LEVDA: Latent Ensemble Variational Data Assimilation via Differentiable Dynamics

Phillip Si **Peng Chen**
School of Computational Science and Engineering
Georgia Institute of Technology, Atlanta, GA 30332
{psi6, pchen402}@gatech.edu

Abstract

Long-range geophysical forecasts are fundamentally limited by chaotic dynamics and numerical errors. While data assimilation can mitigate these issues, classical variational smoothers require computationally expensive tangent-linear and adjoint models. Conversely, recent efficient latent filtering methods often enforce weak trajectory-level constraints and assume fixed observation grids. To bridge this gap, we propose Latent Ensemble Variational Data Assimilation (LEVDA), a nonlinear ensemble-space variational smoother that operates in the low-dimensional latent space of a pretrained differentiable neural dynamics surrogate. By performing four-dimensional ensemble-variational optimization within an ensemble subspace, LEVDA jointly assimilates states and unknown parameters without adjoint code or auxiliary observation-to-latent encoders. Leveraging the fully differentiable, continuous-in-time-and-space nature of the surrogate, LEVDA accommodates highly irregular sampling at arbitrary spatiotemporal locations. Across three challenging geophysical benchmarks, LEVDA matches or outperforms state-of-the-art baselines under severe observational sparsity while providing improved uncertainty quantification. Simultaneously, it achieves substantially improved assimilation accuracy and efficiency compared to alternative smoothing methods. Code is available at https://github.com/ps789/levda.

1 Introduction

The accuracy of long-range forecasts in many geophysical systems is fundamentally limited by intrinsic chaos and a sensitive dependence on initial conditions (Lorenz, 1963, 1969, 1982). Global weather prediction is a prominent example, but similar challenges arise in ocean dynamics, geophysical flows, and other high-dimensional spatiotemporal processes. In both traditional numerical models and newer data-driven surrogates, forecast skill is further degraded by numerical errors stemming from uncertainty and discretization (Bauer et al., 2015; Dueben & Bauer, 2018). Data assimilation (DA) mitigates these issues by systematically incorporating observational data into predictive models to infer initial conditions—and potentially unknown parameters—that are consistent with historical measurements, thereby reducing error growth over longer lead times (Kalnay, 2003; Carrassi et al., 2018; Bannister, 2017).

Recent score-based approaches, such as Latent-EnSF (Si & Chen, 2025) and LD-EnSF (Xiao et al., 2026), have demonstrated strong performance for filtering under sparse observations by leveraging learned generative priors in a low-dimensional latent space. However, because these methods typically operate sequentially, they provide weaker trajectory-level constraints over a multi-time assimilation window—a critical drawback when enforcing dynamical consistency or performing reliable joint state–parameter estimation. Furthermore, they often require the training of additional amortized components, such as observation-to-latent encoders, whose underlying assumptions can be restrictive. While score-based smoothing methods like SDA (Rozet & Louppe, 2023) target the smoothing

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

posterior directly, they can be computationally intensive and frequently treat dynamics implicitly rather than as an explicit, differentiable prior. Similarly, approaches like Appa (Andry et al., 2025) that learn latent dynamics via spatiotemporal diffusion models remain challenging to adapt for joint state–parameter estimation and highly irregular spatiotemporal observation networks.

Conversely, variational data assimilation methods, such as 4DVar (Rabier et al., 1998), enforce strict dynamical consistency by fitting trajectories to all observations within a specified time window, but are bottlenecked by repeated nonlinear model runs and the derivation of gradients via explicit tangent-linear and adjoint models. Ensemble formulations like four-dimensional ensemble-variational (4DEnVar) method (Zhu et al., 2022) mitigate this by replacing adjoints with flow-dependent covariance information derived from an ensemble. While recent machine-learning accelerated methods (e.g., FengWu-4DVar (Xiao et al., 2024)) substantially speed up assimilation by reducing forward and gradient costs, they typically yield a single deterministic MAP trajectory; extracting calibrated uncertainty quantification (UQ) requires additional approximations (Hodyss et al., 2016). Thus, a critical gap remains: achieving the rigorous multi-time constraints of variational smoothing and the reliable UQ of ensemble methods, while maintaining the efficiency of low-dimensional surrogates.

To bridge this gap, we introduce *Latent Ensemble Variational Data Assimilation* (LEVDA), inspired by 4DEnVar, a smoothing-based framework that performs nonlinear ensemble-space optimization directly in the latent space of a pretrained differentiable dynamical surrogate (e.g., an LDNet (Regazzoni et al., 2024)). LEVDA enables gradient-based assimilation through the surrogate dynamics without requiring adjoint models, and it avoids the need to train auxiliary observation-to-latent encoders by computing observation misfits directly from decoded latent trajectories. By leveraging the fully differentiable and continuous-in-time-and-space nature of the surrogate, LEVDA effortlessly accommodates observations at arbitrary, off-grid spatiotemporal locations and natively supports the joint assimilation of states and unknown parameters. Consequently, LEVDA confers several practical benefits that are difficult to achieve simultaneously with prior data-driven approaches: it enforces multi-time dynamical constraints, handles highly irregular observation operators, and substantially reduces computational complexity compared to full-state ensemble variational smoothers. We evaluate LEVDA on three challenging geophysical benchmarks, demonstrating state-of-the-art accuracy under extreme observational sparsity with improved uncertainty quantification, while substantially reducing runtime costs relative to other smoothing methods. Relative to 4DEnVar, LEVDA keeps the nonlinear coupling between the ensemble coefficients and the observation misfit by differentiating through the latent dynamics and the decoder instead of linearizing the coefficient-to-observation map (Section 3.2); Table 7 (Appendix G) summarizes the resulting capabilities.

2 Background

We briefly review the variational frameworks that form the foundation of LEVDA. We first define the standard state-space model and observation process, followed by a summary of variational smoothing (4DVar) and its ensemble-based counterpart (4DEnVar). Finally, we introduce the differentiable latent dynamical surrogates that make efficient, low-dimensional assimilation possible.

2.1 Dynamical Systems

Consider a physical system characterized by a state vector $x_t \in \mathbb{R}^{d_x}$ and potentially time-varying parameters $u_t \in \mathbb{R}^{d_u}$ at discrete time steps $t \in \mathbb{Z}^+$. Given initial conditions (x_0, u_0), we model the forward dynamics as a Markov process:

$$x_t = M(x_{t-1}, u_{t-1}) + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, Q_t), \tag{1}$$

$$u_t = M_u(u_{t-1}) + \eta_t, \quad \eta_t \sim \mathcal{N}(0, Q_t^u), \tag{2}$$

for $t = 1, 2, \ldots$, where the state transition $M : \mathbb{R}^{d_x} \times \mathbb{R}^{d_u} \to \mathbb{R}^{d_x}$ and parameter evolution $M_u : \mathbb{R}^{d_u} \to \mathbb{R}^{d_u}$ may be nonlinear. For strictly static parameters, we simply take $M_u(u) = u$.

Observations of the system are acquired via

$$y_t = H_t(x_t) + \gamma_t, \qquad \gamma_t \sim \mathcal{N}(0, R_t), \tag{3}$$

where the observation operator $H_t : \mathbb{R}^{d_x} \to \mathbb{R}^{d_y}$ can be time-varying to accommodate irregular sampling or moving sensors. We assume the noise sequences $\{\epsilon_t\}$, $\{\eta_t\}$, and $\{\gamma_t\}$ are mutually independent, with covariance matrices that may vary over time.

2.2 4DVar

To unify the notation for joint state-parameter estimation, we define the augmented state vector $z_t = (x_t, u_t) \in \mathbb{R}^{d_x+d_u}$. Under the dynamical model in Equations (1) and (2), the one-step transition densities and observation likelihood are standard Gaussians: $p(x_t \mid x_{t-1}, u_{t-1}) = \mathcal{N}\big(M(x_{t-1}, u_{t-1}), Q_t\big)$, $p(u_t \mid u_{t-1}) = \mathcal{N}\big(M_u(u_{t-1}), Q_t^u\big)$, $p(y_t \mid x_t) = \mathcal{N}\big(H_t(x_t), R_t\big)$. Equivalently, the augmented transition density factors as $p(z_t \mid z_{t-1}) = p(x_t \mid x_{t-1}, u_{t-1})p(u_t \mid u_{t-1})$, and the likelihood depends only on the physical state, $p(y_t \mid z_t) = p(y_t \mid x_t)$.

Given a sequence of past observations $y_{1:t-1}$ and observations $y_{t:t+\tau}$ in a fixed assimilation window of length τ, the smoothing posterior is given by Bayes' rule:

$$p(z_{t:t+\tau} \mid y_{1:t+\tau}) \propto p(z_t \mid y_{1:t-1}) \prod_{i=1}^{\tau} p(z_{t+i} \mid z_{t+i-1}) \prod_{i=0}^{\tau} p(y_{t+i} \mid x_{t+i}). \tag{4}$$

Variational smoothing, or 4DVar (Rabier et al., 1998), computes a maximum a posteriori (MAP) estimate of the trajectory over the assimilation window. In incremental implementations, the filtering prior is approximated by a Gaussian background, $p(z_t \mid y_{1:t-1}) \approx \mathcal{N}(z_t^b, B)$, and the optimal state is found by minimizing the negative log-posterior. Under the Gaussian assumptions established above, the cost function of the strong-constraint 4DVar (assuming $Q_t = 0$ and $Q_t^u = 0$) is given by

$$J(z_{t:t+\tau}) = \frac{1}{2}\|z_t - z_t^b\|^2_{B^{-1}} + \frac{1}{2}\sum_{i=0}^{\tau} \|y_{t+i} - H_{t+i}(x_{t+i})\|^2_{R_{t+i}^{-1}}, \tag{5}$$

where the future states (x_{t+i}, u_{t+i}) are strictly determined by the initial condition (x_t, u_t) via the forward dynamics, allowing the optimization over z_t using tangent-linear or adjoint models.

2.3 4DEnVar

4DEnVar is an ensemble-space analogue of incremental 4DVar that avoids the need for explicit tangent-linear and adjoint models by restricting the analysis increment to a subspace spanned by an ensemble (Courtier et al., 1994; Zhu et al., 2022).

Let z_t^b be a background initial condition (e.g., the mean of the filtering prior) and let $\{z_t^{(k)}\}_{k=1}^K$ be an ensemble of initial conditions with mean $\bar{z}_t = \frac{1}{K}\sum_{k=1}^K z_t^{(k)}$. We define the perturbations as $\delta z_t^{(k)} = z_t^{(k)} - \bar{z}_t$, which form the ensemble perturbation matrix: $P_z = \frac{1}{\sqrt{K-1}}[\delta z_t^{(1)}, \ldots, \delta z_t^{(K)}]$. This yields an empirical background covariance $B \approx P_z P_z^T$, allowing 4DEnVar to parameterize the initial analysis increment as $\delta z_t = P_z \alpha$ for $\alpha \in \mathbb{R}^K$.

To evaluate the observation-space misfit, each ensemble member is propagated through the nonlinear model to obtain trajectories $\{z_{t+i}^{(k)}\}_{k=1}^K$ for $i = 0, \ldots, \tau$. Let $x_{t+i}^{(k)}$ denote the physical state component of $z_{t+i}^{(k)}$. For each observation time, we compute the observation-space perturbations:

$$\delta y_{t+i}^{(k)} = H_{t+i}(x_{t+i}^{(k)}) - \frac{1}{K}\sum_{\ell=1}^{K} H_{t+i}(x_{t+i}^{(\ell)}), \quad i = 0, \ldots, \tau, \tag{6}$$

and assemble the corresponding matrices $P_{y,t+i} = \frac{1}{\sqrt{K-1}}[\delta y_{t+i}^{(1)}, \ldots, \delta y_{t+i}^{(K)}]$. Let x_{t+i}^b be the background trajectory obtained by propagating z_t^b, and define the innovations $d_{t+i} = y_{t+i} - H_{t+i}(x_{t+i}^b)$. The incremental 4DEnVar objective in the reduced ensemble space is then:

$$J(\alpha) = \frac{1}{2}\alpha^T\alpha + \frac{1}{2}\sum_{i=0}^{\tau} \|d_{t+i} - P_{y,t+i}\alpha\|^2_{R_{t+i}^{-1}}. \tag{7}$$

Minimizing this quadratic cost yields the optimal coefficients α^*, and the analysis update is computed as $z_t^a = z_t^b + P_z\alpha^*$ (yielding the updated physical states and parameters (x_t^a, u_t^a)). Operationally, outer loops are often used to update the background trajectory and recompute perturbations, addressing nonlinearities in the dynamics and observation operators. Additionally, localization and inflation are typically applied to maintain filter stability in high-dimensional regimes (Zhu et al., 2022).

2.4 Differentiable Latent Dynamics

Latent Dynamics Networks (LDNets) (Regazzoni et al., 2024) provide a compact, differentiable surrogate for high-dimensional dynamical systems by jointly learning a latent vector field and a decoder to physical space, and have been used for metropolitan flood prediction (Si et al., 2026). We represent the latent state as a continuous-time function $s(t) \in \mathbb{R}^{d_s}$ and let $u(t)$ denote any (known or unknown) time-varying parameters or forcings. A typical LDNet defines the latent evolution via an ordinary differential equation (ODE):

$$\frac{d}{dt}s(t) \;=\; \mathcal{F}_\theta(s(t), u(t)), \tag{8}$$

where the neural vector field $\mathcal{F}_\theta$ is integrated via a differentiable ODE solver. For instance, discretizing time as $t_k = t_0 + k\Delta t$ with piecewise constant $u(t)$ yields the forward-Euler update:

$$s_{k+1} \;=\; s_k + \Delta t\, \mathcal{F}_\theta(s_k, u_k), \qquad k = 0, 1, \ldots, \tag{9}$$

where $s_k \approx s(t_k)$ and $u_k \approx u(t_k)$. Given a latent trajectory $s(t)$, a decoder D_ϕ maps the latent states back to physical quantities. In particular, we use coordinate-based neural field decoders that enable continuous spatial evaluation at any query point $\xi \in \Omega$ via

$$\hat{x}(t, \xi) = D_\phi(s(t), \xi). \tag{10}$$

Following Xiao et al. (2026), we adopt a shifted discrete-time convention with fixed initialization $s_{-1} = 0$ to accommodate varying initial conditions. Additionally, the step size Δt can be treated as a learnable parameter and optimized jointly with weights (θ, ϕ).

Given a dataset of N physical trajectories $\{x^{(n)}(t_k, \cdot)\}_{k=0}^{T}$, the surrogate model is trained to minimize reconstruction error of decoded latent trajectories. To reduce cost and handle irregular spatial data, we sample a subset of query points $\Xi_{n,k} \subset \Omega$ for each trajectory n at time t_k, evaluating the decoder only at these points. With the predicted field $\hat{x}^{(n)}(t_k, \xi) = D_\phi(s^{(n)}(t_k), \xi)$, the objective is:

$$\mathcal{L}(\theta, \phi) = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=0}^{T}\frac{1}{|\Xi_{n,k}|}\sum_{\xi\in\Xi_{n,k}}\left\|\hat{x}^{(n)}(t_k, \xi) - x^{(n)}(t_k, \xi)\right\|_2^2. \tag{11}$$

Here, each latent trajectory $s^{(n)}(t_k)$ is generated by integrating the learned dynamics forward. Crucially, this combination of continuous-time latent evolution, coordinate-based decoding, and end-to-end differentiability forms the technical foundation of LEVDA, enabling gradient-based variational smoothing across arbitrary, grid-free spatiotemporal observation networks.

3 Methodology

LEVDA integrates differentiable latent dynamics with ensemble-variational smoothing to formulate a latent-space analogue of 4DEnVar. For a given assimilation window starting at time t, the method takes three inputs: (i) a fully differentiable pretrained latent surrogate (trained via (11)), (ii) an ensemble representing the background distribution of window-initial latent variables, and (iii) a collection of (potentially spatiotemporally irregular) observations. In turn, LEVDA produces an analysis ensemble of latent states and parameters for the window onset, along with decoded physical trajectories that harmonize the observations with the latent dynamics.

3.1 Latent Variational Data Assimilation (LVDA)

We assume access to a pretrained differentiable latent surrogate that evolves an augmented latent state $z(t) = (s(t), u(t))$ and decodes $s(t)$ to physical space via D_ϕ. Concretely, we model the latent dynamics by

$$\frac{d}{dt}s(t) = \mathcal{F}_\theta(s(t), u(t)), \qquad \frac{d}{dt}u(t) = \mathcal{G}(u(t)), \tag{12}$$

where $\mathcal{G}$ encodes any assumed parameter evolution (e.g., $\mathcal{G} \equiv 0$ for static parameters). The decoded physical state is defined as $x(t) = D_\phi(s(t))$. Observations at time t_i follow the standard state-space model from Equation (3); crucially, however, the observation operator H_{t_i} is applied in physical space *after* decoding. This means LVDA does not require an additional observation-to-latent encoder

as used in LD-EnSF (Xiao et al., 2026). The assimilation process simply differentiates through the composite map $z(t) \mapsto s(t) \mapsto x(t) = D_\phi(s(t)) \mapsto H_{t_i}(x(t_i))$.

For a smoothing window anchored at time t, let $\mathcal{T}_t = \{t_1, \ldots, t_m\}$ denote the sequence of observation times within the window. These observations can be irregular and need not lie on a fixed grid. To handle arbitrary observation times, any $\tilde{t} \in \mathcal{T}_t$ falling between the fixed discrete integration steps t_k and t_{k+1} of (9) is parameterized as $\tilde{t} = t_k + \alpha\Delta t$ with $\alpha \in [0, 1)$. Consistent with a forward-Euler discretization, we approximate the latent state at the observation time via linear interpolation:

$$s(\tilde{t}) \approx s(t_k) + \alpha\Delta t\dot{s}(t_k), \tag{13}$$

where $\dot{s}(t_k) = \mathcal{F}_\theta(s(t_k), u(t_k))$. Because this continuous-time formulation gracefully handles off-grid data, LVDA naturally accommodates a varying number of available observations per assimilation window.

LVDA computes a maximum a posteriori (MAP) estimate of the window-initial latent variables $z(t) = (s(t), u(t))$, which in turn determines the implied trajectory under Equation (12). We approximate the filtering prior with a Gaussian background $z(t) \sim \mathcal{N}(z_t^b, B)$, where $z_t^b = (s_t^b, u_t^b)$, and assume a block-diagonal covariance $B = \mathrm{diag}(B_s, B_u)$. In practice, B_s and B_u can be estimated from the background ensemble (e.g., using a diagonal or shrinkage estimator). The strong-constraint LVDA objective is therefore:

$$J(s(t), u(t)) = \frac{1}{2}\|s(t) - s_t^b\|^2_{B_s^{-1}} + \frac{1}{2}\|u(t) - u_t^b\|^2_{B_u^{-1}} + \frac{1}{2}\sum_{t_i \in \mathcal{T}_t} \left\|y_{t_i} - H_{t_i}\big(D_\phi(s(t_i))\big)\right\|^2_{R_{t_i}^{-1}}, \tag{14}$$

where the state $s(t_i)$ at each observation time is obtained by integrating Equation (12) forward from the initial condition $(s(t), u(t))$.

3.2 Latent Ensemble Variational Data Assimilation

While LVDA provides a clean MAP formulation, its direct application requires specifying (and inverting) background covariances over the augmented latent variables (s, u) and optimizing in the full augmented latent dimension $d_s + d_u$. LEVDA is designed to make latent variational smoothing computationally efficient and robust by: (i) restricting the update to an ensemble subspace, thereby optimizing over only K coefficients; (ii) avoiding the explicit construction or inversion of B_s and B_u by relying on the empirical covariance of the ensemble; (iii) leveraging a flow-dependent, data-adaptive geometry via ensemble perturbations; and (iv) returning an analysis *ensemble* of latent trajectories rather than a single deterministic MAP, from which uncertainty in the decoded physical state can be characterized. In the latent space, specifying an operational B_s and B_u is nontrivial: a static estimate must be regularized and inverted, and misspecification degrades the analysis. The ensemble-subspace parameterization represents $B \approx P_z P_z^T$ implicitly through perturbations, with no estimation, inversion, or manual covariance modeling, while providing flow-dependent structure.

Let $\{z_t^{b,(k)}\}_{k=1}^K$ be an ensemble of background window-initial latent variables, where $z_t^{b,(k)} = (s_t^{b,(k)}, u_t^{b,(k)})$ and the ensemble mean is $\bar{z}_t^b = \frac{1}{K}\sum_{k=1}^K z_t^{b,(k)}$. Operationally, this background ensemble can be obtained by propagating the previous analysis ensemble forward under the latent surrogate, or by encoding an ensemble of full-state estimates into the latent space (if an encoder is available). We define the perturbations as $\delta z_t^{(k)} = z_t^{b,(k)} - \bar{z}_t^b$. To prevent filter collapse, we optionally apply multiplicative and additive inflation:

$$\delta z_t^{(k)} \leftarrow \rho\, \delta z_t^{(k)} + \epsilon^{(k)}, \quad \rho \geq 1, \;\; \epsilon^{(k)} \sim \mathcal{N}(0, \sigma^2 I_{d_s+d_u}),$$

and then form the ensemble perturbation matrix: $P_z = \frac{1}{\sqrt{K-1}}[\delta z_t^{(1)}, \ldots, \delta z_t^{(K)}]$.

To define the analysis, LEVDA constrains the initial conditions to an affine subspace: $z_t(\alpha) = \bar{z}_t^b + P_z\alpha$, where $\alpha \in \mathbb{R}^K$. This formulation implicitly adopts a standard Gaussian prior $\alpha \sim \mathcal{N}(0, I)$ and an empirical background covariance $B \approx P_z P_z^T$. For any candidate α, the latent trajectory and corresponding decoded states are obtained by integrating Equation (12). The optimal analysis is then found by minimizing the nonlinear ensemble-space variational objective:

$$J(\alpha) = \frac{1}{2}\|\alpha\|_2^2 + \frac{1}{2}\sum_{t_i \in \mathcal{T}_t} \left\|y_{t_i} - H_{t_i}\big(D_\phi(s(t_i; \alpha))\big)\right\|^2_{R_{t_i}^{-1}}, \tag{15}$$

where $s(t_i;\alpha)$ denotes the latent state at time t_i obtained by integrating forward from the initial condition $z_t(\alpha)$. In our implementation the regularization of the K coefficient vectors is centered at the members' own backgrounds rather than at the ensemble mean: with $\Delta A = [\alpha^{(1)} - \sqrt{K-1}\,e_1, \ldots, \alpha^{(K)} - \sqrt{K-1}\,e_K]$ the penalty is $\frac{1}{K^2}\|\Delta A\,\Delta A^\top\|_F^2$, plus a quadratic penalty on the parameter increment of each member, and the K coupled problems are solved jointly by one L-BFGS run; member j is thus pulled towards its own background member and the members do not share a common minimizer. We optimize a set of coefficient vectors $\{\alpha^{(j)}\}_{j=1}^K$, where each is initialized to a background member (i.e., $\alpha^{(j)} = \sqrt{K-1}e_j$, such that $\bar{z}_t^b + P_z\alpha^{(j)}$ recovers the j-th background member when $\epsilon^{(k)} = 0$). These problems are solved using L-BFGS via automatic differentiation through the latent solver. Because the observation misfit is nonlinearly coupled to α, these optimization trajectories allow the analysis members to capture local nonlinearities. The forward and backward passes of the K members are batched, so the joint solve costs one batched L-BFGS run. Finally, we construct the analysis ensemble by setting

$$z_t^{a,(j)} = \bar{z}_t^b + P_z\alpha_*^{(j)}, \quad j = 1, \ldots, K,$$

yielding the updated parameters $u_t^{a,(j)}$ and decoded analysis states $x_t^{a,(j)} = D_\phi(s_t^{a,(j)})$. Note that LEVDA never decodes the full physical state during assimilation: the LDNet decoder is a coordinate-based neural field, so the misfit is evaluated only at the observed locations, and the cost of each evaluation scales with the number of observed points times the latent dimension d_s, independent of the grid resolution.

Connection to 4DEnVar. LEVDA adopts the ensemble-subspace philosophy of 4DEnVar, restricting the analysis increment to an affine subspace spanned by background perturbations. The primary distinction lies in sensitivity handling. Classical 4DEnVar avoids explicit tangent-linear or adjoint models by using nonlinear ensemble propagation to form an empirical linear map from coefficients to observation increments (e.g., via P_{y,t_i}); this yields a purely quadratic objective solvable without model differentiation. That quadratic problem is the first Gauss–Newton iterate of Equation (15) at $\alpha = 0$ with the ensemble-estimated sensitivity in place of the exact one, so LEVDA can be read as incremental 4D-Var (Courtier et al., 1994) carried out in the ensemble subspace with exact gradients, in the spirit of the iterative ensemble Kalman smoother (Bocquet & Sakov, 2014). Conversely, LEVDA maintains the nonlinear coupling between the observation misfit and coefficients through the differentiable latent dynamics and decoder, changing the optimization problem itself. This enables assimilation at arbitrary spatiotemporal locations without auxiliary observation-to-latent encoders. It computes $\nabla_\alpha J$ in Equation (15) via exact automatic differentiation through the latent solver, D_ϕ, and H_{t_i}, leveraging the gradients of the pretrained surrogate without requiring adjoint code for the original high-dimensional simulator. While LEVDA mirrors a full-state ensemble-variational smoother as the latent dimension approaches the full state dimension, its low-dimensional formulation with differentiable latent dynamics provides significant computational savings while retaining rigorous, dynamics-aware constraints. Appendix B provides a structural approximation analysis that decomposes LEVDA's error into ensemble-subspace, surrogate-objective, and representational components.

4 Experiments

4.1 Experimental Setup

We evaluate LEVDA on three benchmark settings adapted from Xiao et al. (2026): (i) Kolmogorov flow with uncertain viscosity, (ii) tsunami propagation with an unknown source, and (iii) global atmospheric dynamics with an unknown forcing term. Dataset details are provided in Appendix A.

Implementation Details. For all experiments, we use the LDNets from Xiao et al. (2026) with latent dimensions $D_s = 10$ (Kolmogorov), $D_s = 12$ (Tsunami), and $D_s = 52$ (Atmosphere). Unless otherwise stated, we use an ensemble size of $K = 100$ for all latent methods (LEVDA, LD-EnSF, Latent-EnSF) and $K = 40$ for the full-state 4DEnVar baseline (constrained by high cost of full-state propagation). We optimize Equation 15 using L-BFGS with up to 10 iterations per solve. Step sizes are set to 0.1 for Kolmogorov and tsunami, and 0.001 for the atmospheric setting. We apply additive inflation to the latent ensemble (6×10^{-4} for state, 1×10^{-4} for parameters). A single outer loop is typically sufficient; additional outer loops did not significantly improve performance.

Observations are generated by subsampling the ground truth (see Table 2) and adding Gaussian noise at a 10% noise-to-signal ratio (defined in Appendix D); every method assimilates the same noisy observations, and the objective uses $R_t = \sigma_a^2 I$ with a single assumed noise level σ_a (Appendix D). For LEVDA, the background ensemble is generated by taking a different trajectory from the dataset, perturbing the unknown system parameters, and propagating through the surrogate to the window start.

4.2 Baselines

We evaluate LEVDA against seven baselines spanning filtering and smoothing in both latent and full state spaces. All methods utilize identical observations and noise levels for a fair comparison.

Full-State Filtering. We use local ensemble transform Kalman filter (LETKF) (Hunt et al., 2007) and ensemble score filter (EnSF) (Bao et al., 2024). These operate directly on the high-dimensional grid but often struggle with sparse observations and nonlinear dynamics.

Full-State Smoothing. We use 4DEnVar (Zhu et al., 2022) as a high-dimensional smoothing baseline. We report results only for the tsunami setting because of the substantial engineering and runtime costs required at our target resolutions. Specifically, full-state 4DEnVar operates directly in the state space, requiring the construction of localized covariance operators and careful tuning of localization and inflation parameters. These steps are tightly coupled to the simulator's discretization and boundary conditions, introducing significant implementation complexity and overhead.

Full-State Score-Based Smoothing. We include Score-based Data Assimilation (SDA, Rozet & Louppe (2023)), which directly targets the smoothing posterior via conditional generative modeling in full space. This provides a comparison to a method explicitly designed for trajectory-level smoothing.

Latent-Space Filtering and Smoothing. We compare against Latent-EnSF (Si & Chen, 2025), which filters in a latent space (requiring coupled state and observation encoders), and LD-EnSF (Xiao et al., 2026), which uses the same differentiable dynamics as LEVDA but in a sequential filtering framework relying on an LSTM observation encoder. Additionally, we evaluate GLA (Cheng et al., 2023), a 4D variational smoother that uses a static latent mapping to produce a single MAP trajectory.

4.3 Comparison of Accuracy, Robustness, and Uncertainty

Figure 1 summarizes assimilation accuracy and uncertainty across the three benchmarks, evaluated using relative RMSE (defined in Appendix E.1). Overall, LEVDA matches or outperforms LD-EnSF without requiring an auxiliary LSTM observation encoder, and it consistently surpasses the remaining baselines under sparse observations. Additionally, Appendix D reports that LEVDA achieves more robust assimilation with varying assimilation window length τ and the ensemble size K, varying levels of observation sparsity and noise, as well as more accurate parameter estimation. Improved uncertainty quantification measured by the spread-error ratio and energy score for LEVDA is reported in Appendix E.2.

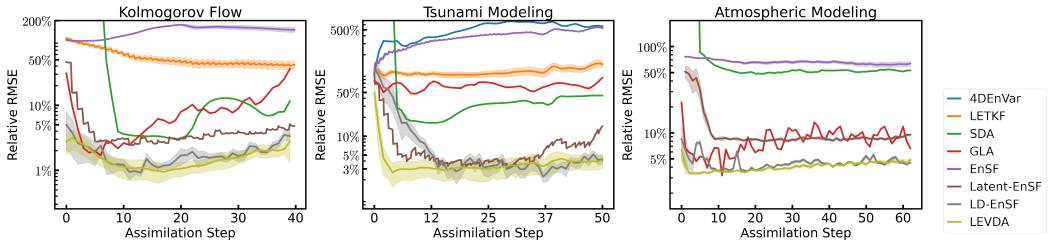


Figure 1: Relative RMSE of the ensemble-mean trajectory over time for Kolmogorov flow (left), tsunami modeling (middle), and atmospheric modeling (right); shaded regions show the standard deviation of member-wise relative RMSE.

Figure 2 visualizes the final-time zonal wind velocity at 500 hPa for the atmospheric benchmark. Despite the sparse observation density, gradient propagation through the latent surrogate remains stable, allowing LEVDA to produce realistic reconstructions comparable to LD-EnSF. Extended temporal visualizations comparing assimilation performance for Kolmogorov flow and tsunami modeling are provided in Appendix H.

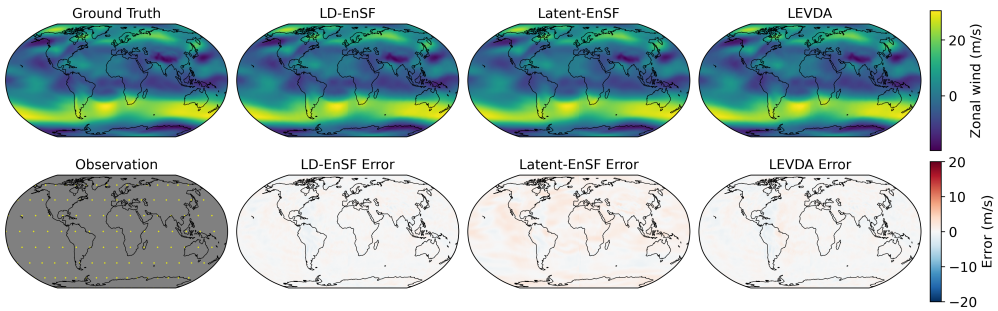


Figure 2: Atmospheric benchmark: final-time zonal wind u at 500 hPa. The visualization compares the ground truth (with sparse observations) to the assimilated states and errors for different methods.

4.4 Spatiotemporally Varying Observation Operators

LEVDA is designed to handle flexible observation patterns, relaxing the standard assumption that measurements must arrive on fixed grids at regular intervals. In practice, data is often collected by moving platforms or sampled asynchronously. We evaluate LEVDA under three progressively more realistic scenarios: time-varying observation locations, irregular observation timestamps, and a joint regime combining both irregularities. As shown in Figure 3, LEVDA maintains performance comparable to the idealized fixed-grid baseline in all cases.

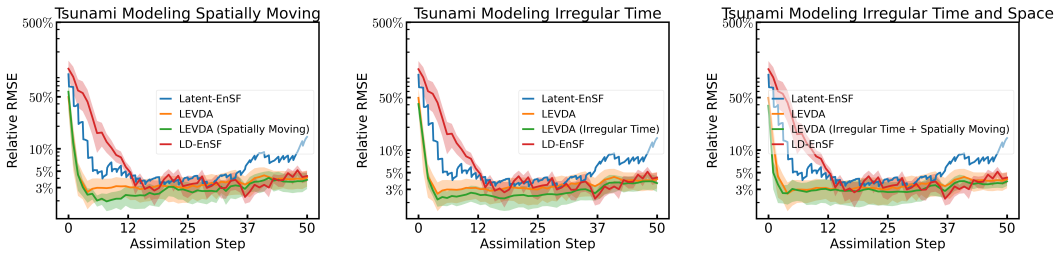


Figure 3: Effect of irregular sampling on tsunami modeling: (a) moving observation locations (randomized at each observed time), (b) irregular observation times with fixed spatial locations, and (c) joint spatiotemporal irregularity. Curves depict the relative RMSE of the ensemble-mean trajectory; shaded regions indicate the standard deviation of member-wise relative RMSE.

Moving Observation Locations. LEVDA overcomes the fixed-grid limitation of encoder-based methods like LD-EnSF, enabling assimilation from time-varying locations typical of moving platforms or drifting sensors, illustrated in Figure 13 (Appendix H). We evaluate this using the tsunami benchmark (Table 2), comparing the standard fixed-grid setup to a scenario where observation coordinates are resampled at each time step (maintaining 0.4% sparsity). Figure 3a shows that LEVDA adapts robustly to these spatially shifting observations, achieving accuracy comparable to the fixed-grid baseline.

Irregular Observation Times. LDNets model the system dynamics via a learned time derivative of the latent state, rather than a fixed-step recurrence. This continuous-time formulation allows the latent state to be propagated using standard numerical integrators (e.g., forward Euler) to any arbitrary timestamp. Consequently, LEVDA can naturally assimilate observations taken at irregular or asynchronous intervals, decoupling the assimilation process from the rigid fixed-time grids required by many discrete-time data-driven models.

In Figure 3b, we examine an irregular-sampling regime by drawing 51 observation times uniformly at random, matching the total count of the fixed-interval baseline. Observation times that do not fall at regular time steps are calculated by (13).

During smoothing, LEVDA assimilates only those observations that fall within the current time window; consequently, the number of available observations per window varies naturally. For valid comparison, all methods are evaluated on the original fixed simulation grid, even though LEVDA (Irregular Time) assimilates the asynchronously sampled data.

Despite the highly nonlinear physical dynamics, the latent trajectories are smooth by construction: one latent step already spans a coarse interval (40 simulator steps on tsunami), so the within-step evolution is locally near-linear and the interpolation in (13) is consistent with the forward-Euler latent solver. Figure 3b is the empirical check: if the interpolation error were significant, the irregular-time curve would degrade relative to the fixed-grid baseline. For stiffer latent dynamics or larger steps, a higher-order integrator with a matching interpolant (e.g., Runge–Kutta with Hermite interpolation) can be substituted without retraining the surrogate, since (13) only requires consistency with the chosen solver.

Joint Spatiotemporal Irregularity. We combine spatial subsampling and temporal irregularity to form a joint spatiotemporal stress test. At each randomly selected observation time, we independently draw a random subset of spatial locations. Consistent with the findings in Figure 3b, Figure 3c demonstrates that LEVDA maintains performance comparable to the fixed-interval baseline even under this fully irregular regime.

4.5 Runtime and Training Cost

Table 1 compares wall-clock runtimes for 4DEnVar, SDA, LD-EnSF, and LEVDA. LEVDA, despite being a smoothing method, avoids the prohibitive cost of other smoothing baselines (4DEnVar and SDA), reducing per-window assimilation by orders of magnitude. 4DEnVar was not feasible for the KF and PlanetSWE examples because its parallelized implementation conflicts with the parallelization strategy used by the underlying simulation packages, leading to incompatible execution and impractical runtime overhead. Even so, the tsunami modeling example demonstrates that latent-subspace optimization removes the full-state propagation cost that dominates 4DEnVar. The comparison is not hardware-controlled (CPU for 4DEnVar, GPU for the latent methods) and uses $K = 40$ against $K = 100$; per member, however, 4DEnVar costs about $616\,\mathrm{s}$ on tsunami against $0.11\,\mathrm{s}$ for LEVDA, a gap of three orders of magnitude that no hardware adjustment closes; the smaller ensemble understates what a $K = 100$ 4DEnVar would cost. Compared with the fastest filtering baseline LD-EnSF, LEVDA is slower per window because it performs L-BFGS optimization with backpropagation through the latent surrogate rather than a single sequential update. In exchange, LEVDA avoids the LSTM observation-to-latent encoder that LD-EnSF must pretrain (and retrain whenever the observation operator changes, see training time in Table 1), enforces multi-time dynamical constraints over a smoothing window, and returns an analysis ensemble suitable for uncertainty quantification (Appendix E.2).

Example	Kolmogorov Flow				Tsunami Modeling				Atmospheric Modeling			
Metric	4DEnVar	SDA	LD-EnSF	LEVDA	4DEnVar	SDA	LD-EnSF	LEVDA	4DEnVar	SDA	LD-EnSF	LEVDA
Encoder Training (s)	0	0	298	0	0	0	312	0	0	0	1429	0
Assimilation (s)	–	6,720	0.35	13.91	24,639	16,800	0.37	11.32	–	–	1.42	19.47
Assimilation dim.	D_x=45,000	D_x=45,000	D_s=10	D_s=10	D_x=67,500	D_x=67,500	D_s=12	D_s=12	D_x=393,216	D_x=393,216	D_s=52	D_s=52

Table 1: Wall-clock runtime comparison. We report latent assimilation dimensions (D_s) for latent methods (LD-EnSF, LEVDA) and full-state dimensions (D_x) for 4DEnVar and SDA. Runtimes reflect full-trajectory assimilation with an ensemble size of $K = 100$ for latent methods, while 4DEnVar is limited to $K = 40$ due to the high cost of full-state propagation. 4DEnVar simulations utilize an AMD 7543 CPU (64 threads; full-state parallelized propagation is CPU-bound), whereas data-driven and latent methods use an NVIDIA A6000 GPU; the comparison is therefore not hardware-controlled (per-member costs are given in the text). SDA runtime for Atmospheric Modeling is omitted because GPU memory constraints necessitated a CPU-only implementation.

5 Related Work

We position our work at the intersection of (i) latent-space filtering for sparse observations, (ii) score-based smoothing via conditional generation, and (iii) surrogate-accelerated variational data assimilation. Latent-EnSF (Si & Chen, 2025) and LD-EnSF (Xiao et al., 2026) demonstrate that score-based priors in a low-dimensional latent space render *filtering* feasible for large-scale systems with sparse observations, and LD-EnFF (Tian et al., 2026) replaces the score-based update with flow matching; as sequential methods, they enforce weaker trajectory-level constraints and typically require training auxiliary observation networks.

Score-based *smoothing* methods such as SDA (Rozet & Louppe, 2023), FlowDAS (Chen et al., 2025), and Appa (Andry et al., 2025) target trajectory-level posteriors via conditional generation. These

approaches avoid explicit adjoints but typically represent dynamics implicitly through a generative model, incur high inference cost from many denoising or flow steps, and do not natively support joint state–parameter inference or irregular spatiotemporal sampling.

Latent and reduced-order variational DA is the most closely related family. GLA (Cheng et al., 2023) performs 4D variational smoothing in a learned latent space but uses a static surrogate mapping and returns a single MAP trajectory. VAE-Var (Xiao et al., 2025) replaces the Gaussian background in 3D-Var with a VAE-learned prior, but is single-time and lacks an explicit differentiable dynamics surrogate. FengWu-4DVar (Xiao et al., 2024) replaces the numerical model and adjoint with a deep weather surrogate, but operates in the full state space and returns a deterministic MAP. Tensor-Var (Yang et al., 2025) learns feature maps in which the dynamics and observation operators become linear, so that 4D-Var reduces to a linear problem in feature space; it accommodates moving observation locations but returns a single deterministic analysis on a fixed time grid and does not estimate parameters (Table 7). LEVDA differs by combining ensemble-subspace optimization with explicit differentiable latent dynamics, yielding a multi-member analysis under irregular spatiotemporal observations and joint state–parameter inference in a single framework. A detailed qualitative comparison is provided in Appendix G.

6 Conclusion

We introduced LEVDA, a nonlinear ensemble-variational smoother inspired by 4DEnVar that optimizes in the low-dimensional latent space of a pretrained differentiable surrogate. Differentiating through the surrogate over an ensemble subspace gives efficient windowed assimilation without tangent-linear or adjoint code for the high-dimensional simulator and without observation-to-latent encoders; LEVDA jointly assimilates states and unknown parameters, accommodates time-dependent observation operators such as moving sensors, and supports observations at arbitrary spatiotemporal locations. On three geophysical benchmarks it matches or outperforms state-of-the-art latent filtering baselines under severe sparsity, with substantial accuracy and runtime advantages over full-state 4DEnVar, and the ablations show robustness to spatiotemporal irregularity and modest ensemble sizes.

Limitations. LEVDA relies on a pretrained differentiable surrogate and inherits its potential failure modes: assimilation accuracy depends on the surrogate's fidelity over the window, and performance may degrade under significant distribution shifts or unmodeled model errors; like any surrogate-based method it also depends on training data that are typically produced by existing simulation or analysis pipelines. The analysis ensemble is not a calibrated posterior and should be interpreted as an optimization-derived approximation rather than a probabilistic sample, which may limit reliability in downstream decision-making tasks such as forecasting or risk assessment. The construction is related in spirit to randomize-then-optimize (RTO) sampling (Bardsley et al., 2014), in which each member solves its own randomized variational problem; RTO recovers the exact posterior for linear-Gaussian problems when every member fits its own perturbed observations from its own background draw, whereas LEVDA's members are regularized towards their own background members but fit the same unperturbed observations, so their dispersion reflects the flow-dependent perturbation geometry, the regularization and the inflation rather than a sampling guarantee (Appendix D shows that it is stable across optimization budgets). Furthermore, LEVDA optimizes a nonconvex objective within an ensemble subspace. While the low latent dimension ensures tractability and Appendix D demonstrates robustness across varying τ, K, and sparsity levels, difficult regimes may still require tuning of hyperparameters such as ensemble size and inflation; the atmospheric benchmark, for instance, required a smaller optimizer step size than the other two. The benefit of iterating the nonlinear objective beyond its first, linearized iterate is expected to be largest when the observation–parameter map is non-convex, as for the tsunami source location, and smaller where it is nearly linear. The benchmarks are two-dimensional; the formulation extends to three dimensions by adding a coordinate to the decoder query. Finally, although LEVDA improves uncertainty quantification, consistent calibration for atmospheric dynamics remains open, and every method evaluated on that benchmark is under-dispersed (Appendix E.2). Future work could extend LEVDA to a weak-constraint formulation that explicitly models errors in the learned latent dynamics, so that observations correct accumulated surrogate inaccuracies beyond adjustments to the initial conditions, an observation-error model that includes the surrogate's reconstruction error at the observed points, and adaptive regularization that balances observation consistency, model fidelity, and ensemble spread.

Acknowledgments and Disclosure of Funding

This work was supported in part by the National Science Foundation under Grant No. 2325631 and by the U.S. Department of Energy's Genesis Mission through Argonne National Laboratory under Research Subcontract No. 6F-60171, issued under Prime Contract No. DE-AC02-06CH11357. We thank Pengpeng Xiao for her collaboration on our work LD-EnSF (Xiao et al., 2026), which provided part of the LDNet implementation and benchmarks for this project. The authors declare no competing interests.

References

Andry, G., Lewin, S., Rozet, F., Rochman, O., Mangeleer, V., Pirlet, M., Faulx, E., Grégoire, M., and Louppe, G. Appa: Bending weather dynamics with latent diffusion models for global data assimilation. *arXiv preprint arXiv:2504.18720*, 2025.

Bannister, R. N. A review of operational methods of variational and ensemble-variational data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 143(703):607–633, 2017.

Bao, F., Zhang, Z., and Zhang, G. An ensemble score filter for tracking high-dimensional nonlinear dynamical systems. *Computer Methods in Applied Mechanics and Engineering*, 432:117447, 2024.

Bardsley, J. M., Solonen, A., Haario, H., and Laine, M. Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems. *SIAM Journal on Scientific Computing*, 36(4):A1895–A1910, 2014.

Bauer, P., Thorpe, A., and Brunet, G. The quiet revolution of numerical weather prediction. *Nature*, 525(7567):47–55, 2015.

Bocquet, M. and Sakov, P. An iterative ensemble Kalman smoother. *Quarterly Journal of the Royal Meteorological Society*, 140(682):1521–1535, 2014.

Carrassi, A., Bocquet, M., Bertino, L., and Evensen, G. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *Wiley Interdisciplinary Reviews: Climate Change*, 9(5):e535, 2018.

Chen, S., Jia, Y., Qu, Q., Sun, H., and Fessler, J. A. FlowDAS: A stochastic interpolant-based framework for data assimilation. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Cheng, S., Chen, J., Anastasiou, C., Angeli, P., Matar, O. K., Guo, Y.-K., Pain, C. C., and Arcucci, R. Generalised latent assimilation in heterogeneous reduced spaces with machine learning surrogate models. *Journal of Scientific Computing*, 94(1):11, 2023.

Courtier, P., Thépaut, J.-N., and Hollingsworth, A. A strategy for operational implementation of 4D-Var, using an incremental approach. *Quarterly Journal of the Royal Meteorological Society*, 120(519):1367–1387, 1994.

Dueben, P. D. and Bauer, P. Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, 11(10):3999–4009, 2018.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al. The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, 146(730):1999–2049, 2020.

Hodyss, D., Bishop, C. H., and Morzfeld, M. To what extent is your data assimilation scheme designed to find the posterior mean, the posterior mode or something else? *Tellus A: Dynamic Meteorology and Oceanography*, 68(1):30625, 2016.

Hunt, B. R., Kostelich, E. J., and Szunyogh, I. Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter. *Physica D: Nonlinear Phenomena*, 230(1-2):112–126, 2007.

Kalnay, E. *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2003.

Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P., and Hoyer, S. Machine learning–accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118 (21):e2101784118, 2021.

Lorenz, E. N. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141, 1963.

Lorenz, E. N. Atmospheric predictability as revealed by naturally occurring analogues. *Journal of the Atmospheric Sciences*, 26(4):636–646, 1969.

Lorenz, E. N. Atmospheric predictability experiments with a large numerical model. *Tellus*, 34(6): 505–513, 1982.

Ohana, R., McCabe, M., Meyer, L., Morel, R., Agocs, F., Beneitez, M., Berger, M., Burkhart, B., Dalziel, S., Fielding, D., et al. The Well: A large-scale collection of diverse physics simulations for machine learning. *Advances in Neural Information Processing Systems*, 37:44989–45037, 2024.

Rabier, F., Thépaut, J.-N., and Courtier, P. Extended assimilation and forecast experiments with a four-dimensional variational assimilation system. *Quarterly Journal of the Royal Meteorological Society*, 124(550):1861–1887, 1998.

Regazzoni, F., Pagani, S., Salvador, M., Dedè, L., and Quarteroni, A. Learning the intrinsic dynamics of spatio-temporal processes through Latent Dynamics Networks. *Nature Communications*, 15(1): 1834, 2024.

Rozet, F. and Louppe, G. Score-based data assimilation. *Advances in Neural Information Processing Systems*, 36:40521–40541, 2023.

Si, P. and Chen, P. Latent-EnSF: A latent ensemble score filter for high-dimensional data assimilation with sparse observation data. In *The Thirteenth International Conference on Learning Representations*, 2025.

Si, P., Qiu, Y., Sallam, O., Feinstein, J., He, Z., Yan, E., and Chen, P. Toward AI-driven digital twins for metropolitan floods: A conditional latent dynamics network surrogate of the shallow water equations. *Journal of Hydrology*, 680:136461, 2026.

Tian, Z., Shen, K., Hao, W., Si, P., Chen, P., and Zhu, W. LD-EnFF: Latent-dynamics ensemble flow filtering for data assimilation with sparse observations. *arXiv preprint arXiv:2610.04034*, 2026.

Xiao, P., Si, P., and Chen, P. LD-EnSF: Synergizing latent dynamics with ensemble score filters for fast data assimilation with sparse observations. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xiao, Y., Bai, L., Xue, W., Chen, H., Chen, K., Chen, K., Han, T., and Ouyang, W. Towards a self-contained data-driven global weather forecasting framework. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54255–54275. PMLR, 2024.

Xiao, Y., Jia, Q., Chen, K., Bai, L., and Xue, W. VAE-Var: Variational autoencoder-enhanced variational methods for data assimilation in meteorology. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yang, Y., Cheng, X., Giles, D., Cheng, S., He, Y., Xue, X., Chen, B., and Hu, Y. Tensor-Var: Efficient four-dimensional variational data assimilation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70649–70679. PMLR, 2025.

Zhu, S., Wang, B., Zhang, L., Liu, J., Liu, Y., Gong, J., Xu, S., Wang, Y., Huang, W., Liu, L., et al. A four-dimensional ensemble-variational (4DEnVar) data assimilation system based on GRAPES-GFS: System description and primary tests. *Journal of Advances in Modeling Earth Systems*, 14(7):e2021MS002737, 2022.

A Datasets

Kolmogorov flow. Canonical sinusoidally forced turbulence with an unknown viscosity (or Reynolds number Re) (Kochkov et al., 2021). The evaluation window covers 200 simulation steps following a 100-step burn-in period.

Tsunami modeling. Simplified shallow-water dynamics where an initial Gaussian bump is placed at a random location to simulate an unknown landslide or earthquake source (Si & Chen, 2025).

Atmospheric modeling. A global spherical simulation adapted from the `planetswe` example in The Well (Ohana et al., 2024), governed by complex dynamics with an unknown forcing term. The simulation spans 21 days (504 hours), initialized from ERA5 500 hPa data (Hersbach et al., 2020). Observations are extremely sparse, covering only $\sim 0.2\%$ of the time steps.

Table 2 summarizes the discretization details and spatial and temporal observation sparsity for each.

Setting	Grid	Simulation steps	Obs. times (interval)	Δt	Obs. grid
Kolmogorov flow	150×150	200	40 (5)	0.04	10×10
Tsunami modeling	150×150	2000	50 (40)	≈ 21 s	10×10
Atmospheric modeling	512×256	30240	63 (480)	≈ 60 s	16×8

Table 2: Summary of experimental setups (adapted from Xiao et al. (2026)). Obs. times represent the total number of observation times while interval in parentheses denotes the spacing (in simulation steps) between consecutive observations. Obs. grid indicates the spatial subsampling resolution.

B Approximation Analysis

LEVDA replaces the original strong-constraint smoothing problem in physical state space with (i) an ensemble-subspace optimization in the latent space of (ii) a learned differentiable surrogate. Each of these substitutions introduces an approximation gap. We characterize these gaps qualitatively here and discuss when they can be expected to be small, building on related analyses for linearized reduced-order DA (Cheng et al., 2023).

Setup. Let $z = (x, u)$ and $\zeta = (s, u)$ denote window-initial augmented variables in physical and latent coordinates, respectively. Let $J^\star(z)$ be the original strong-constraint smoothing cost obtained by propagating z with the true physical dynamics, and let $z^\star = \arg\min_z J^\star(z)$. Let $D_\phi^\#(\zeta) = (D_\phi(s), u)$ denote the initial-time decoding map, with identity action on the parameter component. The representable physical manifold is

$$\mathcal{M}_\phi = \{D_\phi^\#(\zeta) : \zeta = (s, u)\},$$

and we define the best representable physical optimum as

$$z^\dagger = \arg\min_{z \in \mathcal{M}_\phi} J^\star(z).$$

The latent surrogate cost $J_{\text{lat}}(\zeta)$ is obtained by replacing the physical flow and observation map in $J^\star$ with the decoded surrogate trajectory

$$D_\phi^\#\left(\Phi_{t_i \leftarrow t}^\theta(\zeta)\right),$$

where $\Phi_{t_i \leftarrow t}^\theta$ is the latent flow induced by $\mathcal{F}_\theta$. Let $\zeta^\star = \arg\min_\zeta J_{\text{lat}}(\zeta)$. LEVDA further restricts the solution to the affine ensemble subspace

$$\mathcal{S} = \bar{\zeta}^b + \text{range}(P_\zeta),$$

and its exact subspace minimizer is $\zeta_{\mathcal{S}}^\star = \arg\min_{\zeta \in \mathcal{S}} J_{\text{lat}}(\zeta)$. The computed LEVDA member ζ^a may differ from $\zeta_{\mathcal{S}}^\star$ because Equation (15) is nonconvex and solved only approximately.

Decomposition of the analysis gap. The error of the LEVDA decoded analysis $\hat{z}^a = D_\phi^\#(\zeta^a)$ relative to the physical smoothing optimum admits the structural decomposition

$$\underbrace{\|\hat{z}^a - z^\star\|}_{\text{total gap}} \le \underbrace{\|\hat{z}^a - D_\phi^\#(\zeta_{\mathcal{S}}^\star)\| + \|D_\phi^\#(\zeta_{\mathcal{S}}^\star) - D_\phi^\#(\zeta^\star)\|}_{\substack{\text{(i) ensemble-subspace} \\ \text{/optimization gap}}} + \underbrace{\|D_\phi^\#(\zeta^\star) - z^\dagger\|}_{\text{(ii) surrogate-objective gap}} + \underbrace{\|z^\dagger - z^\star\|}_{\text{(iii) representational gap}}, \tag{16}$$

where the norm may be interpreted either at the window-initial state or, after replacing $D_\phi^\#$ by the decoded trajectory map, over the full smoothing window. The three terms isolate distinct sources of error: (i) finite-dimensional ensemble restriction plus numerical/local optimization error, (ii) mismatch between the surrogate objective and the physical objective restricted to the representable manifold, and (iii) irreducible mismatch between the representable manifold and the true physical optimum.

(i) Subspace gap. The subspace term vanishes, up to solver tolerance, when $\zeta^\star \in \bar{\zeta}^b + \mathrm{range}(P_\zeta)$. Because the columns of P_ζ are background ensemble perturbations, this holds approximately whenever the analysis increment is well explained by background flow-dependent variability, the same condition under which classical 4DEnVar updates are accurate. Quantitatively, let $\Pi_{\mathcal{S}}(\zeta^\star)$ denote the Euclidean projection of $\zeta^\star$ onto $\mathcal{S}$. If J_{lat} is locally L-smooth and μ-strongly convex in a convex neighborhood containing $\zeta^\star$, $\Pi_{\mathcal{S}}(\zeta^\star)$, and $\zeta_{\mathcal{S}}^\star$, then exact constrained minimization gives

$$\|\zeta_{\mathcal{S}}^\star - \zeta^\star\| \;\le\; \sqrt{\tfrac{L}{\mu}}\,\|\Pi_{\mathcal{S}}(\zeta^\star) - \zeta^\star\|. \tag{17}$$

If $D_\phi^\#$ is locally L_D-Lipschitz, the decoded contribution is bounded by

$$\|D_\phi^\#(\zeta_{\mathcal{S}}^\star) - D_\phi^\#(\zeta^\star)\| \;\le\; L_D\sqrt{\tfrac{L}{\mu}}\,\|\Pi_{\mathcal{S}}(\zeta^\star) - \zeta^\star\|. \tag{18}$$

The remaining term $\|\hat{z}^a - D_\phi^\#(\zeta_{\mathcal{S}}^\star)\|$ is an optimization residual: it is zero for exact global solution of the restricted objective, but in practice reflects L-BFGS tolerance and possible convergence to a local minimum. Empirically (Appendix D, Figure 6b), accuracy stabilizes for ensemble sizes $K \ge 10$, indicating that on our benchmarks the dominant directions of the analysis increment are captured by a low-rank ensemble subspace.

(ii) Surrogate gap. The surrogate-objective gap is governed by trajectory-level fidelity of the learned latent dynamics and decoder over the assimilation window. The relevant comparison is not between latent variables and physical states, but between decoded surrogate trajectories and physical trajectories initialized from the same representable state:

$$D_\phi^\#\left(\Phi_{t_i\leftarrow t}^\theta(\zeta)\right) \quad \text{versus} \quad \Phi_{t_i\leftarrow t}^\star\Big(D_\phi^\#(\zeta)\Big), \tag{19}$$

where $\Phi^\star$ denotes the true physical flow. If this decoded trajectory mismatch induces a uniform objective perturbation

$$|J_{\mathrm{lat}}(\zeta) - J^\star(D_\phi^\#(\zeta))| \le \varepsilon_{\mathrm{sur}}$$

over a neighborhood of the relevant representable minimizers, then local μ-strong convexity of the restricted physical objective gives a conservative optimizer-distance scale $O(\sqrt{\varepsilon_{\mathrm{sur}}/\mu})$. A sharper $O(\varepsilon_{\mathrm{sur}}/\mu)$ shift requires stronger gradient-level perturbation control, e.g., $\|\nabla J_{\mathrm{lat}}(\zeta) - \nabla(J^\star \circ D_\phi^\#)(\zeta)\| = O(\varepsilon_{\mathrm{sur}})$. Appendix C provides an empirical proxy for this surrogate floor via the LDNet ground-truth optimum, finding a residual relative error of roughly 1% on the tsunami benchmark.

(iii) Representational gap. The representational gap reflects the irreducible mismatch between the surrogate's representable manifold and the true PDE solution manifold. Unlike the previous two terms, this gap cannot be reduced by better optimization or more ensemble members; it is set by the expressivity and training fidelity of the surrogate.

Comparison with prior bounds. Cheng et al. (2023) derive a formal upper bound on the discrepancy between surrogate and original assimilation costs in a linearized reduced-order setting; our setting is

more general (nonlinear latent dynamics, nonlinear decoder, multi-step smoothing, and the additional ensemble-subspace restriction), so a direct extension of that bound is non-trivial. The decomposition (16) above is intended as a structural rather than tight characterization. It separates errors that can be mitigated through ensemble design and optimization, errors controlled by surrogate trajectory fidelity, and errors intrinsic to the learned representable manifold.

C Surrogate-Error Analysis

LEVDA's assimilation accuracy is fundamentally bounded by the fidelity of the pretrained latent surrogate: the analysis is restricted to trajectories the surrogate can represent, so any irreducible surrogate error sets a floor on the achievable analysis error. We characterize this floor empirically on the tsunami benchmark and analyze sensitivity to deliberately degraded surrogates.

Surrogate-induced error floor. We compare LEVDA's analysis error against the *LDNet ground-truth optimum*: the trajectory obtained by initializing the LDNet from the (oracle) true initial condition and parameter and integrating forward, with no observation noise and no assimilation. This provides a lower bound on the achievable analysis error for any method that operates within the LDNet's representable trajectory manifold. As shown in Figure 4, LEVDA's relative-RMSE trajectory tracks the LDNet ground-truth optimum closely throughout the assimilation window, with a residual gap below $\sim 1\%$ in relative error. This indicates that, on this benchmark, the optimization within the latent ensemble subspace is near-optimal given the surrogate, and the dominant source of remaining error is the surrogate's reconstruction/dynamics fidelity rather than the variational solver, the ensemble-subspace restriction, or the warm-start strategy.

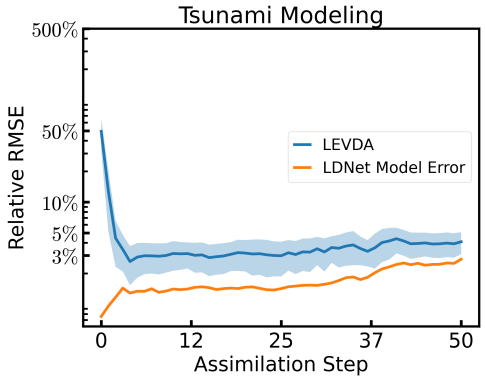


Figure 4: Surrogate-induced error floor on the tsunami benchmark. Solid: LEVDA analysis-mean relative RMSE. Dashed: LDNet ground-truth optimum (obtained by integrating the surrogate from the true initial condition).

Sensitivity to surrogate fidelity. To assess whether LEVDA remains bounded by the accuracy of the underlying surrogate, we evaluate it using a deliberately degraded version of the tsunami LDNet surrogate. In particular, we perturb all trained network parameters with additive Gaussian noise of standard deviation $\sigma_w = 10^{-3}$. This degradation substantially reduces the standalone predictive accuracy of the LDNet model, providing a controlled setting in which the surrogate itself performs poorly.

Figure 5 shows that LEVDA closely tracks the degraded LDNet performance across assimilation steps (the update and observation interval of the latent dynamics model). In the low-fidelity regime, LEVDA neither catastrophically amplifies surrogate errors nor artificially exceeds the representational capability of the surrogate. Instead, its performance remains effectively lower bounded by the LDNet accuracy itself. In this example, where the perturbed LDNet exhibits relatively poor predictive skill, LEVDA achieves nearly identical error levels, demonstrating that the assimilation framework degrades gracefully with surrogate quality.

Implication. These results provide a quantitative characterization of LEVDA's operating regime. When the pretrained surrogate is sufficiently accurate, LEVDA preserves that fidelity while providing efficient variational assimilation and uncertainty quantification. As surrogate quality deteriorates,

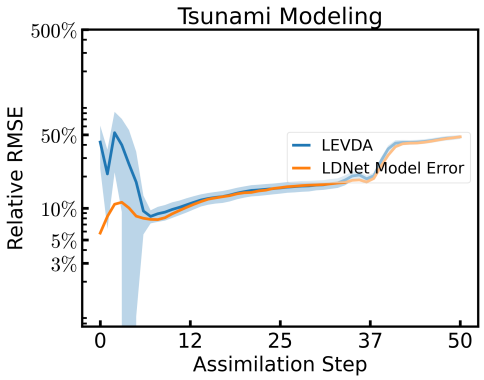


Figure 5: Relative RMSE for the degraded tsunami LDNet surrogate and the corresponding LEVDA analysis across assimilation steps. The pretrained LDNet parameters are perturbed with additive Gaussian noise of standard deviation $\sigma_w = 10^{-3}$.

LEVDA degrades smoothly toward the surrogate error floor rather than diverging, confirming that the dominant limitation is the representational accuracy of the surrogate itself. In practice, this implies that improving assimilation performance beyond this bound requires either retraining the surrogate or introducing explicit model-error corrections.

D Ablation Studies on Tsunami Modeling

We perform ablation studies using the tsunami modeling benchmark to evaluate LEVDA's robustness to three key hyperparameters: smoothing time horizon, ensemble size, and observation sparsity. Figure 6 illustrates the sensitivity of the reconstruction error to variations in these settings.

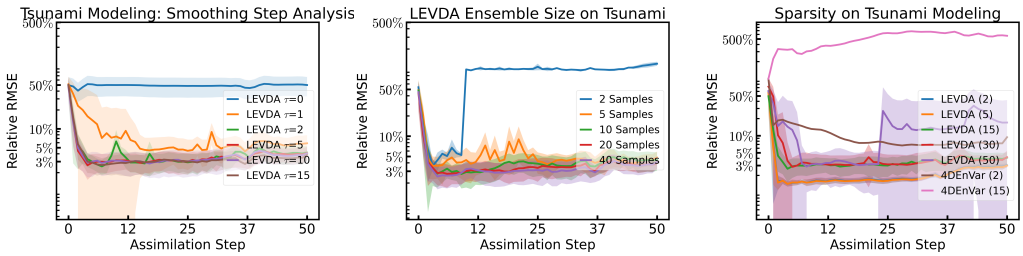


Figure 6: Tsunami benchmark sensitivity analysis: LEVDA Relative RMSE as a function of (a) smoothing time horizon τ, (b) ensemble size K, and (c) spatial observation stride (sparsity).

Smoothing Time Horizon. We vary the smoothing time horizon τ (the length of the assimilation window) to assess robustness to the duration of temporal context utilized. Figure 6a demonstrates that LEVDA is relatively insensitive to τ beyond a critical threshold. The failure observed at $\tau = 0$ is consistent with theoretical expectations: an instantaneous snapshot provides insufficient dynamic information to uniquely identify the governing parameters u.

Ensemble Size. We vary the ensemble size K to evaluate the sample efficiency of the data assimilation process. Figure 6b indicates that performance stabilizes for $K \geq 10$, with moderate degradation at $K = 5$ and divergence at $K = 2$. This efficiency stems from performing optimization within a low-dimensional latent subspace, where a relatively small ensemble is sufficient to span the tangent space required for effective updates.

Spatial Observation Sparsity. To evaluate robustness to data scarcity, we test spatial strides $s \in \{2, 5, 15, 30, 50\}$, corresponding to observation densities ranging from 25% down to an extreme 0.04% on the 2D grid. We benchmark against 4DEnVar (with localization radius $r = 2$) at strides 2 and 15. Figure 6c reveals a stark contrast: while 4DEnVar is competitive at high density ($s = 2$), it suffers a performance collapse at the experimental setting of $s = 15$ (0.4%). Conversely, LEVDA

maintains high accuracy and stability up to $s = 15$, degrades only modestly at $s = 30$, and remains functional even at the hyper-sparse rate of 0.04% ($s = 50$).

Our experiments reveal that 4DEnVar is highly sensitive to the localization radius under sparse conditions. Extreme sparsity creates a fundamental dilemma: a large localization radius introduces sampling noise via spurious long-range correlations, causing filter divergence, whereas a small radius prevents the propagation of information from sparse observations to the vast unobserved regions. At $s = 15$ (0.4% density), we observe a specific failure mode where the analysis fits the observations well (low residual) but fails globally (high state error). This occurs because the localization mask truncates the physical cross-covariances necessary to infer the unobserved state variables, rendering the full-state inverse problem ill-posed, a limitation LEVDA avoids by assimilating in a globally coupled latent space.

Robustness to Temporal Sparsity. We examine LEVDA's sensitivity to temporal data scarcity by increasing the time interval between observations. Figure 7a demonstrates remarkable stability: error rates remain nearly flat even as observations become increasingly infrequent. This resilience stems from optimizing a continuous trajectory within the low-dimensional latent manifold, which effectively bridges the gaps between sparse timepoints.

For this experiment, we extended the smoothing window to $\tau = 10$ (versus the standard $\tau = 5$) to ensure the assimilation horizon remains sufficiently long to capture these wider observation gaps.

Sensitivity to Observation Noise. We analyze LEVDA's robustness to varying magnitudes of observation noise, addressing the reality that real-world sensor statistics are often uncertain.

Because the noise level dictates the balance between the observation likelihood and the learned latent prior, the assimilation framework must remain stable even when this balance shifts. Figure 7b reports performance under noise-to-signal ratios of 5%, 10%, and 20% for the tsunami modeling case. Crucially, we maintain identical hyperparameter weights across all trials, demonstrating that LEVDA handles increased measurement uncertainty without requiring specific retuning.

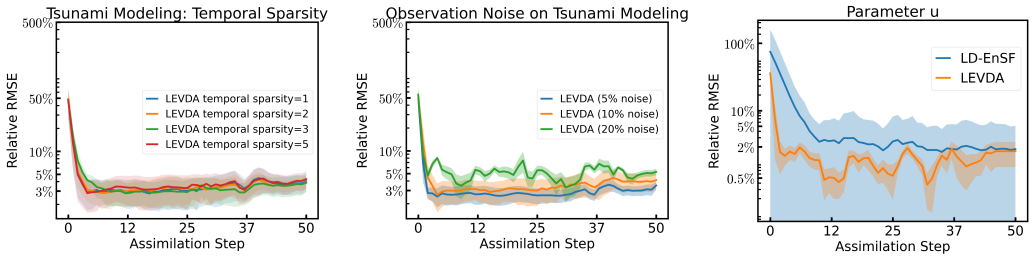


Figure 7: Additional Tsunami benchmark ablations: (a) Impact of temporal sparsity on LEVDA Relative RMSE (using extended smoothing window $\tau = 10$). (b) Robustness to observation noise (5%, 10%, 20% noise-to-signal ratios) under fixed likelihood weighting. (c) Parameter estimation accuracy: Relative RMSE of the parameter u over the assimilation window.

Sensitivity to Misspecified Observation Noise

We additionally include a misspecification ablation in which the noise level assumed in the LEVDA objective (Equation 15) differs from the true noise level; Table 3 reports the final-step relative RMSE of the ensemble mean on the tsunami benchmark.

For a prescribed observation noise level $p\%$, additive Gaussian noise is generated independently at every spatiotemporal point and for each state variable according to

$$y_{\text{obs}} = y + \epsilon, \qquad \epsilon \sim \mathcal{N}\left(0, (p|y|)^2\right), \tag{20}$$

where y denotes the corresponding ground-truth value in the normalized units in which the surrogate operates. Thus, the noise standard deviation is defined locally as a fixed percentage of the magnitude of the underlying (normalized) state at each point, rather than using a global dataset standard deviation or RMS value. In all benchmarks the objective uses $R_t = \sigma_a^2 I$ with a single assumed noise level σ_a, so every observation receives the same weight and the local noise level is not used as a weight;

the same observation model and noise realization are used for every compared method, and the misspecification ablation varies σ_a at a fixed true level.

Table 3: Tsunami benchmark: final-step relative RMSE of the ensemble mean for different true observation noise levels (rows) and noise levels assumed in the objective (columns).

True noise \ Assumed noise	**5%**	**10%**	**20%**
5%	0.0397	0.0441	0.0398
10%	0.0377	0.0378	0.0388
20%	0.0490	0.0411	0.0450

Across all nine combinations, including $4\times$ misspecification in either direction, the relative RMSE stays within $[0.038, 0.049]$, indicating LEVDA is quite insensitive to moderate misspecification of R_t.

Parameter Estimation. Beyond state reconstruction, many inverse problems necessitate the recovery of governing latent parameters. In the tsunami modeling benchmark, we focus on estimating the initial condition parameter u that specifies the starting location of the Gaussian bump. Although u is not directly observed, it dictates the subsequent wave evolution and must be inferred solely from sparse partial observations. Figure 7c tracks the RMSE of the estimated parameter over the course of the assimilation window. LEVDA consistently achieves a lower RMSE than LD-EnSF, despite not utilizing an explicit encoder as in LD-EnSF to map observations to parameters. Instead, LEVDA treats u as an augmented component of the latent state, estimating it jointly via the variational smoothing objective. This approach leverages the full temporal context of the smoothing window, allowing observations from multiple time steps to progressively constrain u, thereby reducing the uncertainty often associated with purely filtering-based updates. The superior accuracy of LEVDA confirms that variational smoothing offers a more robust mechanism for latent parameter discovery than direct encoder-based inversion.

Sensitivity to Hyperparameters With zero inflation, the ensemble can collapse toward a single member in the extreme case, degrading the subspace basis. Empirically, however, LEVDA is robust in the regimes tested: we used identical inflation values (additive, 6×10^{-4} for state, 1×10^{-4} for parameters) for both Kolmogorov flow and tsunami without retuning. Only the atmospheric benchmark required adjusting the magnitude, consistent with its much higher dimensionality and stronger nonlinearity. L-BFGS is used with at most 10 iterations per solve, a single outer loop (additional outer loops gave no significant improvement), and warm starts at the background members ($\alpha_j = \sqrt{K-1}e_j$).

Table 4 reports a sweep of the additive state inflation on its sample trajectory ($K = 100$, 10 iterations, 10% noise, five initializations), from zero to four times the default: the spread–error ratio rises monotonically from 0.46 without inflation to 2.13 at four times the default, while the relative RMSE of the ensemble mean stays within $[0.0374, 0.0384]$ up to twice the default and degrades beyond it (final step 0.0472 at four times the default). Without inflation the ensemble is under-dispersed but does not collapse, and the default sits below the value at which the spread matches the error, so the inflation sets the spread and, past twice the default, costs accuracy.

Table 4: Tsunami benchmark ($K = 100$, 10 iterations, 10% noise, five initializations): sensitivity to the additive state inflation σ (the parameter inflation is kept at 10^{-4}). rRMSE is the relative RMSE of the ensemble mean, SER the spread–error ratio; "mean" averages over assimilation steps, "final" is the last step.

Additive inflation σ	**rRMSE (mean)**	**rRMSE (final)**	**SER (mean)**	**SER (final)**
0	0.0374	0.0347	0.462	0.324
1.5×10^{-4}	0.0375	0.0347	0.479	0.351
3×10^{-4}	0.0380	0.0352	0.552	0.430
6×10^{-4} (default)	0.0380	0.0357	0.764	0.652
1.2×10^{-3}	0.0384	0.0382	1.262	1.101
2.4×10^{-3}	0.0413	0.0472	2.131	1.652

The observation-noise ablation uses fixed hyperparameters across 5%/10%/20% noise levels; and in our testing the final-step rRMSE on the tsunami benchmark varies only between 0.038 and 0.049 when the observation noise assumed in the loss is misspecified by up to $4\times$ relative to the true noise.

Table 5: Tsunami benchmark($K = 100$, 10% noise, five initializations): accuracy, calibration and optimization diagnostics for different maximum numbers of L-BFGS iterations per solve (the default is 10). rRMSE and SER are averaged over assimilation steps; the remaining metrics are defined in Table 6.

Max L-BFGS iterations	**rRMSE**	**SER**	**Grad.** α	**Grad.** u	$\boldsymbol{\Delta\alpha}$	$\boldsymbol{\Delta u}$	**Initial** $\Delta\alpha$	**Initial** Δu
5	0.0420	0.801	0.111	6.26	3.64×10^{-4}	2.01×10^{-2}	4.30×10^{-4}	2.95×10^{-2}
10	0.0380	0.764	7.35×10^{-2}	2.13	2.22×10^{-4}	3.33×10^{-3}	3.22×10^{-4}	2.06×10^{-2}
20	0.0371	0.762	6.95×10^{-2}	0.722	1.67×10^{-4}	7.05×10^{-4}	3.07×10^{-4}	1.88×10^{-2}
50	0.0346	0.754	5.78×10^{-2}	0.45	1.24×10^{-2}	4.28×10^{-4}	2.73×10^{-4}	1.89×10^{-2}

As shown in Table 5, the trajectory-averaged spread–error ratio stays between 0.75 and 0.80 for iteration budgets of 5, 10, 20 and 50, so the ensemble calibration is insensitive to the stopping point of the optimizer, while the accuracy of the ensemble mean improves only slightly with the budget (relative RMSE 0.0420 to 0.0346). Increasing the budget does continue to reduce the optimization residuals: the gradient norm with respect to the model parameter over the final three iterations decreases from 6.26 after 5 iterations to 0.45 after 50, and the corresponding parameter update from 2.01×10^{-2} to 4.28×10^{-4}. The size of the first iteration is independent of the budget, as it must be (2.73×10^{-4} to 4.30×10^{-4} for α and 1.88×10^{-2} to 2.95×10^{-2} for u). The coefficient update $\Delta\alpha$ over the final three iterations is of order 10^{-4} up to 20 iterations and larger at 50, where the runs end inside the line search at the evaluation cap of the optimizer (1.25 times the iteration budget), so the gradient norms rather than the step sizes are the convergence indicator there. These results indicate that LEVDA does not require every ensemble member to be optimized to convergence: a modest budget already yields stable ensemble statistics, and further iterations refine the solution without materially changing the analysis ensemble.

Table 6: Definitions of the optimization diagnostics of Table 5. All norms are Frobenius norms of the $K \times K$ coefficient matrix $A = [\alpha^{(1)}, \ldots, \alpha^{(K)}]$ or of the $K \times d_u$ matrix of parameter increments, evaluated at the accepted iterates and averaged over all assimilation windows and the five initializations.

Metric	**Description**
Grad. α	Norm of the gradient of the objective with respect to the coefficients, averaged over the final three L-BFGS iterations.
Grad. u	Norm of the gradient of the objective with respect to the parameter increments, averaged over the final three L-BFGS iterations.
$\Delta\alpha$	Norm of the change of the coefficients between consecutive L-BFGS iterations, averaged over the final three iterations.
Δu	Norm of the change of the parameter increments between consecutive L-BFGS iterations, averaged over the final three iterations.
Initial $\Delta\alpha$	Norm of the change of the coefficients in the first L-BFGS iteration, as a reference for the scale of the initial step.
Initial Δu	Norm of the change of the parameter increments in the first L-BFGS iteration.

E Metrics

E.1 Relative RMSE

We evaluate reconstruction accuracy using the relative Root Mean Square Error (RMSE) of the analysis ensemble mean $\bar{x}_t$ with respect to the ground truth $x_t^\star$ measured in the Euclidean norm:

$$\text{relative RMSE}(t) \;=\; \frac{\|\bar{x}_t - x_t^\star\|_2}{\|x_t^\star\|_2}.$$

In time-series plots, shaded regions denote the standard deviation of the member-wise relative errors, which are

$$e_t^{(i)} = \frac{\|x_t^{(i)} - x_t^\star\|_2}{\|x_t^\star\|_2}, \qquad i = 1, \ldots, N.$$

E.2 Uncertainty Quantification Metrics

We evaluate the quality of LEVDA's uncertainty estimates using proper scoring rules and calibration diagnostics. Across all test cases, LEVDA consistently improves these metrics, demonstrating a more reliable characterization of ensemble dispersion. However, achieving proper calibration in the atmospheric modeling case remains a significant challenge for all evaluated methods.

Spread-Error Ratio. To evaluate the reliability of the assimilated ensemble, we first employ the Spread-Error Ratio (SER). This metric diagnoses the calibration of the ensemble by comparing its internal diversity (spread) to the actual predictive accuracy of its mean. The SER at time t is defined as:

$$\mathrm{SER}_t = \frac{\mathrm{Spread}_t}{\mathrm{Error}_t} = \frac{\sqrt{\frac{1}{K}\sum_{j=1}^{K}\|x_t^{(j)} - \bar{x}_t\|_2^2}}{\|\bar{x}_t - x_t^\star\|_2}, \tag{21}$$

where $\bar{x}_t = \frac{1}{K}\sum_{j=1}^{K} x_t^{(j)}$ is the ensemble mean, $x_t^{(j)}$ represents individual ensemble members, and $x_t^\star$ is the ground truth. An SER of 1 indicates ideal calibration, whereas SER < 1 signals overconfident under-dispersion, and SER > 1 signals underconfident over-dispersion. We report the evolution of the SER across the three benchmarks in Figure 8.

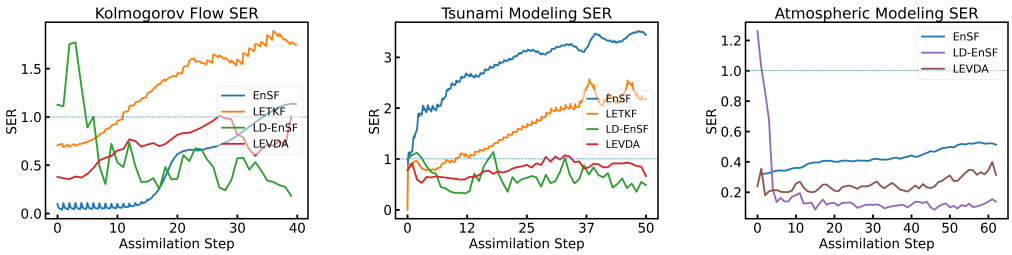


Figure 8: Spread-Error Ratio (SER) over time for Kolmogorov flow, tsunami modeling, and atmospheric modeling (left-to-right). Values closer to 1 indicate better calibration of the ensemble spread.

Continuous Ranked Probability Score. Our second uncertainty metric is the Continuous Ranked Probability Score (CRPS), a proper scoring rule that quantifies the discrepancy between the forecast and observed cumulative distribution functions. A lower CRPS indicates superior probabilistic skill, reflecting an optimal balance between predictive accuracy and ensemble dispersion. For high-dimensional states, we compute the CRPS using an energy-score formulation, averaging across spatial dimensions and channels to evaluate performance over time (Figure 9).

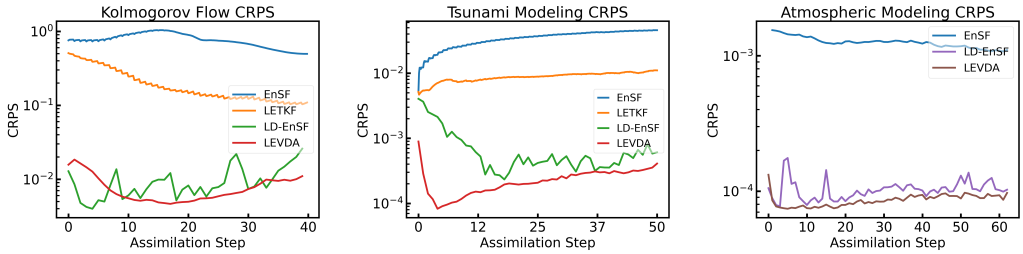


Figure 9: Continuous Ranked Probability Score (CRPS) over time for Kolmogorov flow, tsunami modeling, and atmospheric modeling (left-to-right). Lower values represent higher probabilistic accuracy.

Overall, LEVDA achieves the lowest CRPS and best SER among evaluated baselines, indicating superior probabilistic accuracy. In addition, LEVDA produces SER closer to 1 than baselines in nearly all settings, reflecting improved calibration rather than merely sharper, under-dispersed posteriors.

F Non-Ensemble and Interpolation Baselines

Latent Variational Data Assimilation To strengthen baseline comparisons, we additionally include a surrogate-enhanced variational baseline using the same pretrained LDNet but optimizing a single latent trajectory directly, i.e., without the ensemble-space α-parameterization. This serves as a practical proxy for surrogate-enhanced latent 4DVar / MAP-style assimilation, which we denote as LVDA in Figure 10. Empirically, LVDA achieves average reconstruction errors comparable to LEVDA, but its ensemble spread (obtained from assimilating multiple initializations) is extremely narrow and therefore fails to capture predictive uncertainty. In addition, LVDA appears noticeably more sensitive to observation noise, exhibiting sharper peaks and valleys along the same trajectory. These results suggest that the ensemble-subspace formulation is important not only for preserving uncertainty information rather than collapsing to a single optimized trajectory, but also for regularizing the effect of noisy observations on the final assimilated state. Consequently, LVDA performs worse than LEVDA in terms of CRPS and SER despite similar mean reconstruction errors.

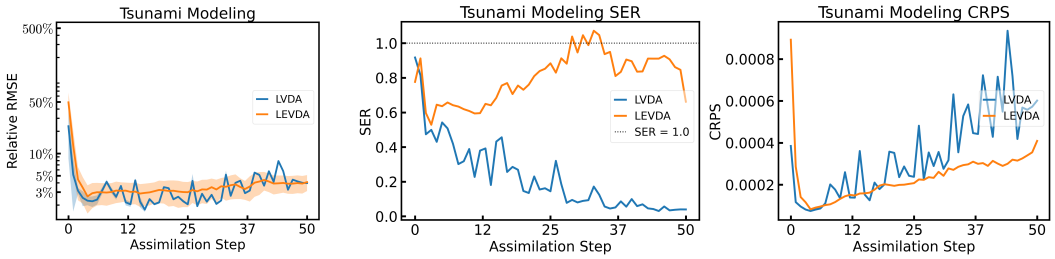


Figure 10: Comparison against a non-ensemble baseline (LVDA) in terms of rRMSE (left), SER (middle), and CRPS (right). While LVDA achieves similar average reconstruction accuracy, it exhibits significantly narrower ensemble spread and degraded uncertainty calibration relative to LEVDA.

Simple Interpolation As an additional sanity-check baseline, we compare against simple interpolation methods applied directly to the sparse observations. In our experimental setting with extremely sparse observations (approximately $225\times$ sparsity), interpolation-based approaches perform particularly poorly. We evaluate both linear interpolation using SciPy's `RegularGridInterpolator` and nonlinear cubic interpolation using `RectBivariateSpline`. Both methods yield substantially larger reconstruction errors, with relative RMSE values approaching 50%, compared to the considerably lower errors achieved by the DA baselines considered in this work. These results highlight that the assimilation problem cannot be solved adequately through local interpolation alone and instead requires learned dynamical priors capable of propagating information globally across space and time.

G Capability Comparison

Table 7 contrasts LEVDA with a broad set of recent latent-space and surrogate-assisted DA methods, including score-based approaches (Latent-EnSF, LD-EnSF, SDA, FlowDAS, Appa) and latent / reduced-order variational approaches (GLA, VAE-Var, FengWu-4DVar). LEVDA distinguishes itself as the only framework in this comparison that simultaneously integrates trajectory-level smoothing, explicit differentiable latent dynamics, ensemble-based uncertainty, irregular observation windows, and joint state–parameter estimation.

H Visualization of Experimental Systems

Figures 12 and 14 show the fields of the tsunami and Kolmogorov benchmarks; Figure 13 shows the observed locations of the moving-sensor experiment of Section 4.4.

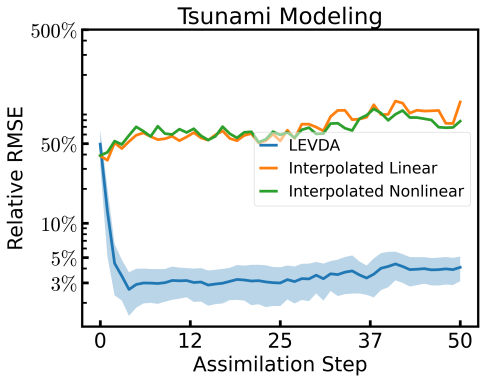


Figure 11: Comparison against simple interpolation baselines. Both linear and cubic interpolation methods perform substantially worse under highly sparse observations, demonstrating the necessity of learned dynamical priors for accurate state reconstruction.

Method	**Smoothing**	**Differentiable dynamics**	**Ensemble UQ**	**Moving observations**	**Irregular-time observations**	**Parameter estimation**
Latent-EnSF (Si & Chen, 2025)	×	×	✓	×	×	×
LD-EnSF (Xiao et al., 2026)	×	✓	✓	×	×	✓
SDA (Rozet & Louppe, 2023)	✓	×	✓	✓	×	×
FlowDAS (Chen et al., 2025)	✓	×	✓	✓	×	×
Appa (Andry et al., 2025)	✓	×	✓	✓	×	×
GLA (Cheng et al., 2023)	✓	×	×	×	×	×
VAE-Var (Xiao et al., 2025)	×	×	×	×	×	×
FengWu-4DVar (Xiao et al., 2024)	✓	✓	×	✓	×	×
Tensor-Var (Yang et al., 2025)	✓	✓	×	✓	×	×
LEVDA (ours)	✓	✓	✓	✓	✓	✓

Table 7: Feature comparison across recent latent-space, score-based, and surrogate-assisted variational DA methods. "Smoothing" denotes trajectory-level inference over an assimilation window (as opposed to sequential filtering or single-time analysis). "Differentiable dynamics" refers to the use of an explicit learned surrogate for state evolution that supports gradient-based optimization through the dynamics, as opposed to implicit trajectory priors or static latent maps. "Ensemble UQ" denotes the production of a multi-member analysis ensemble (rather than a single MAP estimate) that can be used for uncertainty quantification. "Irregular-time observations" denotes native support for data at arbitrary time stamps without resampling to a fixed temporal grid.

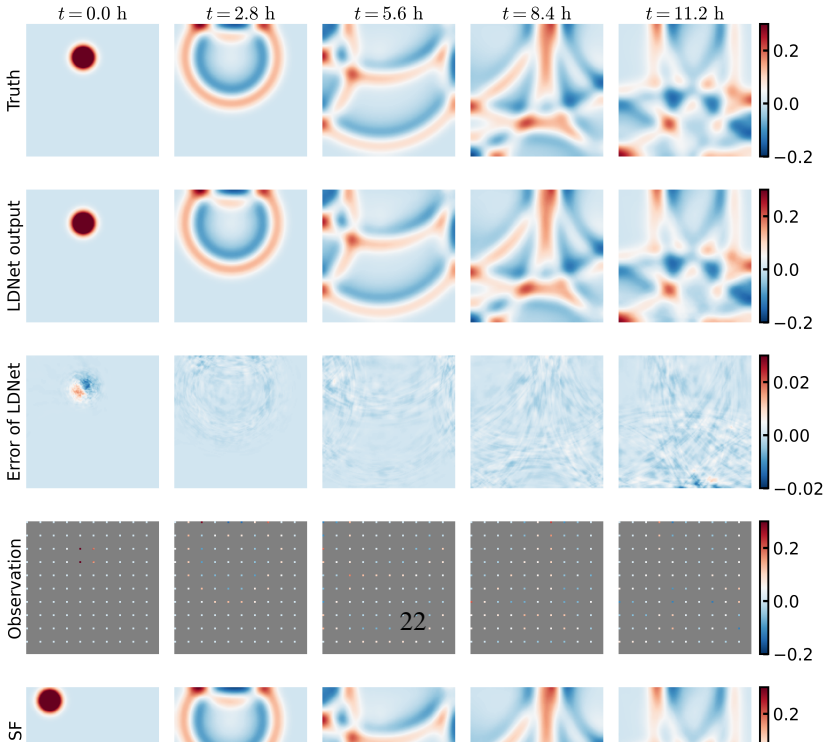




Figure 13: Moving observation locations in the tsunami experiment of Section 4.4: the 0.4% of grid points observed at five observation times, drawn independently at every observation time and colored by the observed surface elevation.

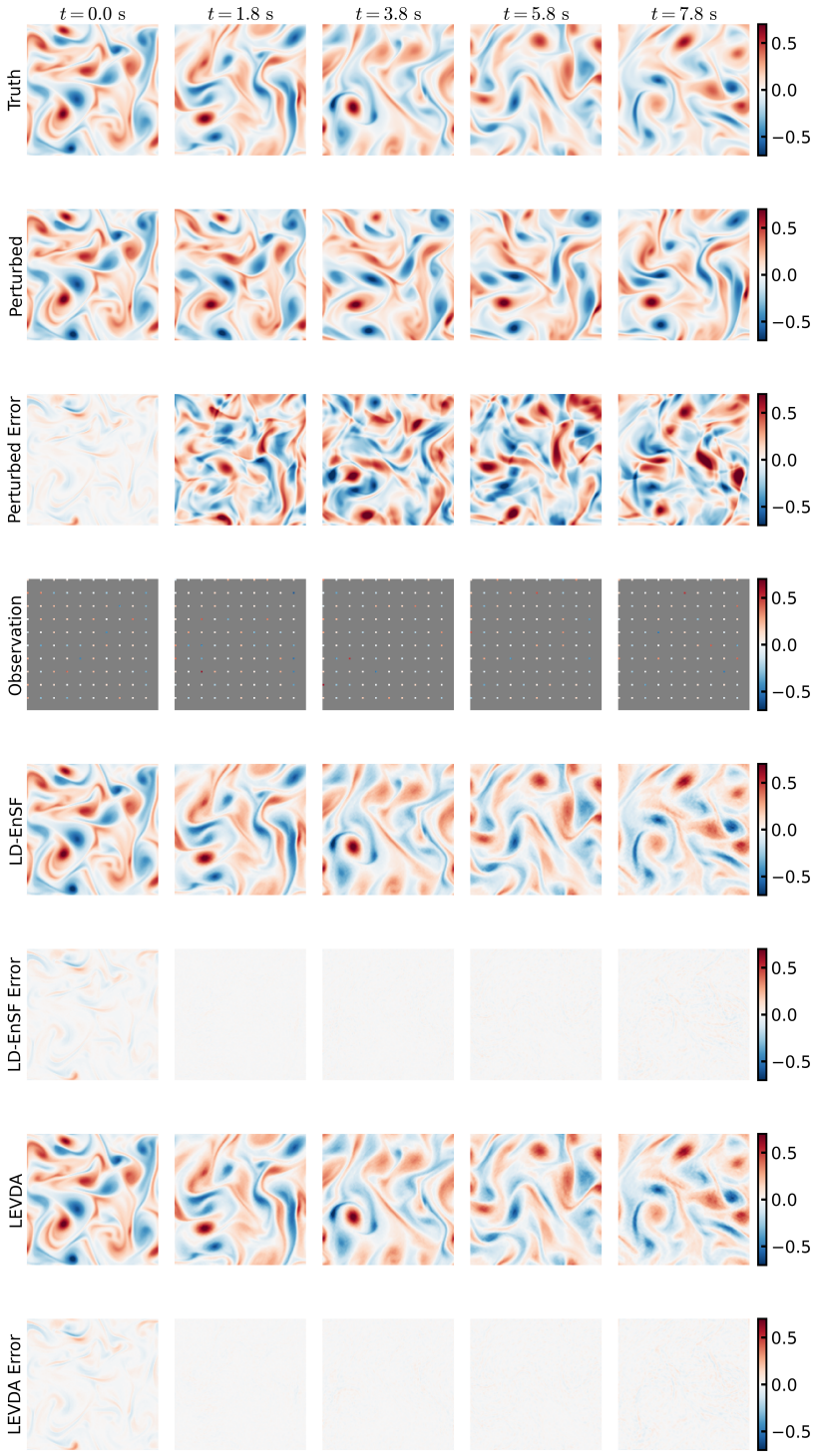


Figure 14: Kolmogorov flow benchmark: vorticity field, comparing ground truth and assimilated reconstruction. The rows display: (1) ground truth; (2) perturbed trajectory with different Reynolds number; (3) error of perturbed trajectory; (4) sparse observations; (10×10 subsampled from a 150×150 grid (5) assimilated results of a single trajectory from the LD-EnSF ensemble ($N = 20$); (6) LD-EnSF error; (7) a single assimilated trajectory from the LEVDA ensemble ($N = 20$); (8) LEVDA error.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

1. **Claims**

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

   Answer: [Yes]

   Justification: All claims made are substantiated in experiments (sometimes in appendix).

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Can be found in the limitations section of Conclusion (section 6).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.

- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: Proofs are shown in appendix where appropriate and terms are broken down.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Experimental details of the dataset and hyperparameters are all given to accurately reproduce, and mathematical details are given. Code will be released upon acceptance.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [No]

   Justification: Code is not released at this time, but we will be releasing code when the paper is accepted.

   Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: Details specified in section 4.1.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification: Standard deviation of error is reported for the trajectories.

Guidelines:

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

8. **Experiments compute resources**

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

Answer: [Yes]

Justification: We analyze speed, and specify our current compute resources in Table 1.

Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: No potential harms in the research progress. In terms of societal impacts, data assimilation and prediction processes that are adopted by the public always have a risk of incorrect prediction, which can lead to misinformed decisions on how to combat events such as natural disasters. Our goal here is to provide enough uncertainty capture that this risk is minimal, but of course, there are certain experiments where our model is still overconfident, which can lead to unforeseen risks. We do note the limitations of these, and also specify that it might not capture the true posterior in the limitations.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We discuss this in the above question.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: THe paper poses no risk for the data or model release.

Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We properly credit all data sources and paper sources.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [N/A]

    Justification: We have not yet released the new assets, these will be released later.

    Guidelines:

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

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

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

Answer: [N/A]

Justification: We have not worked with crowdsourcing or human subjects.

Guidelines:

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

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

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

Answer: [N/A]

Justification: No human subject testing.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: Core methodology was not developed with LLM input.

Guidelines:

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