Let $\boldsymbol{x}_0 \sim p_{\text{data}}$ and define the forward diffusion process

$$q(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}) \;=\; \mathcal{N}\big(\sqrt{\alpha_t}\,\boldsymbol{x}_{t-1},\; (1-\alpha_t)\mathbf{I}\big), \tag{1}$$

with $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, so that

$$q(\boldsymbol{x}_t \mid \boldsymbol{x}_0) = \mathcal{N}\big(\sqrt{\bar{\alpha}_t}\,\boldsymbol{x}_0,\; (1-\bar{\alpha}_t)\mathbf{I}\big). \tag{2}$$

The reverse dynamics are parameterized by a neural network $\varepsilon_\theta(\boldsymbol{x}_t, t)$ with parameters $\theta \in \mathbb{R}^p$ and follow the standard DDPM update (Ho et al., 2020)

$$\begin{aligned} \boldsymbol{x}_{t-1} &= a_t \boldsymbol{x}_t - b_t\, \varepsilon_\theta(\boldsymbol{x}_t, t) + \boldsymbol{\eta}_t, \\ \boldsymbol{\eta}_t &\sim \mathcal{N}(\mathbf{0}, \tilde{\beta}_t \mathbf{I}), \end{aligned} \tag{3}$$

where $a_t = \alpha_t^{-1/2}$, $b_t = \beta_t / \big(\sqrt{\alpha_t}\sqrt{1-\bar{\alpha}_t}\big)$, and $\tilde{\beta}_t = \frac{1-\alpha_{t-1}}{1-\bar{\alpha}_t}\beta_t$ (with $\alpha_0 = 1$). Throughout, we adopt the ε-prediction parameterization. The denoiser ε_θ is trained via score matching on pairs $(\boldsymbol{x}_0, t)$ sampled from the data distribution and a time prior, by minimizing a weighted mean-squared error (MSE) between the predicted and true noise (Hang et al., 2023). During sampling, the diffusion schedule is fixed and randomness is introduced through $\boldsymbol{\eta}_t$, representing *sampling noise* independent of the model parameters θ.

1.2 Formalizing the UQ Problem

We define predictive uncertainty for a single reverse step as the conditional covariance $\mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t)$. By the law of total covariance,

$$\begin{aligned} \mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t) = &\underbrace{\mathbb{E}_\theta\big[\mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \theta)\big]}_{\text{aleatoric}} \\ &+ \underbrace{\mathrm{Cov}_\theta\big(\mathbb{E}[\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \theta]\big)}_{\text{epistemic}}, \end{aligned} \tag{4}$$

where expectations and covariances over θ are taken with respect to a parameter distribution, such as the posterior $p(\theta \mid \mathcal{D})$ given training data $\mathcal{D}$. Here, θ is treated as a random variable encoding uncertainty over which model generated the data. The second term measures how predictions vary across plausible models (epistemic uncertainty), while the first term captures the intrinsic randomness of the reverse step given a fixed model (aleatoric uncertainty). Together, they separate uncertainty due to limited knowledge from irreducible stochasticity in the generative process.

For the DDPM update, the conditional moments are

$$\mathbb{E}[\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \theta] = a_t \boldsymbol{x}_t - b_t\, \varepsilon_\theta(\boldsymbol{x}_t, t), \tag{5}$$

$$\mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \theta) = \tilde{\beta}_t \mathbf{I}. \tag{6}$$

The aleatoric component in Equation (4) therefore corresponds directly to the diffusion noise $\tilde{\beta}_t \mathbf{I}$, while the epistemic component arises solely from uncertainty in the denoiser parameters θ. Unlike aleatoric uncertainty, epistemic uncertainty is *reducible*: it decreases as more data are observed or as the model class better matches the true generative process. Equation (4) thus characterizes the one-step predictive variability of the reverse diffusion. However, in practice, uncertainty is often assessed at the level of entire trajectories or in the variability of the final sample $\boldsymbol{x}_0$. Such quantities can be obtained by propagating and accumulating the one-step uncertainties along the reverse chain, yielding a pathwise notion of uncertainty that remains consistent with the diffusion dynamics.

Throughout, we assume a fixed diffusion schedule, Gaussian forward noise, and a differentiable denoiser ε_θ. Unless otherwise stated, conditioning on $(\boldsymbol{x}_t, t)$ removes randomness from the data distribution and all variables at time steps greater than t, isolating the stochasticity of the reverse transition from uncertainty in θ. Because the denoiser is applied repeatedly, small modeling errors can compound along the reverse trajectory. Thus, epistemic uncertainty is most pronounced in low-density regions, between modes, or under distribution shift, where the model must extrapolate beyond its training data. Aleatoric variability, by contrast, reflects only the randomness of sampling.

2 METHOD

In this section, we introduce a Fisher information-guided method that isolates epistemic (parameter) uncertainty by projecting a Laplace posterior over model parameters into data space at each diffusion reverse step. We then propagate and accumulate these one-step contributions along the denoising trajectory via a simple linear recursion induced by the DDPM update. This recursion tracks parameter sensitivity along the reverse path, yielding a final epistemic covariance at $\boldsymbol{x}_0$ and enabling sample-level diagnostics such as trace-based scores and variance maps. Finally, to make the approach scalable, we introduce a randomized subnetwork estimator that subsamples columns of the generalized Gauss-Newton (GGN) matrix (i.e., the Fisher matrix under a squared-loss objective). This estimator preserves the global structure of the network at a fraction of the computational cost, while avoiding the inaccuracies associated with methods such as LLLA.

Aleatoric randomness and conditioning. We let η denote the aggregate non-parametric randomness in the system, including (i) training-side randomness (e.g., data sampling and stochastic optimization noise), and (ii) sampling-side randomness (e.g., for-

ward diffusion noise used to define $\boldsymbol{x}_t$ from $\boldsymbol{x}_0$, latent trajectory randomness, and, in DDPMs, explicit reverse-step noise). For a fixed realization of η, the reverse trajectory $\{\boldsymbol{x}_t(\theta, \eta)\}_{t=0}^{T}$ is a deterministic function of the model parameters θ.

2.1 Fisher Information-guided Projection

Let $\hat{\theta}$ denote the maximum a posteriori (MAP) estimate of the model parameters given training data $\mathcal{D}$. We approximate the posterior distribution locally using a second-order (Laplace) expansion.

The posterior covariance over parameters is

$$\boldsymbol{\Sigma}_\theta \equiv \mathrm{Cov}[\boldsymbol{\theta} \mid \mathcal{D}] = \mathbb{E}\big[(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \mid \mathcal{D}\big]. \tag{7}$$

Under a local Laplace approximation around $\hat{\theta}$, this covariance takes the form $\boldsymbol{\Sigma}_\theta \approx (\boldsymbol{H} + \lambda \mathbf{I})^{-1}$, where $\boldsymbol{H}$ is the GGN matrix of the training loss and λ is a small damping parameter. The Jacobian of the denoiser with respect to the model parameters is

$$\boldsymbol{J}_t \coloneqq \nabla_\theta \varepsilon_\theta(\boldsymbol{x}_t, t)\big|_{\hat{\theta}} \in \mathbb{R}^{d \times p}. \tag{8}$$

All Jacobians $\boldsymbol{J}_t$ are evaluated along the MAP sampling trajectory $\{\boldsymbol{x}_t(\hat{\theta}, \eta)\}_{t=0}^{T}$ corresponding to a fixed realization of η. Throughout, we adopt the ε-prediction parameterization; alternative parameterizations modify only the scalar coefficients (a_t, b_t).

We define the epistemic covariance of the reverse state $\boldsymbol{x}_t$ as $\boldsymbol{\Sigma}_t^{\mathrm{ep}}(\eta) \coloneqq \mathrm{Cov}_\theta\big(\boldsymbol{x}_t(\theta, \eta) \mid \eta\big)$. Intuitively, this quantity captures how uncertainty in the model parameters propagates into uncertainty in the reverse diffusion state, holding all non-parametric randomness fixed. Equation (9) describes how to compute its contribution over a single reverse step.

One-step Fisher–Laplace Projection

$$\boldsymbol{\Sigma}_{t-1|t}^{\mathrm{ep}}(\eta) = b_t^2\, \boldsymbol{J}_t\, \boldsymbol{\Sigma}_\theta\, \boldsymbol{J}_t^\top. \tag{9}$$

The validity of the one-step projection in Equation (9) is discussed in Section 3.1.

2.2 Propagation through the Denoising Steps

We now describe the central mechanism of our method: a recursion that propagates epistemic uncertainty along the reverse diffusion trajectory. Conditioning on a fixed realization of the non-parametric randomness η, we propagate epistemic covariance backward in time using a linear update induced by the DDPM dynamics. Equation (10) accumulates the effect of parameter uncertainty across successive denoising steps.

Multi-step Epistemic Recursion

$$\boldsymbol{\Sigma}_{t-1}^{\mathrm{ep}}(\eta) = a_t^2\, \boldsymbol{\Sigma}_t^{\mathrm{ep}}(\eta) + b_t^2\, \boldsymbol{J}_t\, \boldsymbol{\Sigma}_\theta\, \boldsymbol{J}_t^\top. \tag{10}$$

The recursion in Equation (10) is justified and analyzed in Section 3.1. Unrolling it yields the closed-form expression in Equation (11).

Unrolled Epistemic Accumulation

$$\boldsymbol{\Sigma}_0^{\mathrm{ep}}(\eta) = \sum_{s=1}^{T} \left(\prod_{j<s} a_j \right)^2 b_s^2\, \boldsymbol{J}_s\, \boldsymbol{\Sigma}_\theta\, \boldsymbol{J}_s^\top. \tag{11}$$

While Equation (11) provides a clear characterization of epistemic uncertainty, directly evaluating this expression requires projecting a full-parameter Laplace posterior through the network at every reverse step.

Computational considerations. The full Fisher–Laplace projection $b_t^2 \boldsymbol{J}_t \boldsymbol{\Sigma}_\theta \boldsymbol{J}_t^\top$ provides the most faithful estimate of epistemic uncertainty, as it accounts for parameter sensitivity throughout the entire network. However, forming, storing, or inverting the full posterior covariance $\boldsymbol{\Sigma}_\theta \in \mathbb{R}^{p \times p}$ is computationally prohibitive in large-scale models, where the number of parameters p is large. This motivates practical approximations that trade accuracy for scalability.

A common simplification (Daxberger et al., 2021) restricts the Laplace posterior to the *final affine layer* of the network. For example, Kou et al. (2024) adopt this approach to quantify uncertainty in diffusion models. Algebraically, this replaces $\boldsymbol{\Sigma}_\theta$ with a covariance $\boldsymbol{\Sigma}_{\theta,\mathrm{last}}$ defined only over the final-layer parameters, and projects uncertainty using the corresponding subset of columns of $\boldsymbol{J}_t$. While computationally efficient, this approximation discards parameter sensitivity originating in earlier layers and propagated through the network. As we show in Section 4, it can underestimate epistemic structure along the reverse trajectory, particularly when uncertainty arises from deep feature representations rather than the final linear head. These limitations motivate an alternative that remains computationally tractable while retaining sensitivity contributions from across the network.

2.3 Randomized Subnetwork Approximation

We propose a scalable alternative that preserves network-wide parameter sensitivity while substantially reducing computational cost. Our approach approximates the Fisher–Laplace projection by operating on a randomly selected subnetwork of parameters at each reverse step. Rather than restricting uncertainty to a fixed subset of parameters, the method randomly sub-

samples columns of both the Jacobian and the GGN matrix. This preserves sensitivity contributions from across the network while avoiding the cost of full Fisher inversion. The procedure is given by:

1. Sample a uniform index set $I \subset \{1, \ldots, p\}$ with cardinality $|I| = m \ll p$.
2. Restrict the GGN matrix to the selected coordinates, forming $\boldsymbol{H}_{I,I} \in \mathbb{R}^{m \times m}$, and define the subnetwork posterior covariance
$$\boldsymbol{\Sigma}_{\text{sub}} = (\boldsymbol{H}_{I,I} + \lambda \mathbf{I})^{-1}.$$
3. Replace the full Jacobian by its selected columns $\boldsymbol{J}_{t,I} \in \mathbb{R}^{d \times m}$ and use this subnetwork within the recursion of Equation (10):
$$\boldsymbol{\Sigma}_{t-1}^{\text{ep}} \;=\; a_t^2\, \boldsymbol{\Sigma}_t^{\text{ep}} \;+\; b_t^2\, \boldsymbol{J}_{t,I}\, \boldsymbol{\Sigma}_{\text{sub}}\, \boldsymbol{J}_{t,I}^\top.$$

This randomized approximation preserves the structure of the original Fisher–Laplace projection while reducing both memory and computational requirements. Unlike LLLA, it captures epistemic contributions originating in intermediate representations and propagated through the network. Algorithm 1 summarizes the resulting Fisher–Laplace Randomized Estimator (FLARE), which integrates the randomized algorithm with our recursion formula to produce both a sample $\hat{\boldsymbol{x}}_0$ and its associated epistemic covariance.

Theoretical guarantees. In Section 3.2, we analyze the randomized subnetwork estimator using tools from randomized numerical linear algebra (Murray et al., 2023). Under mild regularity assumptions, we show that the approximation error of the randomized projection decays as $\mathcal{O}(1/\sqrt{m})$, yielding a principled trade-off between computational cost and estimation accuracy. By contrast, last-layer Laplace approximations do not admit comparable guarantees under the same assumptions, as they restrict uncertainty to a fixed subset of parameters corresponding to the final layer and thus discard network-wide sensitivity.

Approximating the Hessian. Our method requires access to curvature information of the denoiser network $\varepsilon_\theta(\boldsymbol{x}_t, t)$ in the form of a GGN matrix. For the mean-squared error objective used to train the denoiser, the GGN matrix at the MAP estimate satisfies

$$\boldsymbol{H} \;\approx\; \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{J}_i^\top \boldsymbol{J}_i, \qquad \boldsymbol{J}_i := \nabla_\theta \varepsilon_\theta(\boldsymbol{x}_i, t_i)\big|_{\hat{\theta}}, \tag{12}$$

where $(\boldsymbol{x}_i, t_i)$ are training pairs. Equivalently, stacking the Jacobians row-wise to form the population Jacobian $\boldsymbol{J}_{\text{pop}} \in \mathbb{R}^{(nd) \times p}$ yields $\boldsymbol{H} \approx \frac{1}{n} \boldsymbol{J}_{\text{pop}}^\top \boldsymbol{J}_{\text{pop}}$.

Algorithm 1: FLARE

Inputs : diffusion model ε_θ; subnetwork size m; Hessian–vector products for $(\boldsymbol{H} + \lambda \mathbf{I})$; DDPM schedule $\{\beta_t\}_{t=1}^T$; $\boldsymbol{x}_T \sim \mathcal{N}(0, \mathbf{I})$.

Outputs: generated sample $\hat{\boldsymbol{x}}_0$; epistemic covariance $\boldsymbol{\Sigma}^{\text{ep}}(\hat{\boldsymbol{x}}_0)$.

Precompute: $\alpha_t \leftarrow 1 - \beta_t$;
$\bar{\alpha}_t \leftarrow \prod_{s \le t} \alpha_s$;
$a_t \leftarrow \alpha_t^{-1/2}$;
$b_t \leftarrow \beta_t / (\sqrt{\alpha_t}\sqrt{1 - \bar{\alpha}_t})$.

Sample subnetwork (once): draw $I \subset [p]$; define solve operator $\boldsymbol{\Sigma}_{\text{sub}} v := (\boldsymbol{H}_{I,I} + \lambda \mathbf{I})^{-1} v$

Initialize: $\hat{\boldsymbol{x}}_T \leftarrow \boldsymbol{x}_T$,;
$\boldsymbol{\Sigma}^{\text{ep}}(\hat{\boldsymbol{x}}_T) \leftarrow \mathbf{0}$;

for $t = T$ **to** 1 **do**
- $\hat{\varepsilon}_t \leftarrow \varepsilon_\theta(\hat{\boldsymbol{x}}_t, t)$;
- $\hat{\boldsymbol{x}}_{t-1} \leftarrow a_t \Big(\hat{\boldsymbol{x}}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\, \hat{\varepsilon}_t\Big)$;
- $\boldsymbol{J}_{t,I} \leftarrow \nabla_{\theta_I} \varepsilon_\theta(\hat{\boldsymbol{x}}_t, t)$;
- $\Delta_t \leftarrow \boldsymbol{J}_{t,I}\, \boldsymbol{\Sigma}_{\text{sub}}\, \boldsymbol{J}_{t,I}^\top$;
- $\boldsymbol{\Sigma}^{\text{ep}}(\hat{\boldsymbol{x}}_{t-1}) \leftarrow a_t^2\, \boldsymbol{\Sigma}^{\text{ep}}(\hat{\boldsymbol{x}}_t) + b_t^2\, \Delta_t$;

return $\hat{\boldsymbol{x}}_0,\ \boldsymbol{\Sigma}^{\text{ep}}(\hat{\boldsymbol{x}}_0)$

Restricting $\boldsymbol{J}_{\text{pop}}$ to the final-layer coordinates recovers the standard LLLA. In contrast, our randomized subnetwork approach uniformly subsamples m columns of $\boldsymbol{J}_{\text{pop}}$, yielding a reduced GGN that retains sensitivity contributions from across the network. This reduces both compute and memory costs from scaling with p to scaling with m, while avoiding the structural bias introduced by last-layer restriction.

To disentangle the effects of full-curvature modeling from epistemic–aleatoric separation, we include controlled full-Hessian ablations and four-way uncertainty comparisons in Appendix G, Fig. 6.

2.4 Diagnostics

Algorithm 1 produces, for each generated sample $\hat{\boldsymbol{x}}_0$, an associated epistemic covariance $\boldsymbol{\Sigma}_0^{\text{ep}}(\eta)$. We use this covariance to define scalar diagnostics that quantify epistemic uncertainty at the sample level.

Our primary metric is the trace $\operatorname{tr}(\boldsymbol{\Sigma}_0^{\text{ep}}(\eta))$, which equals the sum of per-dimension epistemic variances. When reporting an average variance, we use the normalized trace $\operatorname{tr}(\boldsymbol{\Sigma}_0^{\text{ep}}(\eta))/d$. For cross-dataset comparisons, we apply additional normalization to ensure scale invariance. In all experiments, we rank samples by these scores and retain those with the lowest $\operatorname{tr}(\boldsymbol{\Sigma}_0^{\text{ep}}(\eta))$ as our most confidently generated samples.

In practice, we often seek scalar diagnostics without explicitly forming the full covariance matrix. For example, computing $\operatorname{tr}(\boldsymbol{\Sigma}_0^{\text{ep}}(\eta))$ requires evaluating

$\mathrm{tr}(\boldsymbol{\Sigma}^{\mathrm{ep}}_{t-1|t}(\eta))$ at each reverse step along the denoising trajectory. Let $\boldsymbol{g}_{t,k} \in \mathbb{R}^p$ denote the gradient of the k-th output of $\varepsilon_\theta(\boldsymbol{x}_t, t)$, i.e., the k-th row of $\boldsymbol{J}_t$. Then,

$$\mathrm{tr}(\boldsymbol{\Sigma}^{\mathrm{ep}}_{t-1|t}(\eta)) = b_t^2 \sum_{k=1}^{d} u_{t,k},$$

where $u_{t,k} := \boldsymbol{g}_{t,k}^\top \boldsymbol{\Sigma}_\theta \boldsymbol{g}_{t,k}$. Each $u_{t,k}$ can be computed efficiently using the conjugate gradient method. Specifically, since $\boldsymbol{\Sigma}_\theta \approx (\boldsymbol{H} + \lambda \mathbf{I})^{-1}$, we solve

$$(\boldsymbol{H} + \lambda \mathbf{I})\, \boldsymbol{z} = \boldsymbol{g}_{t,k}, \tag{13}$$

where $\boldsymbol{z} \in \mathbb{R}^p$ denotes the solution to the corresponding linear system, using Hessian–vector products (e.g., the empirical Fisher), evaluate $u_{t,k} = \boldsymbol{g}_{t,k}^\top \boldsymbol{z}$, and sum across dimensions. This avoids explicit matrix inversion and is standard in Laplace-based approximations.

2.5 Relation to BayesDiff

BayesDiff (Kou et al., 2024) estimates *pixel-wise predictive uncertainty* by placing a Laplace posterior on the *noise predictions* of a diffusion model and propagating data-space variances through the reverse diffusion process. Its recursion tracks the total variance of the latent state, combining diffusion noise, output uncertainty of the denoiser, and a data-space cross-covariance $\mathrm{Cov}(x_t, \varepsilon_t)$ estimated via Monte Carlo sampling. In contrast, our method operates directly in *parameter space*. We define epistemic uncertainty as the covariance of the reverse trajectory induced solely by uncertainty in the model parameters, conditioning on a fixed realization of the diffusion randomness. This leads to a recursion that propagates parameter-induced variability through the reverse dynamics via

$$\mathrm{Cov}_\theta\big(\mathbb{E}[x_{t-1} \mid x_t, \theta]\big) = b_t^2\, J_t\, \Sigma_\theta\, J_t^\top,$$

a quantity that is neither estimated nor approximated in BayesDiff. As a result, the two approaches quantify fundamentally different notions of uncertainty. BayesDiff targets total predictive uncertainty in data space, aggregating multiple sources of variability, whereas our method isolates epistemic uncertainty arising from parameter uncertainty and propagates it explicitly through the reverse diffusion process.

3 THEORY

Section 2 introduced two computational primitives: (i) the *one-step Fisher–Laplace projection* in Equation (9), and (ii) the *multi-step epistemic propagation* recursion in Equation (10). This section provides theoretical justification for both constructions, as well as for the randomized subnetwork approximation introduced in Section 2.3. All proofs are deferred to the appendix, as indicated throughout.

3.1 Epistemic Uncertainty and Propagation

We justify the recursion in Equation (10) used to propagate epistemic covariance along the reverse diffusion trajectory. We proceed in three steps: first, we establish the one-step Fisher–Laplace projection term (Equation (9)); second, we derive a general propagation identity that includes a cross-covariance term; and finally, we state a local decoupling condition under which this cross term vanishes, yielding the recursion used in our method.

One-step Fisher–Laplace projection. Our analysis follows from the law of total variance together with the DDPM reverse update (Ho et al., 2020). Let $\hat{\theta}$ denote the maximum a posteriori (MAP) estimate of the model parameters, around which the posterior distribution is locally approximated using a second-order (Laplace) expansion (Daxberger et al., 2021).

Proposition 1 (One-step Fisher–Laplace projection)**.** *Under a local Gaussian posterior $\theta \sim \mathcal{N}(\hat{\theta}, \boldsymbol{\Sigma}_\theta)$, independence of the reverse-step noise η_t from the model parameters θ, and a first-order (delta-method) linearization of the one-step conditional mean around $\hat{\theta}$, the conditional covariance at step t admits the decomposition*

$$\mathrm{Cov}_\theta\big(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \eta\big) = \underbrace{\tilde{\beta}_t \mathbf{I}}_{\textit{aleatoric}} + \underbrace{b_t^2\, \boldsymbol{J}_t\, \boldsymbol{\Sigma}_\theta\, \boldsymbol{J}_t^\top}_{\textit{epistemic}} + o(\|\boldsymbol{\Sigma}_\theta\|).$$

Consequently, the epistemic contribution to the reverse-step uncertainty, which we isolate and propagate in our method, is given by

$$\begin{aligned}\boldsymbol{\Sigma}^{\mathrm{ep}}_{t-1|t}(\eta) &:= \mathrm{Cov}_\theta\big(\mathbb{E}[\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \theta, \eta] \;\big|\; \eta\big) \\ &\approx b_t^2\, \boldsymbol{J}_t\, \boldsymbol{\Sigma}_\theta\, \boldsymbol{J}_t^\top.\end{aligned} \tag{14}$$

This result shows that epistemic uncertainty can be expressed explicitly in terms of parameter sensitivity, without conflating parameter uncertainty with diffusion noise. The above result holds under mild regularity conditions, formalized in Appendix A. In particular, we assume that the denoiser $\varepsilon_\theta(\cdot, t)$ is continuously differentiable in θ with a locally Lipschitz Jacobian, that the diffusion schedule is fixed, and that the injected noise η_t (a component of η) is independent of the model parameters. The posterior covariance $\mathrm{Cov}[\theta \mid \mathcal{D}] = \boldsymbol{\Sigma}_\theta$ is assumed to have finite second moments. Under a Laplace approximation, we take $\boldsymbol{\Sigma}_\theta \approx (\boldsymbol{H} + \lambda \mathbf{I})^{-1}$, where $\boldsymbol{H}$ denotes the Gauss–Newton approximation to the Hessian of the weighted MSE training loss (Ritter et al., 2018).

Propagation through the denoising steps. Equation (14) characterizes the epistemic uncertainty contributed by a single reverse step. To understand how this uncertainty accumulates along the reverse trajectory, we define $\boldsymbol{\Sigma}_t^{\mathrm{ep}}(\eta) \coloneqq \mathrm{Cov}_\theta\big(\boldsymbol{x}_t(\theta,\eta) \mid \eta\big)$, the epistemic covariance of the reverse state at time t, conditioned on a fixed realization of the non-parametric randomness η. Under the ε-prediction DDPM update and a first-order linearization around the MAP estimate, the epistemic covariance evolves according to

$$\begin{aligned}\boldsymbol{\Sigma}_{t-1}^{\mathrm{ep}}(\eta) = a_t^2\,\boldsymbol{\Sigma}_t^{\mathrm{ep}}(\eta) + b_t^2\,\boldsymbol{J}_t\,\boldsymbol{\Sigma}_\theta\,\boldsymbol{J}_t^\top \\ + 2a_t b_t\,\boldsymbol{C}_t(\eta) + o(\|\boldsymbol{\Sigma}_\theta\|),\end{aligned} \tag{15}$$

where the cross-covariance term is defined as

$$\boldsymbol{C}_t(\eta) \coloneqq \mathrm{Cov}_\theta\big(\boldsymbol{x}_t(\theta,\eta),\ \boldsymbol{J}_t(\theta-\hat{\theta}) \mid \eta\big).$$

To obtain a tractable recursion, we consider the regime in which the parameter-induced perturbation of the reverse state $\boldsymbol{x}_t(\theta,\eta)$ is approximately uncorrelated with the linearized denoiser response evaluated at the MAP iterate. Conditioned on a given realization of the non-parametric randomness η, this corresponds to

$$\mathrm{Cov}_\theta\Big(\boldsymbol{x}_t(\theta,\eta),\ \boldsymbol{J}_t(\theta-\hat{\theta}) \,\Big|\, \eta\Big) \approx \mathbf{0}. \tag{16}$$

Under this local decoupling approximation, the cross-covariance term in Equation (15) vanishes, yielding the additive recursion in Equation (10) used in our method. Importantly, this approximation is not imposed arbitrarily. In Appendix G, we show that the cross term is (i) uniformly bounded and absorbable into the leading variance terms, (ii) asymptotically negligible under posterior concentration and local smoothness of the reverse trajectory, and (iii) numerically insignificant under a full-Hessian Laplace posterior, as verified by Monte Carlo evaluation.

3.2 Randomized Estimator

We now provide theoretical support for the randomized subnetwork approximation introduced in Section 2.3. Using tools from randomized numerical linear algebra (Murray et al., 2023; Erichson et al., 2019), we show that the random subnetwork estimator converges rapidly to the full Fisher–Laplace covariance under mild regularity conditions, while last-layer restrictions generally do not admit comparable guarantees. Our main result establishes a relative error bound in trace norm that decays with the subnetwork size m.

Setup. We analyze the approximation of the one-step epistemic covariance $\boldsymbol{\Sigma}_{t-1|t}^{\mathrm{ep}}$. Under the generalized Gauss–Newton modeling assumption and Equation (14), this covariance can be written as

$$\boldsymbol{\Sigma}_{t-1|t}^{\mathrm{ep}} \approx \boldsymbol{J}_t\big(\boldsymbol{J}_{\mathrm{pop}}^\top\boldsymbol{J}_{\mathrm{pop}} + \lambda\mathbf{I}\big)^{-1}\boldsymbol{J}_t^\top, \tag{17}$$

where $\boldsymbol{J}_t$ denotes the Jacobian of the denoiser at step t, and $\boldsymbol{J}_{\mathrm{pop}}$ is the population Jacobian formed by stacking training-set Jacobians.

Since the regularization parameter λ is typically very small (e.g., $\lambda = 10^{-6}$ in our experiments), we simplify the analysis by considering the limit $\lambda \to 0$. This reduces the problem to approximating

$$\boldsymbol{V} \coloneqq \boldsymbol{J}_t\big(\boldsymbol{J}_{\mathrm{pop}}^\top\boldsymbol{J}_{\mathrm{pop}}\big)^+\boldsymbol{J}_t^\top, \tag{18}$$

where $(\cdot)^+$ denotes the Moore–Penrose pseudoinverse.

Directly constructing $\boldsymbol{V}$ is computationally infeasible in large models, as it requires access to the full Jacobians. Both the LLLA and the randomized subnetwork method approximate $\boldsymbol{V}$ by restricting attention to a subset of parameter coordinates. Specifically, they form reduced Jacobians $\tilde{\boldsymbol{J}}_t$ and $\tilde{\boldsymbol{J}}_{\mathrm{pop}}$ and compute

$$\tilde{\boldsymbol{V}} \coloneqq \tilde{\boldsymbol{J}}_t\big(\tilde{\boldsymbol{J}}_{\mathrm{pop}}^\top\tilde{\boldsymbol{J}}_{\mathrm{pop}}\big)^+\tilde{\boldsymbol{J}}_t^\top. \tag{19}$$

The two approaches differ only in how these reduced Jacobians are chosen. LLLA deterministically restricts to the coordinates corresponding to the final affine layer, whereas the randomized subnetwork method samples a uniformly random subset of m coordinates.

Least-squares reformulation. The following observation, proved in Appendix B, allows us to analyze these approximations through an equivalent least-squares perspective.

Lemma 1. *Let $\boldsymbol{A} \in \mathbb{R}^{p\times k}$ and $\boldsymbol{B} \in \mathbb{R}^{p\times \ell}$ be full-rank matrices with $k, \ell < p$. Let $\boldsymbol{X}_\star \in \mathbb{R}^{k\times\ell}$ denote the solution to $\min_{\boldsymbol{X}} \|\boldsymbol{A}\boldsymbol{X}-\boldsymbol{B}\|_{\mathrm{F}}$. Then $\boldsymbol{B}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^+\boldsymbol{B} = \boldsymbol{X}_\star^\top\boldsymbol{X}_\star$.*

Applying this lemma yields the representations

$$\boldsymbol{V} = \boldsymbol{X}_\star^\top\boldsymbol{X}_\star, \quad \boldsymbol{X}_\star \coloneqq \arg\min_{\boldsymbol{X}} \|\boldsymbol{J}_{\mathrm{pop}}^\top\boldsymbol{X} - \boldsymbol{J}_t^\top\|_{\mathrm{F}}^2,$$
$$\tilde{\boldsymbol{V}} = \tilde{\boldsymbol{X}}^\top\tilde{\boldsymbol{X}}, \quad \tilde{\boldsymbol{X}} \coloneqq \arg\min_{\boldsymbol{X}} \|\tilde{\boldsymbol{J}}_{\mathrm{pop}}^\top\boldsymbol{X} - \tilde{\boldsymbol{J}}_t^\top\|_{\mathrm{F}}^2.$$

Consequently, $\tilde{\boldsymbol{V}}$ is a good approximation of $\boldsymbol{V}$ if and only if the reduced least-squares problem provides an accurate approximation of the full one.

From this perspective, the LLLA implicitly assumes that the least-squares problem induced by $\boldsymbol{J}_{\mathrm{pop}}^\top$ and $\boldsymbol{J}_t^\top$ can be well approximated using only the coordinates corresponding to the final layer. This assumption is generally not guaranteed to hold for overdetermined least-squares problems and helps explain the empirical limitations of LLLA observed in practice.

Randomized subnetwork analysis. In contrast, the randomized subnetwork approximation corresponds to solving the same least-squares problem using

a uniformly random subset of rows. This strategy is well studied in randomized numerical linear algebra and is known to yield reliable approximations under mild conditions. Specifically, we assume: **(i) Conditioning:** The population Jacobian matrix $\boldsymbol{J}_{\text{pop}}^{\top}$ has condition number κ; **(ii) Coherence:** The population Jacobian has coherence μ, ruling out concentration of mass on a small subset of coordinates; **(iii) Alignment:** The step-wise Jacobian $\boldsymbol{J}_t$ has nontrivial overlap with the rowspace of $\boldsymbol{J}_{\text{pop}}$, quantified by a constant $\gamma > 0$. Under these conditions, we obtain the following trace-norm approximation guarantee.

Theorem 1. *Let $\boldsymbol{J}_{\text{pop}} \in \mathbb{R}^{nd\times p}$ and $\boldsymbol{J}_t \in \mathbb{R}^{d\times p}$ satisfy the assumptions above. Then, with probability at least $1-\delta$, the randomized subnetwork estimator with subnetwork size m satisfies*

$$\left|\operatorname{tr}(\boldsymbol{V}-\tilde{\boldsymbol{V}})\right| \;\le\; \tilde{\mathcal{O}}\left(\sqrt{\frac{p\,\gamma\,\mu}{m\,\delta}}\,\kappa\right)\operatorname{tr}(\boldsymbol{V}). \tag{20}$$

As m increases, the approximation error decays to zero. By a standard union bound, this guarantee can be extended to hold uniformly over the T reverse steps used in Algorithm 1.

4 EXPERIMENTAL RESULTS

In this section, we evaluate the effectiveness of our proposed method (Algorithm 1) for quantifying epistemic uncertainty in samples generated by diffusion models (research code: `https://github.com/aditiii12/UQ_in_Diffusion`). We use three synthetic time-series benchmarks to probe complementary aspects of epistemic uncertainty. These tasks test multi-modality, ambiguity, extrapolation, and temporal decay.

Baselines and evaluation protocol. We compare against two baseline methods: (i) BayesDiff predictive variance (Kou et al., 2024), and (ii) last-layer Laplace (LLLA) rollouts (Daxberger et al., 2021). All methods are evaluated under matched training and sampling protocols. Each method assigns a scalar uncertainty score u to every generated sample. For a given method, we form a filtered subset by retaining the lowest-uncertainty $p\%$ of samples, with $p = 50$ for the sine and grid datasets and $p = 25$ for the chirp dataset (the latter reduces visual clutter given the longer sequence length $L = 80$). Unfiltered generations, i.e., using all samples, serve as the baseline.

Quantitative evaluation. To assess the effect of uncertainty-based filtering, we train a discriminator D_ϕ to distinguish training samples from generated samples, labeling training data as 1 and generated data as 0. Its accuracy is $\text{Acc} = \frac{1}{N}\sum_{i=1}^{N} 1\{D_\phi(x_i) = y_i\}$, where $y_i \in \{0,1\}$ indicates if x_i is drawn from the training set. To quantify the impact of filtering, we report *gap closure*, defined as

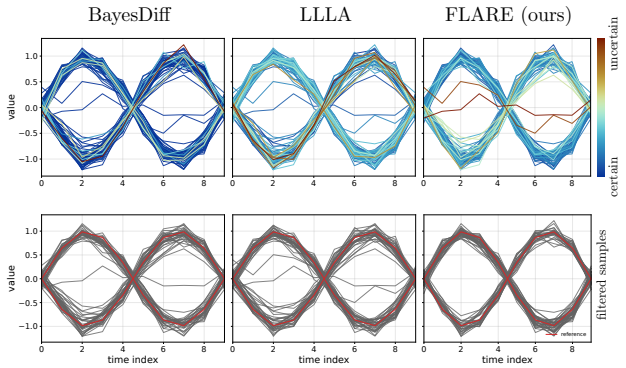
$$\text{Gap-Closure} \;=\; \frac{\left|0.5-\text{Acc}_f\right| - \left|0.5-\text{Acc}_{uf}\right|}{\left|0.5-\text{Acc}_{uf}\right|},$$

where Acc_f denotes the discriminator accuracy on the filtered samples and Acc_{uf} the accuracy on the unfiltered baseline. Subtracting from 0.5 measures by how much we are outperforming random chance, with smaller values indicating less distinguishability between generated and training samples. Gap closure measures how much filtering reduces the discriminator's ability to distinguish generated samples *relative to the unfiltered baseline*. A gap-closure of 100% is good, indicating that the discriminator can no longer distinguish filtered samples from training data. Values between 0% and 100% indicate partial improvement, while negative values indicate that filtering actually removes high quality samples and degrades realism. We also compute the Receiver Operating Characteristic Area Under the Curve (ROC-AUC), defined as $\text{AUC} = \Pr\big(s(x^+) > s(x^-)\big)$, where $s(\cdot)$ denotes the discriminator score, x^+ is a training sample, and x^- is a generated sample. An AUC of 0.5 corresponds to chance-level discrimination (good), while larger values indicate greater separability (bad). Statistical significance is assessed using bootstrap resampling. For each method, we estimate the distribution of gap-closure values under resampling and report p-values, indicating whether improvements are consistent.

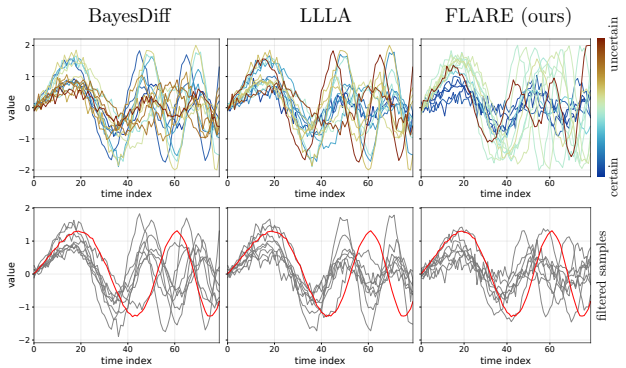
Results. As shown in Table 1, BayesDiff predictive variance yields modest improvements, with +13.4% gap closure on the sine dataset and +41.7% on the chirp dataset, both with statistical significance. The last-layer Laplace approximation (LLLA) often underperforms, exhibiting negative or marginal gap closures and relatively high ROC-AUC values. In contrast, our method consistently achieves the strongest gains, with +93.0% gap closure on the sine dataset, +74.3% on

Table 1: Comparison of uncertainty-based filtering. Higher gap-closure and lower ROC-AUC indicate improved alignment between filtered generations and the training data. Best values are highlighted in bold.

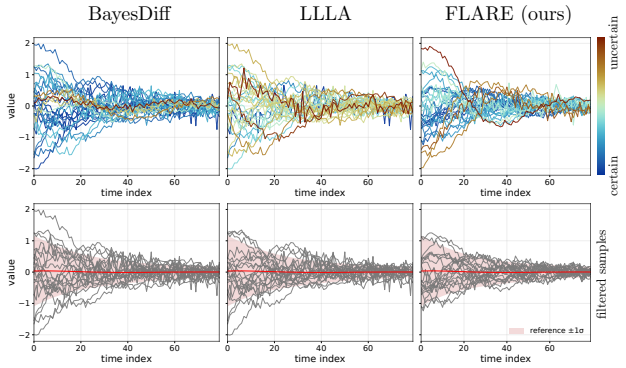
Dataset	Method	Gap-Closure (%)	ROC-AUC (↓)	p (bootstrap)
Sines	BayesDiff	+13.3654	0.6153	0.0001
	LLLA	+47.1873	0.5814	0.0201
	FLARE (ours)	**+93.0814**	**0.5003**	**0.0012**
Chirp	BayesDiff	+41.7337	0.6616	0.0030
	LLLA	+59.0583	0.5891	0.0116
	FLARE (ours)	**+74.3137**	**0.5345**	**0.0002**
Damped Sines	BayesDiff	+10.3984	0.6861	0.0001
	LLLA	−18.7540	0.7754	0.0066
	FLARE (ours)	**+85.0**	**0.5085**	**0.0002**



(a) Sinusoidal time series dataset.



(b) Chirp time series dataset.



(c) Damped Sinusoidal time series dataset.

Figure 2: Generated time-series samples before and after uncertainty-based filtering. Panel (a) shows the sinusoidal dataset and panel (b) the chirp dataset. **Top:** generated trajectories, colored by epistemic uncertainty (blue = low, red = high). **Bottom:** trajectories retained after filtering by uncertainty. Filtering removes implausible, off-manifold samples while preserving diverse, on-distribution trajectories.

the chirp dataset, and +76.2% on the damped sine dataset, alongside the lowest ROC-AUC values. These results indicate that filtering with FLARE substantially reduces the discriminator's ability to distinguish generated samples from training data across all three settings. All improvements are statistically significant under bootstrap testing. Overall, the quantitative results align with qualitative inspection, indicating that FLARE provides a more informative epistemic uncertainty signal than existing baselines.

Qualitative evaluation. Uncertainty-based filtering also improves the visual fidelity of generated samples. In the 2D grid dataset (Figure 1), filtered samples concentrate tightly around the mixture components, with reduced spillover into low-density regions. On the sinusoidal time-series benchmark (Figure 2a), low-uncertainty subsets cluster around the two sinusoidal modes, while samples in ambiguous regions between modes are pruned. For both the chirped and damped sinusoid datasets (Figure 2b, Figure 2c), filtering suppresses spurious oscillations and frequency drift, yielding smoother trajectories that better match the training distribution. *These visual patterns mirror the quantitative findings and provide evidence that FLARE identifies uncertainty in regions where the model must interpolate, extrapolate, or resolve ambiguity.*

5 CONCLUSIONS AND LIMITATIONS

In this work, we argue that reliable uncertainty quantification in diffusion models requires explicitly tracking how parameter uncertainty propagates through the reverse dynamics. To this end, we introduced a Fisher–Laplace formulation that projects parameter uncertainty into data space via the Jacobian of the denoiser and then accumulates this uncertainty along a realized reverse trajectory. Building on this formulation, we proposed FLARE, a randomized subnetwork estimator that makes Fisher–Laplace uncertainty estimation scalable to modern diffusion models. FLARE preserves network-wide sensitivity while reducing computational cost, and we provide both theoretical guarantees and empirical evidence that it yields more informative epistemic uncertainty estimates than last-layer or predictive-variance-based alternatives.

Limitations. We evaluated FLARE primarily on synthetic time-series benchmarks designed to probe specific epistemic failure modes; extending the approach to higher-dimensional domains such as multivariate time series or image generation remains an important direction for future work. In addition, while randomized subnetwork sampling significantly reduces computational overhead, approximating curvature information in very large diffusion models is still demanding. Future work could explore structured or adaptive subsampling strategies, low-rank curvature approximations, or tighter integrations with efficient second-order optimization methods to further scale FLARE.

Acknowledgments

NBE would like to acknowledge support from the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, EXPRESS: 2025 Exploratory Research for Extreme-Scale Science program, and the Scientific Discovery through Advanced Computing (SciDAC) program, under Contract Number DE-AC02-05CH11231 at Berkeley Lab. EC would like to acknowledge support from the National Science Foundation under Award No. 2412577. We would also like to acknowledge supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-25-2-0011. The views, opinions, and/or findings expressed are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

Aithal, S. K., Maini, P., Lipton, Z., and Kolter, J. Z. (2024). Understanding hallucinations in diffusion models through mode interpolation. *Advances in Neural Information Processing Systems*, 37:134614–134644.

Berry, L., Brando, A., and Meger, D. (2024). Shedding light on large generative networks: Estimating epistemic uncertainty in diffusion models. In *The 40th Conference on Uncertainty in Artificial Intelligence*.

Chan, M., Molina, M., and Metzler, C. (2024). Estimating epistemic and aleatoric uncertainty with a single model. *Advances in Neural Information Processing Systems*, 37:109845–109870.

Chen, S., Chewi, S., Li, J., Li, Y., Salim, A., and Zhang, A. R. (2022). Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *arXiv preprint arXiv:2209.11215*.

Chewi, S., Kalavasis, A., Mehrotra, A., and Montasser, O. (2025). Ddpm score matching and distribution learning. *arXiv preprint arXiv:2504.05161*.

Daxberger, E., Kristiadi, A., Immer, A., Eschenhagen, R., Bauer, M., and Hennig, P. (2021). Laplace redux-effortless bayesian deep learning. *Advances in neural information processing systems*, 34:20089–20103.

De Vita, M. and Belagiannis, V. (2025). Diffusion model guided sampling with pixel-wise aleatoric uncertainty estimation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3844–3854. IEEE.

Depeweg, S., Hernández-Lobato, J. M., Doshi-Velez, F., and Udluft, S. (2018). Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *International Conference on Machine Learning*, pages 1184–1193. PMLR.

Drineas, P. and Mahoney, M. W. (2018). Lectures on randomized numerical linear algebra. *American Mathematical Society*, 3(4).

Erichson, N. B., Voronin, S., Brunton, S. L., and Kutz, J. N. (2019). Randomized matrix decompositions using r. *Journal of Statistical Software*, 89:1–48.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. (2024). Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*.

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR.

Hang, T., Gu, S., Li, C., Bao, J., Chen, D., Hu, H., Geng, X., and Guo, B. (2023). Efficient diffusion training via min-snr weighting strategy. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7441–7451.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851.

Hüllermeier, E. and Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110:457–506.

Jazbec, M., Wong-Toi, E., Xia, G., Zhang, D., Nalisnick, E., and Mandt, S. (2025). Generative uncertainty in diffusion models. In *Proceedings of the 41st Conference on Uncertainty in Artificial Intelligence (UAI)*, Proceedings of Machine Learning Research.

Kendall, A. and Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*, volume 30.

Kou, S., Gan, L., Wang, D., Li, C., and Deng, Z. (2024). Bayesdiff: Estimating pixel-wise uncertainty in diffusion via bayesian inference. In *The Twelfth International Conference on Learning Representations*.

Krueger, D., Huang, C.-W., Islam, R., Turner, R., Lacoste, A., and Courville, A. (2017). Bayesian hypernetworks. *arXiv preprint arXiv:1710.04759*.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30.

Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., and Le, M. (2023). Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*.

Lipman, Y., Havasi, M., Holderrieth, P., Shaul, N., Le, M., Karrer, B., Chen, R. T., Lopez-Paz, D., Ben-Hamu, H., and Gat, I. (2024). Flow matching guide and code. *arXiv preprint arXiv:2412.06264*.

Liu, X., Gong, C., and qiang liu (2023). Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. (2022). Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in neural information processing systems*, 35:5775–5787.

MacKay, D. J. C. (1992). A practical bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. (2019). A simple baseline for bayesian uncertainty in deep learning. *Advances in neural information processing systems*, 32.

Meyer, R. A. (2023). Subspace embedding via leverage score sampling. *RandNLA Proof Wiki*.

Murray, R., Demmel, J., Mahoney, M. W., Erichson, N. B., Melnichenko, M., Malik, O. A., Grigori, L., Luszczek, P., Dereziński, M., Lopes, M. E., et al. (2023). Randomized numerical linear algebra: A perspective on the field with an eye to software. *arXiv preprint arXiv:2302.11474*.

Nichol, A. and Dhariwal, P. (2021). Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*.

Ritter, H., Botev, A., and Barber, D. (2018). A scalable laplace approximation for neural networks. In *International Conference on Learning Representations*.

Shu, D. and Farimani, A. B. (2024). Zero-shot uncertainty quantification using diffusion probabilistic models. *arXiv preprint arXiv:2408.04718*.

Song, J., Meng, C., and Ermon, S. (2020). Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2021). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

Woodruff, D. P. (2014). Sketching as a tool for numerical linear algebra. *Foundations and Trends in Theoretical Computer Science*, 10(1–2):1–157.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes.]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes.]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes, see supplement.]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes.]
   (b) Complete proofs of all theoretical results. [Yes, see Appendix.]
   (c) Clear explanations of any assumptions. [Yes.]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes.]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes, see Appendix and code.]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes.]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [No.]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes.]
   (b) The license information of the assets, if applicable. [Yes/No/Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes.]
   (d) Information about consent from data providers/curators. [Not Applicable.]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable.]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable.]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable.]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable.]

A Proof of Proposition 1

In this section, we prove Proposition 1. We start by restating the setup and formalizing notation further. Let d be the data dimension and p the number of parameters. We adopt the ε-prediction reverse update

$$\mathbf{x}_{t-1} = a_t\,\mathbf{x}_t - b_t\,\varepsilon_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + \boldsymbol{\eta}_t, \qquad \boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0}, \tilde{\beta}_t \mathbf{I}_d). \tag{21}$$

where $a_t, b_t, \tilde{\sigma}_t > 0$ are schedule constants. We take $\hat{\boldsymbol{\theta}}$ to be the maximum a posteriori (MAP) estimate of the denoiser's parameters $\boldsymbol{\theta}$ given training data D and prior $p(\boldsymbol{\theta})$:

$$\hat{\boldsymbol{\theta}} \;:=\; \arg\max_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta} \mid D) \;=\; \arg\max_{\boldsymbol{\theta}} \left\{\log p(D \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})\right\}.$$

We define the parameter–Jacobian of the denoiser at the current sampler state as

$$\mathbf{J}_t \;:=\; \nabla_{\boldsymbol{\theta}}\, \varepsilon_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\big|_{\hat{\boldsymbol{\theta}}} \in \mathbb{R}^{d\times p},$$

which depends on t because it is evaluated at the realized pair $(\mathbf{x}_t, t)$ along the reverse trajectory.

In order to prove Proposition 1, we will make a mild assumption about the smoothness of our model.

Assumption 1. *For all time steps t, we assume the denoiser $\epsilon_{\boldsymbol{\theta}}(\cdot, t)$ is continuously differentiable in a neighborhood around the MAP estimate $\hat{\boldsymbol{\theta}}$ that includes $\boldsymbol{\theta}$.*

We can now prove the main result:

Proposition 1, Restated. *Under a local Gaussian posterior $\boldsymbol{\theta} \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}})$, independence of the schedule noise $\boldsymbol{\eta}_t$ from $\boldsymbol{\theta}$, and Assumption 1, the conditional covariance at reverse step t can be decomposed as*

$$\mathrm{Cov}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, t) = \tilde{\beta}_t\,\mathbf{I}_d \;+\; b_t^2\,\mathbf{J}_t\,\boldsymbol{\Sigma}_\theta\,\mathbf{J}_t^\top \;+\; o(\|\boldsymbol{\Sigma}_\theta\|).$$

Consequently, the epistemic contribution to the covariance is

$$\boldsymbol{\Sigma}^{\mathrm{ep}}_{t-1|t} \;:=\; \mathrm{Cov}_{\boldsymbol{\theta}}\big(\,\mathbb{E}[\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta}]\,\big) \;=\; b_t^2\,\mathbf{J}_t\,\boldsymbol{\Sigma}_\theta\,\mathbf{J}_t^\top \;+\; o(\|\boldsymbol{\Sigma}_\theta\|).$$

Proof. Recall Equation (4), which uses the law of total variance to say

$$\mathrm{Cov}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, t) = \mathbb{E}_{\boldsymbol{\theta}}\big[\,\mathrm{Cov}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta})\,\big] \;+\; \mathrm{Cov}_{\boldsymbol{\theta}}\big(\,\mathbb{E}[\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta}]\,\big).$$

We will analyze these two terms separately.

Recall the reverse update rule under ϵ-prediction, stated in Equation (21). Conditioned on $(\boldsymbol{x}_t, t)$ and a fixed parameter vector $\boldsymbol{\theta}$, the only randomness in the update rule originates from the injection noise $\boldsymbol{\eta}_t$, and so

$$\mathbb{E}_{\boldsymbol{\theta}}[\mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[\mathrm{Cov}(\boldsymbol{\eta}_t)] = \mathbb{E}_{\boldsymbol{\theta}}[\tilde{\beta}_t \mathbf{I}_d] = \tilde{\beta}_t \mathbf{I}_d. \tag{22}$$

That is, the aleatoric portion of the covariance of $\boldsymbol{x}_{t-1}$ given $(\boldsymbol{x}_t, t)$ is exactly $\tilde{\beta}_t \mathbf{I}_d$.

We now turn to the second term from the law of total variance. From the reverse update rule Equation (21), we have $\mathbb{E}[\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t, \boldsymbol{\theta}] = a_t \boldsymbol{x}_t - b_t \epsilon_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}_t, t)$. By Assumption 1, we linearly approximate $\epsilon_{\boldsymbol{\theta}}$ around the MAP estimate $\hat{\boldsymbol{\theta}}$:

$$\epsilon_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = \epsilon_{\hat{\boldsymbol{\theta}}}(\boldsymbol{x}_t, t) + \boldsymbol{J}_t(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) + \boldsymbol{r}_t(\theta) \tag{23}$$

where $\boldsymbol{J}_t = \nabla_{\boldsymbol{\theta}} \epsilon_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)|_{\hat{\theta}}$ as defined earlier and the remainder function satisfies $\|\boldsymbol{r}_t(\boldsymbol{\theta})\| = O(\|\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\|^2)$. Substituting this into the conditional mean, we can define

$$\mathbf{c}_t := a_t \mathbf{x}_t - b_t\,\varepsilon_{\hat{\boldsymbol{\theta}}}(\mathbf{x}_t, t), \qquad \text{so that} \quad \boldsymbol{\mu}_{\boldsymbol{\theta}} := \mathbb{E}[\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta}] = \mathbf{c}_t - b_t \mathbf{J}_t(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) - b_t \mathbf{r}_t(\boldsymbol{\theta}). \tag{24}$$

We now examine the covariance of all three terms in $\mu_{\boldsymbol{\theta}}$ with respect to $\boldsymbol{\theta} \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_{\boldsymbol{\theta}})$. Since $\boldsymbol{c}_t$ is independent of $\boldsymbol{\theta}$, we know $\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{c}_t) = \mathbf{0}$. We also directly compute $\mathrm{Cov}(b_t \boldsymbol{J}_t(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})) = b_t^2 \boldsymbol{J}_t \boldsymbol{\Sigma}_{\boldsymbol{\theta}} \boldsymbol{J}_t^\top$.

To tackle the third term, recall that the norm of $\boldsymbol{r}_t(\boldsymbol{\theta})$ is quadratic in $\|\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}\|$. So, standard delta-method bounds imply that $\|\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{r}_t(\boldsymbol{\theta}))\| = o(\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\|_2)$ and that the cross-covariance between $\boldsymbol{r}_t(\boldsymbol{\theta})$ and $b_t^2\boldsymbol{J}_t(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})$ is $o(\|\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\|_2)$. Therefore,

$$\mathrm{Cov}_{\boldsymbol{\theta}}\big(\mathbb{E}[\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta}]\big) = b_t^2\, \mathbf{J}_t \boldsymbol{\Sigma}_\theta \mathbf{J}_t^\top \;+\; o(\|\boldsymbol{\Sigma}_\theta\|_2), \tag{25}$$

which is the epistemic contribution. Combining Equations (22) and (25) yields

$$\mathrm{Cov}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, t) = \tilde{\beta}_t \mathbf{I}_d \;+\; b_t^2\, \mathbf{J}_t \boldsymbol{\Sigma}_\theta \mathbf{J}_t^\top \;+\; o(\|\boldsymbol{\Sigma}_\theta\|_2).$$

We see that, the predictive covariance $\mathrm{Cov}(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t, t)$ decomposes into an aleatoric term from the injected noise and an epistemic term obtained by projecting the parameter posterior through the parameter–Jacobian at the current state.

□

Explicit DDPM coefficients. For completeness, one may take (Ho et al., 2020; Nichol and Dhariwal, 2021)

$$a_t = \frac{1}{\sqrt{\alpha_t}}\left(1 - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\right), \quad b_t = \frac{\sqrt{1-\alpha_t}}{\sqrt{\alpha_t}}, \quad \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\,\beta_t,$$

so that $m_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) = a_t \boldsymbol{x}_t - b_t \varepsilon_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ and $\mathrm{Var}(\boldsymbol{\eta}_t) = \tilde{\beta}_t \mathbf{I}$.

Remarks. (i) The DDPM reverse update yields an *exact* structural split $\mathrm{Cov}(x_{t-1} \mid x_t, t) = \tilde{\beta}_t I + \mathrm{Cov}_\theta(m_\theta)$ before approximation; the "aleatoric vs. epistemic" terminology follows Kendall and Gal (2017). (ii) The epistemic term is a first-order pushforward of the parameter posterior: $b_t^2 J_t \Sigma_\theta J_t^\top$, requiring only first-order network derivatives; the Laplace link between Σ_θ and (approximate) Fisher curvature is classical (MacKay, 1992; Daxberger et al., 2021). (iii) For mean- or x_0-prediction, replace $b_t J_t$ by the Jacobian of the corresponding one-step mean m_θ; the proof is unchanged. (iv) In the broader asymptotic/theoretical context for diffusion estimators and sampling, see Chewi et al. (2025); Chen et al. (2022) for likelihood identities, efficiency guarantees, and sampling convergence results that justify using Fisher-type curvature as a proxy for epistemic uncertainty in DDPMs. (v) The argument extends to DDIM (Song et al., 2020) and flow–matching/ODE samplers (Lipman et al., 2024): their one–step updates take the form $\mathbf{x}_{t-1} = m_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ (i.e., $\tilde{\sigma}_t = 0$), so the aleatoric term $\tilde{\beta}_t \mathbf{I}_d$ disappears and the epistemic contribution reduces to the first–order pushforward through the mean map

$$\mathrm{Cov}_{\boldsymbol{\theta}}\big(\mathbb{E}[\mathbf{x}_{t-1} \mid \mathbf{x}_t, t, \boldsymbol{\theta}]\big) \;=\; \big(\nabla_{\boldsymbol{\theta}} m_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\big)\, \boldsymbol{\Sigma}_\theta\, \big(\nabla_{\boldsymbol{\theta}} m_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\big)^\top \;+\; o(\|\boldsymbol{\Sigma}_\theta\|_2).$$

B Proofs for Section 3

In this appendix, we prove Lemma 1, which relates the covariance matrix construction to overdetermined least squares problems, and Theorem 1, which uses this least squares perspective to prove that the random subnetwork approach

Lemma 1, Restated. *Let $\boldsymbol{A} \in \mathbb{R}^{p\times k}, \boldsymbol{B} \in \mathbb{R}^{p\times \ell}$ be full rank with $k, \ell < p$. Let $\boldsymbol{X}_\star \in \mathbb{R}^{k\times \ell}$ be the solution to the least squares problem $\min_{\boldsymbol{X}} \|\boldsymbol{A}\boldsymbol{X} - \boldsymbol{B}\|_{\mathrm{F}}$. Then we have $\boldsymbol{B}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^+\boldsymbol{B} = \boldsymbol{X}_\star^\top \boldsymbol{X}_\star$.*

Proof. This follows directly from $(\boldsymbol{A}\boldsymbol{A}^\top)^+ = (\boldsymbol{A}^+)^\top \boldsymbol{A}^+$ and the fact that $\boldsymbol{X}_\star = \boldsymbol{A}^+\boldsymbol{B}$, since we can expand

$$\boldsymbol{B}^\top(\boldsymbol{A}\boldsymbol{A}^\top)^+\boldsymbol{B} = (\boldsymbol{A}^+\boldsymbol{B})^\top(\boldsymbol{A}^+\boldsymbol{B}) = \boldsymbol{X}_\star^\top \boldsymbol{X}_\star.$$

□

We next turn our attention to Theorem 1. Before proving this result, we formally define the terms involved. Let $\boldsymbol{A} \in \mathbb{R}^{p\times k}$ be full-rank matrix with $p \geq k$ with economic SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}_\theta \boldsymbol{V}^\top$ and singular values $\sigma_1 \geq \ldots \geq \sigma_k$. The *condition number* of $\boldsymbol{A}$ is the ratio $\kappa(\boldsymbol{A}) \coloneqq \frac{\sigma_1}{\sigma_k}$. Since we expanded the economic SVD of $\boldsymbol{A}$, we have $\boldsymbol{U} \in \mathbb{R}^{p\times k}$ with orthonormal columns but not orthonormal rows. The *coherence* of $\boldsymbol{A}$ is the maximum squared row norm of $\boldsymbol{U}$. That is, letting $\boldsymbol{u}_i^\top$ be the ith row of $\boldsymbol{U}$, we have $\mu(\boldsymbol{A}) \coloneqq \max_{i\in\{1,\ldots,p\}} \|\boldsymbol{u}_i\|_2^2 \in [\frac{k}{p}, 1]$.

The result Theorem 1 follows from the well-understood facts about the sketch-and-solve algorithm from randomized numerical linear algebra (Woodruff, 2014; Drineas and Mahoney, 2018; Meyer, 2023).

Imported Theorem 1. *Let $\boldsymbol{A} \in \mathbb{R}^{p\times k}$ be full-rank with $p \geq k$ and let $\boldsymbol{B} \in \mathbb{R}^{p\times \ell}$. Assume that $\|\boldsymbol{P}\boldsymbol{B}\|_{\mathrm{F}}^2 \geq \gamma\|(\mathbf{I}-\boldsymbol{P})\boldsymbol{B}\|_{\mathrm{F}}^2$ for some $\gamma > 0$, where $\boldsymbol{P} \in \mathbb{R}^{p\times p}$ is the orthogonal projection onto the range of $\boldsymbol{A}$. Sample a uniformly random set $I \subseteq \{1,\ldots,p\}$ with $|I| = m$ for some m with $k \leq m \leq p$. Let $\tilde{\boldsymbol{A}} \in \mathbb{R}^{m\times k}$ and $\tilde{\boldsymbol{B}} \in \mathbb{R}^{m\times \ell}$ be the restrictions of $\boldsymbol{A}$ and $\boldsymbol{B}$ to the rows indexed by I. Define $\boldsymbol{X}_\star$ to be the solution of the full least squares problem*

$$\boldsymbol{X}_\star \coloneqq \operatorname*{arg\,min}_{\boldsymbol{X}\in\mathbb{R}^{k\times\ell}} \|\boldsymbol{A}\boldsymbol{X}-\boldsymbol{B}\|_{\mathrm{F}}^2,$$

and define $\tilde{\boldsymbol{X}}$ to be the solution of the subsampled problem

$$\tilde{\boldsymbol{X}} \coloneqq \operatorname*{arg\,min}_{\boldsymbol{X}\in\mathbb{R}^{k\times\ell}} \|\tilde{\boldsymbol{A}}\boldsymbol{X}-\tilde{\boldsymbol{B}}\|_{\mathrm{F}}^2.$$

Then, with probability at least $1-\delta$ we have that

$$\|\boldsymbol{X}_\star - \tilde{\boldsymbol{X}}\|_{\mathrm{F}} \leq \tilde{\mathcal{O}}\left(\sqrt{\frac{p\gamma\mu(\boldsymbol{A})}{m\delta}}\kappa(\boldsymbol{A})\right)\|\boldsymbol{X}_\star\|_{\mathrm{F}}.$$

Remark 1. This result does not appear exactly as stated above in the cited works. To recover this theorem statement, we first use the Corollary 1 from (Meyer, 2023) to show that the uniformly random subsampling procedure produces an *OSE Guarantee*. We then use the OSE Guarantee to activate Lemma 68 from (Drineas and Mahoney, 2018). We further note the Lemma 68 is only stated for the case where $\ell = 1$. However, the proof of Lemma 68 goes through without change if we allow $\ell > 1$ and use the Frobenius norm in place of the vector ℓ_2 norm. These analyses are all standard, and have been presented in some form or another in many other works.

From this result, we can now recover our main theorem statement.

Theorem 1, Restated. *Let $\boldsymbol{J}_{\mathrm{pop}} \in \mathbb{R}^{nd\times p}$ and $\boldsymbol{J}_t \in \mathbb{R}^{d\times p}$. Let $\mu \in [\frac{nd}{p}, 1]$ and $\kappa \geq 1$ be the* coherence *and* condition number *of $\boldsymbol{J}_{\mathrm{pop}}^\top$, respectively. Let $\boldsymbol{P} \in \mathbb{R}^{p\times p}$ be the orthogonal projection onto the rowspace of $\boldsymbol{J}_{\mathrm{pop}}$, and assume that $\|\boldsymbol{J}_t\boldsymbol{P}\|_{\mathrm{F}}^2 \geq \gamma\|\boldsymbol{J}_t(\mathbf{I}-\boldsymbol{P})\|_{\mathrm{F}}^2$ for some $\gamma > 0$. Then, then random subnetwork approximation with subnetwork size m has*

$$|\operatorname{tr}(\boldsymbol{V}-\tilde{\boldsymbol{V}})| \leq \tilde{\mathcal{O}}\left(\sqrt{\frac{p\gamma\mu}{m\delta}}\kappa\right)\operatorname{tr}(\boldsymbol{V})$$

with probability at least $1-\delta$.

Proof. By Lemma 1, we know that

$$\boldsymbol{V} = \boldsymbol{X}_\star^\top\boldsymbol{X}_\star \quad \text{where} \quad \boldsymbol{X}_\star = \operatorname*{arg\,min}_{\boldsymbol{X}} \|\boldsymbol{J}_{\mathrm{pop}}^\top\boldsymbol{X} - \boldsymbol{J}_t^\top\|_{\mathrm{F}}^2, \text{ and}$$
$$\tilde{\boldsymbol{V}} = \tilde{\boldsymbol{X}}^\top\tilde{\boldsymbol{X}} \quad \text{where} \quad \tilde{\boldsymbol{X}} = \operatorname*{arg\,min}_{\boldsymbol{X}} \|\tilde{\boldsymbol{J}}_{\mathrm{pop}}^\top\boldsymbol{X} - \tilde{\boldsymbol{J}}_t^\top\|_{\mathrm{F}}^2.$$

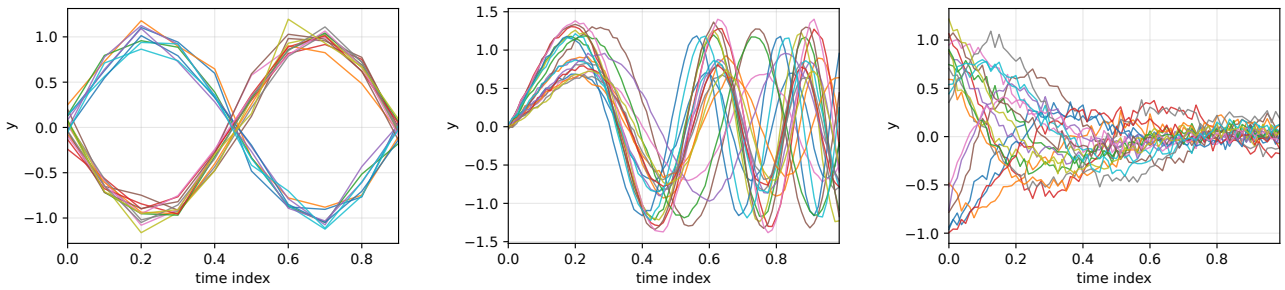
By Imported Theorem 1, we know that $\|\boldsymbol{X}_\star - \tilde{\boldsymbol{X}}\|_{\mathrm{F}} \leq \varepsilon\|\boldsymbol{X}_\star\|_{\mathrm{F}}$ with probability at least $1-\delta$, where $\varepsilon = \tilde{\mathcal{O}}(\sqrt{\frac{p\gamma\mu}{m\delta}}\kappa)$. We thereby infer that $|\|\boldsymbol{X}_\star\|_{\mathrm{F}}^2 - \|\tilde{\boldsymbol{X}}\|_{\mathrm{F}}^2| \leq 2\varepsilon\|\boldsymbol{X}_\star\|_{\mathrm{F}}^2$ for small enough ε. Recalling that $\operatorname{tr}(\boldsymbol{A}^\top\boldsymbol{A}) = \|\boldsymbol{A}\|_{\mathrm{F}}^2$, we observe that $\operatorname{tr}(\boldsymbol{V}) = \|\boldsymbol{X}_\star\|_{\mathrm{F}}^2$ and therefore

$$|\operatorname{tr}(\boldsymbol{V}-\tilde{\boldsymbol{V}})| = \left|\|\boldsymbol{X}_\star\|_{\mathrm{F}}^2 - \|\tilde{\boldsymbol{X}}\|_{\mathrm{F}}^2\right| \leq 2\epsilon\|\boldsymbol{X}_\star\|_{\mathrm{F}}^2 = \tilde{\mathcal{O}}\left(\sqrt{\frac{p\gamma\mu}{m\delta}}\kappa\right)\operatorname{tr}(\boldsymbol{V})$$

□

C Dataset Description

We used four simple synthetic data sets to test different aspects of uncertainty behavior. Each dataset is designed to expose a particular modeling challenge while remaining interpretable and visually intuitive. The collection includes both spatial and temporal domains, covering multimodal distributions as well as dynamic signals with varying frequency, phase, and decay. Together, these benchmarks provide controlled settings in which to study how uncertainty captures ambiguity, extrapolation, and signal degradation. They serve as minimal yet diverse examples that highlight where models should express high confidence and where epistemic uncertainty should naturally arise.



(a) Bimodal Sinusoidal **(b)** Chirp **(c)** Damped Sinusoidal

Figure 3: Illustrations of the synthetic datasets used for training. Each dataset highlights a distinct modeling challenge: mode ambiguity, frequency drift, and amplitude drift.

1. **Grid (2D)**: a 3×3 modal Gaussian mixture with centers on $\{-1, 0, 1\}^2$ and per-mode covariance $0.05^2 I_2$, yielding high-support cells separated by low-density corridors. This setup highlights the need for spatially aware uncertainty, especially between modes where epistemic error should spike.

2. **Bimodal Sinusoidal time series** (length L=10): each sequence is drawn from a balanced mixture of $x_t = \pm \sin(2\pi\tau_t) + \varepsilon_t$, where τ_t is a uniform grid on $[0, 1]$ and $\varepsilon_t \sim \mathcal{N}(0,\, 0.1^2)$. This creates two separated modes with ambiguous regions in between.

3. **Chirp time series** (length L=80): sequences follow $x_t = A \sin\big(2\pi(f_0\tau_t + \frac{1}{2}k\tau_t^2)\big) + \varepsilon_t$, with $A \sim \mathcal{U}[0.6, 1.4]$, $f_0 \sim \mathcal{U}[0.5, 1.0]$, $k \sim \mathcal{U}[2.0, 5.0]$, τ_t an 80-point grid on $[0, 1]$, and $\varepsilon_t \sim \mathcal{N}(0,\, 0.02^2)$. The time-varying frequency stresses extrapolation and off-manifold behavior.

4. **Damped sinusoidal time series** (length L=40): sequences decay smoothly in amplitude, following $x_t = Ae^{-dt} \sin\big(2\pi f t + \phi\big) + \varepsilon_t$, with A, f, d, and ϕ drawn from uniform ranges. This benchmark evaluates how well uncertainty tracks vanishing signal energy and long-horizon extrapolation under structured decay.

D Model Details and Hyperparameters

For each dataset, we train a diffusion denoiser with output dimension matching the data. For the denoiser we use a ScoreNet-style architecture composed of FiLM–residual MLP blocks, trained with the ε-prediction objective used in diffusion models. The reported last-layer parameter counts correspond to the regression head `Linear`(hidden → data_dim), which maps the hidden representation to the data dimension and includes bias parameters.

On the bimodal sinusoidal dataset, the network has **8,330** total parameters; the LLLA head is `net.out` with weight 10×32 and bias (**330** last-layer parameters).

On both the chirp and damped sine datasets, the model has **72,496** total parameters; the LLLA head is 80×128 plus bias (**10,320** last-layer parameters). For the full-parameter variant on the chirp dataset, we use the randomized version of the algorithm with a subsampled parameter set of **4,412** directions to control cost.

On the 2D multimodal grid dataset, the network totals **7,810** parameters, with a **66**-parameter last layer. This setup lets us contrast (i) BayesDiff predictive variance, (ii) last-layer epistemic rollouts, and (iii) subsampled projected epistemic rollouts under matched training and sampling protocols. Table 2 summarizes the key experimental settings for each dataset, including input dimensionality, model size, diffusion parameters, and optimization details used throughout our experiments. We report both last-layer Laplace (LLLA) and full-parameter Fisher/Laplace variants.

Table 2: Model architectures, dataset sizes, and parameter counts for all datasets.

	2D Grid	Bimodal Sinusoid (L=10)	Chirp (L=80)	Damped Sine (L=40)
Dataset size	6000/mode	5000	8000	8000
Input dimension (d)	2	10	80	40
Model width	32	32	128	128
Diffusion steps (T)	800	600	600	600
Beta schedule	cosine	cosine	cosine	cosine
Learning rate	5×10^{-6}	5×10^{-4}	5×10^{-4}	5×10^{-4}
Optimizer (β_1, β_2)	(0.9, 0.999)	(0.9, 0.999)	(0.9, 0.999)	(0.9, 0.999)
Batch size	256	512	256	256
Total parameters (p)	7,810	4,218	72,496	72,496
Last-layer parameters	66	330	10,320	10,320

E Implementation Details

We reuse the same architectures and datasets across all runs. Training follows standard DDPM practice: **AdamW** optimizer with cosine learning–rate decay, gradient clipping, and an exponential moving average (EMA) of parameters. Unless stated otherwise, evaluation uses EMA weights; the EMA decay is tuned per dataset in the range $[0.999, 0.9999]$. We use diffusion loss with log–SNR weighting across timesteps (placing more mass around mid–SNR), and sample training timesteps from the same distribution used by the loss. Data preprocessing and augmentations (normalization, *etc.*) match the baseline DDPM setup.

Noise schedule and sampler We adopt a cosine (or linear, matching the baseline) β-schedule; the corresponding $(a_t, b_t, \tilde{\sigma}_t)$ are computed exactly from the schedule definitions.

MAP estimate and posterior approximation. After training, we take the MAP point $\hat{\boldsymbol{\theta}}$ (the final EMA weights unless noted) and approximate the local parameter posterior with a Gaussian using the `laplace-torch` (Laplace) library. Curvature is computed as generalized Gauss–Newton / empirical Fisher on the training loss; damping (prior precision) is tuned on a small validation split. The posterior is computed once at $\hat{\boldsymbol{\theta}}$ and reused for all timesteps.

Jacobian handling. We never materialize $\mathbf{J}_t = \nabla_{\boldsymbol{\theta}} \varepsilon_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$. All computations only require its action on vectors, obtained via autograd as vector–Jacobian and Jacobian–vector products (VJPs/JVPs), each costing a single forward+backward pass. We instantiate the posterior with the low-rank Laplace variant (via the `laplace` library), yielding $\boldsymbol{\Sigma}_\theta \approx \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$ with rank $r \ll p$. The epistemic term is then formed without trace estimators:

$$b_t^2 \, \mathbf{J}_t \boldsymbol{\Sigma}_\theta \mathbf{J}_t^\top \;=\; b_t^2 \sum_{i=1}^{r} \lambda_i \, (\mathbf{J}_t \mathbf{u}_i)(\mathbf{J}_t \mathbf{u}_i)^\top,$$

computed by pushing the basis vectors $\{\mathbf{u}_i\}_{i=1}^r$ through $\mathbf{J}_t$ using r JVPs per step t. We cache per-t intermediates (e.g., $\varepsilon_{\hat{\boldsymbol{\theta}}}(\mathbf{x}_t, t)$ and b_t) to avoid redundant passes; the overall cost scales linearly with the chosen rank r and is independent of the total parameter count p.

F Additional Experiments

(a) DDIM results. We repeat our analysis with the deterministic DDIM sampler (setting the injected noise to zero, $\boldsymbol{\eta}_t \equiv \mathbf{0}$). In this case the aleatoric component vanishes and the one–step predictive covariance is dominated by the epistemic pushforward

$$\mathrm{Cov}(\mathbf{x}_{t-1} \mid \mathbf{x}_t, t) \;\approx\; b_t^2 \, \mathbf{J}_t \, \boldsymbol{\Sigma}_\theta \, \mathbf{J}_t^\top,$$

composed with the DDIM step's linear map when applicable. Qualitatively, the trajectories and uncertainty bands closely track the DDPM case: epistemic bands concentrate around the two ground–truth modes throughout the reverse sweep. Figure 4 shows results for 2D grid data. Again, it can be seen that FLARE faithfully filters the generated data.

(b) Parameter–budget ablation. We study the effect of restricting the parameter posterior to a fraction of weights using FullSubnetLaplace, keeping $\{1\%, 5\%, 10\%, 30\%, 50\%\}$ of parameters. As the kept fraction increases, samples tighten around the data manifolds and the epistemic covariance $b_t^2 \mathbf{J}_t \mathbf{\Sigma}_\theta \mathbf{J}_t^\top$ contracts smoothly; already at 10–30% we observe stable mode coverage, with diminishing returns beyond 30–50%. Representative trajectories across the grid are shown in Figure 5.

G Discussion around Parameter-Covariance Term

This appendix empirically evaluates the parameter cross-covariance term that arises in the full epistemic variance recursion and provides an independent Monte-Carlo (MC) validation of our analytic computation under a full-Hessian Laplace posterior.

G.1 Theoretical Upper Bound on Cross-Covariance Term

Let $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma_\theta)$ denote the parameter posterior induced by a full-Hessian Laplace approximation. Linearizing the reverse diffusion step around $\hat{\theta}$ yields

$$x_{t-1} \;\approx\; a_t x_t - b_t \varepsilon_\theta(x_t, t), \tag{26}$$

where a_t, b_t are scalar schedule coefficients and $\varepsilon_\theta(\cdot)$ denotes the model's noise predictor.

Conditioning on all non-parameter randomness (i.e. on a fixed reverse trajectory), the epistemic variance propagates as

$$\mathrm{Var}_\theta[x_{t-1}] = a_t^2 \,\mathrm{Var}_\theta[x_t] + b_t^2 \,\mathrm{Var}_\theta[\varepsilon_t] - 2 a_t b_t \,\mathrm{Cov}_\theta(x_t, \varepsilon_t). \tag{27}$$

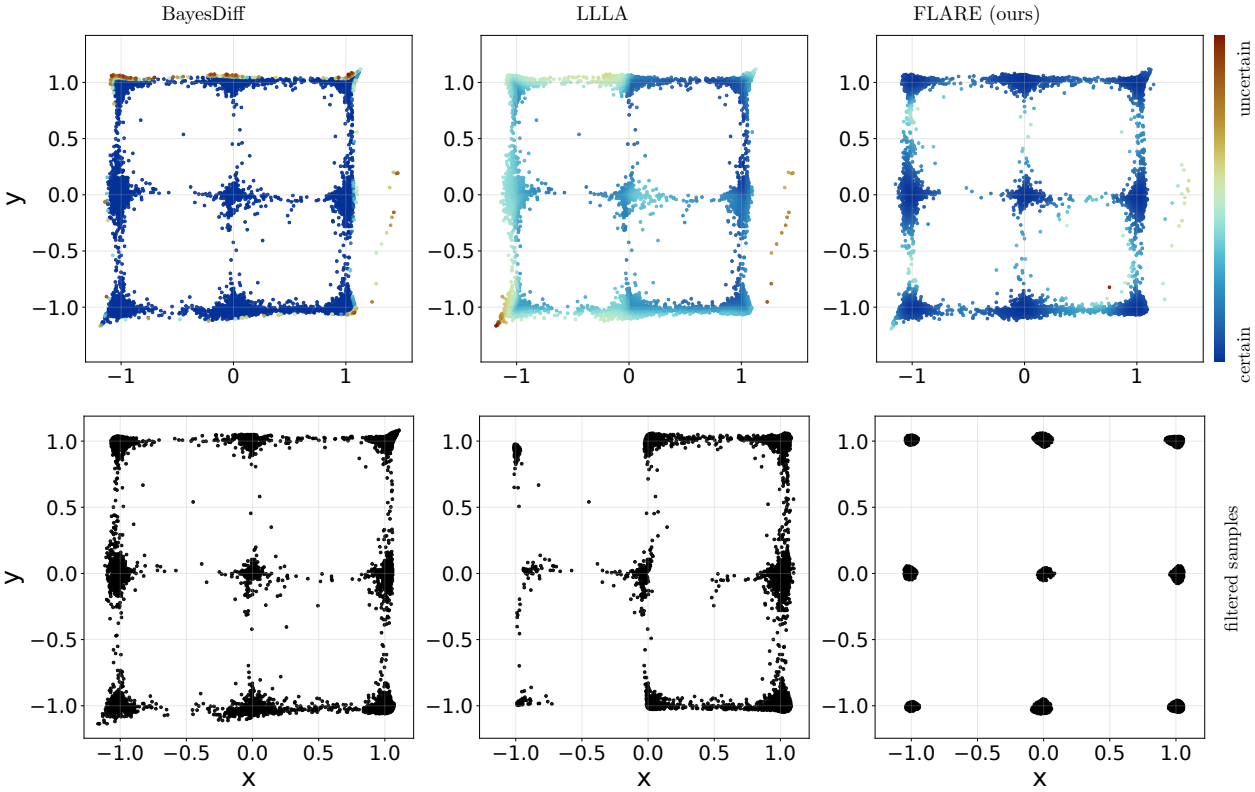


Figure 4: DDIM - Mode interpolation on a 2D Gaussian mixture adapted from Aithal et al. (2024) and Jazbec et al. (2025). The dataset consists of 9 Gaussian modes arranged on a square grid. The top row shows uncertainty scores from BayesDiff (left), LLLA (middle), and our method (right) for generated samples. The bottom row shows the same generated samples after filtering by a fixed uncertainty threshold. BayesDiff fails to assign high scores to points between modes, while LLLA performs even worse, leading to unreliable filtering. In contrast, our method produces faithful uncertainty estimates, enabling consistent removal of low-confidence samples.

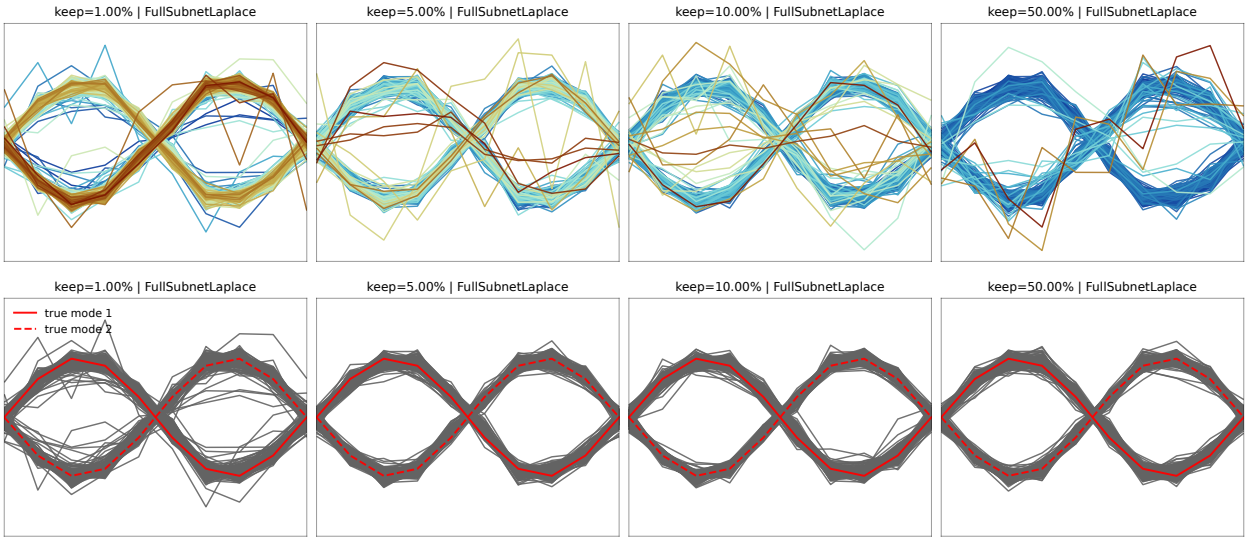


Figure 5: Parameter–budget ablation with FullSubnetLaplace. We retain $\{1\%, 5\%, 10\%, 30\%, 50\%\}$ of parameters. As the kept fraction increases, trajectories tighten around the data manifolds and the epistemic covariance $b_t^2 \mathbf{J}_t \mathbf{\Sigma}_\theta \mathbf{J}_t^\top$ contracts smoothly; 10–30% already yields stable mode coverage, with diminishing returns beyond 30–50%.

Bounding and absorbing the cross term. Condition on a fixed realization of aleatoric randomness η, so that all randomness is over $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma_\theta)$. For any unit vector $u \in \mathbb{S}^{d-1}$, Cauchy–Schwarz yields

$$\left|\mathrm{Cov}_\theta\big(u^\top x_t,\ u^\top \varepsilon_t\big)\right| \le \sqrt{\mathrm{Var}_\theta(u^\top x_t)}\ \sqrt{\mathrm{Var}_\theta(u^\top \varepsilon_t)}. \tag{28}$$

Applying Young's inequality $2\sqrt{ab} \le \delta a + \delta^{-1} b$ (for any $\delta > 0$) gives

$$2\left|\mathrm{Cov}_\theta\big(u^\top x_t,\ u^\top \varepsilon_t\big)\right| \le \delta\, \mathrm{Var}_\theta(u^\top x_t) + \delta^{-1}\, \mathrm{Var}_\theta(u^\top \varepsilon_t). \tag{29}$$

Taking the directional variance of equation 27 and bounding the cross term yields the sandwich bound

$$\begin{aligned}\Big(a_t^2 - |a_t b_t|\delta\Big)\mathrm{Var}_\theta(u^\top x_t) + \Big(b_t^2 - |a_t b_t|\delta^{-1}\Big)\mathrm{Var}_\theta(u^\top \varepsilon_t) &\le \mathrm{Var}_\theta(u^\top x_{t-1}) \\ &\le \Big(a_t^2 + |a_t b_t|\delta\Big)\mathrm{Var}_\theta(u^\top x_t) + \Big(b_t^2 + |a_t b_t|\delta^{-1}\Big)\mathrm{Var}_\theta(u^\top \varepsilon_t).\end{aligned} \tag{30}$$

Thus the cross covariance term cannot introduce an uncontrolled contribution: it is always dominated by (and can be absorbed into) the same two variance terms already present in the recursion.

When is the cross term small? Beyond being absorbable via Cauchy–Schwarz and Young's inequality, the cross term is *asymptotically negligible* under posterior concentration and local smoothness of the sampler.

Conditioned on a fixed realization of all aleatoric randomness, suppose the reverse trajectory $x_t(\theta)$ admits a local linearization around the MAP,

$$x_t(\theta) = x_t(\hat{\theta}) + G_t(\theta - \hat{\theta}) + r_t(\theta),$$

where $G_t := \nabla_\theta x_t(\theta)|_{\hat{\theta}}$ and the remainder satisfies $\mathbb{E}_\theta \|r_t(\theta)\|^2 = o(\mathrm{tr}(\Sigma_\theta))$. Then

$$\mathrm{Cov}_\theta\big(x_t(\theta), J_t(\theta - \hat{\theta})\big) = G_t \Sigma_\theta J_t^\top + o(\|\Sigma_\theta\|),$$

and therefore

$$\big\|\mathrm{Cov}_\theta(x_t(\theta), J_t(\theta - \hat{\theta}))\big\| \ \le\ \|G_t\|\,\|\Sigma_\theta\|\,\|J_t\| + o(\|\Sigma_\theta\|).$$

Under posterior concentration ($\Sigma_\theta \to 0$), the cross term vanishes at the same order as the leading epistemic variance term $J_t \Sigma_\theta J_t^\top$.

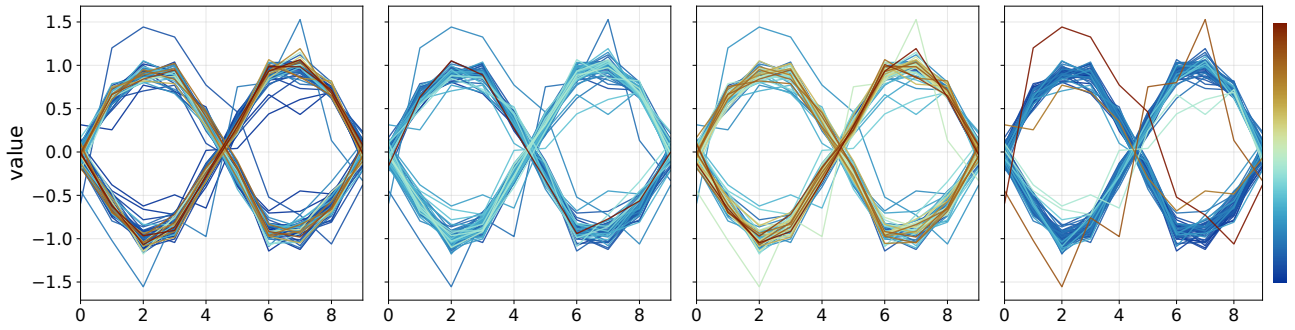


Figure 6: **Four-way epistemic UQ comparison in the full-curvature regime.** We compare uncertainty estimates from (left to right): *BayesDiff (last-layer Laplace, total recursion)*, *Fuller-Hessian Laplace (Bayesdiff recursion)*, *BayesDiff (epistemic-only recursion)*, and *FLARE (Fisher–Laplace projection)*. In this controlled setting, full-curvature computation is feasible and allows direct inspection of epistemic structure. The scalable approximation of Kou et al. (2024) (BayesDiff) already breaks down in this regime, producing uncertainty maps that are not epistemically meaningful relative to the full-Hessian reference, whereas FLARE closely tracks the curvature-based baseline while avoiding full-Hessian computation.

Importantly, G_t is not an independent quantity: since the sampler depends on parameters only through the denoiser, G_t is itself a linear combination of past denoiser Jacobians $\{J_s\}_{s>t}$ propagated through the linear reverse dynamics. Thus $\|G_t\|$ is controlled by the same local smoothness and stability assumptions used to bound $\|J_t\|$.

A more direct sufficient condition is Lipschitz stability of the reverse trajectory in parameters: assume that on the posterior mass,

$$\|x_t(\theta) - x_t(\hat{\theta})\| \le L_t \|\theta - \hat{\theta}\|.$$

Then for any unit vector u,

$$\mathrm{Var}_\theta(u^\top x_t) \le L_t^2 \,\mathrm{tr}(\Sigma_\theta), \qquad \mathrm{Var}_\theta\big(u^\top J_t(\theta - \hat{\theta})\big) \le \|J_t\|_2^2 \,\lambda_{\max}(\Sigma_\theta),$$

and by Cauchy–Schwarz,

$$\big|\mathrm{Cov}_\theta(u^\top x_t, u^\top J_t(\theta - \hat{\theta}))\big| \le L_t \|J_t\|_2 \sqrt{\mathrm{tr}(\Sigma_\theta)\,\lambda_{\max}(\Sigma_\theta)}.$$

Thus the cross term is small whenever the parameter posterior is concentrated and the reverse trajectory is stable to parameter perturbations.

Specialization to the linearized term. In our Monte-Carlo study we replace ε_t by the linearized perturbation $J_t(\theta - \hat{\theta})$. In this case, $\mathrm{Var}_\theta(u^\top J_t(\theta - \hat{\theta})) = u^\top J_t \Sigma_\theta J_t^\top u$, so the same absorption bound applies with $\mathrm{Var}_\theta(u^\top \varepsilon_t)$ replaced by $u^\top J_t \Sigma_\theta J_t^\top u$.

Our main method (FLARE) drops the final cross term under a local decoupling assumption. Here we empirically verify that this term is numerically negligible.

G.2 Empirical Monte Carlo Cross-Covariance Term Evaluation

We empirically assess the "missing" cross-covariance term from the reviewer comment,

$$-2a_t b_t \,\mathrm{Cov}_\theta\big(x_t(\theta),\, J_t(\theta - \hat{\theta})\big), \tag{31}$$

where $\mathrm{Cov}_\theta(\cdot)$ denotes covariance over θ *conditioned on a fixed realization of all aleatoric randomness* ξ (initialization x_T and diffusion noises $\{\eta_t\}$), as defined in our rebuttal. We take $\hat{\theta}$ to be the MAP estimate and approximate the parameter posterior by a full-Hessian Laplace, $\theta \sim \mathcal{N}(\hat{\theta}, \Sigma_\theta)$, with precision $P = \Sigma_\theta^{-1}$.

Shared stochastic reverse path. We fix a single realization ξ and run stochastic DDPM sampling to obtain a trajectory $\{x_t(\hat{\theta})\}_{t=0}^{T}$. All Monte Carlo evaluations below reuse the same ξ, so that variability arises only from sampling θ.

Local linearization term. At each step t, we compute the Jacobian at the MAP (evaluated along the shared path),

$$J_t \;=\; \nabla_\theta \varepsilon_\theta(x_t(\hat{\theta}), t)\big|_{\theta=\hat{\theta}}, \tag{32}$$

and use it to form the linearized perturbation $J_t(\theta - \hat{\theta})$ for sampled θ.

Monte Carlo estimator of the cross-covariance. We draw S samples $\delta\theta^{(s)} \sim \mathcal{N}(0, \Sigma_\theta)$ and set $\theta^{(s)} = \hat{\theta} + \delta\theta^{(s)}$. For each s, we recompute the DDPM reverse iterate $x_t(\theta^{(s)})$ using the *same* aleatoric randomness ξ. We then estimate the cross-covariance in equation 31 dimensionwise:

$$\widehat{\mathrm{Cov}}_{\theta,j}(t) = \frac{1}{S-1}\sum_{s=1}^{S}\Big(x_{t,j}(\theta^{(s)}) - \bar{x}_{t,j}\Big)\Big(\big(J_t\delta\theta^{(s)}\big)_j - \overline{(J_t\delta\theta)_j}\Big), \tag{33}$$

with $\bar{x}_{t,j}$ and $\overline{(J_t\delta\theta)_j}$ the sample means across s.

Epistemic recursion with and without the MC cross term. Using the same coefficients (a_t, b_t) as in the main text, we compare:

$$\mathrm{Var}_{t-1}^{\mathrm{noX}} = a_t^2\,\mathrm{Var}_t^{\mathrm{noX}} + b_t^2\,\mathrm{diag}\big(J_t\Sigma_\theta J_t^\top\big), \tag{34}$$

$$\mathrm{Var}_{t-1}^{\mathrm{withX}} = a_t^2\,\mathrm{Var}_t^{\mathrm{withX}} + b_t^2\,\mathrm{diag}\big(J_t\Sigma_\theta J_t^\top\big) - 2a_t b_t\,\widehat{\mathrm{Cov}}_\theta(t), \tag{35}$$

where $\widehat{\mathrm{Cov}}_\theta(t) \in \mathbb{R}^D$ stacks the dimensionwise estimates in equation 33. Per-sample epistemic scores are $u = \sum_{j=1}^{D}\mathrm{Var}_j$.

Result (baseline small model). On a baseline setting with $n = 20$ shared-path samples and $S = 128$ posterior draws per step, the MC cross term changes the epistemic score by approximately $10^{-3}\%$–$10^{-2}\%$ on average. A one-sided test against a practical threshold of 0.01% rejects $H_0 : \mathbb{E}[\Delta u/u] \geq 0.01\%$ with $p \approx 8.3 \times 10^{-13}$, indicating the cross term is negligible at this scale.

G.3 Quantitative Impact of the Cross Term

Across $n = 20$ samples on a shared stochastic DDPM trajectory, we observe that the relative contribution of the cross term is extremely small.

Specifically, the mean percent change

$$\frac{u_{\mathrm{withX}} - u_{\mathrm{noX}}}{u_{\mathrm{noX}}}$$

lies in the range

$$10^{-3}\% \text{ to } 5 \times 10^{-3}\%,$$

with the maximum observed per-sample contribution below $3 \times 10^{-2}\%$.

We further test whether the mean contribution exceeds a practically meaningful tolerance. For a threshold of $\epsilon = 0.01\%$, we conduct a one-sided t-test:

$$H_0 :\; \mathbb{E}[\Delta u/u_{\mathrm{noX}}] \geq \epsilon, \qquad H_1 :\; \mathbb{E}[\Delta u/u_{\mathrm{noX}}] < \epsilon,$$

and reject H_0 with $p \approx 8 \times 10^{-13}$. This confirms that the cross term is statistically and practically negligible.

G.4 Implication for FLARE

These experiments demonstrate that even under a full-Hessian Laplace posterior and explicit Jacobian evaluation, the parameter cross-covariance term contributes less than $10^{-2}\%$ to the epistemic uncertainty. Dropping this term therefore has negligible numerical impact while substantially simplifying the recursion and reducing computational cost, justifying the epistemic propagation used by FLARE.

Table 3: Effect size of the Monte Carlo cross-covariance term. Statistics are computed over $N = 80$ stochastic DDPM trajectories. All values are reported in percent units.

Dataset	Mean (%)	Std (%)	Max (%)	SE (%)	df
Sine	0.00338	0.00440	0.02220	0.00049	79
Chirp	0.00495	0.00610	0.02840	0.00068	79

Table 4: One-sided t-tests assessing whether the mean cross-covariance contribution μ is below a user-chosen practical threshold τ. Negative t-statistics indicate evidence that $\mu < \tau$.

τ Threshold	Sine t	Sine p	Conclusion	Chirp t	Chirp p	Conclusion
0.005%	-3.29	0.9993	$\mu < 0.005\%$	-0.10	0.539	$\mu < 0.005\%$
0.01%	-13.45	1.0000	$\mu < 0.01\%$	-7.43	1.000	$\mu < 0.01\%$