A UNIFIED FRAMEWORK FOR BAYESIAN DATA ASSIMILATION WITH GENERATIVE MODELS AND OBSERVATION INTERPOLANTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian data assimilation combines model forecasts with noisy observations, but sampling high-dimensional, non-Gaussian posteriors remains challenging. We introduce an observation-interpolant framework that turns pretrained stochastic interpolant, flow matching, and diffusion models into posterior samplers without retraining. Conditioning the interpolant path on observations yields a shared likelihood-score correction to the drift or velocity, unifying stochastic and deterministic posterior sampling. The resulting SDEs and ODEs sample the exact posterior when the intermediate likelihood score is known. For practical computation, we approximate this score using a closed-form Gaussian surrogate with a bias-corrected mean and covariance inflated by the model's source covariance. Jacobian-free and ensemble-shared approximations make the method tractable in high dimensions. We evaluate the framework on linear–Gaussian dynamics, stochastic two-dimensional Navier–Stokes, and urban airflow with up to $\mathcal{O}(10^4)$ degrees of freedom.

1 INTRODUCTION

Data assimilation (DA) estimates unknown states of dynamical systems by combining prior model predictions with noisy observations. Bayesian DA combines these sources of information in a posterior distribution, which can be approximated by an ensemble of samples. Ensemble Kalman filters provide an efficient approximation but rely on Gaussian assumptions, whereas particle filters are more general but typically require many more samples (Evensen et al., 2022). Neural network surrogates can be used to accelerate the forward model (Mücke et al., 2024; Cheng et al., 2023; Chen et al., 2023), but the learned surrogates themselves are typically deterministic and therefore require a priori model-error distributions that are hard to specify for high-dimensional and nonlinear stochastic systems.

Generative models offer an alternative, by learning to sample directly from a distribution of interest. Two flow-based families dominate: stochastic interpolants (SIs) (Albergo et al., 2025; Albergo & Vanden-Eijnden, 2023; Chen et al., 2024), whose probability paths admit both ODE and SDE samplers, and flow matching (FM) (Lipman et al., 2023), which is typically sampled through a deterministic ordinary differential equation (ODE). Denoising diffusion (Ho et al., 2020; Song et al., 2021) corresponds to SDE sampling of an FM-type model. In DA, and posterior sampling in general, they are trained on the prior and modified at sampling time to incorporate observations. All three are increasingly applied to physical forecasting, yet methods for turning them into *posterior* samplers have been developed model by model, each tied to whether the generator is stochastic or deterministic.

A unified view. For data assimilation, we view stochastic interpolants, flow matching, and denoising diffusion as different samplers within one posterior-sampling framework, rather than distinct methods. Each can be described by an interpolant that connects a tractable source distribution to the transition prior $p(\boldsymbol{x}_1 \mid \boldsymbol{x}_0)$ (Albergo et al., 2025; Lipman et al., 2023; Ma et al., 2024; Holderrieth et al., 2025). Given a pretrained model that samples from this prior, we modify its sampling dynamics to target $p(\boldsymbol{x}_1 \mid \boldsymbol{y}, \boldsymbol{x}_0)$ without retraining. Conditioning the interpolant on observations gives a common likelihood-score correction to its velocity or drift (Proposition B.1). Different choices of diffusion coefficient yield deterministic or stochastic samplers. Adjusting the drift accordingly preserves

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

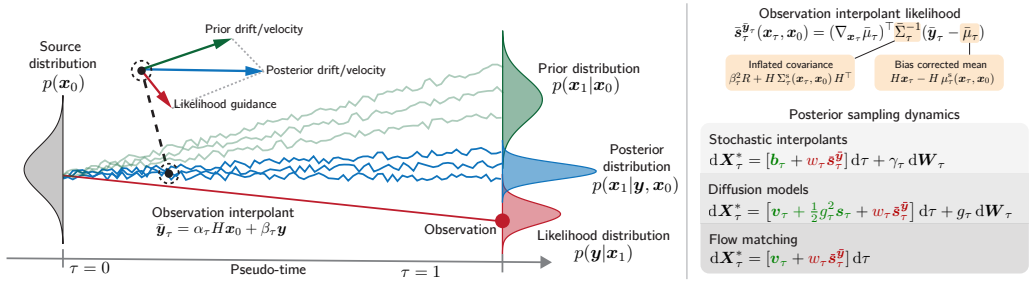


Figure 1: **Unified posterior sampling for stochastic interpolants, flow matching, and diffusion models.** *Left:* interpolating the observation yields closed-form guidance that steers the pretrained prior flow $p(\boldsymbol{x}_1 \mid \boldsymbol{x}_0)$ toward the posterior $p(\boldsymbol{x}_1 \mid \boldsymbol{y}, \boldsymbol{x}_0)$, without retraining. *Right:* the weighted guidance w_τ shifts the prior drift to the posterior drift.

the conditioned interpolant's distribution at each pseudo-time, so all resulting samplers target the same posterior. In practice, one closed-form Gaussian surrogate approximates the likelihood score for all three. The samplers differ only in their diffusion coefficient and likelihood-score weighting (Figure 1).

Contributions. Our contributions are summarized as follows:

- **Unified posterior-sampler family.** One guidance term with weight $w_\tau = \kappa_\tau + \frac{1}{2}g_\tau^2$ turns any interpolant-based generative model into a family of posterior samplers.
- **Observation-interpolant likelihood surrogate.** We interpolate the observation with the same schedules as the state and derive its exact conditional mean and covariance, including source-noise bias and covariance inflation. A Gaussian approximation then gives a common likelihood-score surrogate for all three samplers.
- **Scalable covariance approximations.** A Jacobian-free covariance and an ensemble-shared Jacobian make the sparse-observation regime feasible at field scale.

2 Related Work

Flow-based generative models in physics applications. Flow-based generative models are increasingly used for probabilistic physical forecasting, sampling the next state given the current one via noise around neural-operator predictions (Lippe et al., 2023), latent autoregressive dynamics with a diffusion decoder (Gao et al., 2024), or direct conditional sampling with denoising diffusion (Kohl et al., 2026), flow matching (Fotiadis et al., 2025), and stochastic interpolants (Chen et al., 2024; Mücke & Sanderse, 2026).

Posterior sampling and data assimilation with generative models. Generative models are increasingly used to solve inverse problems (Daras et al., 2024). A common approach is to train a flow-based model to sample from the prior and, at inference time, condition its dynamics on sparse and noisy observations by modifying the velocity or drift through the score decomposition, $\nabla \log p_\tau(\boldsymbol{x} \mid \boldsymbol{y}) = \nabla \log p_\tau(\boldsymbol{x}) + \nabla \log p_\tau(\boldsymbol{y} \mid \boldsymbol{x})$. This principle underlies posterior-sampling methods based on diffusion models (Song et al., 2021; Chung et al., 2023; Bao et al., 2024; Qu et al., 2024; Rozet & Louppe, 2023), flow matching (Pokle et al., 2024; Yan et al., 2025; Martin et al., 2025), and stochastic interpolants (Chen et al., 2025). These methods differ mainly in how they approximate the likelihood score. Most use a Tweedie-type estimate of the terminal sample and differentiate through it to obtain the score (Song et al., 2021; Rozet & Louppe, 2023; Qu et al., 2024; Pokle et al., 2024; Martin et al., 2025), while others combine related approximations with Monte Carlo estimation (Chen et al., 2025). They also differ in their treatment of the likelihood covariance, using either the original covariance (Chung et al., 2023) or an inflated one (Song et al., 2023). Closest to our work is Yan et al. (2025), who interpolate the observations within a flow-matching formulation. In contrast, we develop a unified framework for diffusion models, flow matching, and stochastic interpolants. We interpolate the observations using the same schedules as the state, rather than evaluating the original likelihood at a single estimated terminal state. Deriving the conditional source-noise moments then

gives a bias-corrected mean and inflated covariance for a common Gaussian likelihood surrogate across all three samplers.

3 PRELIMINARIES

Bayesian inverse problems and data assimilation For $n = 1, \dots, N$, consider the Markov model (Carrassi et al., 2018; Evensen et al., 2022):

$$\boldsymbol{u}^n \sim p\big(\boldsymbol{u}^n \,\big|\, \boldsymbol{u}^{n-1}\big), \quad \boldsymbol{y}^n = H\boldsymbol{u}^n + \boldsymbol{\eta}^n, \qquad \boldsymbol{\eta}^n \sim p_{\boldsymbol{\eta}}, \tag{1}$$

where $\boldsymbol{u}^n \in \mathbb{R}^{N_{\boldsymbol{u}}}$ is the state, $\boldsymbol{y}^n \in \mathbb{R}^{N_{\boldsymbol{y}}}$ the observation, and $H \in \mathbb{R}^{N_{\boldsymbol{y}} \times N_{\boldsymbol{u}}}$ the observation matrix. The transition prior and noise-induced likelihood define the *one-step posterior*:

$$p\big(\boldsymbol{u}^n \,\big|\, \boldsymbol{y}^n, \boldsymbol{u}^{n-1}\big) = \frac{p(\boldsymbol{y}^n \,|\, \boldsymbol{u}^n)\, p\big(\boldsymbol{u}^n \,\big|\, \boldsymbol{u}^{n-1}\big)}{p(\boldsymbol{y}^n \,|\, \boldsymbol{u}^{n-1})}. \tag{2}$$

Feeding each posterior sample into the next step gives an autoregressive assimilation procedure.

Interpolation-based generative modelling To model the transition prior, let $\boldsymbol{x}_0 = \boldsymbol{u}^{n-1}$ be the conditioning state and $\boldsymbol{x}_1 = \boldsymbol{u}^n$ the target. We use the general interpolant

$$\boldsymbol{x}_\tau = \alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1 + \sigma_\tau \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \qquad \tau \in [0, 1], \tag{3}$$

where τ is pseudo-time, $\boldsymbol{\varepsilon}$ is independent of $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, and $\beta_0 = 0$, $\beta_1 = 1$, and $\alpha_1 = \sigma_1 = 0$. The schedule σ_τ specifies the noise standard deviation conditional on both endpoints.

Flow matching (FM) typically sets $\alpha_\tau \equiv 0$ and $\sigma_0 = 1$, connecting a standard Gaussian source to $p(\boldsymbol{x}_1 \,|\, \boldsymbol{x}_0)$. Its marginal velocity is learned through (Lipman et al., 2023; Liu et al., 2023)

$$\mathcal{L}_{\mathrm{vel}}(\theta) = \mathbb{E}_{\tau, \boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{\varepsilon}} \left[\left\| \boldsymbol{v}_\tau^\theta(\boldsymbol{x}_\tau, \boldsymbol{x}_0) - \left(\dot{\beta}_\tau \boldsymbol{x}_1 + \dot{\sigma}_\tau \boldsymbol{\varepsilon} \right) \right\|^2 \right], \tag{4}$$

using data pairs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, $\tau \sim \mathcal{U}(0, 1)$, and Gaussian noise. Sampling then consists of drawing $\boldsymbol{X}_0$ from the source and integrating the ODE from $\tau = 0$ to $\tau = 1$:

$$\text{FM-ODE:} \qquad \mathrm{d}\boldsymbol{X}_\tau = \boldsymbol{v}_\tau(\boldsymbol{X}_\tau, \boldsymbol{x}_0)\, \mathrm{d}\tau, \qquad \boldsymbol{X}_0 \sim \mathcal{N}(0, \boldsymbol{I}). \tag{5}$$

With the exact velocity and well-posed dynamics, $\boldsymbol{X}_1 \sim p(\boldsymbol{x}_1 \,|\, \boldsymbol{x}_0)$. We omit the learned parameters θ below.

Diffusion models (DM) typically learn the marginal score $\boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) := \nabla_{\boldsymbol{x}} \log p_\tau(\boldsymbol{x} \,|\, \boldsymbol{x}_0)$ on Gaussian-source paths (Ho et al., 2020; Song et al., 2021). However, for the general interpolant (3), the score and velocity satisfy

$$\boldsymbol{v}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) = \frac{\dot{\beta}_\tau}{\beta_\tau} \boldsymbol{x} + \left(\dot{\alpha}_\tau - \frac{\dot{\beta}_\tau}{\beta_\tau} \alpha_\tau \right) \boldsymbol{x}_0 + \kappa_\tau \boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0), \qquad \kappa_\tau := \frac{\dot{\beta}_\tau}{\beta_\tau} \sigma_\tau^2 - \sigma_\tau \dot{\sigma}_\tau, \tag{6}$$

for differentiable schedules with $\beta_\tau, \sigma_\tau > 0$ on $(0, 1)$ (Appendix A). Thus, FM and DM are equivalent, and a velocity trained via flow matching can be used to sample via the SDE:

$$\text{DM-SDE:} \quad \mathrm{d}\boldsymbol{X}_\tau = \big[\boldsymbol{v}_\tau(\boldsymbol{X}_\tau, \boldsymbol{x}_0) + \tfrac{1}{2} g_\tau^2 \boldsymbol{s}_\tau(\boldsymbol{X}_\tau, \boldsymbol{x}_0) \big]\, \mathrm{d}\tau + g_\tau\, \mathrm{d}\boldsymbol{W}_\tau, \quad \boldsymbol{X}_0 \sim \mathcal{N}(0, \boldsymbol{I}). \tag{7}$$

Here, $\boldsymbol{W}_\tau$ is standard Brownian motion, and $g_\tau > 0$ is the diffusion term. Unlike σ_τ, g_τ controls sampling dynamics rather than interpolant marginals; setting $g_\tau = 0$ recovers the FM ODE. Our labels FM-ODE and DM-SDE distinguish these sampling choices, independently of which field was trained.

Stochastic interpolants (SI) can also connect the conditioning state directly to the transition prior. Following Chen et al. (2024), we set $\alpha_0 = 1$ and $\sigma_0 = 0$ and replace the Gaussian noise in (3) by the Brownian source $\gamma_\tau \boldsymbol{W}_\tau$. Since $\boldsymbol{W}_\tau \sim \mathcal{N}(0, \tau \boldsymbol{I})$, this preserves the interpolant's one-time marginals when $\sigma_\tau = \gamma_\tau \sqrt{\tau}$. In this setting, one can learn the SDE drift directly via:

$$\mathcal{L}_{\mathrm{drift}}(\theta) = \mathbb{E}_{\tau, \boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{\varepsilon}} \left[\left\| \boldsymbol{b}_\tau^\theta(\boldsymbol{x}_\tau, \boldsymbol{x}_0) - \left(\dot{\alpha}_\tau \boldsymbol{x}_0 + \dot{\beta}_\tau \boldsymbol{x}_1 + \dot{\gamma}_\tau \sqrt{\tau}\, \boldsymbol{\varepsilon} \right) \right\|^2 \right]. \tag{8}$$

Here, $\dot{\gamma}_\tau\sqrt{\tau} = \dot{\sigma}_\tau - \sigma_\tau/(2\tau)$, so this objective is fully determined by the schedules in (3). Sampling proceeds by integrating

$$\text{SI-SDE:} \qquad \mathrm{d}\boldsymbol{X}_\tau = \boldsymbol{b}_\tau(\boldsymbol{X}_\tau, \boldsymbol{x}_0)\,\mathrm{d}\tau + \gamma_\tau\,\mathrm{d}\boldsymbol{W}_\tau, \qquad \boldsymbol{X}_0 = \boldsymbol{x}_0, \tag{9}$$

from $\tau = 0$ to $\tau = 1$. Note that γ_τ is natively chosen as the diffusion term. The SI drift can be decomposed into velocity and score, $\boldsymbol{b}_\tau = \boldsymbol{v}_\tau + \frac{1}{2}\gamma_\tau^2\boldsymbol{s}_\tau$. It therefore shares the velocity–score structure of the preceding samplers.

4 Unified Posterior Sampler

We use the general interpolant of Section 3 to sample the one-step posterior $p(\boldsymbol{x}_1 \mid \boldsymbol{y}, \boldsymbol{x}_0)$ without retraining the prior model by conditioning the drift/velocity on the observations.

Define the intermediate likelihood

$$\Phi_\tau^{\boldsymbol{y}}(\boldsymbol{x}, \boldsymbol{x}_0) := p(\boldsymbol{y} \mid \boldsymbol{X}_\tau = \boldsymbol{x}, \boldsymbol{x}_0) = \mathbb{E}[p(\boldsymbol{y} \mid \boldsymbol{x}_1) \mid \boldsymbol{X}_\tau = \boldsymbol{x}, \boldsymbol{x}_0]\,, \tag{10}$$

and its score $\boldsymbol{s}_\tau^{\boldsymbol{y}} := \nabla_{\boldsymbol{x}} \log \Phi_\tau^{\boldsymbol{y}}$. Conditioning preserves the interpolant's schedules and anchor, while shifting its score and velocity by $\boldsymbol{s}_\tau^{\boldsymbol{y}}$ and $\kappa_\tau \boldsymbol{s}_\tau^{\boldsymbol{y}}$, respectively (Proposition B.1). Applying marginal-preserving ODE/SDE dynamics to this conditioned path (Albergo et al., 2025; Ma et al., 2024) yields the following unified posterior sampler:

Theorem 4.1 (Unified posterior sampler)**.** *Let the prior model define an interpolant path with velocity $\boldsymbol{v}_\tau$ and score $\boldsymbol{s}_\tau$ whose terminal law is $p(\boldsymbol{x}_1 \mid \boldsymbol{x}_0)$, and let $\Phi_\tau^{\boldsymbol{y}}$ be the intermediate likelihood* (10)*. Assume the posterior-path density p_τ^* is positive and C^2 on $(0,1) \times \mathbb{R}^{N_u}$. Let $g_\tau \geq 0$ be a chosen sampler diffusion, equal to γ_τ for the native SI sampler and zero for the FM ODE, measurable with $g \in L^2([0,1])$, and assume that the SDE below and its Fokker–Planck equation are well posed with unique solutions for the stated initial law. Then*

$$\mathrm{d}\boldsymbol{X}_\tau^* = \underbrace{\left[\boldsymbol{v}_\tau(\boldsymbol{X}_\tau^*, \boldsymbol{x}_0) + \tfrac{1}{2}g_\tau^2\,\boldsymbol{s}_\tau(\boldsymbol{X}_\tau^*, \boldsymbol{x}_0)\right]}_{\text{prior drift } \boldsymbol{b}_\tau}\mathrm{d}\tau + \underbrace{\left(\kappa_\tau + \tfrac{1}{2}g_\tau^2\right)}_{=:\,w_\tau}\boldsymbol{s}_\tau^{\boldsymbol{y}}(\boldsymbol{X}_\tau^*, \boldsymbol{x}_0)\,\mathrm{d}\tau + g_\tau\,\mathrm{d}\boldsymbol{W}_\tau, \tag{11}$$

initialised from the source law $p_0^{\boldsymbol{y}} = p_0$ (unchanged by conditioning because $\beta_0 = 0$), has marginals $p_\tau^{\boldsymbol{y}}(\boldsymbol{x} \mid \boldsymbol{x}_0) \propto \Phi_\tau^{\boldsymbol{y}}(\boldsymbol{x}, \boldsymbol{x}_0)\,p_\tau(\boldsymbol{x})$ for all $\tau \in [0,1]$, and in particular generates $\boldsymbol{X}_1^* \sim p(\boldsymbol{x}_1 \mid \boldsymbol{y}, \boldsymbol{x}_0)$.

See Appendix B for a proof. While analogous results exist for individual model families, to our knowledge this is the first unified posterior-sampling result explicitly covering stochastic interpolants, flow matching, and diffusion models. As a result, given a trained drift, $\boldsymbol{b}_\tau$, velocity, $\boldsymbol{v}_\tau$, or score $\boldsymbol{s}_\tau$, one can sample from the posterior distribution via one of the following ODEs/SDEs:

$$\begin{aligned}
&\text{SI-SDE } (g_\tau = \gamma_\tau): && \mathrm{d}\boldsymbol{X}_\tau^* = [\boldsymbol{b}_\tau + w_\tau \boldsymbol{s}_\tau^{\boldsymbol{y}}]\,\mathrm{d}\tau + \gamma_\tau\,\mathrm{d}\boldsymbol{W}_\tau, && \boldsymbol{X}_0^* = \boldsymbol{x}_0,\\
&\text{DM-SDE } (g_\tau > 0): && \mathrm{d}\boldsymbol{X}_\tau^* = \left[\boldsymbol{v}_\tau + \tfrac{1}{2}g_\tau^2\boldsymbol{s}_\tau + w_\tau \boldsymbol{s}_\tau^{\boldsymbol{y}}\right]\mathrm{d}\tau + g_\tau\,\mathrm{d}\boldsymbol{W}_\tau, && \boldsymbol{X}_0^* \sim \mathcal{N}(0, \boldsymbol{I}),\\
&\text{FM-ODE } (g_\tau = 0): && \mathrm{d}\boldsymbol{X}_\tau^* = [\boldsymbol{v}_\tau + \kappa_\tau \boldsymbol{s}_\tau^{\boldsymbol{y}}]\,\mathrm{d}\tau, && \boldsymbol{X}_0^* \sim \mathcal{N}(0, \boldsymbol{I}).
\end{aligned}$$

4.1 A closed-form likelihood score from observation interpolation

The posterior drift in Eq. (11) depends on the likelihood score $\boldsymbol{s}_\tau^{\boldsymbol{y}}$ at every intermediate state, yet the conditional expectation that defines it in Eq. (10) is intractable. Estimating it by Monte Carlo or by differentiating through simulated trajectories becomes prohibitive when repeated at every sampler step. We remove this barrier by interpolating the observation alongside the state, so that at every pseudo-time the sampler conditions on an observation matched to its current state:

$$\bar{\boldsymbol{y}}_\tau(\boldsymbol{x}_0, \boldsymbol{y}) = \alpha_\tau H\boldsymbol{x}_0 + \beta_\tau \boldsymbol{y}, \qquad \bar{\boldsymbol{y}}_0 = \alpha_0 H\boldsymbol{x}_0, \quad \bar{\boldsymbol{y}}_1 = \boldsymbol{y}. \tag{12}$$

which reduces to $\beta_\tau \boldsymbol{y}$ for FM and DM ($\alpha_\tau \equiv 0$). This scales the measured endpoint contribution by β_τ and includes the source contribution $\alpha_\tau H\boldsymbol{x}_0$. We can then replace the likelihood score from (11) with the interpolated likelihood score, $\bar{\boldsymbol{s}}_\tau^{\bar{\boldsymbol{y}}_\tau} := \nabla_{\boldsymbol{x}_\tau} \log \bar{\Phi}_\tau^{\bar{\boldsymbol{y}}_\tau} = \nabla_{\boldsymbol{x}_\tau} \log p(\bar{\boldsymbol{y}} \mid \boldsymbol{X}_\tau = \boldsymbol{x}, \boldsymbol{x}_0)$.

Linearity of H lets us write $\bar{\boldsymbol{y}}_\tau$ as $H\boldsymbol{x}_\tau$ plus a residual containing both observation and source noise. Its exact conditional moments follow from the prior score and its derivative. They differ from the naive mean $H\boldsymbol{x}_\tau$ and covariance $\beta_\tau^2 R$ because the source noise is correlated with the intermediate state:

Lemma 4.2 (Interpolated-observation likelihood)**.** *Let $\boldsymbol{H}$ be linear with matrix H and $\boldsymbol{\eta} \sim \mathcal{N}(0, R)$ independent of $(\boldsymbol{x}_1, \boldsymbol{\varepsilon})$ given $\boldsymbol{x}_0$. Then, writing the zero-mean source noise process as $\boldsymbol{\chi}^{\mathrm{s}}_\tau := \sigma_\tau \boldsymbol{\varepsilon}$ reveals:*

$$\bar{\boldsymbol{y}}_\tau = H\boldsymbol{x}_\tau + \bar{\boldsymbol{\eta}}_\tau, \qquad \bar{\boldsymbol{\eta}}_\tau := \beta_\tau \boldsymbol{\eta} - H\boldsymbol{\chi}^{\mathrm{s}}_\tau, \tag{13}$$

and the conditional mean and covariance of $p(\bar{\boldsymbol{y}}_\tau \mid \boldsymbol{x}_0, \boldsymbol{x}_\tau)$ are

$$\bar{\mu}_\tau(\boldsymbol{x}_\tau, \boldsymbol{x}_0) = H\boldsymbol{x}_\tau - H\,\mu^{\mathrm{s}}_\tau(\boldsymbol{x}_\tau, \boldsymbol{x}_0), \quad \bar{\Sigma}_\tau(\boldsymbol{x}_\tau, \boldsymbol{x}_0) = \beta_\tau^2 R + H\,\Sigma^{\mathrm{s}}_\tau(\boldsymbol{x}_\tau, \boldsymbol{x}_0)\,H^\top, \tag{14}$$

with source moments $\mu^{\mathrm{s}}_\tau := \mathbb{E}[\boldsymbol{\chi}^{\mathrm{s}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0]$ and $\Sigma^{\mathrm{s}}_\tau := Cov(\boldsymbol{\chi}^{\mathrm{s}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0)$.

Lemma 4.3 (Source conditional moments)**.** *For the path* (3)*, the conditional moments of the source process are*

$$\mu^{\mathrm{s}}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) = -\sigma_\tau^2\, \boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0), \qquad \Sigma^{\mathrm{s}}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) = \sigma_\tau^2\, \boldsymbol{I} + \sigma_\tau^4\, \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0). \tag{15}$$

Both proofs are in Appendix C. The source mean and covariance for the SI ($\boldsymbol{\chi}^{\mathrm{s}}_\tau = \gamma_\tau \boldsymbol{W}_\tau$, $\sigma_\tau = \gamma_\tau\sqrt{\tau}$) are computed as follows:

$$\mu^{\mathrm{s}}_\tau = -\gamma_\tau^2 \tau A_\tau(\beta_\tau \boldsymbol{b}_\tau - \boldsymbol{c}_\tau)\,, \qquad \Sigma^{\mathrm{s}}_\tau = \gamma_\tau^2 \tau \boldsymbol{I} + \gamma_\tau^4 \tau^2 A_\tau \left(\beta_\tau \nabla_{\boldsymbol{x}} \boldsymbol{b}_\tau - \dot{\beta}_\tau \boldsymbol{I}\right),$$

A_τ and $\boldsymbol{c}_\tau$ defined in (29). The FM/DM score is recovered from the velocity through (28).

4.2 Practical approximations

Gaussian likelihood. The moments in Lemmas 4.2–4.3 are exact for an exact prior score, but the conditional likelihood is not necessarily Gaussian. However, we approximate it by a Gaussian with these moments and differentiate through its mean, holding its covariance fixed:

$$\bar{\boldsymbol{s}}^{\bar{\boldsymbol{y}}_\tau}_\tau(\boldsymbol{x}_\tau, \boldsymbol{x}_0) = (\nabla_{\boldsymbol{x}_\tau} \bar{\mu}_\tau)^\top \bar{\Sigma}_\tau^{-1}(\bar{\boldsymbol{y}}_\tau - \bar{\mu}_\tau), \qquad \bar{\Sigma}_\tau = \beta_\tau^2 R + H\Sigma^{\mathrm{s}}_\tau H^\top. \tag{16}$$

This closure is exact in the linear–Gaussian regime. Section 5 evaluates its accuracy empirically. Compared with the naive likelihood $\mathcal{N}(H\boldsymbol{x}_\tau, \beta_\tau^2 R)$, the mean correction $-H\mu^{\mathrm{s}}_\tau$ removes the expected source contribution, while the additional covariance captures its residual uncertainty. Retaining prior correlations allows observations to inform unobserved degrees of freedom, particularly under sparse sensing.

Jacobian-free approximation. Dropping the score Jacobian consistently from the mean Jacobian and covariance gives $\bar{\Sigma}_\tau \approx \beta_\tau^2 R + \sigma_\tau^2 H H^\top$ and $(\nabla_{\boldsymbol{x}} \bar{\mu}_\tau)^\top \approx H^\top$. This approximation is accurate when $\sigma_\tau^2 \|\nabla^2_{\boldsymbol{x}} \log p_\tau\| \ll 1$ and is best suited to dense, informative observations. The covariance is state-independent and precomputable, requiring only an $N_{\boldsymbol{y}} \times N_{\boldsymbol{y}}$ solve per member beyond prior-model evaluations.

Ensemble-shared Jacobian. Sparse observations benefit from the prior correlations discarded above. Computing the required Jacobian projections separately costs $N_{\boldsymbol{y}}$ Jacobian–vector products per member per step. We instead evaluate them at the ensemble mean and share the resulting $N_{\boldsymbol{y}} \times N_{\boldsymbol{y}}$ factorisation across all E members, reducing the Jacobian work to $\mathcal{O}(N_{\boldsymbol{y}})$ network passes and one factorisation per step. Refreshing the shared Jacobian every k-th step further reduces its evaluation cost by a factor of k, while schedule coefficients are updated at every step.

Jacobian damping. For stability with strongly nonlinear priors, we introduce $\lambda \in [0, 1]$ through the parameterisation-specific covariances

$$\widetilde{\Sigma^{\mathrm{s}}_\tau}^{(\lambda),\mathrm{SI}} = \gamma_\tau^2 \tau \boldsymbol{I} + \gamma_\tau^4 \tau^2 A_\tau \left(\lambda \beta_\tau J_{\boldsymbol{b}_\tau} - \dot{\beta}_\tau \boldsymbol{I}\right), \qquad \widetilde{\Sigma^{\mathrm{s}}_\tau}^{(\lambda),\mathrm{FM}} = \sigma_\tau^2 \boldsymbol{I} + \lambda \sigma_\tau^4 \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau. \tag{17}$$

For SI, damping affects only the learned drift Jacobian; for FM, it scales the full score-Jacobian correction. Thus $\lambda = 0$ recovers the Jacobian-free covariance for FM, whereas SI retains the analytical term $-\gamma_\tau^4 \tau^2 A_\tau \dot{\beta}_\tau \boldsymbol{I}$.

4.3 Implementation

All three samplers need only the pre-trained model and an ODE/SDE solver, and share the assimilation loop in Algorithm 1. Recovered coefficients degenerate at $\tau = 0$ (in particular, the FM guidance

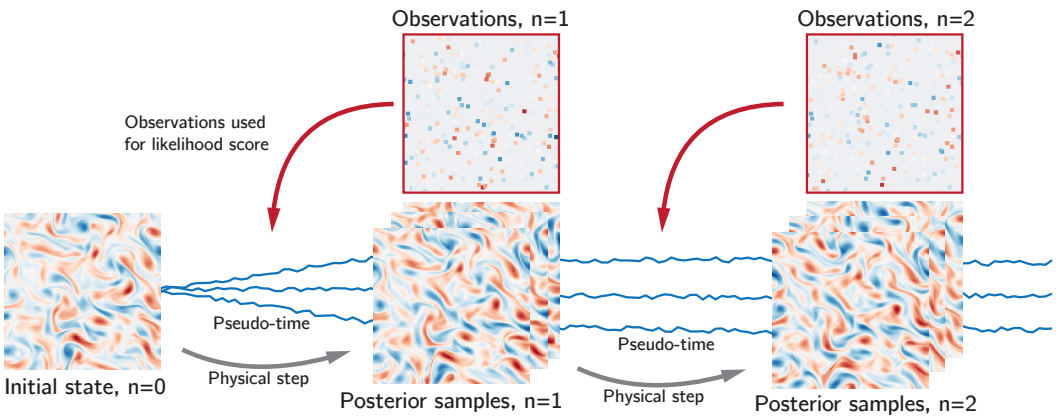


Figure 2: **Sequential assimilation with the unified sampler.** From a single initial state, $\boldsymbol{u}^0$, an ensemble of posterior samples are generated via the guided SDE/ODE (Algorithm 1). Each posterior member then initializes the next forecast, yielding an autoregressive rollout of (2).

representation is singular because $\beta_0 = 0$), so the first step uses the prior drift only and the correction is applied from $\tau = \Delta\tau$ onwards. Independent draws give an ensemble. Feeding each posterior sample $\boldsymbol{u}^n$ back before assimilating $\boldsymbol{y}^{n+1}$ advances the autoregressive assimilation in physical time n (see Figure 2). The solver and number of pseudo-time steps are chosen post-training, here forward Euler for the guided ODE and Euler–Maruyama for the SDEs. Schedules, architectures, and training details are in Appendix D.

Algorithm 1 Unified posterior assimilation step

Input: $\boldsymbol{x}_0 = \boldsymbol{u}^{n-1}$, $\boldsymbol{y} = \boldsymbol{y}^n$, trained model, schedules, diffusion g_τ, steps M
Initialise $\boldsymbol{X}^*$ from sampler source
Set $\Delta\tau = 1/M$
for $\tau = 0, \Delta\tau, \ldots, 1 - \Delta\tau$ **do**
  Form prior drift $\boldsymbol{b}_\tau$ (use native $\boldsymbol{b}_\tau$ for SI; otherwise $\boldsymbol{v}_\tau + \frac{1}{2} g_\tau^2 \boldsymbol{s}_\tau$)
  if $\tau = 0$ **then**
    Set likelihood correction $\boldsymbol{r}_\tau = \boldsymbol{0}$
  else
    Compute $\bar{\boldsymbol{y}}_\tau = \alpha_\tau H \boldsymbol{x}_0 + \beta_\tau \boldsymbol{y}$
    Compute source moments $\mu_\tau^{\mathrm{s}}, \Sigma_\tau^{\mathrm{s}}$
    Set $\bar{\mu}_\tau = H\boldsymbol{X}^* - H\mu_\tau^{\mathrm{s}}$
    Set $\bar{\Sigma}_\tau = \beta_\tau^2 R + H\Sigma_\tau^{\mathrm{s}} H^\top$
    Compute $\bar{\boldsymbol{s}}_\tau^{\bar{\boldsymbol{y}}_\tau} = (\nabla_{\boldsymbol{x}} \bar{\mu}_\tau)^\top \bar{\Sigma}_\tau^{-1} (\bar{\boldsymbol{y}}_\tau - \bar{\mu}_\tau)$
    Set $\boldsymbol{r}_\tau = (\kappa_\tau + \frac{1}{2} g_\tau^2) \bar{\boldsymbol{s}}_\tau^{\bar{\boldsymbol{y}}_\tau}$
  end if
  Draw noise $\boldsymbol{z} \sim \mathcal{N}(0, \boldsymbol{I})$
  $\boldsymbol{X}^* \leftarrow \boldsymbol{X}^* + (\boldsymbol{b}_\tau + \boldsymbol{r}_\tau)\,\Delta\tau + g_\tau \sqrt{\Delta\tau}\,\boldsymbol{z}$
end for
Return: $\boldsymbol{u}^n = \boldsymbol{X}^*$

5 Results

We evaluate three increasingly realistic problems: a linear–Gaussian system with an exact posterior, stochastically forced 2D Navier–Stokes, and a 2D slice of a 3D urban airflow. All cases use synthetic data, which means that we have direct access to the full ground truth states for evaluation. The samplers of Section 4.3 are reported as *Ours SI-SDE*, *DM-SDE*, and *FM-ODE*, using either a Jacobian-free approximation (*Jac-free*) or a shared Jacobian updated every tenth pseudo-step (*shared*). The field experiments are assimilated autoregressively with ensembles of 64 members.

Baselines, scenarios, and metrics. We compare with FlowDAS (Chen et al., 2025), SDA (Rozet & Louppe, 2023), D-Flow SGLD (Parikh et al., 2026; Ben-Hamu et al., 2024), Guided FM (Yan et al., 2025), and SURGE (Wei et al., 2026). All methods use the same prior trained SI or FM (DM is derived from the trained FM) to isolate the effect of the posterior sampling techniques rather than the quality of the training. See Appendix D for details on the neural network architecture and training. According to Wei et al. (2026), both SDA and FlowDAS perform better when combined with SURGE, so we report only the SURGE-enhanced versions. For Navier–Stokes, we additionally include an ensemble Kalman filter (Evensen et al., 2022) and bootstrap particle filter (Carrassi et al., 2018) using the true solver. Hyperparameters are tuned on separate short trajectories. Details are given in Appendix F.

We consider super-resolution from 32^2 or 16^2 to 128^2 (where the observation operator is simply subsampling grid point), and random sensors covering 5% or 1.5625% of the domain for the Navier–

Stokes case and 1.5625% or 0.78125% for the urban case, with Gaussian observation noise. Navier–Stokes uses all four settings and urban airflow the two sparse-sensor settings. For the field cases, we report RMSE, CRPS, and runtime per assimilation step, and additional metrics in the appendix. For the linear–Gaussian case, we report Gaussian KL divergence to the exact posterior. Metric definitions are given in Appendix E. For the Navier-Stokes and the Urban test case, the metrics are computed and aggregated over several autoregressive posterior sampling steps. All runs are performed an NVIDIA GeForce RTX 3090 with 24 GB memory.

5.1 Simple analytical test case

The first case is a 2D linear–Gaussian system with $N_{\boldsymbol{u}} = N_{\boldsymbol{y}} = 2$ of Appendix G: $\boldsymbol{u}^1 = \boldsymbol{u}^0 + \boldsymbol{\xi}$, $\boldsymbol{y}^1 = H\boldsymbol{u}^1 + \boldsymbol{\eta}$ with standard-normal noises and $H = R = \boldsymbol{I}$. The exact posterior, SI drift, and FM velocity are known in closed form, so no network is trained and the samplers are isolated from model error. We use 100,000 samples for all methods.

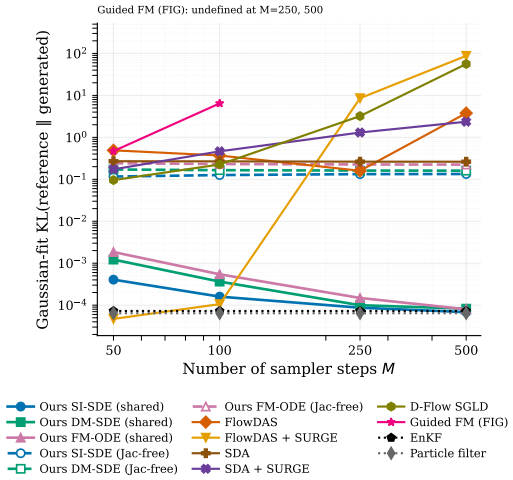


Figure 3: KL to the exact posterior vs. sampler steps M (five seeds).

Figure 3 shows the KL divergence to the exact posterior versus the number of sampler steps M. With the full inflated covariance, all three of our samplers converge to the exact posterior within sampling constraints (KL $\approx 10^{-4}$), matching the EnKF and particle filter in their ideal linear–Gaussian regime; the Jacobian-free variant performs significantly worse (KL ≈ 0.12–0.24). Among the baselines, only FlowDAS+SURGE also recovers the exact posterior. However, it is worth noting that with increasing number of SDE steps, it diverges, suggesting high sensitivity to hyperparameters. Guidance alone does not converge at any step count (SDA KL ≈ 0.27, FlowDAS ≥ 0.49), Guided FM collapses, and SDA+SURGE and D-Flow SGLD perform modestly at moderate M but diverge as M grows at fixed per-step hyperparameters. Density panels and a full table of results are in Appendix G.

5.2 Stochastic Navier–Stokes

The second case is an incompressible fluid on the torus $\mathcal{T}^2 = [0, 2\pi]^2$, governed by the stochastically forced two-dimensional vorticity equation,

$$\mathrm{d}\omega + \boldsymbol{v} \cdot \nabla\omega \,\mathrm{d}t = \nu\Delta\omega \,\mathrm{d}t - \alpha\omega \,\mathrm{d}t + \varepsilon \,\mathrm{d}\xi, \tag{18}$$

where $\boldsymbol{v} = \nabla^{\perp}\psi$ with $-\Delta\psi = \omega$ and $\mathrm{d}\xi$ a temporally white forcing on four Fourier modes. Trajectories are generated with JAX-CFD (Kochkov et al., 2021) on a 256^2 pseudo-spectral grid at $\nu = 10^{-3}$ and down-sampled to 128^2; we assimilate 15 consecutive steps at interval $\Delta t = 0.5$, observing vorticity through $\boldsymbol{y} = \boldsymbol{H}(\omega) + \boldsymbol{\eta}$ with $\boldsymbol{\eta} \sim \mathcal{N}(0, 0.05^2\boldsymbol{I})$ (data generation details in Appendix H).

Table 1 summarizes accuracy and cost, while Figure 4 shows reconstructions for the 5%-sensor case. In dense super-resolution settings, our samplers achieve the best RMSE and CRPS; at 32^2, the Jacobian-free covariance matches the shared approximation at roughly one-tenth the cost. Under sparse observations, however, covariance quality becomes critical: the shared inflated covariance reduces SI-SDE RMSE by 2–4$\times$ relative to the Jacobian-free variant, which, like Guided FM, over-smooths toward the prior mean. SI-SDE is therefore the strongest of our samplers in this regime. SDA+SURGE and D-Flow SGLD remain competitive, but SDA+SURGE collapses the ensemble, while D-Flow costs $\approx$700 s per step—about an order of magnitude more than SI-SDE (shared) and two orders more than the Jacobian-free samplers. SI-SDE (shared) also achieves the best or tied-best CRPS in three of four scenarios among methods using $E = 64$. Classical filters with $E = 64$ are not competitive; the $E = 1000$ EnKF reference attains the lowest sparse-scenario errors but requires the true solver and $>$100 s per step. Additional convergence, error, and enstrophy results are given in Appendix H.

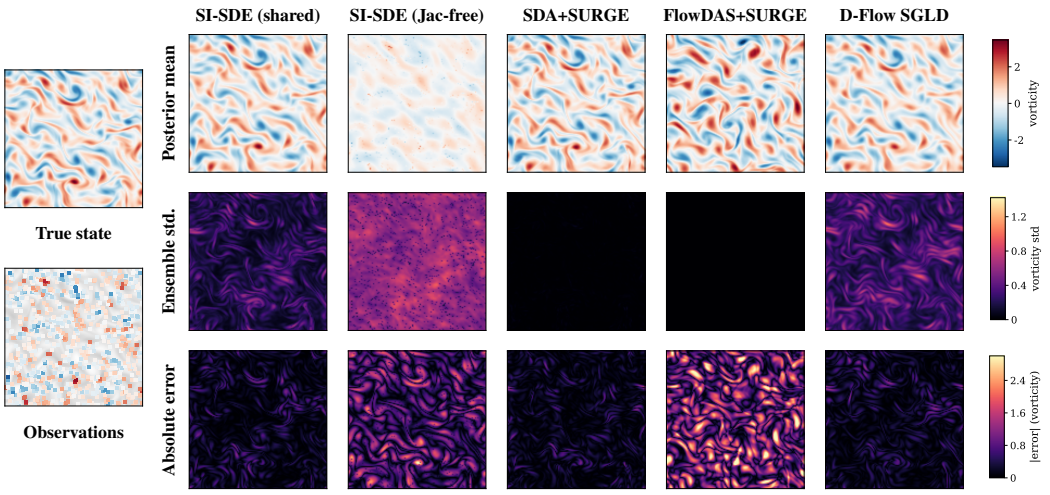


Figure 4: Navier–Stokes, sparse sensors at 5%: truth and observations (left), and the posterior mean, ensemble standard deviation, and absolute error at the final assimilation step. One colour scale per row. All methods and scenarios: Figures 7–10.

Table 1: Navier–Stokes: vorticity RMSE and CRPS across the four observation scenarios, with per-step cost (lower is better). $E = 64$ and $M = 100$ sampler steps. **Bold** marks the best value in each column, together with every value equal to it to 2 decimal places in unscaled units.

	Vorticity RMSE ($\times 10^{-1}$)				**Vorticity CRPS** ($\times 10^{-1}$)				**Cost (s/step)**			
	16^2	32^2	5%	$\frac{1}{64}$	16^2	32^2	5%	$\frac{1}{64}$	16^2	32^2	5%	$\frac{1}{64}$
Ours – ensemble-shared inflated covariance (Jacobian refreshed every $k = 10$ steps)												
Ours (SI-SDE)	**2.024**	0.667	1.731	3.854	**1.062**	**0.347**	**0.872**	2.073	41.3	77.7	68.4	41.8
Ours (DM-SDE)	2.162	**0.638**	3.424	6.834	1.154	**0.333**	1.823	3.830	87.2	164.1	144.3	88.6
Ours (FM-ODE)	2.102	**0.626**	3.828	7.038	**1.128**	**0.327**	2.051	3.945	89.1	164.9	144.7	89.3
Ours – Jacobian-free covariance												
Ours (SI-SDE)	2.765	0.670	6.969	8.500	1.518	0.355	3.952	4.818	**6.9**	7.1	7.1	6.9
Ours (DM-SDE)	2.759	0.672	6.756	8.362	1.501	0.351	3.754	4.681	13.0	13.3	13.2	13.0
Ours (FM-ODE)	2.733	0.686	7.026	8.460	1.492	0.359	3.897	4.732	13.0	13.2	13.1	13.0
FlowDAS + SURGE	5.066	1.540	8.632	11.896	3.853	1.164	6.742	9.426	32.2	33.5	33.2	33.7
SDA + SURGE	3.699	1.577	**1.616**	2.970	2.642	1.080	1.130	2.092	64.7	68.3	69.5	69.6
D-Flow SGLD	2.455	2.010	1.974	**2.313**	1.272	1.004	0.992	**1.179**	671.6	693.7	701.5	704.2
Guided FM (FIG)	2.986	0.780	7.231	8.461	1.623	0.415	4.082	4.756	13.6	14.0	14.1	14.2
EnKF	2.683	2.161	2.768	6.222	1.680	1.320	1.384	3.319	7.5	7.5	7.5	7.5
Particle filter	8.753	8.388	8.240	8.577	6.914	6.613	6.478	6.751	7.3	**6.8**	**6.8**	**6.8**
EnKF reference ($E = 1000$)	1.442	0.941	0.935	1.053	0.779	0.511	0.516	0.565	102.0	102.4	103.0	122.2

5.3 URBAN AIRFLOW

The third case considers coupled velocity (u, v, w) and temperature fields over an urban area with several buildings, simulated with the building-resolving LES code uDALES (Suter et al., 2022). The $128 \times 128 \times 128\,\mathrm{m}^3$ domain has $1\,\mathrm{m}$ resolution and is periodic in the horizontal directions. Simulations are spun up for 100 s and sampled every 2 s, retaining a horizontal slice at 2 m height. We initialize from a different held-out flow state and forecast 145 physical steps, assimilating sparse velocity observations every third step (49 updates). White noise with standard deviation 0.05 is added to all velocity observations. Both velocity and temperature are reconstructed, so temperature is inferred only through its coupling to the observed flow. Table 2 reports rollout-averaged results at both sensor densities, and Figure 5 shows temperature reconstructions at 0.78125% density. Further details are in Appendix I.

Table 2 reveals a split between observed and unobserved variables. The computational cost per step is omitted as timings are approximately the same as for the Navier-Stokes case. DM-SDE gives the lowest velocity RMSE and CRPS and best calibration (see spread-skill figures in Figure 21) among our samplers, whereas SI-SDE gives the lowest temperature RMSE and CRPS at both sensor densities. The shared covariance improves SI-SDE temperature RMSE from $0.38/0.43$ to $0.36/0.42$. D-Flow

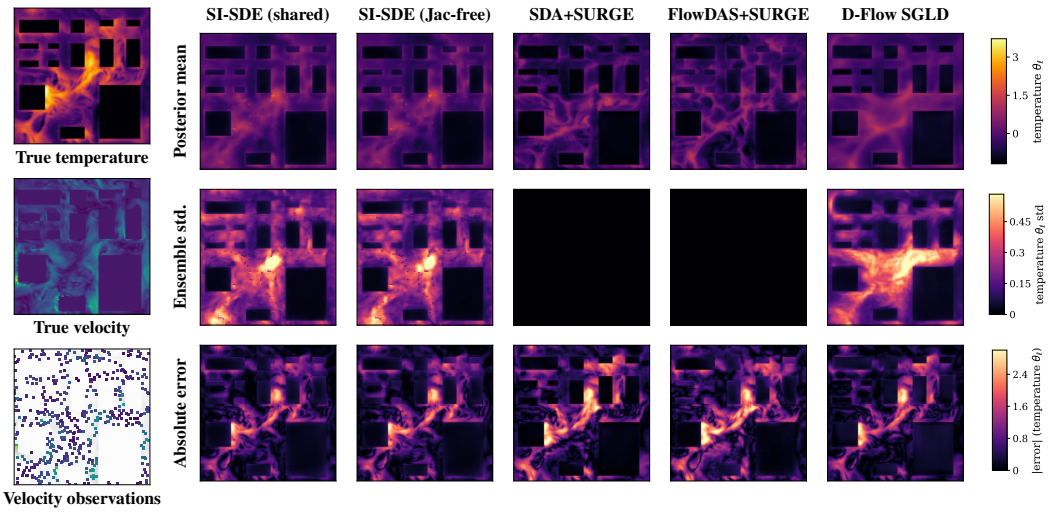


Figure 5: Urban airflow, sparse sensors at 0.78125%: true potential temperature, true velocity magnitude, and observed velocity sensors (left), followed by the posterior mean, ensemble standard deviation, and absolute error of the unobserved *temperature* at the final forecast step.

SGLD is the closest baseline, matching the best velocity or CRPS values but costing significantly more per step. Both SURGE ensembles are severely underdispersed. Over the long rollout, velocity errors decrease as the observations correct the mismatched initial state, while errors in the unobserved temperature accumulate (Appendix I). Figure 5 nevertheless shows that SI-SDE recovers the broad thermal-plume structure and places uncertainty around unresolved fine scales.

6 Conclusion

We introduced a unified posterior-sampling framework for stochastic interpolants, flow matching, and diffusion models based on a shared observation-interpolant likelihood correction. With the full covariance, all three samplers closely approximate the exact posterior in the linear–Gaussian test. On Navier–Stokes, our methods achieve the best accuracy under dense observations among all the generatibe posterior samplers, while the shared covariance substantially improves SI-SDE under sparse observations. In the urban case, DM-SDE performs best on observed velocity, while SI-SDE best limits error growth in temperature inferred from velocity alone. There, the Jacobian-free approximation remains close to the shared variant at 3–4$\times$ lower cost. Thus the best sampler and covariance approximation are problem-dependent: prior correlations are decisive for sparse Navier–Stokes, but provide only modest gains in the urban case.

Table 2: Urban (uDALES): rollout-averaged velocity RMSE (V-RMSE), temperature RMSE (T-RMSE), velocity CRPS (V-CRPS), and temperature CRPS (T-CRPS). Mean over 4–5 held-out trajectories (cell-dependent), $E = 64$, and $M = 50$ sampler steps. **Bold** marks the best value in each column, including ties to 2 decimal places.

	V-RMSE		T-RMSE		V-CRPS		T-CRPS	
	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{64}$	$\frac{1}{128}$
Ours – shared covariance (Jacobian refresh $k = 10$)								
Ours (SI-SDE)	0.60	0.67	**0.36**	**0.42**	0.33	0.38	**0.21**	**0.25**
Ours (DM-SDE)	**0.56**	**0.62**	0.45	0.55	**0.30**	**0.34**	0.28	0.36
Ours (FM-ODE)	0.59	0.66	0.40	0.51	0.33	0.37	0.25	0.33
Ours – Jacobian-free covariance								
Ours (SI-SDE)	0.61	0.68	0.38	0.43	0.34	0.38	0.23	0.26
Ours (DM-SDE)	0.57	0.63	0.47	0.56	**0.30**	**0.34**	0.29	0.37
Ours (FM-ODE)	0.60	0.66	0.42	0.52	0.33	0.37	0.26	0.34
FlowDAS + SURGE	0.88	0.89	0.53	0.55	0.66	0.66	0.40	0.42
SDA + SURGE	0.73	0.80	0.47	0.55	0.53	0.58	0.34	0.41
D-Flow SGLD	0.57	**0.62**	**0.36**	0.43	0.32	0.35	**0.21**	**0.25**
Guided FM (FIG)	0.59	0.65	0.41	0.51	0.33	0.36	0.25	0.33

Limitations. The exactness result assumes an exact prior and likelihood score, while the practical method uses a Gaussian approximation, stopped covariance gradients, and approximate Jacobians. These approximations can introduce bias, as seen for the Jacobian-free variant in the analytical and sparse Navier–Stokes cases. The current formulation also assumes linear Gaussian observations, and the shared covariance becomes costly as the observation dimension grows. Finally, one-step autoregressive assimilation neither reweights past states nor uses future observations, so model and approximation errors may accumulate over long rollouts. Extending the method to nonlinear observations, smoothing, and broader model mismatch remains future work.

Reproducibility statement

Theoretical claims and assumptions from Section 4 are formally proven in Appendices A, B, and C. Comprehensive experimental configurations—including architectures, metrics, baselines, and data generation—are detailed in Appendices D through I (including specific configurations in H.1). An anonymized supplementary code archive containing the core sampler library, test harnesses, and pre-trained priors is currently provided to assist the review process. Due to file size limits, this archive omits the training data and code, external solvers (JAX-CFD (Kochkov et al., 2021) and uDALES (Suter et al., 2022)), true-solver filters, and the urban case files. The entire codebase and data will be made available upon acceptance.

Ethics statement

The research presented here introduces a unified framework for Bayesian data assimilation using flow-based generative models. The methodology is evaluated exclusively on physical systems, specifically linear-Gaussian dynamics, two-dimensional Navier-Stokes equations, and urban airflow simulations. This work does not involve human subjects, sensitive personal datasets, or applications that risk generating harmful insights or propagating societal bias. The foundational contributions strictly advance physical forecasting methodologies and present no foreseeable negative societal impacts under ICLR ethics guidelines.

AI use statement

In this work, we used generative AI tools for formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, supporting qualitative and thematic data analysis, and interpreting results. We have not used generative AI for generating synthetic data sets, developing theoretical models, or cleaning datasets; translation is not applicable to this work.

Additionally, we used AI for creating scientific figures, suggesting experimental parameters, writing/editing code, creating artifacts, drafting text, analyzing literature, searching for information, editing for readability, identifying literature, and formatting references.

We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Michael S. Albergo and Eric Vanden-Eijnden. Building Normalizing Flows with Stochastic Interpolants. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=li7qeBbCR1t. arXiv:2209.15571.

Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic Interpolants: A Unifying Framework for Flows and Diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025. URL https://jmlr.org/papers/v26/23-1605.html. arXiv:2303.08797.

Feng Bao, Zezhong Zhang, and Guannan Zhang. An Ensemble Score Filter for Tracking High-Dimensional Nonlinear Dynamical Systems. *Computer Methods in Applied Mechanics and Engineering*, 432:117447, December 2024. doi: 10.1016/j.cma.2024.117447. arXiv:2309.00983.

Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-Flow: Differentiating through Flows for Controlled Generation. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 3462–3483, July 2024. URL https://proceedings.mlr.press/v235/ben-hamu24a.html. arXiv:2402.14017.

Alberto Carrassi, Marc Bocquet, Laurent Bertino, and Geir Evensen. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, 9(5): e535, September 2018. doi: 10.1002/wcc.535. arXiv:1709.02798.

Siyi Chen, Yixuan Jia, Qing Qu, He Sun, and Jeffrey A. Fessler. FlowDAS: A Stochastic Interpolant-based Framework for Data Assimilation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, 2025. URL https://openreview.net/forum?id=1nWqhiulqD. arXiv:2501.16642 [eess].

Yifan Chen, Mark Goldstein, Mengjian Hua, Michael S. Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Probabilistic Forecasting with Stochastic Interpolants and Föllmer Processes. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 6728–6756. PMLR, July 2024. URL https://proceedings.mlr.press/v235/chen24n.html. arXiv:2403.13724.

Yuming Chen, Daniel Sanz-Alonso, and Rebecca Willett. Reduced-order autodifferentiable ensemble Kalman filters. *Inverse Problems*, 39(12):124001, October 2023. ISSN 0266-5611. doi: 10.1088/1361-6420/acff14.

Sibo Cheng, Jianhua Chen, Charitos Anastasiou, Panagiota Angeli, Omar K. Matar, Yi-Ke Guo, Christopher C. Pain, and Rossella Arcucci. Generalised Latent Assimilation in Heterogeneous Reduced Spaces with Machine Learning Surrogate Models. *Journal of Scientific Computing*, 94 (1):11, 2023. ISSN 1573-7691. doi: 10.1007/s10915-022-02059-4.

Hyungjin Chung, Jeongsol Kim, Michael T. McCann, Marc L. Klasky, and Jong Chul Ye. Diffusion Posterior Sampling for General Noisy Inverse Problems. In *International Conference on Learning Representations (ICLR)*, 2023. URL http://arxiv.org/abs/2209.14687. arXiv:2209.14687 [stat.ML].

Giannis Daras, Hyungjin Chung, Chieh-Hsin Lai, Yuki Mitsufuji, Jong Chul Ye, Peyman Milanfar, Alexandros G. Dimakis, and Mauricio Delbracio. A Survey on Diffusion Models for Inverse Problems, September 2024. URL http://arxiv.org/abs/2410.00083. arXiv:2410.00083 [cs].

Geir Evensen, Femke C. Vossepoel, and Peter Jan van Leeuwen. *Data Assimilation Fundamentals: A Unified Formulation of the State and Parameter Estimation Problem*. Springer Textbooks in Earth Sciences, Geography and Environment. Springer International Publishing, Cham, 2022. ISBN 978-3-030-96708-6. doi: 10.1007/978-3-030-96709-3. URL https://library.oapen.org/handle/20.500.12657/54434.

Stathi Fotiadis, Noah D. Brenowitz, Tomas Geffner, Yair Cohen, Michael Pritchard, Arash Vahdat, and Morteza Mardani. Adaptive Flow Matching for Resolving Small-Scale Physics. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 17489–17521, July 2025. URL https://proceedings.mlr.press/v267/fotiadis25a.html. arXiv:2410.19814 [cs].

Han Gao, Sebastian Kaltenbach, and Petros Koumoutsakos. Generative learning for forecasting the dynamics of high-dimensional complex systems. *Nature Communications*, 15(1):8904, October 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-53165-w. URL https://www.nature.com/articles/s41467-024-53165-w.

Tilmann Gneiting and Adrian E. Raftery. Strictly Proper Scoring Rules, Prediction, and Estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising Diffusion Probabilistic Models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851. Curran Associates, Inc., December 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html. arXiv:2006.11239 [cs].

Peter Holderrieth, Marton Havasi, Jason Yim, Neta Shaul, Itai Gat, Tommi Jaakkola, Brian Karrer, Ricky T. Q. Chen, and Yaron Lipman. Generator Matching: Generative modeling with arbitrary Markov processes. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=RuP17cJtZo. arXiv:2410.20587 [cs.LG].

Dmitrii Kochkov, Jamie A. Smith, Peter Norgaard, Gideon Dresdner, Ayya Alieva, and Stephan Hoyer. JAX-CFD: Computational fluid dynamics in JAX. https://github.com/google/jax-cfd, 2021. GitHub repository.

Georg Kohl, Li-Wei Chen, and Nils Thuerey. Benchmarking Autoregressive Conditional Diffusion Models for Turbulent Flow Simulation. *Neural Networks*, 199:108641, July 2026. ISSN 0893-6080. doi: 10.1016/j.neunet.2026.108641. arXiv:2309.01745 [cs].

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t. arXiv:2210.02747 [cs].

Phillip Lippe, Bastiaan S. Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving Accurate Long Rollouts with Neural PDE Solvers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 67398–67433, 2023. doi: 10.52202/075280-2946. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d529b943af3dba734f8a7d49efcb6d09-Abstract-Conference.html. arXiv:2308.05732 [cs].

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z. arXiv:2209.03003 [cs].

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring Flow and Diffusion-based Generative Models with Scalable Interpolant Transformers. In *Computer Vision – ECCV 2024*, volume 15135 of *Lecture Notes in Computer Science*, pp. 23–40. Springer, 2024. doi: 10.1007/978-3-031-72980-5_2. arXiv:2401.08740 [cs].

Ségolène Martin, Anne Gagneux, Paul Hagemann, and Gabriele Steidl. PnP-Flow: Plug-and-Play Image Restoration with Flow Matching. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/708e58b0b99e3e62d42022b4564bad7a-Abstract-Conference.html. arXiv:2410.02423 [cs].

Nikolaj T. Mücke and Benjamin Sanderse. Physics-aware generative models for turbulent fluid flows through energy-consistent stochastic interpolants. *Computers & Fluids*, 314:107098, June 2026. doi: 10.1016/j.compfluid.2026.107098. arXiv:2504.05852 [cs].

Nikolaj T. Mücke, Sander M. Bohté, and Cornelis W. Oosterlee. The deep latent space particle filter for real-time data assimilation with uncertainty quantification. *Scientific Reports*, 14(1): 19447, August 2024. ISSN 2045-2322. doi: 10.1038/s41598-024-69901-7. URL https://www.nature.com/articles/s41598-024-69901-7. Publisher: Nature Publishing Group.

Meet Hemant Parikh, Yaqin Chen, and Jian-Xun Wang. D-Flow SGLD: Source-Space Posterior Sampling for Scientific Inverse Problems with Flow Matching, 2026. URL http://arxiv.org/abs/2602.21469. arXiv:2602.21469.

Ashwini Pokle, Matthew J. Muckley, Ricky T. Q. Chen, and Brian Karrer. Training-free Linear Image Inverses via Flows. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=PLIt3a4yTm. arXiv:2310.04432 [cs].

Yongquan Qu, Juan Nathaniel, Shuolin Li, and Pierre Gentine. Deep Generative Data Assimilation in Multimodal Setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 449–459, June 2024. doi: 10.1109/CVPRW63382.2024.00050. URL https://openaccess.thecvf.com/content/CVPR2024W/EarthVision/html/Qu_Deep_Generative_Data_Assimilation_in_Multimodal_Setting_CVPRW_2024_paper.html. arXiv:2404.06665 [cs].

François Rozet and Gilles Louppe. Score-based Data Assimilation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://openreview.net/forum?id=VUvLSnMZdX. arXiv:2306.10574.

Jiaming Song, Arash Vahdat, Morteza Mardani, and Jan Kautz. Pseudoinverse-Guided Diffusion Models for Inverse Problems. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://openreview.net/forum?id=9_gsMA8MRKQ.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations (ICLR)*, February 2021. URL https://openreview.net/forum?id=PxTIG12RRHS. arXiv:2011.13456 [cs].

Ivo Suter, Tom Grylls, Birgit S. Sützl, Sam O. Owens, Chris E. Wilson, and Maarten van Reeuwijk. uDALES 1.0: a large-eddy simulation model for urban environments. *Geoscientific Model Development*, 15(13):5309–5335, 2022. ISSN 1991-9603. doi: 10.5194/gmd-15-5309-2022. URL https://gmd.copernicus.org/articles/15/5309/2022/.

Lifu Wei, Yinuo Ren, Naichen Shi, and Yiping Lu. SURGE: Approximation and Training Free Particle Filter for Diffusion Surrogate, 2026. URL http://arxiv.org/abs/2605.18745. arXiv:2605.18745.

Yici Yan, Yichi Zhang, Xiangming Meng, and Zhizhen Zhao. FIG: Flow with Interpolant Guidance for Linear Inverse Problems. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=fs2Z2z3GRx.

A VELOCITY–SCORE DUALITY AND THE MARGINAL-PRESERVING SDE FAMILY

This appendix establishes the structural facts on which the model-agnostic method rests: for an interpolant path the velocity and the score determine each other in closed form (Proposition A.2), the probability-flow ODE belongs to a one-parameter family of SDEs sharing the same marginals (Proposition A.3), and both velocity and score can be read off the trained network for each model class (Section A.1). The first two are known (Albergo et al., 2025; Song et al., 2021; Ma et al., 2024). We state and prove them in the notation of Section 3 for completeness.

Throughout, consider the interpolant path of Eq. (3),

$$\boldsymbol{x}_\tau = \alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1 + \sigma_\tau \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}) \text{ independent of } \boldsymbol{x}_1, \tag{19}$$

with $\boldsymbol{x}_0$ deterministic, $\mathbb{E}\|\boldsymbol{x}_1\|^2 < \infty$, $\beta_\tau > 0$ for $\tau \in (0, 1]$, and schedules continuously differentiable with $\sigma_\tau > 0$ on the open interval $(0, 1)$. All identities below are stated on $(0, 1)$. Whenever the score is recovered from the velocity, we additionally require $\dot\beta_\tau \sigma_\tau - \dot\sigma_\tau \beta_\tau \neq 0$. (For stochastic interpolants $\sigma_\tau = \gamma_\tau \sqrt{\tau}$ vanishes and is not differentiable at $\tau = 0$. The endpoints are covered by the initial condition and the terminal constraint $\boldsymbol{x}_{\tau=1} = \boldsymbol{x}_1$, consistent with the hypotheses of Theorem 4.1.) Let p_τ denote the marginal density of $\boldsymbol{x}_\tau$ and $\boldsymbol{s}_\tau = \nabla_{\boldsymbol{x}} \log p_\tau$ its score.

Lemma A.1 (Tweedie identities for the interpolant path)**.** *Writing $\boldsymbol{m}_\tau := \alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1$, so that $\boldsymbol{x}_\tau = \boldsymbol{m}_\tau + \sigma_\tau \boldsymbol{\varepsilon}$ with $\sigma_\tau \boldsymbol{\varepsilon}$ Gaussian with covariance $\sigma_\tau^2 \boldsymbol{I}$ independent of $\boldsymbol{m}_\tau$,*

$$\mathbb{E}[\boldsymbol{m}_\tau \mid \boldsymbol{x}_\tau] = \boldsymbol{x}_\tau + \sigma_\tau^2 \, \boldsymbol{s}_\tau(\boldsymbol{x}_\tau), \tag{20}$$

$$\mathbb{E}[\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau] = -\sigma_\tau \, \boldsymbol{s}_\tau(\boldsymbol{x}_\tau), \tag{21}$$

$$\mathbb{E}[\boldsymbol{x}_1 \mid \boldsymbol{x}_\tau] = \tfrac{1}{\beta_\tau}\big(\boldsymbol{x}_\tau + \sigma_\tau^2 \boldsymbol{s}_\tau(\boldsymbol{x}_\tau) - \alpha_\tau \boldsymbol{x}_0\big), \tag{22}$$

$$\mathit{Cov}(\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau) = \boldsymbol{I} + \sigma_\tau^2 \, \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau(\boldsymbol{x}_\tau). \tag{23}$$

Proof. Equation (20) is Tweedie's formula for the additive-Gaussian-noise model $\boldsymbol{x}_\tau = \boldsymbol{m}_\tau + \sigma_\tau \boldsymbol{\varepsilon}$. Since $\boldsymbol{\varepsilon} = (\boldsymbol{x}_\tau - \boldsymbol{m}_\tau)/\sigma_\tau$, (21) follows, and $\boldsymbol{x}_1 = (\boldsymbol{m}_\tau - \alpha_\tau \boldsymbol{x}_0)/\beta_\tau$ gives (22). For (23), differentiate $\mathbb{E}[\boldsymbol{m}_\tau \mid \boldsymbol{x}_\tau = \boldsymbol{x}]$ in $\boldsymbol{x}$: on the one hand, by (20), $\nabla_{\boldsymbol{x}} \mathbb{E}[\boldsymbol{m}_\tau \mid \boldsymbol{x}] = \boldsymbol{I} + \sigma_\tau^2 \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau$. On the other, differentiating the conditional Gaussian mean directly gives the standard second-order Tweedie identity $\nabla_{\boldsymbol{x}} \mathbb{E}[\boldsymbol{m}_\tau \mid \boldsymbol{x}] = \sigma_\tau^{-2} \mathrm{Cov}(\boldsymbol{m}_\tau \mid \boldsymbol{x})$. Hence $\mathrm{Cov}(\boldsymbol{m}_\tau \mid \boldsymbol{x}) = \sigma_\tau^2 \boldsymbol{I} + \sigma_\tau^4 \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau$, and $\mathrm{Cov}(\boldsymbol{\varepsilon} \mid \boldsymbol{x}) = \sigma_\tau^{-2} \mathrm{Cov}(\boldsymbol{m}_\tau \mid \boldsymbol{x})$ (as $\boldsymbol{\varepsilon} = (\boldsymbol{x} - \boldsymbol{m}_\tau)/\sigma_\tau$ with $\boldsymbol{x}$ fixed) yields (23). □

Proposition A.2 (Velocity–score identity)**.** *The marginal velocity and the score of the path* (19) *satisfy the exact linear relation*

$$\boldsymbol{v}_\tau(\boldsymbol{x}) = \frac{\dot\beta_\tau}{\beta_\tau}\boldsymbol{x} + \left(\dot\alpha_\tau - \frac{\dot\beta_\tau\alpha_\tau}{\beta_\tau}\right)\boldsymbol{x}_0 + \sigma_\tau\left(\frac{\dot\beta_\tau}{\beta_\tau}\sigma_\tau - \dot\sigma_\tau\right)\boldsymbol{s}_\tau(\boldsymbol{x}), \tag{24}$$

where the score coefficient is the velocity–score coefficient $\kappa_\tau = \sigma_\tau(\frac{\dot\beta_\tau}{\beta_\tau}\sigma_\tau - \dot\sigma_\tau)$ *of Eq.* (6)*. When* $\dot\beta_\tau\sigma_\tau - \dot\sigma_\tau\beta_\tau \neq 0$*, this is equivalently*

$$\boldsymbol{s}_\tau(\boldsymbol{x}) = \frac{\beta_\tau\,\boldsymbol{v}_\tau(\boldsymbol{x}) - \dot\beta_\tau\,\boldsymbol{x} - (\dot\alpha_\tau\beta_\tau - \dot\beta_\tau\alpha_\tau)\,\boldsymbol{x}_0}{\sigma_\tau\,(\dot\beta_\tau\sigma_\tau - \dot\sigma_\tau\beta_\tau)}. \tag{25}$$

Proof. By definition $\boldsymbol{v}_\tau(\boldsymbol{x}) = \mathbb{E}[\dot\alpha_\tau\boldsymbol{x}_0 + \dot\beta_\tau\boldsymbol{x}_1 + \dot\sigma_\tau\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau = \boldsymbol{x}]$. Since $\boldsymbol{x}_0$ is deterministic, substitute (22) and (21):

$$\begin{aligned}\boldsymbol{v}_\tau(\boldsymbol{x}) &= \dot\alpha_\tau\boldsymbol{x}_0 + \frac{\dot\beta_\tau}{\beta_\tau}\big(\boldsymbol{x} + \sigma_\tau^2\boldsymbol{s}_\tau - \alpha_\tau\boldsymbol{x}_0\big) - \dot\sigma_\tau\sigma_\tau\boldsymbol{s}_\tau \\ &= \frac{\dot\beta_\tau}{\beta_\tau}\boldsymbol{x} + \left(\dot\alpha_\tau - \frac{\dot\beta_\tau\alpha_\tau}{\beta_\tau}\right)\boldsymbol{x}_0 + \left(\frac{\dot\beta_\tau}{\beta_\tau}\sigma_\tau^2 - \dot\sigma_\tau\sigma_\tau\right)\boldsymbol{s}_\tau,\end{aligned} \tag{26}$$

which is (24). Multiplying the isolated score term by β_τ and rearranging yields (25). □

Proposition A.3 (Marginal-preserving SDE family)**.** *Let* p_τ *be the marginals of the path* (19) *with velocity* $\boldsymbol{v}_\tau$ *and score* $\boldsymbol{s}_\tau$*, and suppose* $p \in C^2((0,1)\times\mathbb{R}^{N_u})$*. Let* $g_\tau \geq 0$ *be measurable with* $g \in L^2([0,1])$*. Assume that the SDE*

$$\mathrm{d}\boldsymbol{X}_\tau = \big[\boldsymbol{v}_\tau(\boldsymbol{X}_\tau) + \tfrac{1}{2}g_\tau^2\boldsymbol{s}_\tau(\boldsymbol{X}_\tau)\big]\,\mathrm{d}\tau + g_\tau\,\mathrm{d}\boldsymbol{W}_\tau, \qquad \boldsymbol{X}_0 \sim p_0, \tag{27}$$

and its Fokker–Planck equation are well posed with unique solutions for the initial law p_0 *(allowing the point mass* $p_0 = \delta_{\boldsymbol{x}_0}$ *used by SI, which has no density). Then the SDE has marginal law* p_τ *for all* $\tau \in [0,1]$*. Setting* $g_\tau = 0$ *gives the probability-flow ODE whenever that ODE is well posed.*

Proof. Since $\boldsymbol{s}_\tau = \nabla\log p_\tau$, we have $p_\tau\boldsymbol{s}_\tau = \nabla p_\tau$. Hence the score drift and diffusion cancel in the Fokker–Planck equation:

$$-\nabla\cdot\big(\big[\boldsymbol{v}_\tau + \tfrac{1}{2}g_\tau^2\boldsymbol{s}_\tau\big]\,p_\tau\big) + \tfrac{1}{2}g_\tau^2\Delta p_\tau = -\nabla\cdot(\boldsymbol{v}_\tau p_\tau) = \partial_\tau p_\tau,$$

where the last equality is the interpolant's continuity equation. Thus p_τ solves the Fokker–Planck equation with initial law p_0 (interpreted weakly when p_0 is a point mass). Uniqueness gives the claimed marginals. Taking $g_\tau = 0$ yields the probability-flow ODE. □

A.1 Reading the velocity and score from the trained network

The two model classes train different objects. Proposition A.2 converts either into the pair $(\boldsymbol{v}_\tau, \boldsymbol{s}_\tau)$ that the samplers use.

Flow matching. The trained object is the velocity $\boldsymbol{v}_\theta$ itself, and the path has $\alpha_\tau = 0$ with source schedule σ_τ. Specialising (25),

$$\boldsymbol{s}_\tau(\boldsymbol{x}) = \frac{\beta_\tau\,\boldsymbol{v}_\theta(\boldsymbol{x}, \boldsymbol{u}^{n-1}, \tau) - \dot\beta_\tau\,\boldsymbol{x}}{\sigma_\tau\,(\dot\beta_\tau\sigma_\tau - \dot\sigma_\tau\beta_\tau)}, \tag{28}$$

which for rectified flow ($\sigma_\tau = 1-\tau$, $\beta_\tau = \tau$) reduces to $\boldsymbol{s}_\tau(\boldsymbol{x}) = (\tau\,\boldsymbol{v}_\theta - \boldsymbol{x})/(1-\tau)$. Note that (28) degenerates as $\tau \to 1$ (where $\sigma_\tau \to 0$), which is why the DM-SDE lift uses an endpoint-vanishing diffusion g_τ.

Stochastic interpolants. The trained object is the SDE drift $\boldsymbol{b}_\tau = \mathbb{E}[\dot\alpha_\tau \boldsymbol{x}_0 + \dot\beta_\tau \boldsymbol{x}_1 + \dot\gamma_\tau \boldsymbol{W}_\tau \mid \boldsymbol{X}_\tau = \boldsymbol{x}]$ of Eq. (9), with $\sigma_\tau = \gamma_\tau\sqrt{\tau}$. It decomposes as $\boldsymbol{b}_\tau = \boldsymbol{v}_\tau + \frac{1}{2}\gamma_\tau^2 \boldsymbol{s}_\tau$ (Chen et al., 2024), which follows from Lemma A.1: since $\gamma_\tau \boldsymbol{W}_\tau := \sigma_\tau \boldsymbol{\varepsilon}$ at fixed τ, Eq. (21) gives

$$\mathbb{E}[\dot\gamma_\tau \boldsymbol{W}_\tau \mid \boldsymbol{X}_\tau = \boldsymbol{x}] = \tfrac{\dot\gamma_\tau}{\gamma_\tau}\,\mathbb{E}[\gamma_\tau \boldsymbol{W}_\tau \mid \boldsymbol{X}_\tau = \boldsymbol{x}] = -\dot\gamma_\tau\gamma_\tau\tau\, \boldsymbol{s}_\tau(\boldsymbol{x}),$$

while the noise term of the drift carries $\dot\sigma_\tau\,\mathbb{E}[\boldsymbol{\varepsilon} \mid \boldsymbol{X}_\tau = \boldsymbol{x}] = -\dot\sigma_\tau\sigma_\tau\, \boldsymbol{s}_\tau(\boldsymbol{x})$ with $\dot\sigma_\tau\sigma_\tau = \frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}\tau}(\gamma_\tau^2\tau) = \dot\gamma_\tau\gamma_\tau\tau + \frac{1}{2}\gamma_\tau^2$. The remaining terms coincide, so $\boldsymbol{b}_\tau - \boldsymbol{v}_\tau = \frac{1}{2}\gamma_\tau^2\boldsymbol{s}_\tau$. This is also the reconciliation of the two noise parametrisations: the $\dot\gamma_\tau\boldsymbol{W}_\tau$ regression target (the Itô drift of $\mathrm{d}(\gamma_\tau\boldsymbol{W}_\tau)$ for a genuine Wiener process) and the pathwise derivative $\dot\sigma_\tau\boldsymbol{\varepsilon}$ of the equal-in-law source $\sigma_\tau\boldsymbol{\varepsilon}$ differ, upon conditioning, by exactly the score term that the SDE's diffusion restores. Adding $\frac{1}{2}\gamma_\tau^2\boldsymbol{s}_\tau$ to (24) and solving for the score,

$$\boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) = A_\tau[\beta_\tau \boldsymbol{b}_\tau(\boldsymbol{x}, \boldsymbol{x}_0, \tau) - \boldsymbol{c}_\tau(\boldsymbol{x}, \boldsymbol{x}_0)]\,, \tag{29}$$

with $A_\tau = \Big[\tau\gamma_\tau(\dot\beta_\tau\gamma_\tau - \beta_\tau\dot\gamma_\tau)\Big]^{-1}$ and $\boldsymbol{c}_\tau(\boldsymbol{x}, \boldsymbol{x}_0) = \dot\beta_\tau\boldsymbol{x} + (\beta_\tau\dot\alpha_\tau - \dot\beta_\tau\alpha_\tau)\boldsymbol{x}_0$ as stated by Chen et al. (2024). The velocity, when needed separately, is $\boldsymbol{v}_\tau = \boldsymbol{b}_\tau - \frac{1}{2}\gamma_\tau^2\boldsymbol{s}_\tau$.

B Conditioning the interpolant path: proofs

Throughout, $\boldsymbol{x}_0$ is fixed and $\boldsymbol{y} = \boldsymbol{H}(\boldsymbol{x}_1) + \boldsymbol{\eta}$, with $\boldsymbol{\eta}$ independent of $(\boldsymbol{x}_1, \boldsymbol{\varepsilon})$ given $\boldsymbol{x}_0$.

B.1 Posterior path

Proposition B.1 (Conditioning identities)**.** *The conditioned path retains the interpolant representation* (3)*, with the endpoint drawn from the posterior:*

$$\boldsymbol{x}_\tau = \alpha_\tau\boldsymbol{x}_0 + \beta_\tau\boldsymbol{x}_1 + \sigma_\tau\boldsymbol{\varepsilon}, \qquad \boldsymbol{x}_1 \sim p(\boldsymbol{x}_1 \,|\, \boldsymbol{y}, \boldsymbol{x}_0)\,, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \tag{30}$$

with independent $\boldsymbol{x}_1$ and $\boldsymbol{\varepsilon}$. Denoting its score and velocity by $\boldsymbol{s}_\tau^\star$ and $\boldsymbol{v}_\tau^\star$, respectively, we have, for $\tau \in (0, 1)$,

$$\boldsymbol{s}_\tau^\star = \boldsymbol{s}_\tau + \nabla_{\boldsymbol{x}} \log \Phi_\tau^{\boldsymbol{y}}, \tag{31}$$

$$\boldsymbol{v}_\tau^\star = \boldsymbol{v}_\tau + \kappa_\tau \nabla_{\boldsymbol{x}} \log \Phi_\tau^{\boldsymbol{y}}, \tag{32}$$

where arguments $(\boldsymbol{x}, \boldsymbol{x}_0)$ are suppressed, $\Phi_\tau^{\boldsymbol{y}}$ is defined in (10)*, and κ_τ in* (6)*.*

Proof. Since $\boldsymbol{\varepsilon}$ is independent of $(\boldsymbol{x}_1, \boldsymbol{y})$ given $\boldsymbol{x}_0$, conditioning changes only the law of $\boldsymbol{x}_1$, giving (30). By Bayes' rule,

$$p_\tau(\boldsymbol{x} \mid \boldsymbol{y}, \boldsymbol{x}_0) = \frac{\Phi_\tau^{\boldsymbol{y}}(\boldsymbol{x}, \boldsymbol{x}_0)\, p_\tau(\boldsymbol{x} \mid \boldsymbol{x}_0)}{p(\boldsymbol{y} \,|\, \boldsymbol{x}_0)}. \tag{33}$$

Taking $\nabla_{\boldsymbol{x}} \log$ gives (31). Applying the velocity–score identity (24) to both paths, whose schedules and source coincide, then gives (32). □

B.2 Proof of Theorem 4.1

Proof. By Proposition B.1, the posterior marginals form an interpolant path. Under the theorem's well-posedness assumptions, Proposition A.3 therefore gives a sampler with these marginals:

$$\mathrm{d}\boldsymbol{X}_\tau^* = \big[\boldsymbol{v}_\tau^\star(\boldsymbol{X}_\tau^*, \boldsymbol{x}_0) + \tfrac{1}{2}g_\tau^2\, \boldsymbol{s}_\tau^\star(\boldsymbol{X}_\tau^*, \boldsymbol{x}_0)\big]\,\mathrm{d}\tau + g_\tau\,\mathrm{d}\boldsymbol{W}_\tau, \qquad \boldsymbol{X}_0^* \sim p_0^{\boldsymbol{y}}.$$

Substituting (31)–(32) into its drift and suppressing the common arguments gives

$$\boldsymbol{v}_\tau^\star + \tfrac{1}{2}g_\tau^2\boldsymbol{s}_\tau^\star = \underbrace{\boldsymbol{v}_\tau + \tfrac{1}{2}g_\tau^2\boldsymbol{s}_\tau}_{\text{prior drift}} + \underbrace{\big(\kappa_\tau + \tfrac{1}{2}g_\tau^2\big)}_{w_\tau} \nabla_{\boldsymbol{x}} \log \Phi_\tau^{\boldsymbol{y}},$$

which is exactly (11). At $\tau = 1$, the interpolant equals $\boldsymbol{x}_1$, so the sampler's terminal law is $p(\boldsymbol{x}_1 \,|\, \boldsymbol{y}, \boldsymbol{x}_0)$.

Finally, the initial law is unchanged by conditioning. Since $\beta_0 = 0$, the source $\boldsymbol{x}_{\tau=0} = \alpha_0\boldsymbol{x}_0 + \sigma_0\boldsymbol{\varepsilon}$ is independent of $\boldsymbol{y}$ given $\boldsymbol{x}_0$. Thus $\Phi_0^{\boldsymbol{y}} = p(\boldsymbol{y} \,|\, \boldsymbol{x}_0)$ on the source support and $p_0^{\boldsymbol{y}} = p_0$. The sampler starts from $\mathcal{N}(0, \boldsymbol{I})$ for FM/DM and from $\delta_{\boldsymbol{x}_0}$ for SI, including when the source has no density. □

C Proof of Lemma 4.2 and Lemma 4.3

Proof of Lemma 4.2. Recall the general state interpolant and the observation interpolant,

$$\boldsymbol{x}_\tau = \alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1 + \boldsymbol{\chi}^{\mathrm{s}}_\tau, \qquad \boldsymbol{\chi}^{\mathrm{s}}_\tau = \sigma_\tau \boldsymbol{\varepsilon}, \tag{34}$$

$$\bar{\boldsymbol{y}}_\tau = \alpha_\tau H \boldsymbol{x}_0 + \beta_\tau \boldsymbol{y}, \tag{35}$$

where $\boldsymbol{y} = H\boldsymbol{x}_1 + \boldsymbol{\eta}$ with $\boldsymbol{\eta} \sim \mathcal{N}(0, R)$ independent of the source driver $\boldsymbol{\varepsilon}$. Substituting $\boldsymbol{y} = H\boldsymbol{x}_1 + \boldsymbol{\eta}$ into (35),

$$\bar{\boldsymbol{y}}_\tau = H(\alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1) + \beta_\tau \boldsymbol{\eta}. \tag{36}$$

From (34), $\alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{x}_1 = \boldsymbol{x}_\tau - \boldsymbol{\chi}^{\mathrm{s}}_\tau$, giving

$$\bar{\boldsymbol{y}}_\tau = H\boldsymbol{x}_\tau + \underbrace{\beta_\tau \boldsymbol{\eta} - H\boldsymbol{\chi}^{\mathrm{s}}_\tau}_{=\bar{\boldsymbol{\eta}}_\tau}, \tag{37}$$

which is Eq. (13). Since $\boldsymbol{\eta}$ is independent of $\boldsymbol{x}_1$ and of the source driver, $\boldsymbol{\eta} \perp (\boldsymbol{\chi}^{\mathrm{s}}_\tau, \boldsymbol{x}_\tau) \mid \boldsymbol{x}_0$, so

$$\mathbb{E}[\boldsymbol{\eta} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0] = 0, \qquad \mathrm{Cov}(\boldsymbol{\eta} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0) = R. \tag{38}$$

Taking conditional expectation of (37),

$$\mathbb{E}[\bar{\boldsymbol{y}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0] = H\boldsymbol{x}_\tau + \beta_\tau \mathbb{E}[\boldsymbol{\eta} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0] - H\,\mathbb{E}[\boldsymbol{\chi}^{\mathrm{s}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0] = H\boldsymbol{x}_\tau - H\,\mu^{\mathrm{s}}_\tau, \tag{39}$$

which is the mean in (14). Because the cross terms between $\boldsymbol{\eta}$ and $\boldsymbol{\chi}^{\mathrm{s}}_\tau$ vanish upon conditioning,

$$\mathrm{Cov}(\bar{\boldsymbol{y}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0) = \beta_\tau^2\,\mathrm{Cov}(\boldsymbol{\eta} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0) + H\,\mathrm{Cov}(\boldsymbol{\chi}^{\mathrm{s}}_\tau \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0)\,H^\top = \beta_\tau^2 R + H\,\Sigma^{\mathrm{s}}_\tau\,H^\top, \tag{40}$$

which is the covariance in (14). □

Proof of Lemma 4.3. The source process is $\boldsymbol{\chi}^{\mathrm{s}}_\tau = \sigma_\tau \boldsymbol{\varepsilon}$, so $\mu^{\mathrm{s}}_\tau = \sigma_\tau\,\mathbb{E}[\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0]$ and $\Sigma^{\mathrm{s}}_\tau = \sigma_\tau^2\,\mathrm{Cov}(\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau, \boldsymbol{x}_0)$. The Tweedie identities of Lemma A.1 give $\mathbb{E}[\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau] = -\sigma_\tau \boldsymbol{s}_\tau$ (21) and $\mathrm{Cov}(\boldsymbol{\varepsilon} \mid \boldsymbol{x}_\tau) = \boldsymbol{I} + \sigma_\tau^2 \nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau$ (23), hence

$$\mu^{\mathrm{s}}_\tau = -\sigma_\tau^2\,\boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0), \qquad \Sigma^{\mathrm{s}}_\tau = \sigma_\tau^2\,\boldsymbol{I} + \sigma_\tau^4\,\nabla_{\boldsymbol{x}} \boldsymbol{s}_\tau(\boldsymbol{x}, \boldsymbol{x}_0), \tag{41}$$

which is Eq. (15). □

D Implementation details

Interpolant schedules. SI priors use the quadratic schedule $\alpha_\tau = 1 - \tau$, $\beta_\tau = \tau^2$, $\gamma_\tau = 1 - \tau$ (so $\sigma_\tau = (1-\tau)\sqrt{\tau}$), and the FM priors use linear schedules with only a noise source term, $\alpha_\tau = 0$, $\beta_\tau = \tau$, $\sigma_\tau = 1 - \tau$.

Pseudo-time integration. Sampling uses M uniform pseudo-time steps: Euler–Maruyama for the SDE members and forward Euler for the guided ODE.

Architecture. The field-case priors (SI drift and FM velocity networks alike) are conditional U-Nets: four resolution levels with two ConvNeXt-style blocks per level (inner channel multiplier 2), circular padding, and multi-head spatial attention (four heads, rotary positional embedding) in the bottleneck. Pseudo-time enters through a learned scalar embedding, and the five most recent physical states are supplied as additional input channels to the first convolution, realising the conditioning on $\boldsymbol{u}^{n-1}$ (and a short history). The initial conditioning history consists of five consecutive states from the trajectory supplying the initial condition. Thereafter, each ensemble member updates its own history with its generated states. Channel widths per level: Navier–Stokes $(8, 16, 32, 64)$ for the SI prior and $(16, 32, 64, 128)$ for the FM prior, one field channel (vorticity). Urban $(32, 64, 128, 256)$, four field channels (three velocity components and temperature), with solid cells excluded via the building mask. Hyperparameters were chosen based on hyperparameter tuning.

Training. All priors are trained by simulation-free regression of their target (SI drift or FM velocity) under an MSE loss, with per-channel standardised fields: AdamW (learning rate 10^{-4}, weight decay 10^{-6}) with cosine annealing to 10^{-5}, batch size 32, up to 10^3 epochs with early stopping, dropout 0.1, adaptive EMA-based gradient clipping, and mixed precision. Navier–Stokes uses 160 training trajectories and 20 validation trajectories for the early stopping. The urban case uses 150 LES runs for training and 20 runs for validation. The same trained FM checkpoint per case serves every FM and DM sampler and the same trained SI checkpoint serve every SI sampler. No prior is ever retrained or fine-tuned for assimilation.

E EVALUATION METRICS

All metrics are computed per physical step from the posterior ensemble $\{\boldsymbol{x}^{(e)}\}_{e=1}^{E}$ against the truth $\boldsymbol{x}^*$ (or the exact posterior in the analytical case) and then averaged over steps and trajectories. Spatial averages run over the kept grid points $i \in \mathcal{I}$ (in the urban case, solid cells are excluded); $\bar{\boldsymbol{x}} = \frac{1}{E}\sum_e \boldsymbol{x}^{(e)}$ denotes the ensemble mean. The urban per-variable metrics (velocity, temperature) apply the same formulas with $\mathcal{I}$ restricted to the channels of that variable.

Ensemble-mean RMSE. The RMSE of the ensemble mean,

$$\mathrm{RMSE} = \left(\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\left(\bar{x}_i - x_i^*\right)^2\right)^{1/2}, \tag{42}$$

not the member-averaged RMSE, which would additionally count ensemble spread.

CRPS. Per grid point, the unbiased ensemble estimator of the continuous ranked probability score (Gneiting & Raftery, 2007),

$$\mathrm{CRPS}_i = \frac{1}{E}\sum_{e=1}^{E}\left|x_i^{(e)} - x_i^*\right| - \frac{1}{2E(E-1)}\sum_{e\neq e'}\left|x_i^{(e)} - x_i^{(e')}\right|, \tag{43}$$

averaged over $\mathcal{I}$; the $1/(E(E-1))$ normalisation makes the spread term unbiased for $\mathbb{E}|X - X'|$.

Spread–skill. The ratio of corrected mean ensemble spread to skill,

$$\mathrm{ratio} = \sqrt{\tfrac{E+1}{E}}\left(\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}} s_i\right)\Big/\,\mathrm{RMSE}, \tag{44}$$

with s_i the ensemble standard deviation at point i and $\sqrt{(E+1)/E}$ the finite-ensemble correction. We report $|1 - \mathrm{ratio}|$, so 0 is perfectly calibrated.

KL from the exact posterior (analytical case). With sample moments (μ_p, Σ_p) of the method ensemble and exact posterior moments (μ_q, Σ_q), we report the Gaussian KL from the reference to the fitted ensemble distribution:

$$D_{\mathrm{KL}}(q\|p) = \tfrac{1}{2}\big(\operatorname{tr}(\Sigma_p^{-1}\Sigma_q) + (\mu_p - \mu_q)^\top\Sigma_p^{-1}(\mu_p - \mu_q) - d + \log\det\Sigma_p - \log\det\Sigma_q\big), \tag{45}$$

where d is the state dimension. Because μ_p and Σ_p are estimated from a finite ensemble, this reported KL is generally nonzero even for independent samples from the exact posterior. Increasing the number of integration steps alone does not remove this sampling error.

Sliced W_1. The 1-Wasserstein distance averaged over random one-dimensional projections: for unit directions θ_j, $j = 1, \ldots, 128$, the projected distance $W_1(\theta_j)$ is computed by integrating the absolute difference between the empirical quantile functions of the two projected sample sets, and we report $\frac{1}{128}\sum_{j=1}^{128} W_1(\theta_j)$.

Cost. Wall-clock seconds per assimilation step, at matched ensemble size on the same hardware.

Enstrophy spectrum (Navier–Stokes). The spectral diagnostic of the reconstructions, the radially averaged enstrophy spectrum $Z(k)$, is defined where it is used, in Eq. (53) of Appendix H.3.

F METHOD DESCRIPTIONS AND COMPARISON TO ALTERNATIVES

Our method shares building blocks with the baselines below but assembles them differently. For each method we state how the state is evolved, how the likelihood term is approximated, the per-step network cost, and which hyperparameters were tuned; quantitative comparisons are in Section 5. Tuned hyperparameters were selected by RMSE sweeps per observation scenario (and, where indicated, per step count M) on a held-out trajectory not used for evaluation; Table 3 collects all sampler hyperparameters. Throughout, $\hat{\boldsymbol{u}}_\tau^1 = \mathbb{E}[\boldsymbol{x}_1 \mid \boldsymbol{x}_\tau, \boldsymbol{u}^0]$ denotes the one-step denoiser.

FlowDAS. FlowDAS (Chen et al., 2025), commits to the SI-SDE and estimates the likelihood score by Monte Carlo: the gradient of a softmax-weighted squared residual through J one-step predictions $\hat{\boldsymbol{u}}_\tau^{1,j}$, entering the drift with a tuned constant step size ζ in place of our analytic weight w_τ. We instead read the score off the closed-form observation interpolant, removing sampling variance and per-step trajectory backprop; the trade-off is our Gaussian surrogate, whose bias is controlled by the covariance choice rather than the number of samples. *Cost:* two forwards (prior drift and denoiser endpoint) and one backward per step, independent of J. *Key limitation:* ζ is scenario-dependent and may require tuning, and there is no inflated source covariance. In our implementation, stability can also depend on the solver step count M, and overly large ζ can lead to divergence. *Tuned:* the guidance step size ζ, per scenario and M; the number of one-step samples is fixed at $J = 25$.

Guided FM (FIG). FIG (Yan et al., 2025) guides the deterministic FM probability-flow ODE for static linear inverse problems: after each prior-velocity step, k inner corrector steps push the state along $H^\top$ times the *normalised* observation residual with magnitude $c\,(1-\tau)/\tau$, using a measurement interpolant equivalent to ours. *Cost:* one forward per step, no backward (the corrector differentiates only through the linear H) – the cheapest guided sampler in the comparison. *Key limitation:* treats $\boldsymbol{x}_\tau \approx \boldsymbol{x}_1$, uses no covariance, and the corrector is not derived from a probability path, so the guidance magnitude $c(1-\tau)/\tau$ is not data-adaptive. *Tuned:* the corrector scale c, per scenario and M; the number of inner corrector steps k is set by the observation operator (sweep-selected: $k = 1$ for super-resolution, $k = 3$ for sparse sensors), and the measurement-interpolant noise is fixed to zero as in the original work.

SDA. SDA (Rozet & Louppe, 2023) applies Tweedie-based guidance on a diffusion prior: the gradient of a Gaussian likelihood at the Tweedie denoiser, taken through the network, with a ΠGDM-style inflated isotropic covariance scaled by a tunable factor γ_{sda}. Its original form composes locally learned segment scores into a joint trajectory score and conditions all states at once. We use its single-window form and are autoregressive, keeping per-step cost and prior dimension fixed as the horizon grows (but not propagating cross-window correlations as a joint smoother would). *Cost:* two forwards and one backward per step. *Tuned:* the covariance inflation γ_{sda}, per scenario and M. *Key limitation:* SDA is tied to the native reverse diffusion. Its bias therefore stems entirely from the Gaussian–Tweedie likelihood surrogate, whose isotropic covariance carries no model source information. Because the guidance weight is inherited from the fixed diffusion rather than derived from a probability path, it does not extend to samplers with modified or vanishing diffusion ($w_\tau \neq g_\tau^2$), unlike Theorem 4.1.

D-Flow SGLD. Original D-Flow (Ben-Hamu et al., 2024) optimizes the FM source $\boldsymbol{x}_0$ by gradient descent through the full FM-ODE rollout, with prior regularization supplied implicitly by the learned flow. D-Flow SGLD (Parikh et al., 2026), used here, instead runs a preconditioned SGLD chain over $\boldsymbol{x}_0 \sim \mathcal{N}(0, I)$ through the rollout (6-step RK2, backpropagated with gradient checkpointing), with energy given by the squared data misfit plus an explicit ℓ_2 source penalty $\lambda\|\boldsymbol{x}_0\|^2$; zero Langevin noise recovers MAP-style source optimization. *Cost:* the entire rollout repeats at every SGLD step (12 forwards and 12 backwards per step at our settings) – the only baseline backpropagating through a full multi-step generative rollout, and by far the most expensive. *Tuned:* the SGLD step size η and the source penalty λ, per case (in our sweeps η is the dominant knob); the Langevin noise scale and the preconditioner constants follow the original paper.

SURGE. SURGE (Wei et al., 2026) is a sequential Monte-Carlo wrapper: a guided reverse-SDE proposal with Girsanov-corrected importance weights that telescope to the exact terminal likelihood, making the target asymptotically exact as the number of particles $N \to \infty$. The guidance controls proposal efficiency and therefore finite-N degeneracy. In our implementation, particles resample when the effective sample size drops below $N/2$. We report it wrapping the FlowDAS and SDA guidance. *Measured implementation cost:* one additional denoiser forward per step relative to the wrapped method, with no additional backward pass. *Tuned:* nothing SURGE-specific – the resampling threshold is fixed at $\mathrm{ESS} < N/2$, and the wrapped guidance keeps the hyperparameters tuned for its standalone counterpart (ζ, γ_{sda}). *Key limitation:* particle degeneracy in high dimension.

Diffusion prior for DM-based methods. SDA, SURGE, and DM-SDE require a diffusion prior; we construct it from the trained FM velocity via the velocity–score identity (Appendix A), so all generative methods use the same network.

Table 3: Sampler hyperparameters for all methods. *Tuned* entries were selected by RMSE sweeps on a held-out trajectory, per observation scenario and – where a range is given – per step count M; ranges span the values used across the scenario/M grid. *Fixed* entries follow the original works. In the linear–Gaussian case our samplers use the exact covariance without damping ($\lambda = 1$) and the baselines reuse the Navier–Stokes tuning (FIG: $k = 1$, $c = 1$). The full per-scenario, per-M matrices are provided in the configuration files released with the code.

Method	Hyperparameter	Role	Selection	Navier–Stokes	Urban
Ours (all samplers)	λ	Jacobian damping (shared cov.)	tuned	0.9 (SR) / 0.95 (sparse)	0.95
	k	Jacobian refresh cadence	fixed (cost dial)	10	10
FlowDAS	ζ	guidance step size	tuned	2×10^{-3}–0.15	5×10^{-4}
	J	Monte-Carlo $\boldsymbol{x}_1$ samples	fixed	25	25
Guided FM (FIG)	c	corrector step-size scale	tuned	5–20	160
	k	inner corrector steps	operator type	1 (SR) / 3 (sparse)	3
	w	measurement-interpolant noise	fixed	0	0
SDA	γ_{sda}	likelihood covariance inflation	tuned	10^{-4}–10^{-2}	5×10^{-4}
D-Flow SGLD	η	SGLD step size	tuned	5×10^{-3}	5×10^{-3}
	λ	source penalty	tuned	10^{-4}	10^{-3}
	s	Langevin noise scale	fixed	10^{-3}	10^{-3}
	ω, δ	pSGLD preconditioner	fixed	0.99, 10^{-3}	0.99, 10^{-3}
	K	SGLD steps (6-step RK2 rollout)	$= M$	M	M
SURGE	ESS threshold	resampling trigger	fixed	$N/2$	$N/2$
	ζ, γ_{sda}	wrapped guidance	carried over	as above	as above
EnKF	inflation / localisation	analysis regularisation	none	1.0 / none	–
Particle filter	resampling	scheme	fixed	systematic	–

G Analytical test case: setup and additional results

G.1 Setup

The analytical test case is the linear-Gaussian system with $N_{\boldsymbol{u}} = N_{\boldsymbol{y}} = 2$

$$p(\boldsymbol{u}^1 \,|\, \boldsymbol{u}^0) = \mathcal{N}(\boldsymbol{u}^0, \boldsymbol{I}), \qquad p(\boldsymbol{y}^1 \,|\, \boldsymbol{u}^1) = \mathcal{N}(H\boldsymbol{u}^1, R), \qquad H = \boldsymbol{I},\ R = \boldsymbol{I}, \tag{46}$$

i.e. Eq. (1) with identity forward and observation operators and unit Gaussian noise.

For a Gaussian conditional target $p(\boldsymbol{x}_1 \,|\, \boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{\mu}_1(\boldsymbol{x}_0), \boldsymbol{\Sigma}_1)$, the SI drift is available in closed form (Proposition B.9 of Chen et al., 2024). Specialised to $\boldsymbol{\mu}_1(\boldsymbol{x}_0) = \boldsymbol{x}_0$ and $\boldsymbol{\Sigma}_1 = \boldsymbol{I}$, it reads

$$\boldsymbol{b}(\boldsymbol{x}, \boldsymbol{x}_0, \tau) = \dot{\alpha}_\tau \boldsymbol{x}_0 + \dot{\beta}_\tau \boldsymbol{\mu}_1(\boldsymbol{x}_0) + (\beta_\tau \dot{\beta}_\tau + \tau\gamma_\tau\dot{\gamma}_\tau)\, \bar{\boldsymbol{\Sigma}}_\tau^{-1}(\boldsymbol{x} - \bar{\boldsymbol{\mu}}_\tau(\boldsymbol{x}_0)), \tag{47}$$

where

$$\bar{\boldsymbol{\mu}}_\tau(\boldsymbol{x}_0) = \alpha_\tau \boldsymbol{x}_0 + \beta_\tau \boldsymbol{\mu}_1(\boldsymbol{x}_0), \quad \bar{\boldsymbol{\Sigma}}_\tau = (\beta_\tau^2 + \tau\gamma_\tau^2)\, \boldsymbol{I}, \tag{48}$$

are the mean and covariance of the interpolant $\boldsymbol{x}_\tau$ conditioned on $\boldsymbol{x}_0$. Note that the second drift term involves the prior mean $\boldsymbol{\mu}_1(\boldsymbol{x}_0) = \boldsymbol{x}_0$, not $\bar{\boldsymbol{\mu}}_\tau(\boldsymbol{x}_0)$; the two coincide only for schedules with $\alpha_\tau + \beta_\tau = 1$.

The flow matching velocity is available in closed form for the same target. For linear schedules ($\sigma_\tau = 1 - \tau$, $\beta_\tau = \tau$) and $\boldsymbol{\mu}_1(\boldsymbol{x}_0) = \boldsymbol{x}_0$ this reduces to

$$\boldsymbol{v}(\boldsymbol{x}, \boldsymbol{x}_0, \tau) = \boldsymbol{x}_0 + \frac{2\tau - 1}{\tau^2 + (1-\tau)^2}\,(\boldsymbol{x} - \tau \boldsymbol{x}_0). \tag{49}$$

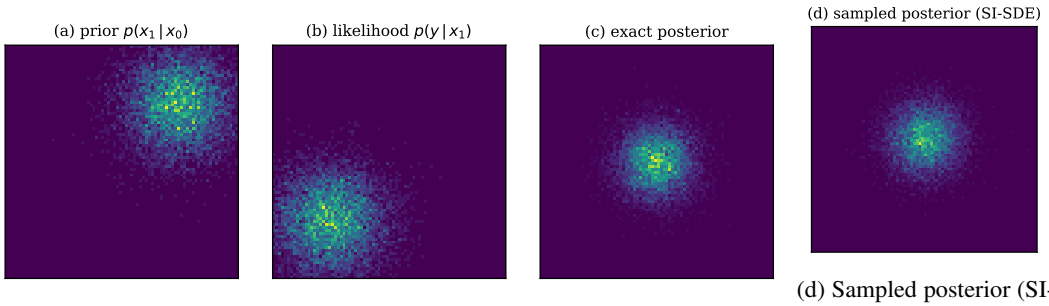
The same velocity drives both the FM-ODE and, via the score identity (28), the DM-SDE sampler with the endpoint-vanishing diffusion $g_\tau = \sqrt{\sigma_\tau \beta_\tau}$.

The true posterior is

$$p(\boldsymbol{u}^1 \,|\, \boldsymbol{y}^1, \boldsymbol{u}^0) = \mathcal{N}\left(\boldsymbol{u}^0 + K(\boldsymbol{y}^1 - H\boldsymbol{u}^0),\, \boldsymbol{I} - KH\right), \tag{50}$$

where

$$K = H^\top (HH^\top + R)^{-1} = \tfrac{1}{2}\,\boldsymbol{I}. \tag{51}$$



(a) Prior conditional. (b) Likelihood. (c) Exact posterior. (d) Sampled posterior (SI-SDE).

Figure 6: Analytical linear–Gaussian case.

Table 4: Analytical case: Gaussian-fit KL divergence and sliced-W_1 to the exact posterior at $M = 50$ and $M = 500$ sampler steps (mean over five seeds, 100,000-sample ensembles), and wall-clock seconds per assimilation at $M = 500$. The EnKF and particle filter perform a single analysis step (no M); both operate in their ideal linear–Gaussian regime here. A dash marks a collapsed ensemble with no finite KL.

	Gaussian KL		**Sliced-W_1**		**Cost**
	M=50	M=500	M=50	M=500	s
Ours (SI-SDE)	0.0004	6.8×10^{-5}	0.011	0.005	4.9
Ours (DM-SDE)	0.0012	8.2×10^{-5}	0.014	0.005	8.1
Ours (FM-ODE)	0.0018	8.0×10^{-5}	0.017	0.004	6.1
Ours (SI-SDE), Jac.-free	0.116	0.134	0.155	0.167	3.5
Ours (DM-SDE), Jac.-free	0.171	0.161	0.182	0.183	3.8
Ours (FM-ODE), Jac.-free	0.241	0.224	0.208	0.209	1.6
FlowDAS	0.491	3.78	0.481	0.438	67.7
FlowDAS + SURGE	4.7×10^{-5}	87.8	0.005	0.649	69.1
SDA	0.270	0.263	0.322	0.318	6.1
SDA + SURGE	0.173	2.36	0.150	0.339	8.5
D-Flow SGLD	0.096	56.1	0.190	0.628	9.9
Guided FM (FIG)	0.470	–	0.354	0.654	1.9
EnKF	7.3×10^{-5}		0.005		0.01
Particle filter	6.4×10^{-5}		0.004		0.01

G.2 Additional results

Figure 6 visualises the case in two dimensions: the prior conditional, likelihood, exact posterior, and the posterior sampled with the closed-form drift. Table 4 reports, for every method, the KL divergence and the sliced-W_1 distance to the exact posterior at $M = 50$ and $M = 500$, alongside the wall-clock cost of one assimilation.

H Navier–Stokes test case: setup and additional results

This appendix outlines the setup in greater detail and presents additional results: the data generation and forcing, qualitative reconstructions for each observation scenario, convergence in the number of sampler steps M, per-assimilation-step error curves, the enstrophy spectra of the reconstructions, and the ablations. Throughout: five held-out trajectories, $E = 64$ ensemble members, 15 assimilation steps, observation noise $\boldsymbol{\eta} \sim \mathcal{N}(0, 0.05^2\boldsymbol{I})$.

H.1 Forcing and data generation

The stochastic forcing in Eq. (18) is temporally white and supplied to four Fourier modes,

$$\mathrm{d}\xi(t,\boldsymbol{x}) = \sum_{\boldsymbol{k}\in\mathcal{K}} \left[\cos(\boldsymbol{k}\cdot\boldsymbol{x})\,\mathrm{d}W_t^{\boldsymbol{k},c} + \sin(\boldsymbol{k}\cdot\boldsymbol{x})\,\mathrm{d}W_t^{\boldsymbol{k},s}\right], \qquad \mathcal{K} = \{(6,0),(7,0),(5,5),(8,8)\}, \tag{52}$$

with independent standard Wiener processes $W^{\boldsymbol{k},c}$ and $W^{\boldsymbol{k},s}$ and forcing amplitude $\varepsilon = 1$.

The trajectories are simulated with JAX-CFD (Kochkov et al., 2021) with a pseudo-spectral (Fourier) discretisation of Eq. (18) on a 256^2 grid over the torus $\mathcal{T}^2 = [0, 2\pi]^2$, with viscosity $\nu = 10^{-3}$ and linear drag $\alpha = 0.1$. Time integration uses forward Euler–Maruyama on the vorticity Fourier coefficients with a fine step of $\mathrm{d}t = 10^{-4}$. Initial conditions are random divergence-free velocity fields with peak wavenumber $k = 4$ and maximum velocity 7.

Each trajectory is integrated for 500000 time steps, corresponding to a time horizon of $T = 50$, and stored every 0.5 time units, i.e at every 5000 time step. This yields 100 coarse snapshots per trajectory, of which the first 50 are discarded so that the retained states are statistically spun up. The stored fields are finally down-sampled from 256^2 to the 128^2 resolution used for training and assimilation, so that the observation operators of Section 5 act on a 128^2 state.

H.2 Ablations

On the Navier–Stokes case, we study the computational approximations introduced in Section 4.2. For the ensemble-shared covariance, we vary the Jacobian refresh rate k to quantify the trade-off between accuracy and computational cost. Table 5 collects the results.

Table 5: Navier–Stokes ablation of the covariance approximation of Section 4.2: the ensemble-shared inflated covariance at Jacobian refresh cadences $k \in \{1, 5, 10\}$ (the shared Jacobian is recomputed every k of the M sampler steps) against the Jacobian-free covariance. Same metrics and scenarios as Table 1, with per-step cost (lower is better). $E = 64$ and $M = 50$ sampler steps. **Bold** marks the best value in each column, together with every value equal to it to 2 decimal places in unscaled units.

	Vorticity RMSE ($\times 10^{-1}$)				**CRPS** ($\times 10^{-1}$)				**Spread–skill** ($\times 10^{-1}$)				**Cost (s/step)**			
	16^2	32^2	5%	$\frac{1}{64}$	16^2	32^2	5%	$\frac{1}{64}$	16^2	32^2	5%	$\frac{1}{64}$	16^2	32^2	5%	$\frac{1}{64}$
Ensemble-shared inflated covariance – Jacobian refreshed every step ($k = 1$)																
Ours (SI-SDE)	2.087	0.701	**1.330**	**2.885**	1.105	0.367	**0.672**	**1.532**	2.742	2.509	2.577	2.503	64.9	215.2	174.5	65.5
Ours (DM-SDE)	2.248	**0.640**	1.470	4.548	1.178	**0.334**	**0.740**	2.480	1.572	**0.637**	**1.079**	1.929	134.2	442.4	360.3	135.2
Ours (FM-ODE)	**1.895**	**0.607**	1.612	4.870	**1.005**	**0.318**	0.821	2.675	**1.448**	1.110	1.385	1.996	134.8	440.4	360.4	135.3
Ensemble-shared inflated covariance – Jacobian refreshed every $k = 5$ steps																
Ours (SI-SDE)	2.076	0.689	1.727	3.925	1.104	0.362	0.886	2.157	2.827	2.701	2.676	2.861	25.8	56.2	48.3	25.8
Ours (DM-SDE)	2.166	**0.642**	3.125	6.621	1.157	**0.335**	1.656	3.723	1.490	0.796	1.639	2.111	53.5	115.9	99.7	53.5
Ours (FM-ODE)	2.085	**0.623**	5.506	6.839	1.121	**0.327**	2.784	3.849	1.651	1.632	1.986	2.085	53.6	116.0	99.8	53.7
Ensemble-shared inflated covariance – Jacobian refreshed every $k = 10$ steps																
Ours (SI-SDE)	2.142	0.684	2.247	5.066	1.152	0.360	1.182	2.873	2.825	2.822	2.943	3.269	20.7	38.8	34.2	20.9
Ours (DM-SDE)	2.344	**0.649**	4.389	7.339	1.261	**0.339**	2.387	4.138	1.557	0.921	1.909	1.988	43.6	82.1	72.2	44.3
Ours (FM-ODE)	2.279	**0.641**	5.263	7.517	1.234	**0.336**	2.868	4.236	1.656	1.843	2.061	1.933	44.5	82.4	72.4	44.6
Jacobian-free covariance																
Ours (SI-SDE)	2.905	0.693	7.069	8.520	1.618	0.373	4.079	4.910	2.694	3.368	3.256	2.811	**3.4**	**3.6**	**3.6**	**3.5**
Ours (DM-SDE)	2.845	0.679	6.805	8.370	1.553	0.354	3.804	4.705	1.623	1.209	2.172	**1.539**	6.5	6.6	6.6	6.5
Ours (FM-ODE)	2.817	0.694	7.076	8.471	1.545	0.365	3.948	4.756	1.670	2.022	2.114	**1.478**	6.5	6.6	6.6	6.5

H.3 Enstrophy spectra

Pointwise metrics do not reveal which scales are misrepresented. Figure 14 therefore compares the radially averaged enstrophy spectrum of each posterior mean with the truth:

$$Z(k) = \left\langle \tfrac{1}{2}|\hat{\omega}(\boldsymbol{k})|^2 \right\rangle_{|\boldsymbol{k}|\in \mathrm{shell}(k)}, \tag{53}$$

using 60 radial bins and averaging over the 15 assimilated steps. This diagnostic is computed directly from vorticity.

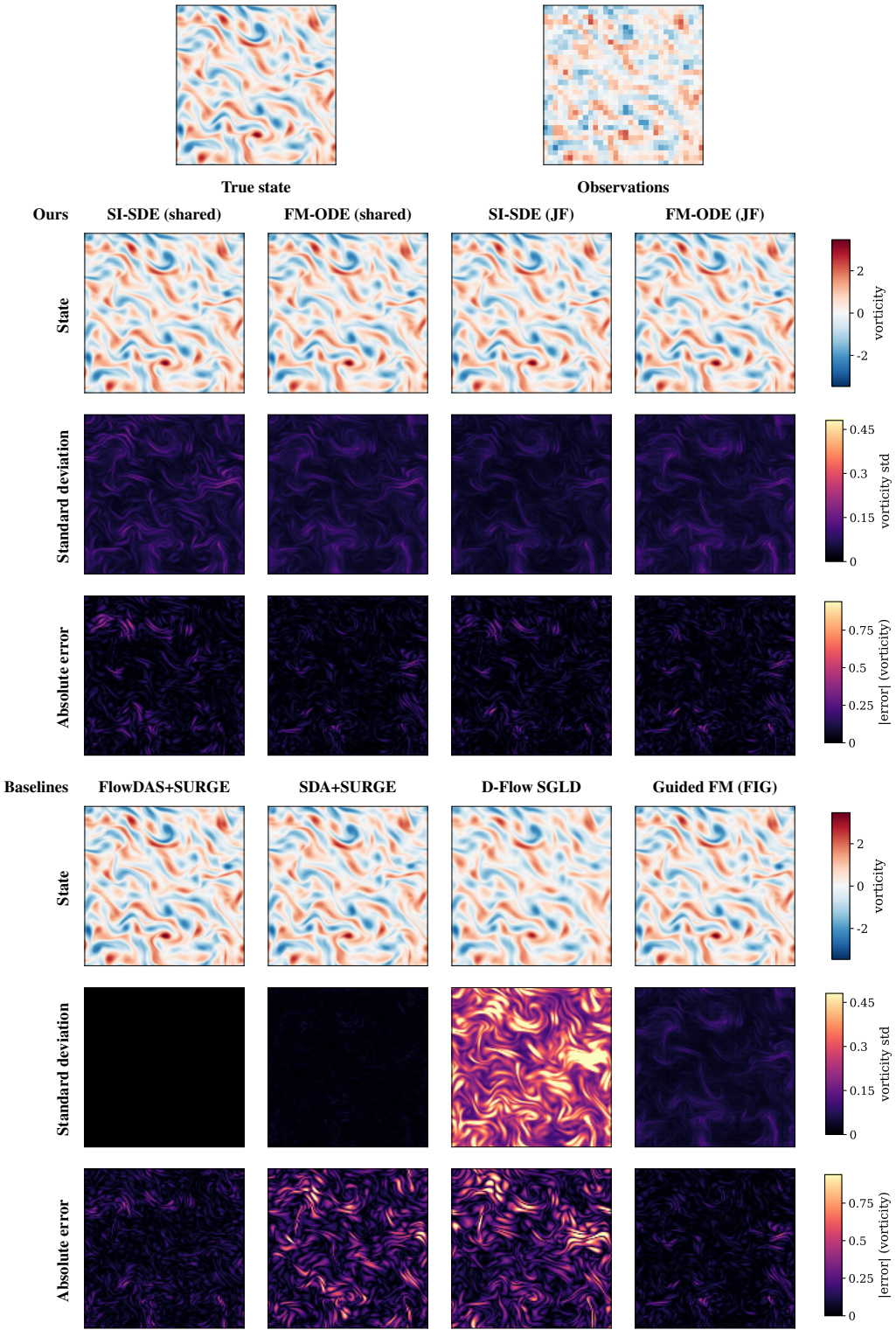


Figure 7: Navier–Stokes, super-resolution $32^2 \to 128^2$: true vorticity and observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final assimilation step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance.

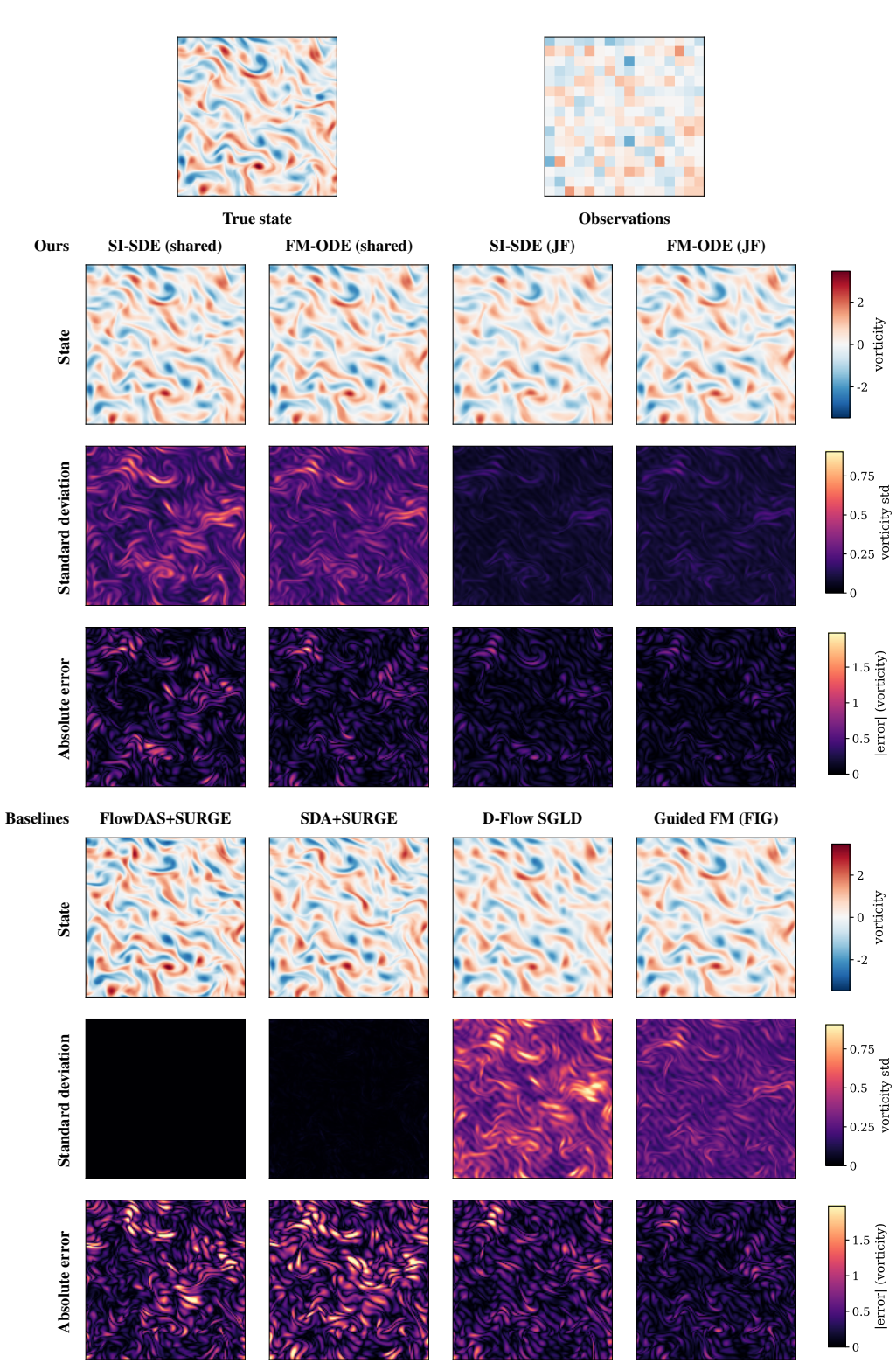


Figure 8: Navier–Stokes, super-resolution $16^2 \to 128^2$: true vorticity and observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final assimilation step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance.

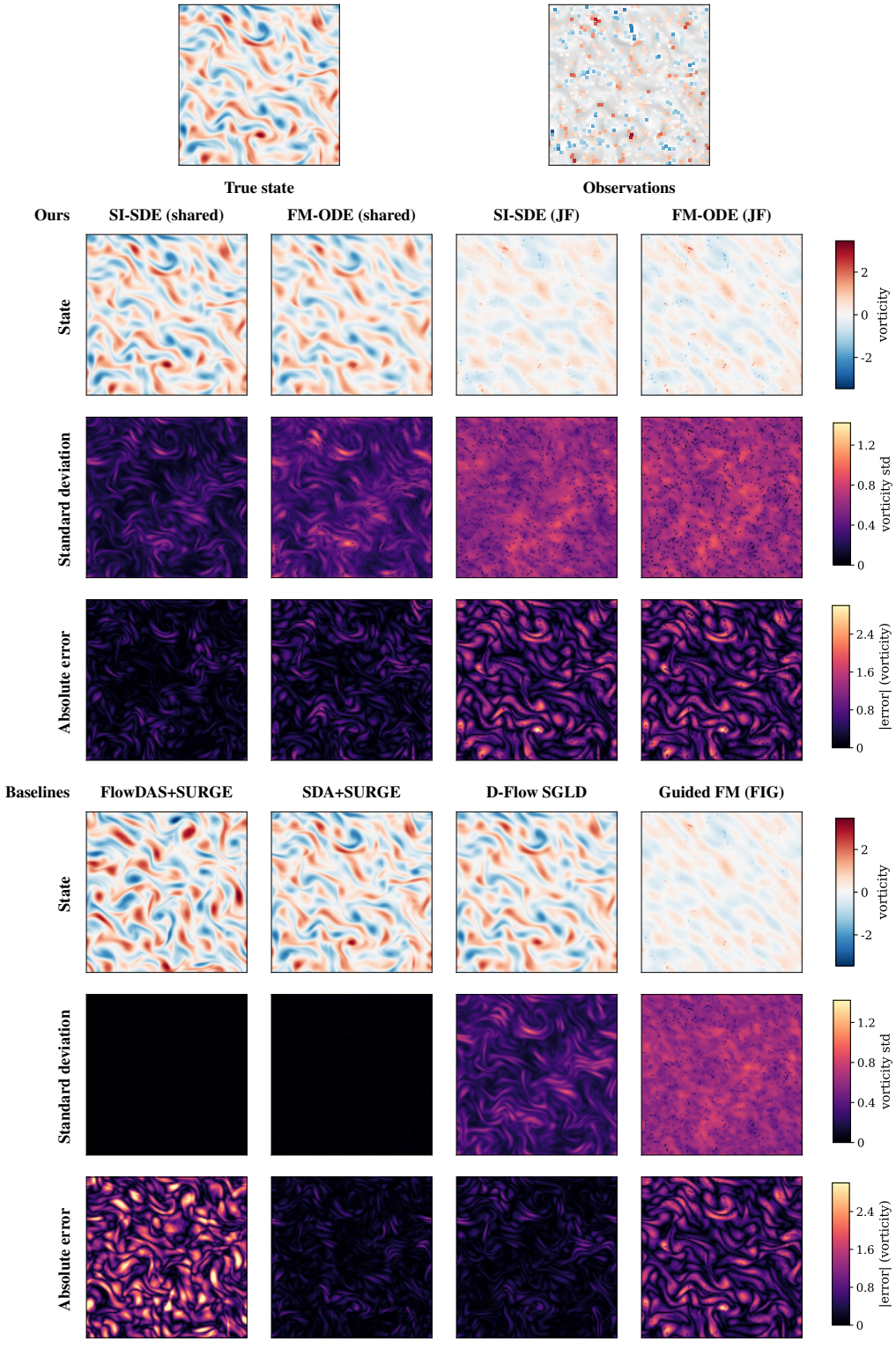


Figure 9: Navier–Stokes, sparse sensors (5%): true vorticity and observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final assimilation step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance.

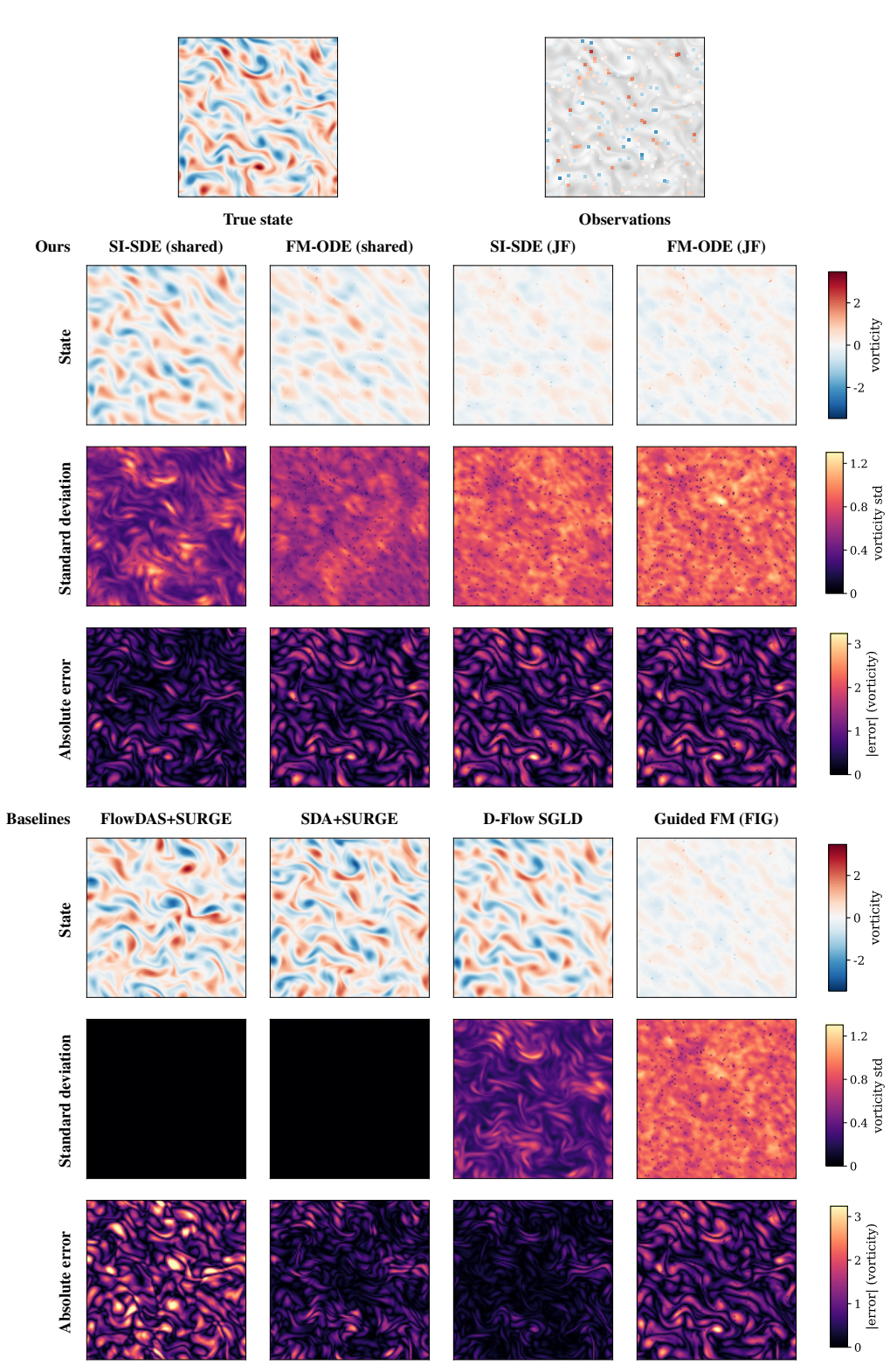
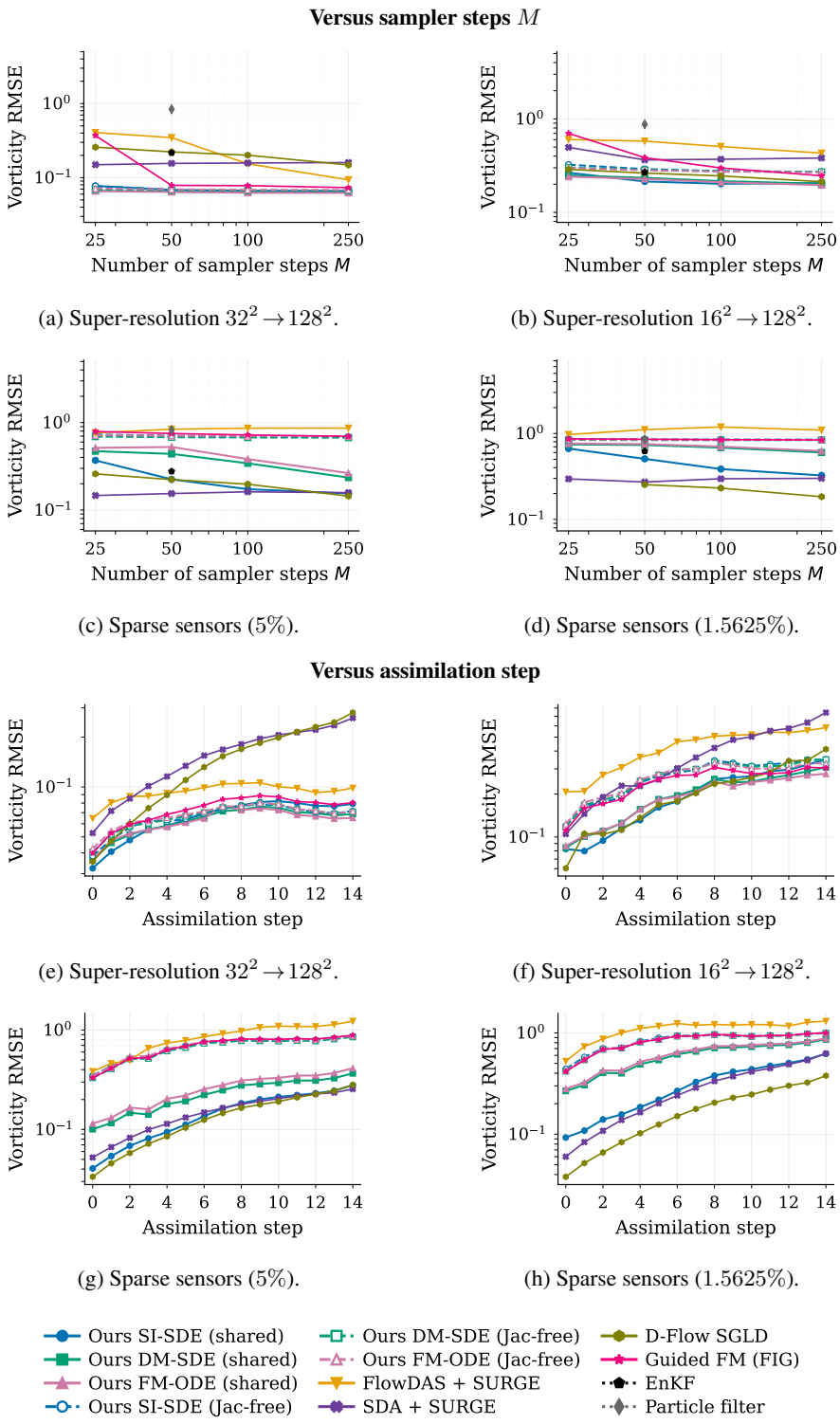


Figure 10: Navier–Stokes, sparse sensors (1.5625%): true vorticity and observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final assimilation step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance.



(a) Super-resolution $32^2 \to 128^2$.

(b) Super-resolution $16^2 \to 128^2$.

(c) Sparse sensors (5%).

(d) Sparse sensors (1.5625%).

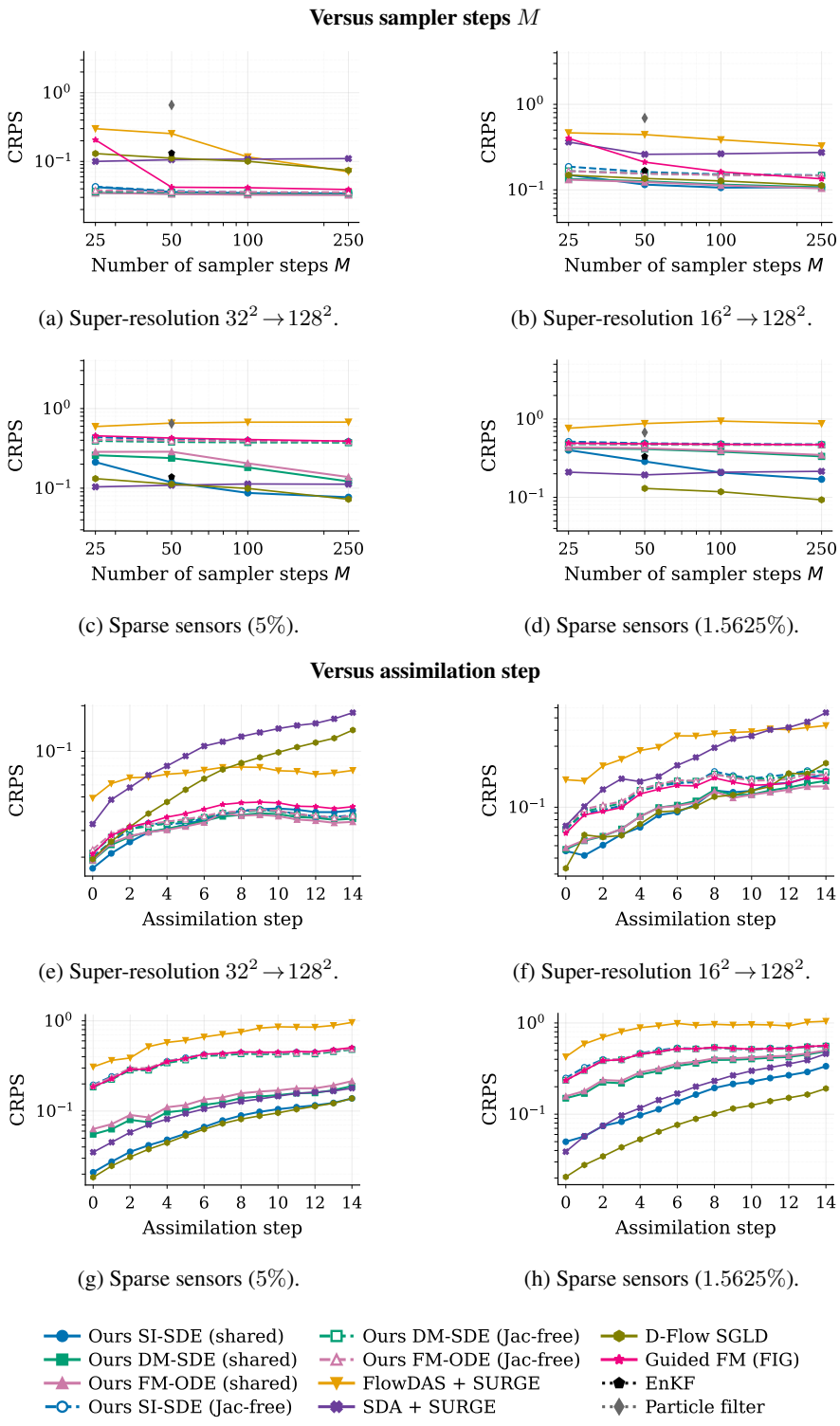
(e) Super-resolution $32^2 \to 128^2$.

(f) Super-resolution $16^2 \to 128^2$.

(g) Sparse sensors (5%).

(h) Sparse sensors (1.5625%).

Figure 11: Navier–Stokes: ensemble-mean vorticity RMSE as a function of the number of sampler steps M (top two rows) and assimilation step at the largest M (bottom two rows), for all observation scenarios.



(a) Super-resolution $32^2 \to 128^2$.

(b) Super-resolution $16^2 \to 128^2$.

(c) Sparse sensors (5%).

(d) Sparse sensors (1.5625%).

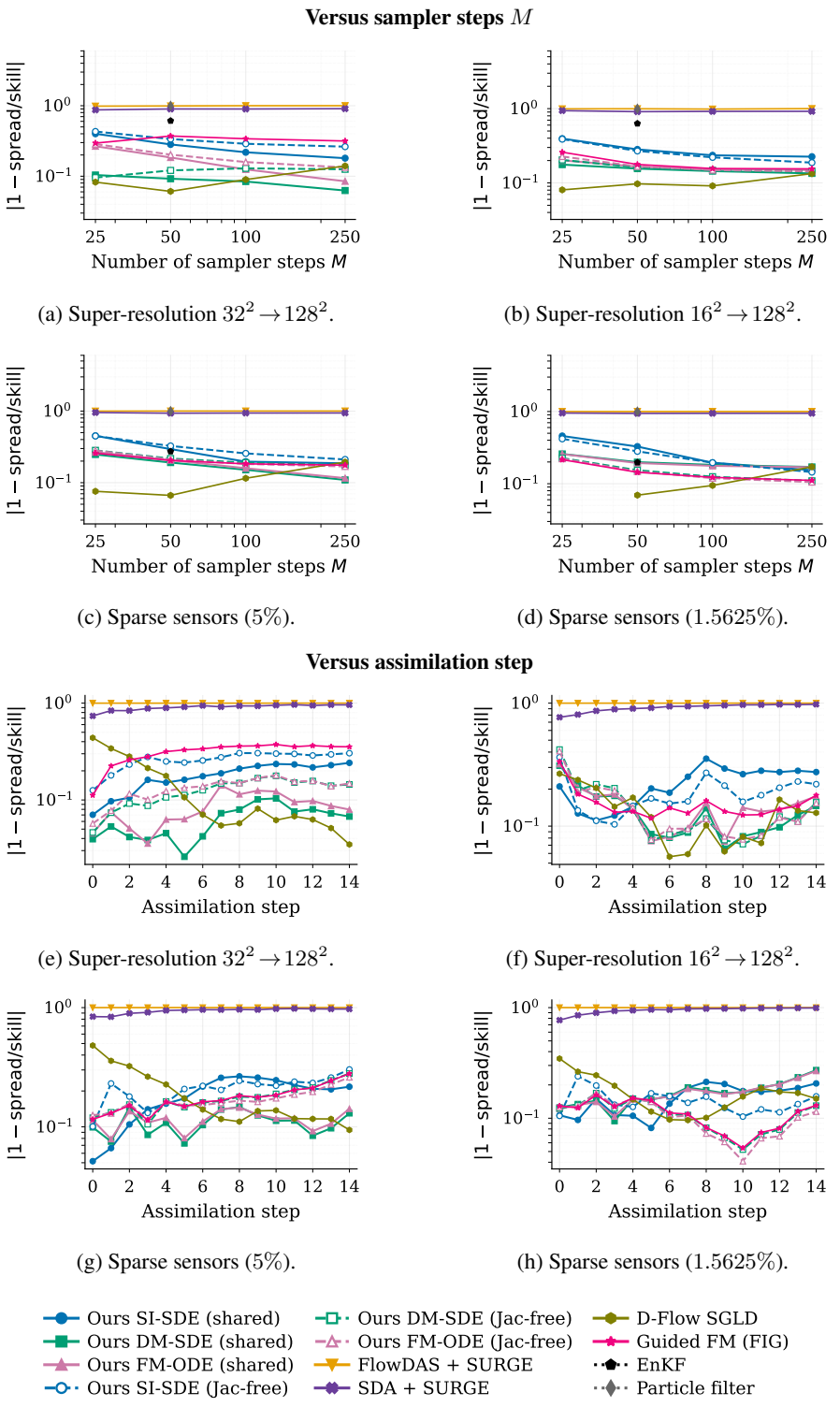
(e) Super-resolution $32^2 \to 128^2$.

(f) Super-resolution $16^2 \to 128^2$.

(g) Sparse sensors (5%).

(h) Sparse sensors (1.5625%).

Figure 12: Navier–Stokes: CRPS as a function of the number of sampler steps M (top two rows) and assimilation step at the largest M (bottom two rows), for all observation scenarios.



(a) Super-resolution $32^2 \to 128^2$.

(b) Super-resolution $16^2 \to 128^2$.

(c) Sparse sensors (5%).

(d) Sparse sensors (1.5625%).

(e) Super-resolution $32^2 \to 128^2$.

(f) Super-resolution $16^2 \to 128^2$.

(g) Sparse sensors (5%).

(h) Sparse sensors (1.5625%).

Figure 13: Navier–Stokes: Spread-skill as a function of the number of sampler steps M (top two rows) and assimilation step at the largest M (bottom two rows), for all observation scenarios.

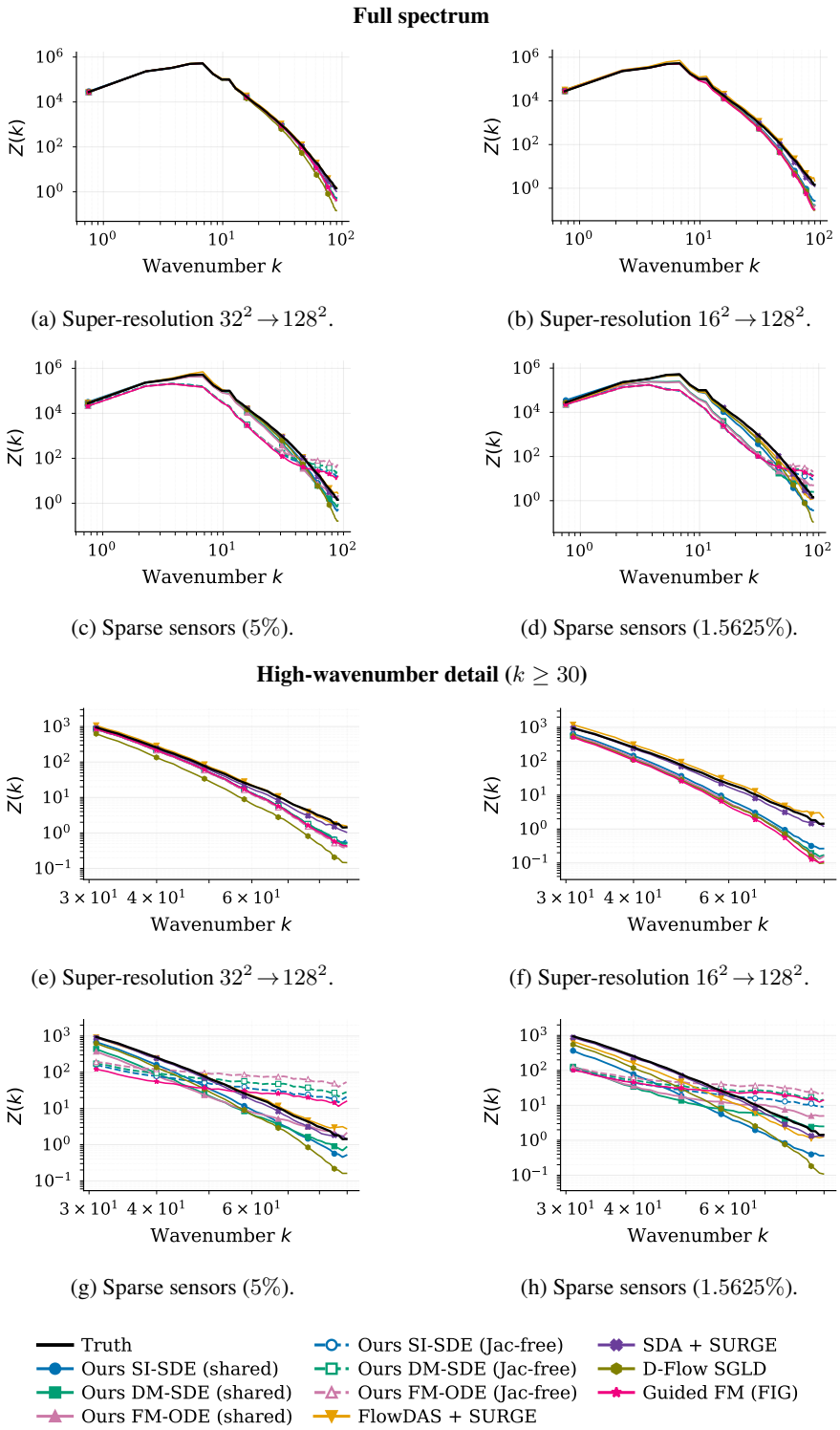


Figure 14: Navier–Stokes: radially-averaged enstrophy spectra $Z(k)$ of the posterior mean for every method (colours) against the true spectrum (black), for all four observation scenarios. Top two rows: the full wavenumber range. Bottom two rows: the same panels zoomed to $k \geq 30$, where the methods separate – a curve below the truth has an over-smoothed posterior mean, one above it carries spurious small-scale enstrophy. Spectra are averaged over the 15 assimilated steps of one held-out trajectory at the largest sampler-step count M. A method with no saved run at this M, or whose posterior diverged there, carries no curve.

I URBAN AIRFLOW TEST CASE: SETUP AND ADDITIONAL RESULTS

This appendix provides additional information and results for the urban-airflow case, including qualitative reconstructions and error trajectories over 145 forecast steps. The reported run uses one held-out trajectory, $E = 64$ ensemble members, $M = 50$ sampler steps, and sensor densities of 1.5625% and 0.78125% of fluid cells. Starting from a different held-out flow state, observations are assimilated every third forecast step, for 49 posterior updates. Metrics are evaluated at every step over fluid cells only, with buildings masked. Due to computational complexities, this case includes only solver-free generative methods; no conventional-filter or KL reference is reported.

The uDALES solver. The data are generated with uDALES (Suter et al., 2022), an open-source, building-resolving large-eddy simulation code for urban microclimate and air quality derived from DALES. It solves the filtered incompressible Navier–Stokes and scalar-transport equations on a staggered Cartesian grid, using the Vreman model for subgrid stresses and an immersed-boundary representation of buildings. Momentum, temperature, and building-surface energy are coupled within a periodic $128 \times 128 \times 128\,\mathrm{m}^3$ domain discretized at $1\,\mathrm{m}$ resolution.

Qualitative reconstructions. Figures 15–18 show the true field, observations, reconstruction, uncertainty, and error for each retained method. Each figure corresponds to one variable and sensor density. As in the Navier–Stokes figures, we omit the DM-SDE variants and show only the SURGE versions of FlowDAS and SDA.

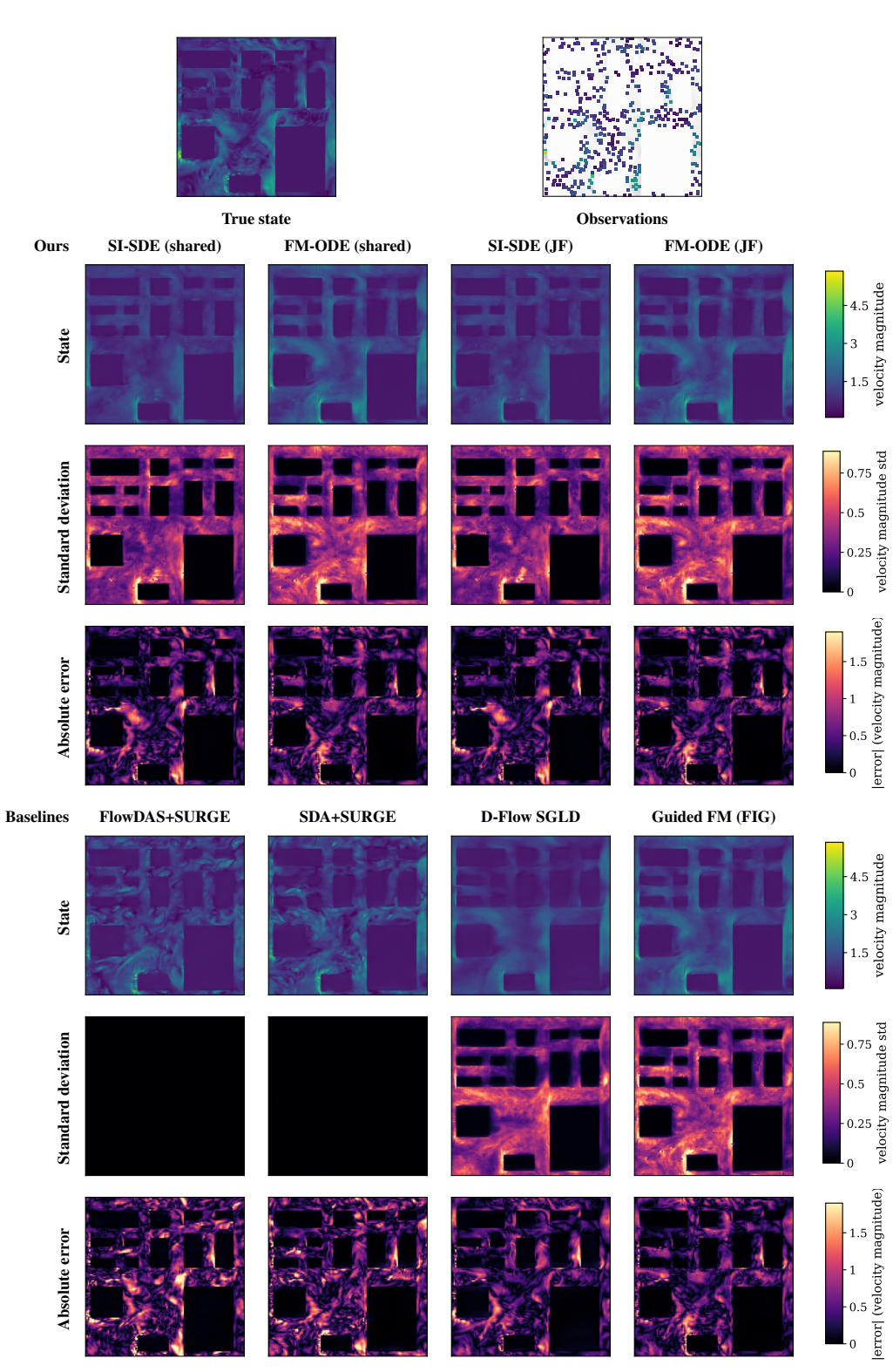


Figure 15: Urban airflow, velocity magnitude, sparse sensors (0.78125%): true state and velocity observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final forecast step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance; solid cells are masked.

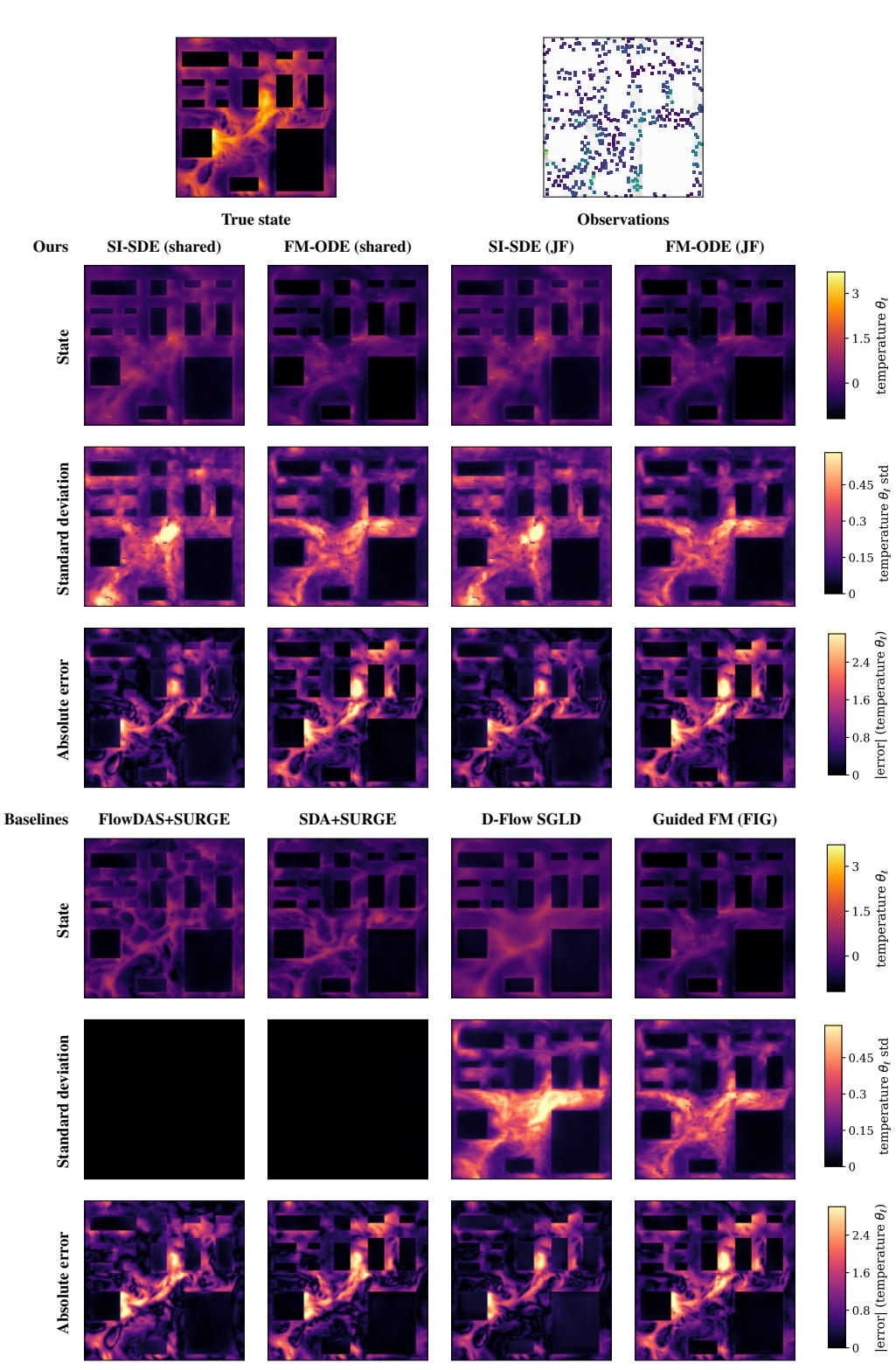


Figure 16: Urban airflow, potential temperature θ_ℓ, sparse sensors (0.78125%): true state and velocity observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final forecast step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance; solid cells are masked.

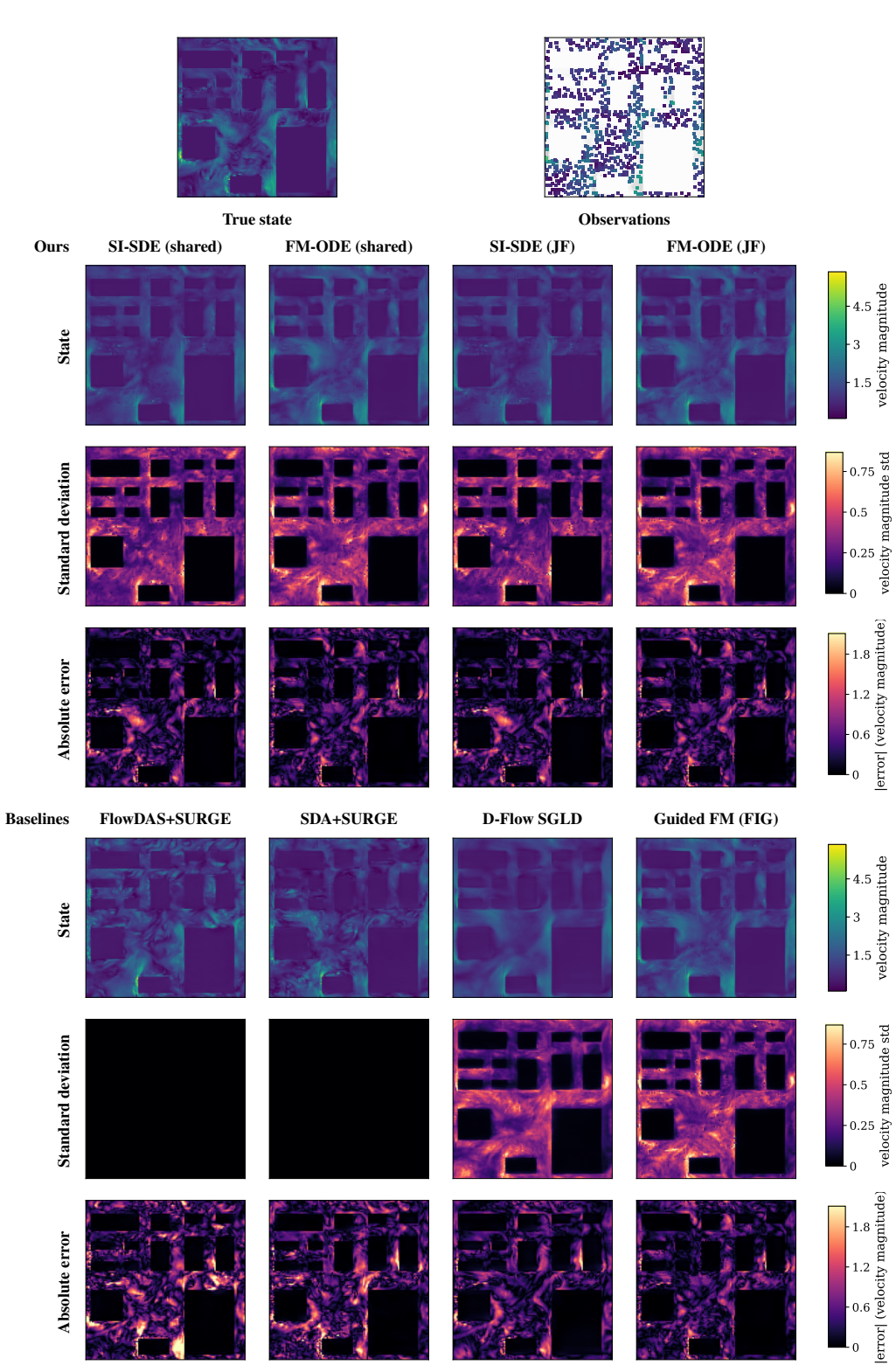


Figure 17: Urban airflow, velocity magnitude, sparse sensors (1.5625%): true state and velocity observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final forecast step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance; solid cells are masked.

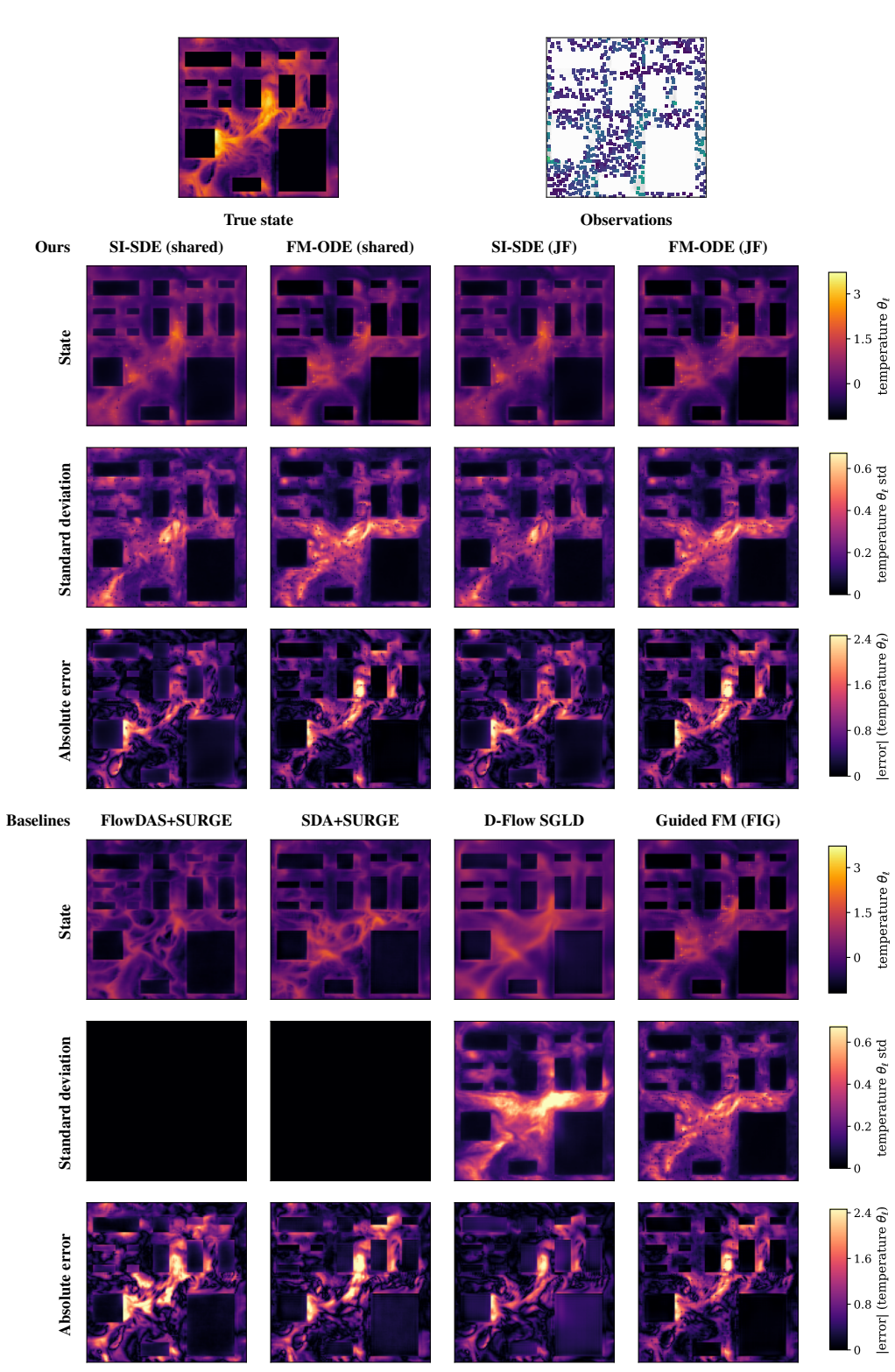
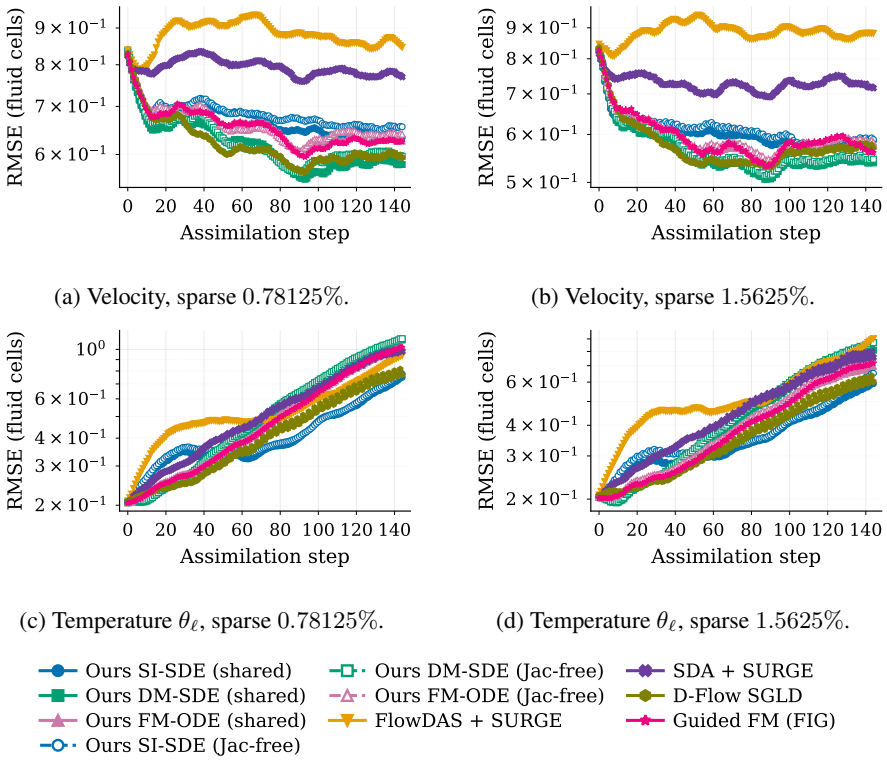


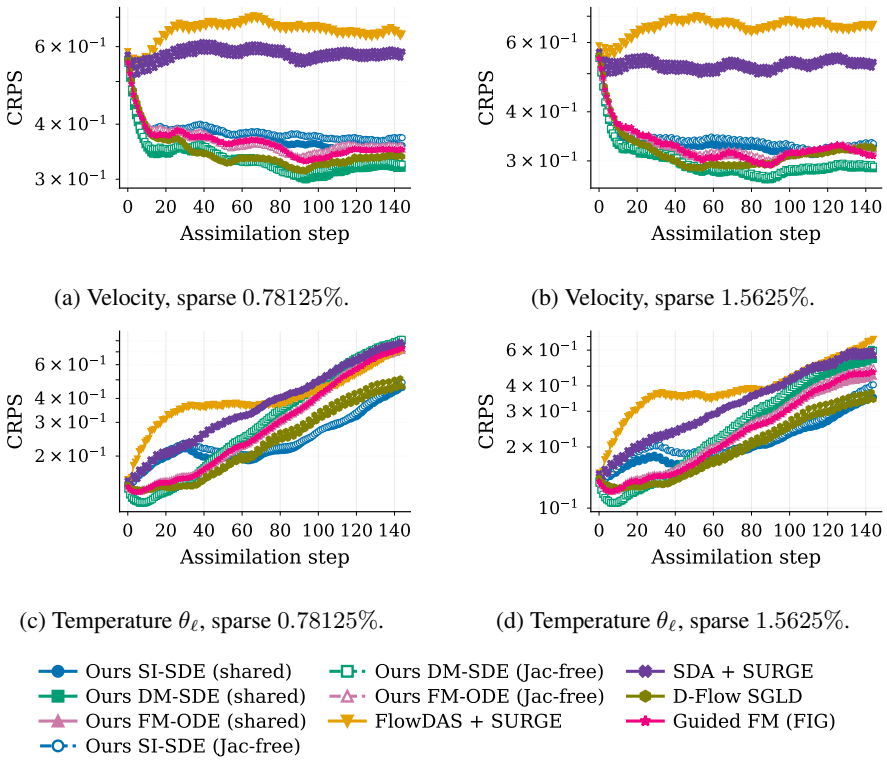
Figure 18: Urban airflow, potential temperature θ_ℓ, sparse sensors (1.5625%): true state and velocity observations (top), followed by the posterior-mean state, ensemble standard deviation, and absolute error for our methods and the retained baselines at the final forecast step. "Shared" denotes the ensemble-shared covariance and "JF" the Jacobian-free covariance; solid cells are masked.



(a) Velocity, sparse 0.78125%. (b) Velocity, sparse 1.5625%.

(c) Temperature θ_ℓ, sparse 0.78125%. (d) Temperature θ_ℓ, sparse 1.5625%.

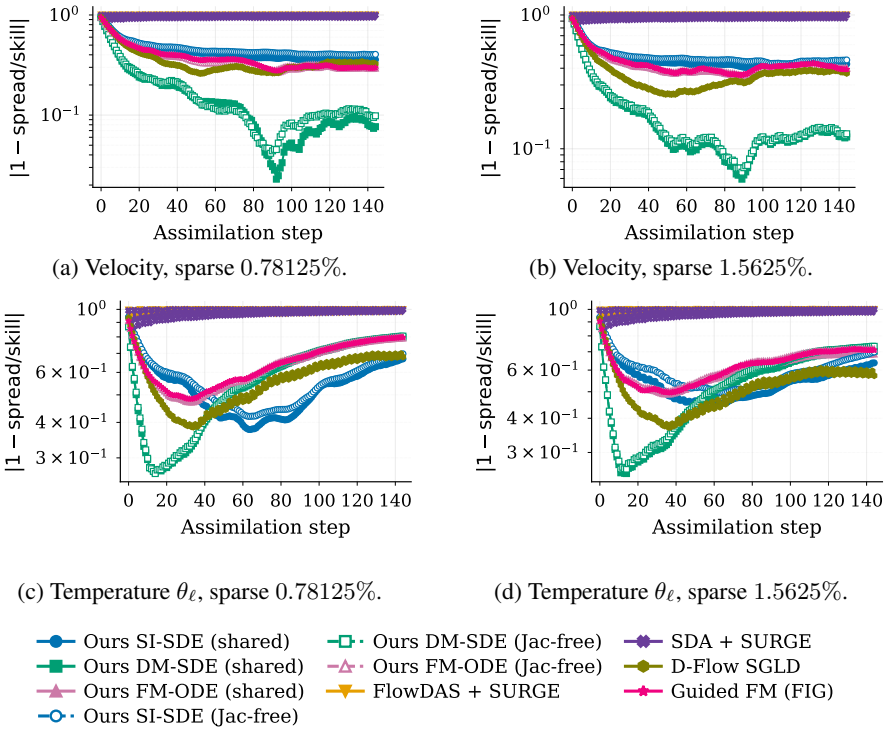
Figure 19: Urban airflow: RMSE (fluid cells) vs. forecast step for velocity and temperature at both sensor densities; observations are assimilated every third step.



(a) Velocity, sparse 0.78125%. (b) Velocity, sparse 1.5625%.

(c) Temperature θ_ℓ, sparse 0.78125%. (d) Temperature θ_ℓ, sparse 1.5625%.

Figure 20: Urban airflow: CRPS vs. forecast step for velocity and temperature at both sensor densities; observations are assimilated every third step.



(a) Velocity, sparse 0.78125%.

(b) Velocity, sparse 1.5625%.

(c) Temperature θ_ℓ, sparse 0.78125%.

(d) Temperature θ_ℓ, sparse 1.5625%.

Figure 21: Urban airflow: spread–skill deviation vs. forecast step for velocity and temperature at both sensor densities; observations are assimilated every third step.