Neural variational Data Assimilation with Uncertainty Quantification using SPDE priors

Maxime Beauchamp, Ronan Fablet, Simon Benaichouche, Pierre Tandeo, Nicolas Desassis, Bertrand Chapron

Neural variational Data Assimilation with Uncertainty Quantification using SPDE priors

Maxime Beauchamp,[a,b] Ronan Fablet,[a] Simon Benaichouche,[a] Pierre Tandeo,[a] Nicolas Desassis,[c] Bertrand Chapron,[d]

[a] *IMT Atlantique, 655 Av. du Technopôle, 29280 Plouzané, France*

[b] *Danish Meteorological Institute, Lyngbyvej 100, 2100 Copenhagen, Denmark*

[c] *Mines ParisTech, Centre de Géosciences, 35 Rue Saint-Honoré, 77300 Fontainebleau, France*

[d] *IFREMER, 1625 Rte de Sainte-Anne, 29280 Plouzané, France*

Corresponding author: Maxime Beauchamp, maxime.beauchamp@imt-atlantique.fr

Simon Benaichouche's current affiliation: INRIA, Rennes, France

ABSTRACT: The spatio-temporal interpolation of large geophysical datasets has historically been addressed by Optimal Interpolation (OI) and more sophisticated equation-based or data-driven Data Assimilation (DA) techniques. Recent advances in the deep learning community enables to address the interpolation problem through a neural architecture incorporating a variational data assimilation framework. The reconstruction task is seen as a joint learning problem of the prior involved in the variational inner cost, seen here as a projection operator of the state, and the gradient-based minimization of the latter. Both prior models and solvers are stated as neural networks with automatic differentiation which can be trained by minimizing a loss function, typically the mean squared error between some ground truth and the reconstruction. Such a strategy turns out to be very efficient to improve the mean state estimation, but still needs complementary developments to quantify its related uncertainty. In this work, we use the theory of Stochastic Partial Differential Equations (SPDE) and Gaussian Processes (GP) to estimate both space-and time-varying covariance of the state. Our neural variational scheme is modified to embed an augmented state formulation with both state and SPDE parametrization to estimate. We demonstrate the potential of the proposed framework on a spatio-temporal GP driven by diffusion-based anisotropies and on realistic Sea Surface Height (SSH) datasets. We show how our solution reaches the OI baseline in the Gaussian case. For nonlinear dynamics, as almost always stated in DA, our solution outperforms OI, while allowing for fast and interpretable online parameter estimation.

1. Introduction

Over the last decade, the emergence of large spatio-temporal datasets both coming from remote sensing satellites and equation-based numerical simulations has been noticed in Geosciences. As a consequence, the need for statistical methods able to handle both the size and the underlying physics of these data is growing. Data assimilation (DA) is the traditional framework used by geoscientists to merge these two types of information, data and model, by propagating information from observational data to areas with missing data. When no deterministic numerical outputs are available, the classic approach stems from the family of the so-called Optimal Interpolation (OI) techniques, also being at the core of the statistical DA methods (Asch et al. 2016). For the production of gridded geophysical maps, most of operational products either derived from OI and DA schemes allow us to estimate the mesoscale components of the targeted variables. In the specific case of Sea Surface Height (SSH) for instance, even if satellite altimetry provides capabilities to inform the mesoscale ocean geostrophic currents, the gridded product fails in merging observations and background model with consistent temporal and spatial resolutions able to retrieve fine mesoscale structures lower than 150-200 km at mid-latitudes, while they are key for general ocean circulation (Su et al. 2018). Then, recent efforts of the DA community have been made to counteract this lack in the gridded SSH products, amongst them the dynamical optimal interpolation (DOI) (Ubelmann et al. 2015), Multiscale Interpolation Ocean Science Topography (MIOST) (Ubelmann et al. 2021) or BFN-QG (Le Guillou et al. 2023).

From another point of view, deep learning frameworks are currently knowing an intense period of scientific contributions to revisit statistical methods with neural network formulation. State-of-the-art methods leverage DL to better extract the information from the observations compared to the classical DA and OI approaches, in which fine-tuning of the DA scheme parametrizations (background and observation error covariances) demands itself some methodological and computational effort, see e.g. Tandeo et al. (2020), all together with the popular Gaussian simplifications involved in most of DA operational schemes (Asch et al. 2016), though some approaches exist to alleviate the Gaussian assumptions in nonlinear geophysical problems, see e.g. Kurosawa and Poterjoy (2023). End-to-end learning architectures are also designed being backboned on DA schemes (Boudier et al. 2023; Rozet and Louppe 2023; Fablet et al. 2021), so

that they draw from the bayesian formalism to learn all the components of the DA procedure (prior model, observation operator, numerical solver, etc.) at once. Last, over the last few years, the successfull applications of generative models to geophysical datasets paves the way to combine DL together with UQ by efficient sampling strategies of the posterior distribution. Though, to our knowledge, no end-to-end-combination of DA, DL and UQ has yet been proposed.

That is why in this work, we draw from preliminary neural schemes inspired by variational data assimilation, see e.g. (Fablet et al. 2021), to jointly learn the SPDE parametrization of a surrogate stochastic prior model of the evolution equation, together with the solver of the minimization problem. Because the parameters of the SPDE remain initially unknown, they are embedded in the optimization process using an augmented state formulation, commonly used in data assimilation, see e.g. (Ruiz et al. 2013). The solver is still based on an iterative residual scheme (Fablet et al. 2021) to update the analysis state. Here, the analysis stands for the expectation of the state given the observations, together with the SPDE parametrization maximizing their likelihood given the true states used during the training. The SPDE equation can then be seen as a tangent linear model of the prior along the data assimilation window, from which we provide uncertainty quantification (UQ) throughout its precision matrix. Using such a stochastic prior entails the possibility of generating huge members in the posterior pdf, after conditioning of the prior samples by our neural variational scheme. Also, if the training dataset is large enough, the method provides an efficient way to estimate *online* the SPDE parametrization for any new sequence of input observations, without any additional inference to make. In the end, the key contributions of this work are four-fold:

- We develop the explicit solver of the considered SPDE prior. It relies on the analytical expression for the SPDE-based precision matrix of any state trajectory, based on a finite-difference discretization of the grid covered by the tensors involved in our neural scheme;
- We exploit this SPDE parametrization as surrogate prior model in the proposed variational formulation and leverage a trainable gradient-based solver to address jointly the interpolation of the state trajectory and the estimation of SPDE parameters from irregularly-sampled observations. The end-to-end training of the solver targets both the expectation of the state given the observations, together with the SPDE parametrization maximizing its likelihood given the true states;

- The SPDE prior paves the way to uncertainty quantification through the sampling of the prior pdf and the conditioning by the neural gradient-based solver;

- Two applications of this framework are provided: first, a GP driven by a spatio-temporal SPDE with spatially varying diffusion parameters is compared to the optimal solution; then a Sea Surface Height realistic dataset is used to demonstrate how the proposed framework is also relevant for non-gaussian and non-linear dynamics.

To present these contributions, the paper is structured as follows. In Section 2, we provide a preliminary background on SPDE-based Optimal Interpolation, Data assimilation and Deep Learning for DA, that we aims to bridge together in this work. We also remind how UQ is usually handled in this different fields. In Section 3, we present how we extend neural solvers to embed SPDE prior parametrizations for UQ. Finally, Section 4 provides two applications of this framework in the Gaussian case and for realistic Sea Surface Height datasets.

2. Background: Data Assimilation, Uncertainty Quantification and Machine Learning

In this work, we target the reconstruction of a probabilistic spatio-temporal state sequence $\mathbf{x} = \{\mathbf{x}_k(\mathcal{D})\}, \mathbf{x}_k \in \mathbb{R}^m$ given the partial and potentially noisy observational dataset $\mathbf{y}(\Omega) = \{\mathbf{y}_k(\Omega_k)\}, \mathbf{y}_k \in \mathbb{R}^{p_k}$, where $\Omega = \{\Omega_k\} \subset \mathcal{D}$ denotes the subdomain with observations and index k refers to time t_k. To do so, we aim at bridging Data Assimilation (DA), Uncertainty Quantification (UQ) and deep learning-based (DL) methods, see Fig. 1 for a conceptual illustration of these connections, to propose SPDE-based extensions of the so-called 4DvarNet neural variational scheme (Fablet et al. 2021), as a generic interpolation and short-term forecasting tools. We provide here a brief presentation of these three literatures, focusing only on what is useful in our framework, with an additional description of how UQ is usually handled among these communities. For a more exhaustive review, Cheng et al. (2023) provides a detailed presentation of machine learning techniques with data assimilation and uncertainty quantification for dynamical systems.

a. Optimal Interpolation (OI) and Data Assimilation (DA)

(i) Classic formulations. As very basic details to ease the link with the other components of this work, we remind that when no dynamical model is available as a prior information, the

covariance-based Optimal Interpolation, see e.g. (Chilès and Delfiner 2012), is given by:

$$\mathbf{x}^{\star} = \mathbf{P}_{\mathbf{xy}}\mathbf{P}_{\mathbf{yy}}^{-1}\mathbf{y} \tag{1}$$

where $\mathbf{P}_{\mathbf{xy}}$ and $\mathbf{P}_{\mathbf{yy}}$ are covariance matrices coming from the covariance $\tilde{\mathbf{P}}$ of the observation and state vector $\begin{bmatrix}\mathbf{y} & \mathbf{x}\end{bmatrix}^{\mathrm{T}}$:

$$\tilde{\mathbf{P}} = \begin{bmatrix}\mathbf{P}_{\mathbf{xx}} & \mathbf{P}_{\mathbf{xy}} \\ \mathbf{P}_{\mathbf{xy}}^{\mathrm{T}} & \mathbf{P}_{\mathbf{yy}}\end{bmatrix} \tag{2}$$

Broadly speaking, when the prior information is available, typically as high dimensional numerical models in geosciences, see e.g. Carrassi et al. (2018), two main categories of DA (Evensen 2009; Evensen et al. 2022) exists: variational and sequential methods. They both aims at minimizing some energy or functional involving an equation-based dynamical prior and an observation term. Drawing from the link established with Gaussian Processes, we can also consider the case of noisy observations and ease the link with data assimilation formalism, see Särkka and Hartikainen (2012); Särkka et al. (2013); Grigorievskiy et al. (2016). The state space model corresponding to the GP regression problem writes:

$$\begin{cases}\mathbf{x}_{t+dt} &= \mathbf{M}_{t+dt}\mathbf{x}_t + \boldsymbol{\eta}_t \\ \mathbf{y}_t &= \mathbf{H}_t\mathbf{x}_t + \varepsilon_t\end{cases} \tag{3}$$

where $\boldsymbol{\eta}_t$ is the m-dimensional noise process and the evolution equation is defined by the feedback linear operator matrix $\mathbf{M}_{t+dt}$. $\mathbf{H}_t$ is the observation operator at time t mapping the state space to the observation space and ε_t the observational error with covariance matrix $\mathbf{R}_t$. Based on this time-dependent notations, we also consider global observation operator $\mathbf{H}$ with dimensions $(L \times p_k) \times (L \times m)$ and global observational error covariance matrix $\mathbf{R}$ with dimensions $(L \times p_k) \times (L \times p_k)$ as block diagonal matrices whose each block respectively contains the time-dependent observation operator and observational error covariance matrix $\mathbf{H}_t$ and $\mathbf{R}_t$.

(ii) The Stochastic Partial Differential Equation (SPDE) approach in DA. The Optimal Interpolation implies to factorize dense covariance matrices which is an issue when the size of

spatio-temporal datasets is large. Reduced rank approximations, see e.g. (Cressie and Wikle 2015) have already been investigated to tackle this specific problem. More recently, the use of sparse precision matrices has also been proposed by using tapering strategies (Furrer et al. 2006; Bolin and Wallin 2016) or by making use of the link seen by Lindgren et al. (2011) between Stochastic Partial Differential Equations (SPDE) and Gaussian Processes. For the latter, if the original link was made through the Poisson SPDE equation (Whittle 1953):

$$(\kappa^2 - \Delta)^{\alpha/2} X = \tau Z \; ; \tag{4}$$

where $\Delta = \sum_{i=1}^{d} \frac{\partial^2}{\partial s_i^2}$ denotes the Laplacian operator, Z is a standard Gaussian white noise, $\kappa = 1/a$, a denotes the range of the GPX, $\alpha = \nu + d/2$ and τ relates to the marginal variance of X. It can be extended to more complex linear SPDE involving physical processes like advection or diffusion (Lindgren et al. 2011; Fuglstad et al. 2015a; Clarotto et al. 2022). The SPDE-based OI formulation uses precision matrix formalism, as the inverse of the covariance matrix $\mathbf{Q} = \tilde{\mathbf{P}}^{-1} = (dxdy)/\tau^2 \cdot \mathbf{B}^{\mathrm{T}}\mathbf{B}$, see Eq. (2) where $\mathbf{B}$ is the discretized version of the fractional differential operator $(\kappa^2 - \Delta)^{\alpha/2}$, and dx, dy are the spatial grid step sizes:

$$\mathbf{x}^{\star} = -\mathbf{Q}_{\mathbf{xx}}^{-1}\mathbf{Q}_{\mathbf{xy}}\mathbf{y} \tag{5}$$

By construction, $\mathbf{Q}$ is sparse which means that we solve a system with sparse Cholesky of complexity $\mathcal{O}(m^{3/2})$, while the general Cholesky algorithm is of complexity $\mathcal{O}(m^3)$. Thus, it opens new avenue to cope with massive observational datasets in geosciences while making use of the underlying physics of such processes. Let note that the so-called SPDE-based approach can also be used as a general spatio-temporal model, even if it is not physically motivated, since it provides a flexible way to handle local anisotropies of a large set of geophysical processes. It has known numerous applications in the past few years, see e.g. Sigrist et al. (2015); Fuglstad et al. (2015b). Though, when considering a spatio-temporal advection-diffusion SPDE, the parameters generally vary continuously across space and/or time making their estimation an additional problem to the original interpolation task. This estimation generally relies on off-line strategies (Fuglstad et al. 2015b) embedded in hierarchical models, which can be another computational issue while the set of parameters estimated does not automatically transfer to a new dataset.

Regarding the transfer of SPDE formulation in the DA formalism, rewriting the covariance-based Eq. (2) in terms of precision matrix leads to:

$$\tilde{\mathbf{P}} = \begin{bmatrix} \mathbf{P}_{\mathbf{xx}} & \mathbf{P}_{\mathbf{xx}}\mathbf{H}^{\mathrm{T}} \\ \mathbf{H}\mathbf{P}_{\mathbf{xx}} & \mathbf{H}\mathbf{P}_{\mathbf{xx}}\mathbf{H}^{\mathrm{T}} + \mathbf{R} \end{bmatrix}, \quad \tilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q}_{\mathbf{xx}} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H} & -\mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1} \\ -\mathbf{R}^{-1}\mathbf{H} & \mathbf{R}^{-1} \end{bmatrix} \tag{6}$$

which gives an other version of Eq. (5):

$$\mathbf{x}^{\star} = \left(\mathbf{Q}_{\mathbf{xx}} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}\right)^{-1} \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y} \tag{7}$$

whose posterior precision matrix of state $\mathbf{x}^{\star}$ is $\mathbf{Q}(\mathbf{x}|\mathbf{y}) = \mathbf{Q}_{\mathbf{xx}} + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}$. This type of formulation makes the link between DA framework and SPDE-based formalism maing extensive use of precision matrix formulations.

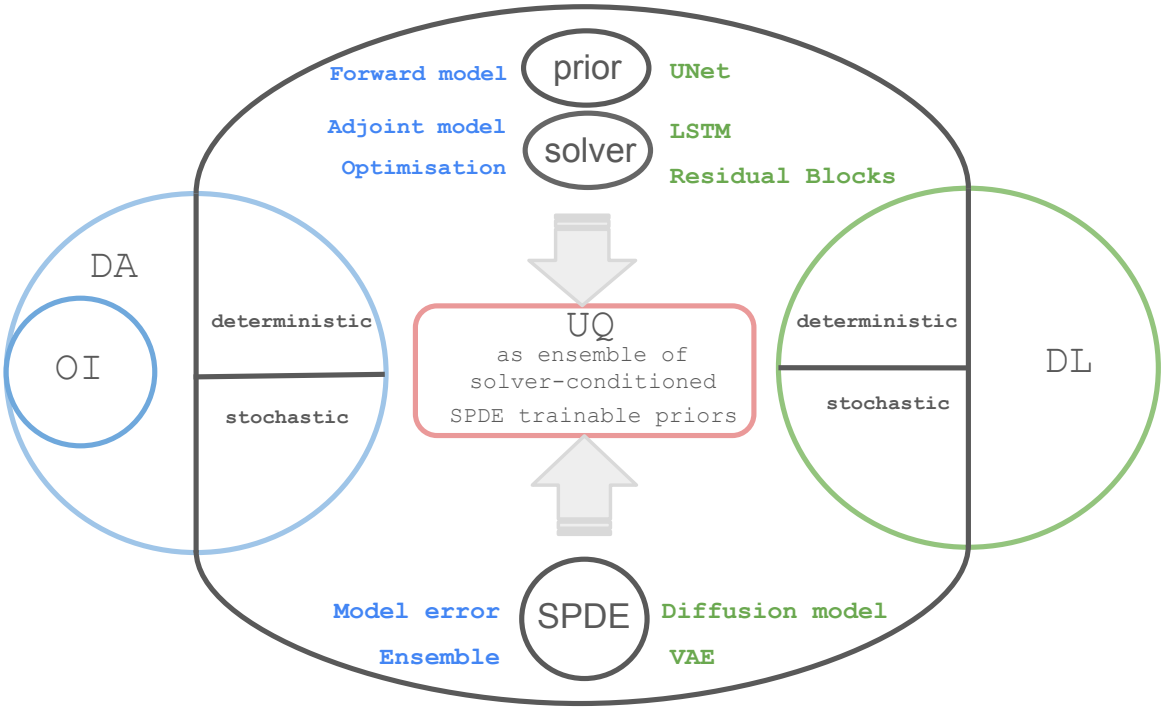


FIG. 1: Combination of SPDE-based formalism, Data Assimilation and Deep Learning to address the challenge of providing UQ in neural variational scheme. We decompose both DA and DL in deterministic and stochastic components to understand how our approach relates to state-of-the-art methods in these fields and how we provide UQ as ensemble of solver-conditioned trainable SPDE priors.

b. Machine and Deep Learning (ML/DL)

From another point of view, deep learning frameworks are currently knowing an intense period of scientific contribution to revisit statistical methods with neural network formulation. The latter enables to use automatic differentiation embedded in the gradient-based optimization as a way of solving traditional inverse problems. Several approaches have been investigated, among others we can think of using DL as a substitute for one component of the DA procedure: surrogate dynamical models making extensive use of state-of-the-art neural networks such as LSTM (Nakamura et al. 2021), UNet (Doury et al. 2023) or Transformers (Tong et al. 2022), reduced order model to compute DA schemes in latent space or with super-resolution component (Brajard et al. 2021), modeling of model/observation error covariance (Cheng and Qiu 2022; Sacco et al. 2022, 2024). The other way rely on end-to-end learning of the entire DA system instead of using DL techniques to address one aspect of DA algorithm among its three main blocks: forward model and its error, observation operator and its error, and DA scheme (EnKF, 4DVar, etc.). Such approaches may rely on applying state-of-the-art neural architectures to map observation data to the hidden state sequence, see e.g. UNet (Li et al. 2022) ,Transformers (Shi et al. 2022) or LSTM architectures (Martin et al. 2023). In particular, when the problem relates to space-time interpolation of partial and noisy observation of geophysical fields, DA-inspired neural schemes have been recently proposed, see e.g. Boudier et al. (2023); Rozet and Louppe (2023); Fablet et al. (2021). The latter specifically suggests a joint learning of prior models and solvers as an end-to-end data-driven variational data assimilation scheme. The so-called 4DVarNet neural scheme is introduced: it involves an implicit iterative gradient-based LSTM solver to minimize a variational cost, close to what is encountered in 4DVar data assimilation (Carrassi et al. 2018). In this variational cost, the dynamical prior is no longer equation-based but is stated as a trainable neural network learnt during the training process. Then, automatic differentiation is used to compute the gradient of the variational cost during the gradient-based iterations, instead of requiring the computation of complex and costly adjoint models (Asch et al. 2016). Drawing from this framework, a neural optimal interpolation scheme has also been proposed (Beauchamp et al. 2022) to reach OI performance with a linear scaling of the solution on the number of space-time variables, leading to a significant speed up in the computation of the solution.

c. Uncertainty Quantification

(i) UQ in DA. Based on its bayesian formalism, Optimal Interpolation provides uncertainty through a posterior covariance matrix $\mathbf{P}_{\mathbf{x}|\mathbf{y}}$. Because the background $\mathbf{x}^b$ is generally considered as stationary, this covariance matrix is mainly driven by the sampling of the observation, see e.g. (Zhen et al. 2020), which may not be realistic for dynamical systems. Relying on a similar framework, DA schemes provide the deterministic part of the evolution equation in Eq. 3 as numerical model outputs, and another probabilistic model has to be given for the distribution of the noise. In addition, under Gaussian assumption of the model likelihood, closed forms exist for the mean and standard deviation of the posterior prediction. Ensemble-based sequential schemes also provide a simple way to compute flow-dependant empirical posterior covariance (Song et al. 2013) that counteracts the sampling issue used in OI or in most of variational schemes. Hybrid methods in DA (Asch et al. 2016) aims at combining both ensemble-based methods together with variational schemes to benefit from the assets of each methods.

(ii) UQ for ML approaches. As stated above, DL for UQ generally relies on the estimation of the stochastic components of the DA scheme, namely the covariance model and/or observation errors. More recently, ML schemes aims at substituting the entire DA scheme, either with an explicit model of the posterior distribution, or by a sampling strategy of the latter. On the first approach, because the true posterior is both computationally and analytically intractable, a popular strategy is to estimate an approximate posterior distribution model with trainable parametrization that involves the minimization of the Kullback-Leibler Divergence between the two distributions, which is the similar that maximizing the Evidence Lower Bound (ELBO) (Huang et al. 2019) referred as variational inference (Zhang et al. 2021). More recently, generative models are extensively used to draw samples in the prior distribution. Among them, GANs (Goodfellow et al. 2014), VAEs (Kingma and Welling 2022), normalizing flows (Dinh et al. 2017) and diffusion models (Ho et al. 2020) are the most popular. Once a way of sampling the prior is available, the posterior pdf can be obtained after conditioning through a classic DA scheme or method-related conditional generative models. This is the case in score-based data assimilation (Rozet and Louppe 2023) stated as diffusion models where Langevin iterative optimisation scheme can also embed the observation likelihood (Ho et al. 2020) to give a direct access to the posterior.

3. Neural variational schemes with SPDE priors

We present here how we draw from the neural variational scheme of Fablet et al. (2021) to embed an augmented state with SPDE prior parametrization. We explain how to parametrize the SPDE with advection-diffusion schemes compliant with many geophysical processes, and how the latter leads to the use of sparse prior precision matrix, which is a key aspect in our use of SPDE priors for surrogate models in DA. A full overview of the neural solver is also presented together with its learning scheme. Last, a presentation of the UQ scheme to generate huge ensembles of the posterior pdf is given.

a. Neural MAP solver

Because data assimilation is fundamentally bayesian, most of the methods used to interpolate an observational dataset involve the use of a model prior $\mathbf{x}$ and the computation of the posterior $\mathrm{E}[\mathbf{x}|\mathbf{y}]$ given the observations. For the latter, the computational time might be expensive, even prohibitive, because it implies to solve linear systems with matrices in high dimensions. Recently, alternate solutions have been proposed to compute the posterior. Rather than using linear algebra, we can use a traditional variational data assimilation scheme (Asch et al. 2016) and the state analysis $\mathbf{x}^\star = \mathrm{E}[\mathbf{x}|\mathbf{y}]$ is obtained by solving the minimization problem:

$$\mathbf{x}^\star = \arg\min_{\mathbf{x}} \mathcal{J}(\mathbf{x})$$

where the variational cost function $\mathcal{J}(\mathbf{x}) = \mathcal{J}_\Phi(\mathbf{x}, \mathbf{y}, \Omega)$ is generally the sum of an observation term and a regularization term involving an operator Φ which is typically a dynamical prior:

$$\begin{aligned} \mathcal{J}_\Phi(\mathbf{x}, \mathbf{y}, \Omega) &= \mathcal{J}^o(\mathbf{x}, \mathbf{y}, \Omega) + \mathcal{J}^b_\Phi(\mathbf{x}) \\ &= ||\mathcal{H}(\mathbf{x}) - \mathbf{y}||^2_\Omega + \lambda ||\mathbf{x} - \Phi(\mathbf{x})||^2 \end{aligned} \tag{8}$$

with $\mathcal{H}$ the observation operator and λ is a predefined or learnable scalar weight. This formulation of functional $\mathcal{J}_\Phi(\mathbf{x}, \mathbf{y}, \Omega)$ directly relates to weakly constraint 4DVar, see e.g. Carrassi et al. (2018). When both prior $\Phi(\mathbf{x}) = \mathbf{x}^b$ and observations $\mathbf{y}$ are assumed to be Gaussian with covariance matrices

B and **R**, $\mathcal{J}_\Phi(\mathbf{x},\mathbf{y},\Omega)$ can be written:

$$\mathcal{J}_\Phi(\mathbf{x},\mathbf{y},\Omega) = (\mathbf{H}\mathbf{x}-\mathbf{y})^{\mathrm{T}}\,\mathbf{R}^{-1}\,(\mathbf{H}\mathbf{x}-\mathbf{y}) + \lambda\left(\mathbf{x}-\mathbf{x}^b\right)^{\mathrm{T}}\mathbf{B}^{-1}\left(\mathbf{x}-\mathbf{x}^b\right) \tag{9}$$

This is well known that equating to zero the gradient of this cost function at a single time t_k produces the exact same analysis formulation that the Kalman Filter analysis step or the BLUE (simple kriging) equations. For inverse problems with time-related processes, the minimization of functional $\mathcal{J}_\Phi$ usually involves iterative gradient-based algorithms and in particular request to consider the adjoint method in classic equation-based variational data assimilation schemes (Asch et al. 2016) where operator Φ identifies to a deterministic model $\mathbf{x}_{k+1} = \mathcal{M}(\mathbf{x}_k)$:

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} - \alpha\nabla_{\mathbf{x}}\mathcal{J}_\Phi(\mathbf{x}^{(i)},\mathbf{y},\Omega)$$

Fablet et al. (2020) shows that the so-called 4DVarNet scheme, an end-to-end deep learning framework can be built based on the above variational formulation where both prior operator Φ and Maximum a posterior (MAP) solver Γ, i.e. the operator solving for the gradient-based minimization of the variational cost, are neural networks. For the latter, following meta-learning schemes (Andrychowicz et al. 2016), a residual LSTM-based representation of operator Γ is considered where the i^{th} iterative update of the solver is given by:

$$\begin{cases} \mathbf{g}^{(i+1)} &= LSTM\left[\alpha\cdot\nabla_{\mathbf{x}}\mathcal{J}_\Phi(\mathbf{x}^{(i)},\mathbf{y},\Omega), h(i), c(i)\right] \\ \\ \mathbf{x}^{(i+1)} &= \mathbf{x}^{(i)} - \mathcal{T}\left(\mathbf{g}^{(i+1)}\right) \end{cases} \tag{10}$$

with $\mathbf{g}^{(i+1)}$ is the LSTM output using as input gradient $\nabla_{\mathbf{x}}\mathcal{J}_\Phi(\mathbf{x}^{(i)},\mathbf{y},\Omega)$, while $h(i)$ and $c(i)$ denotes the internal states of the LSTM (Arras et al. 2019), α is a normalization scalar and $\mathcal{T}$ a linear or convolutional mapping.

In this formulation, the prior term Φ is jointly estimated so that the reconstruction fulfills at most the loss function $\mathcal{L}$ used during the training process. Typically, such a loss function may be stated

as:

$$\mathcal{L}(\mathbf{x}, \mathbf{x}^\star) = ||\mathbf{x} - \mathbf{x}^\star||^2 + \mathcal{L}_{\text{regul}}, \tag{11}$$

the mean squared error between the reconstruction and the ground truth, with additional regularization terms, depending on both the specific application and the targeted ouputs of the end-to-end neural scheme. This implies a bi-level optimization scheme in which we refer to Eqs. (8) and (11) resp. as the inner variational cost and the outer training loss function of 4DVarNet schemes. When replacing the outer training loss function by the same inner variational cost, such a scheme would lead to learn the exact 4DVar scheme, but with additional speeding up the optimization process. Instead, using this dual optimization enables to escape from the Gaussian and Linear hypotheses used in the 4DVar formalism so that the training of (11) acts as an additional constraint in the minimization process of (8).

b. SPDE parametrization

Though, the prior Φ is not easily interpretable: it acts as an encoding of the state $\mathbf{x}$ that helps in the gradient-based minimization process. In this work, we aim at bringing both explainability and stochasticity in the neural scheme by considering as a surrogate model for prior Φ an stochastic PDE (SPDE) instead of a given neural-based architecture. The continuous process associated with our state space describes the dynamical evolution of the spatio-temporal prior process $X_{\boldsymbol{\theta}}(\mathbf{s}, t)$ as a spatio-temporal SPDE embedding the estimation of its parametrization $\boldsymbol{\theta}$, the latter controlling key physical behaviours such as local anisotropy, correlation range, and marginal variance:

$$\mathcal{F}_{t,\boldsymbol{\theta}(t)}\{X(\mathbf{s}, t)\} = \tau(\mathbf{s}, t) Z(\mathbf{s}, t) \tag{12}$$

where operator $\mathcal{F}_{t,\boldsymbol{\theta}(t)}$ is here considered as linear, then the solution of Eq. (12) is a spatio-temporal Gaussian Process (Lindgren et al. 2011). Regarding the right-hand side noise of the SPDE, it is assumed separable, i.e. $Z(\mathbf{s}, t) = Z_t(t) \otimes Z_s(\mathbf{s})$ with $Z_t(t)$ a temporal white noise. $Z_s(\mathbf{s})$ can also be a spatial white noise or we may consider a colored noise to ensure more regularity on process X accross time.

Following a state space formalism, the discretization of the stochastic process $X(\mathbf{s},t)$ is a multivariate gaussian vector $\mathbf{x}$:

$$\mathbf{x} \sim \mathcal{N}\left(\mathbf{x}^b, \mathbf{Q}_{\boldsymbol{\theta}}^b\right) \tag{13}$$

where $\mathbf{x}^b$ is the deterministic mean of the state, typically *the background*, a coarse approximation of the field in stationary formulations, or the *forecast* in a dynamical data assimilation scheme. $\mathbf{Q}_{\boldsymbol{\theta}}^b$ is the precision matrix (inverse of the covariance matrix) of the state sequence $\{\mathbf{x}_0, \cdots, \mathbf{x}_{Ldt}\}$. Because of the explicit link between linear stochastic PDEs and Gaussian Processes, we modify the variational formulation used in 4DVarNet schemes, see e.g. Beauchamp et al. (2023c), by rewriting the matrix formulation of the regularization prior term of Eq. 8 as:

$$\mathbf{x}^\star = \text{argmin}_{\mathbf{x}} \mathcal{J}(\mathbf{x}, \mathbf{y}, \Omega) = \text{argmin}_{\mathbf{x}} ||\mathbf{H}\mathbf{x} - \mathbf{y}||_\Omega^2 + \lambda \left[\mathbf{x} - \mathbf{x}^b\right]^{\mathrm{T}} \mathbf{Q}_{\boldsymbol{\theta}}^b \left[\mathbf{x} - \mathbf{x}^b\right]$$

When $\mathbf{x}^b = \mathbf{0}$ (for simplification) and when denoting $\mathbf{L}$ the square root of the precision matrix $\mathbf{Q}_{\boldsymbol{\theta}}^b$, the identification with 4DVarNet schemes is direct for $\Phi = (1 - \mathbf{L})$. Indeed, $||\mathbf{x}||_{\mathbf{Q}_{\boldsymbol{\theta}}}^2 = \mathbf{x}^{\mathrm{T}}\mathbf{L}^{\mathrm{T}}\mathbf{L}\mathbf{x} = ||\mathbf{L}\mathbf{x}||^2 = ||\mathbf{x} - \Phi \cdot \mathbf{x}||^2$. In this derived formulation, it is clear that if the SPDE is known, it can be embedded in the inner variational cost used by the LSTM iterative solver to optimize the outer training loss function. Drawing from the usual neural variational framework, the trainable prior is now SPDE-based, and the parameters $\boldsymbol{\theta}$ are embedded in the following augmented state formalism:

$$\tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{x} & \boldsymbol{\theta} \end{bmatrix}^{\mathrm{T}} \tag{14}$$

The latent parameter $\boldsymbol{\theta}$ is potentially non stationary in both space and time and its size is directly related to the size of the data assimilation window L.

When dealing with geophysical fields, a generic class of non-stationary models generated by stochastic PDEs shall introduce some diffusion and/or advection terms. They are respectively obtained by introducing a local advection operator $\mathbf{m}(\mathbf{s},t) \cdot \nabla$ where $\mathbf{m}$ is a velocity field and a local

diffusion operator $\nabla \cdot \mathbf{H}(\mathbf{s},t) \cdot \nabla$ where $\mathbf{H}$ acts as a two-dimensional diffusion tensor:

$$\frac{\partial X}{\partial t} + \left\{\boldsymbol{\kappa}^2(\mathbf{s},t) + \mathbf{m}(\mathbf{s},t) \cdot \nabla - \nabla \cdot \mathbf{H}(\mathbf{s},t) \cdot \nabla\right\}^{\alpha/2} X(\mathbf{s},t) = \boldsymbol{\tau}(\mathbf{s},t) Z(\mathbf{s},t) \tag{15}$$

This way of handling spatio-temporal non-stationarities in SPDE models has been first mentioned in the original paper of Lindgren et al. (2011), but because of the challenge of estimating the full set of space-time parameters, no works have been published to our knowledge pushing this framework into its completeness. Though, some parametrization of purely spatial diffusion process (Fuglstad et al. 2015c) or stationary advection-dominated SPDE (Clarotto et al. 2022) has been successfully applied.

Let stress that the advection-diffusion scheme is a good candidate for many geophysical datasets: this is the case in quasi-geostrophic approximation of Sea Surface Temperature (SST), see e.g. (Ubelmann et al. 2014), but also in the dispersion of atmospheric pollutants (Menut et al. 2021) for instance. Then, this framework provides a generic and convenient way to bring more explainability in terms of space-time covariances of dynamical processes.

Such a model implies to estimate new parameters $\kappa(\mathbf{s},t)$, $\mathbf{H}_{2\times2}(\mathbf{s},t)$ and $\mathbf{m}_{2\times1}(\mathbf{s},t) = \begin{bmatrix}\mathbf{m}^1 & \mathbf{m}^2\end{bmatrix}^{\mathrm{T}}$, all varying across space and time along the data assimilation window. In addition, κ needs to be continuous while $\mathbf{m}$ and $\mathbf{H}$ additionnaly requires to be continuously differentiable. Regarding the diffusion tensor, we draw from the spatial statistics literature, see e.g. (Fuglstad et al. 2015a), to introduce the scalars $\gamma(\mathbf{s},t)$, $\beta(\mathbf{s},t)$, $\mathbf{v_1}(\mathbf{s},\mathbf{t})$ and $\mathbf{v_2}(\mathbf{s},\mathbf{t})$ as a generic decomposition of $\mathbf{H}(\mathbf{s},t)$ through the equation:

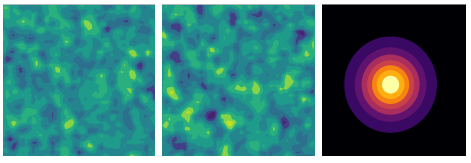
$$\mathbf{H}(\mathbf{s},t) = \begin{bmatrix}\mathbf{H}^{1,1} & \mathbf{H}^{1,2} \\ \mathbf{H}^{1,2} & \mathbf{H}^{2,2}\end{bmatrix}(\mathbf{s},t) = \gamma(\mathbf{s},t)\mathbf{I}_2 + \beta(\mathbf{s},t)\mathbf{v}\mathbf{v}^{\mathrm{T}}$$

with $\mathbf{v}^{\mathrm{T}} = \begin{bmatrix}\mathbf{v_1}(\mathbf{s}) & \mathbf{v_2}(\mathbf{s})\end{bmatrix}$, which models the diffusion tensor as the sum of an isotropic and anisotropic effects, the latter being described by its amplitude and magnitude. This is a valid decomposition for any symmetric positive-definite 2×2 matrix. This leads to the SPDE hyper-parametrization $\boldsymbol{\theta}$ of size $m \times L \times 8$ parameters ($\mathbf{H}^{1,2} = \mathbf{H}^{2,1}$): it grows linearly with the potentially

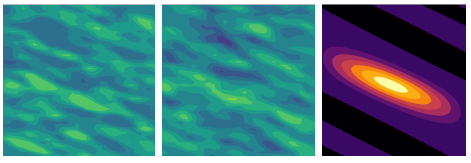
high dimensional state space. In the end, the SPDE hyperparametrization states as:

$$\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\kappa} & \mathbf{m} & \mathbf{H} & \boldsymbol{\tau} \end{bmatrix}^{\mathrm{T}}$$

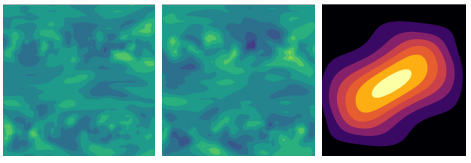
Fig.2 provide some specific examples to show how the modifications in the fractional differential operator leads to more complex spatio-temporal anisotropies. In this four SPDE parametrizations, $\kappa = 0.33$, $\tau = 1$ and $\alpha = 4$.



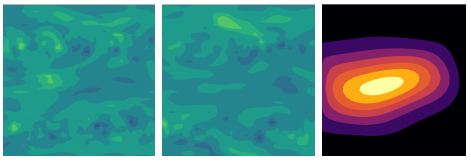
(a) Isotropic model: $\frac{\partial \mathbf{x}}{\partial t} + (\kappa^2(\mathbf{s},t) - \Delta)^{\alpha/2}\mathbf{x}(\mathbf{s},t) = \tau\mathbf{z}(\mathbf{s},t))$



(b) Global anisotropy: $\frac{\partial \mathbf{x}}{\partial t} + \left\{\kappa^2(\mathbf{s},t) - \nabla\cdot\mathbf{H}\nabla\right\}^{\alpha/2}\mathbf{x}(\mathbf{s},t) = \tau\mathbf{z}(\mathbf{s},t)$



(c) Local anisotropy with diffusion: $\frac{\partial \mathbf{x}}{\partial t} + \left\{\kappa^2(\mathbf{s},t) - \nabla\cdot\mathbf{H}(\mathbf{s})\nabla\right\}^{\alpha/2}\mathbf{x}(\mathbf{s},t) = \tau\mathbf{z}(\mathbf{s},t)$



(d) Local anisotropy + Global advection: $\frac{\partial \mathbf{x}}{\partial t} + \left\{\kappa^2(\mathbf{s},t) + \mathbf{m}\cdot\nabla - \nabla\cdot\mathbf{H}(\mathbf{s})\nabla\right\}^{\alpha/2}\mathbf{x}(\mathbf{s},t) = \tau\mathbf{z}(\mathbf{s},t)$

FIG. 2: For (a), (b), (c) and (d), one realization of the corresponding SPDE-driven GP at time $t = 10$ (left panel), $t = 20$ (middle panel) and covariance (right panel) with central point of domain $\mathcal{D} = [0,1]\times[0,1]$

The space-time SPDE is discretized based on a numerical implicit Euler scheme:

$$\frac{\mathbf{x}_{t+dt} - \mathbf{x}_t}{dt} + \mathbf{B}_{t+dt}\mathbf{x}_{t+dt} = \frac{\tau}{\sqrt{dt}}\mathbf{z}_{t+dt}$$

where $\mathbf{x}_t$ and $\mathbf{B}_t$ respectively denote the state space and the finite difference discretization of the fractional differential operator $\frac{\partial}{\partial t} + \left\{\boldsymbol{\kappa}^2(\mathbf{s},t) + \mathbf{m}(\mathbf{s},t)\cdot\nabla - \nabla\cdot\mathbf{H}(\mathbf{s},t)\nabla\right\}^{\alpha/2}$ at time $t = 0,\cdots,L$. The noise $\mathbf{z}_{t+dt}$ is white in space and $\mathbf{M}_{t+dt} = (\mathbf{I} + dt\mathbf{B}_{t+dt})^{-1}$ denotes the matrix operator that emulates the dynamical evolution of state $\mathbf{x}$ from time t to $t+dt$. In this case, $\mathbf{T}_{t+dt} = \tau\sqrt{dt}\mathbf{M}_{t+dt}$ corresponds to the dynamical linear model regularized by the product of the noise variance with the square root of the SPDE time step. In case of a more general right-hand term with non-uniform regularization

variance $\{\tau_t,\, t>0\}$, and $\{\mathbf{z}_t,\, t>0\}$ are independent realizations of a colored noised driven by a spatial isotropic SPDE:

$$(\kappa_s^2-\Delta)^{\alpha_s/2}Z_s(\mathbf{s})=W(\mathbf{s}) \tag{16}$$

with $W(\mathbf{s})$ a white noise with unit variance, the spatial FDM together with time Euler discretization leads to:

$$\mathbf{x}_{t+dt}=\mathbf{M}_{t+dt}\mathbf{x}_t+\widetilde{\mathbf{T}}_{t+dt}\mathbf{z}_{t+dt} \tag{17}$$

where $\widetilde{\mathbf{T}}_{t+dt}=\sqrt{dt}\mathbf{M}_{t+dt}\tau_t\mathbf{L}_s$ and $\mathbf{L}_s$ stands for the Cholesky decomposition of the discretized spatial precision matrix $\mathbf{Q}_s$ introduced of the stochastic process Z_s introduced by Eq. (16). In the case of advection-dominated SPDEs, we involve state-of-the-art upwind schemes (UFDM) for stabilization of the numerical system, by letting the advective transport term, which is the dominating term, collect its information in the flow direction, i.e., upstream or upwind of the point in question. All the calculation details are given in Appendix 5.

In a compact formulation, using centered finite differences on the diffusion term, and by denoting $\mathbf{a}_{i,j}^{1,t,+}=\max(\mathbf{m}_{i,j}^{1,t},0)$, $\mathbf{a}_{i,j}^{1,t,-}=\min(\mathbf{m}_{i,j}^{1,t},0)$, $\mathbf{a}_{i,j}^{2,t,+}=\max(\mathbf{m}_{i,j}^{2,t},0)$, $\mathbf{a}_{i,j}^{2,t,-}=\min(\mathbf{m}_{i,j}^{2,t},0)$, the resulting UFDM scheme is:

$$\begin{aligned}\mathbf{x}_{i,j}^{t+1}=\mathbf{x}_{i,j}^{t}+dt\Big[&\kappa_{i,j}^{t}\mathbf{x}_{i,j}^{t}+\left(\mathbf{a}_{i,j}^{1,t,+}\mathbf{m}_{i,j}^{1,t,-}+\mathbf{a}_{i,j}^{1,t,-}\mathbf{m}_{i,j}^{1,t,+}\right)+\left(\mathbf{a}_{i,j}^{2,t,+}\mathbf{m}_{i,j}^{2,t,-}+\mathbf{a}_{i,j}^{2,t,-}\mathbf{m}_{i,j}^{2,t,+}\right)\\&+\mathbf{H}_{i,j}^{1,1,t}\frac{\mathbf{x}_{i+1,j}^{t}-2\mathbf{x}_{i,j}^{t}+\mathbf{x}_{i-1,j}^{t}}{dx^2}+\mathbf{H}_{i,j}^{2,2,t}\frac{\mathbf{x}_{i,j+1}^{t}-2\mathbf{x}_{i,j}^{t}+\mathbf{x}_{i,j-1}^{t}}{dy^2}\\&+\mathbf{H}_{i,j}^{1,2,t}\frac{\mathbf{x}_{i+1,j+1}^{t}-\mathbf{x}_{i+1,j-1}^{t}-\mathbf{x}_{i-1,j+1}^{t}+\mathbf{x}_{i-1,j-1}^{t}}{2dxdy}+\tau_{i,j}^{t}\mathbf{z}_{i,j}^{t+1}\Big]\end{aligned}$$

Starting from this numerical scheme, the modified 4DVarNet scheme requires the precision matrix $\mathbf{Q}_{\boldsymbol{\theta}}^{b}$ of the state sequence $\{\mathbf{x}_0,\cdots,\mathbf{x}_{Ldt}\}$. Here, $\mathbf{x}_0\sim\mathcal{N}(\mathbf{0},\mathbf{P_0})$ denotes the initial state and $\mathbf{Q}_0=\mathbf{P_0}^{-1}$ is always taken as the precision matrix obtained after a given stabilization run, i.e. the evolution of the dynamical system over N timesteps using as stationary parameters the initial

parametrization $\boldsymbol{\theta}_0$ of the SPDE at time $t = 0$, then we can rewrite :

$$\{\mathbf{x}_0, \cdots, \mathbf{x}_{Ldt}\} = \mathbf{M}_G \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{z} \end{bmatrix}$$

with $\mathbf{z} = [\mathbf{z}_1, \ldots, \mathbf{z}_t]^{\mathrm{T}}$ and

$$\mathbf{M}_G = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_1 & \mathbf{T}_1 & 0 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_2\mathbf{M}_1 & \mathbf{M}_2\mathbf{T}_1 & \mathbf{T}_2 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_3\mathbf{M}_2\mathbf{M}_1 & \mathbf{M}_3\mathbf{M}_2\mathbf{T}_1 & \mathbf{M}_3\mathbf{T}_2 & \mathbf{T}_3 & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \mathbf{T}_L \end{bmatrix}$$

With the additional notation $\mathbf{S}_k = \mathbf{T}_k \mathbf{T}_k^{\mathrm{T}}$, see Eq. (17), the precision matrix $\mathbf{Q}^b$ writes, see Appendix c for all the details:

$$\mathbf{Q}_{\boldsymbol{\theta}}^b = \frac{1}{dt} \begin{bmatrix} \mathbf{P}_0^{-1}+\widetilde{\mathbf{Q}}_{s,1} & -\widetilde{\mathbf{Q}}_{s,1}\mathbf{M}_1^{-1} & 0 & 0 & 0 & ... & 0 \\ -\left(\mathbf{M}_1^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,1} & \mathbf{M}_1^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,1}\mathbf{M}_1+\widetilde{\mathbf{Q}}_{s,2} & -\widetilde{\mathbf{Q}}_{s,2}\mathbf{M}_2^{-1} & 0 & 0 & ... & 0 \\ 0 & -\left(\mathbf{M}_2^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,2} & \mathbf{M}_2^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,2}\mathbf{M}_2+\widetilde{\mathbf{Q}}_{s,3} & -\widetilde{\mathbf{Q}}_{s,3}\mathbf{M}_3^{-1} & 0 & ... & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\left(\mathbf{M}_{L-1}^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,L-1} & \mathbf{M}_L^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,L-1}\mathbf{M}_{L-1}+\widetilde{\mathbf{Q}}_{s,L} & -\widetilde{\mathbf{Q}}_{s,L}\mathbf{M}_L^{-1} \\ 0 & \ddots & \ddots & \ddots & 0 & -\left(\mathbf{M}_L^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,L} & \mathbf{M}_L^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,L}\mathbf{M}_L \end{bmatrix} \tag{18}$$

where $\widetilde{\mathbf{Q}}_{s,t}$ is the precision matrix of the colored noise weighted by the non-uniform regularization variance $\boldsymbol{\tau}_t$. As clearly visible, the sparsity of $\mathbf{Q}_{\boldsymbol{\theta}}^b$ is high, which is key in traditional SPDE-based GP inference, but also in our approach, see Section c.

c. Neural solver with augmented state

Overall, let denote by $\Psi_{\boldsymbol{\theta},\Gamma}(\tilde{\mathbf{x}}^{(0)}, \mathbf{y}, \Omega)$ the output of the end-to-end learning scheme given the SPDE-based dynamical model with parameters $\boldsymbol{\theta}$ and the neural residual architecture for the solver

Γ, see Fig. 3 and Algorithm 1, the initialization $\tilde{\mathbf{x}}^{(0)}$ of augmented state $\tilde{\mathbf{x}}$ and the observations $\mathbf{y}$ on domain Ω.

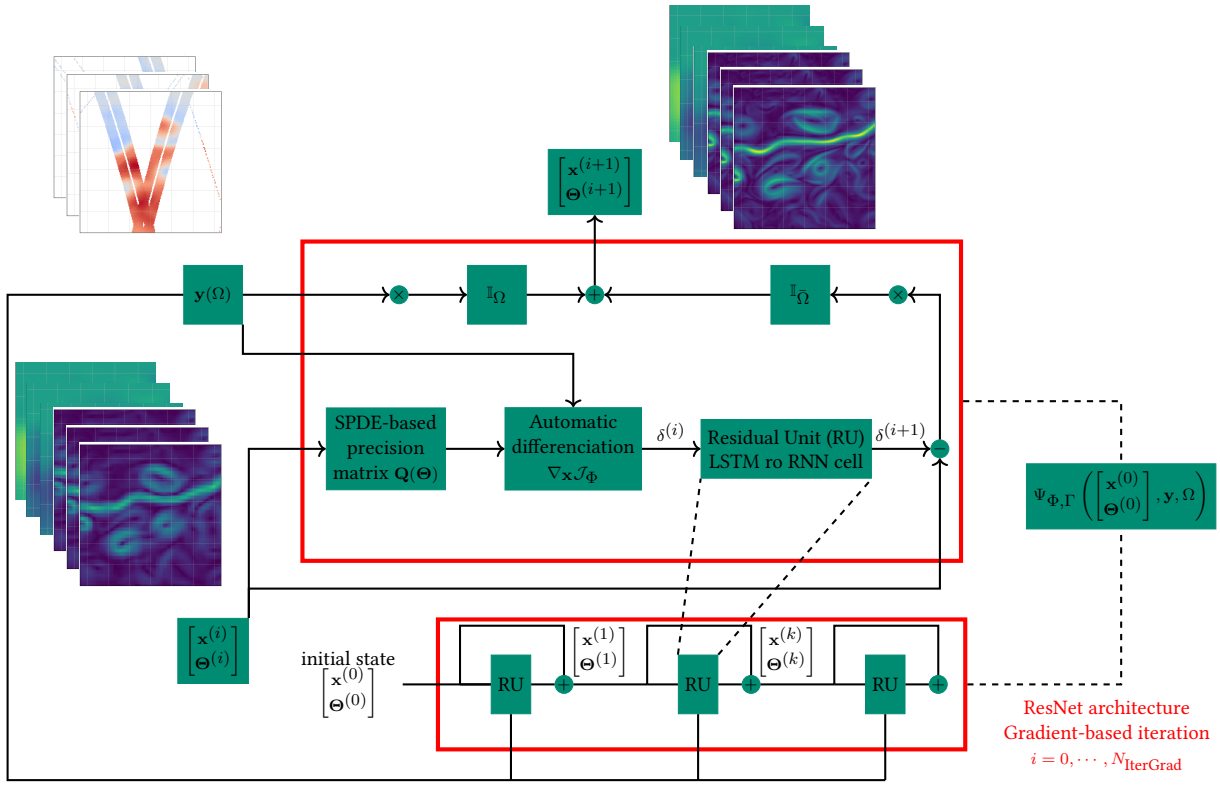


FIG. 3: Sketch of the gradient-based algorithm. $\mathbf{I}_\Omega$ acts as a masking operator for any spatio-temporal location not in Ω.

Algorithm 1: Variational scheme with SPDE-based GP prior and implicit neural solver

Data:

$\mathbf{x} \in \mathbb{R}^{T \times m} = \{\mathbf{x}_k\},\ k = 1, \cdots, T$

$\mathbf{y}_\Omega = \{\mathbf{y}_{k,\Omega_k}\},\ k = 1, \cdots, T$: observations on domains $\Omega_k \subset \mathcal{D}$

N_I: number of iterations

η: gradient step

Init:

$\tilde{\mathbf{x}}^{\star,(0)}$

List of procedures:

Train_$\Psi_{\boldsymbol{\theta},\Gamma}$: end-to-end learning procedure with:

$\boldsymbol{\theta}$: parameter of the SPDE-based prior operator;

$GradLSTM$: residual NN-based representation of $\nabla_{\mathbf{x}}\mathcal{J}(\mathbf{x})$

Γ: iterative gradient-based update operator:

$i = 0$

while $i < N_I$ **do**

$$\mathbf{Q}^b_{\boldsymbol{\theta}^{\star,(i)}} = \mathbf{P}_G^{-1}(\boldsymbol{\theta}^{\star,(i)}) = \mathbf{M}_G^{-1}(\boldsymbol{\theta}^{\star,(i)})^{\mathrm{T}} \begin{bmatrix} \mathbf{P}_0^{-1} & 0 & \dots & 0 \\ 0 & \mathbf{I} & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{I} \end{bmatrix} \mathbf{M}_G^{-1}(\boldsymbol{\theta}^{\star,(i)})$$

$\tilde{\mathbf{x}}^{(i+1)} \leftarrow \tilde{\mathbf{x}}^{(i)} - \eta \times GradLSTM(\tilde{\mathbf{x}}^{(i)})$

$N_I \nearrow\ ; \eta \searrow\ ;\ i \leftarrow i+1$

end

for $i \in 0, \cdots, n_{epochs}$ **do**

$\omega^{(i+1)}_{\Psi_{\boldsymbol{\theta},\Gamma}} \leftarrow \omega^{(i)}_{\Psi_{\boldsymbol{\theta},\Gamma}} - lr \times \nabla\mathcal{L}(\mathbf{x}, \mathbf{x}^{\star,(i)}, \boldsymbol{\theta}^{(i)})$

end

Result: $\tilde{\mathbf{x}}^{\star} \leftarrow \Psi_{\boldsymbol{\theta},\Gamma}(\tilde{\mathbf{x}}^{(0)}, \mathbf{y}, \Omega)$

Then, the joint learning for the weights $\omega_{\boldsymbol{\theta},\Gamma}$ of the neural scheme given the SPDE formulation that is chosen (isotropic or not, non-stationary or not, etc.) and the architecture of operator Γ is

stated as the minimization of the mixed loss function $\mathcal{L}(\mathbf{x}, \mathbf{x}^\star, \boldsymbol{\theta}^\star)$, fully explained in Section d:

$$\omega^\star_{\Psi_{\boldsymbol{\theta},\Gamma}} = \arg\min_{\boldsymbol{\theta},\Gamma} \left[\mathcal{L}(\mathbf{x}, \mathbf{x}^\star, \boldsymbol{\theta}^\star) \right] \text{ s.t. } \tilde{\mathbf{x}}^\star = \Psi_{\boldsymbol{\theta},\Gamma}(\tilde{\mathbf{x}}^{(0)}, \mathbf{y}, \Omega) \tag{19}$$

(i) Initialization of the augmented state. The initial state $\tilde{\mathbf{x}}^{(0)}$ should be filled with 0 in the non-observed part of domain $\mathcal{D}$. Because there is obviously no observation of the SPDE parameters, we might guide the training process at the first iteration of the solver. This proved to be particularly helpful when dealing with realistic geophysical datasets for which the advection diffusion scheme is meaningful. In that case, see for instance Application 2 on real SSH datasets in Section 4, we used first and second-order derivatives of the initial state $\mathbf{x}^{(0)}$ as initial parametrizations for the advection and diffusion process:

$$\mathbf{m}^{(0)} = \begin{pmatrix} \frac{\partial \mathbf{x}^{(0)}}{\partial x} \\ \frac{\partial \mathbf{x}^{(0)}}{\partial y} \end{pmatrix}, \text{ and } \mathbf{v}^{(0)} = \begin{pmatrix} \frac{\partial^2 \mathbf{x}^{(0)}}{\partial x^2} \\ \frac{\partial^2 \mathbf{x}^{(0)}}{\partial y^2} \end{pmatrix}$$

$\boldsymbol{\kappa}^{(0)}$ and $\boldsymbol{\tau}^{(0)}$ are both using absolute values of the normalized gradient norms, while $\beta^{(0)}$ and $\gamma^{(0)}$ are resp. set to 1 and 0. Because κ, τ and γ are both strictly positive, see Section b, we use ReLu activation function on these three parameters to ensure their consistency.

(ii) Computational aspects. The sparse formulation of the precision matrix $\mathbf{Q}^b_{\boldsymbol{\theta}}$ is key in the memory-saving component of the algorithm because the latter relies on a set of N_I gradient-based iterations, meaning that for a single interpolation task, the precision matrix $\mathbf{Q}^b_{\boldsymbol{\theta}}$ is stored N_I times along the computational graph with updated values of the SPDE parameters $\boldsymbol{\theta}^{(i)}$.

d. Learning scheme

(i) Training loss. The joint problem of estimating the best reconstruction and infering realistic SPDE parametrizations is difficult because according to the size and nature of the dataset, the spatio-temporal interpolation may not always benefit from knowing the exact set of true SPDE parameters. Indeed, if the degree of sparsity of the observation dataset is low, the reconstruction may be good despite a poor estimation of the covariance matrix. In the other way, if the degree of sparsity is high, much more difficult will be the estimation of the SPDE underlying parameters.

In this section, we benefit from the supervised configuration of neural variational scheme to train this joint problem. In this Section, we show how to embed the global precision matrix defined by Eq. 18 in our neural scheme and define which training loss is the more appropriate to handle the bi-level optimization scheme (of both the inner variational cost and the training loss).

For the training process, we may consider different loss functions:

- $\mathcal{L}_1(\mathbf{x}, \mathbf{x}^\star) = ||\mathbf{x} - \mathbf{x}^\star||^2$ is the L2-norm of the difference between state $\mathbf{x}$ and reconstruction $\mathbf{x}^\star$

- $\mathcal{L}_2(\mathbf{x}, \boldsymbol{\theta}^\star) = -|\mathbf{Q}^b_{\boldsymbol{\theta}^\star}| + \mathbf{x}^{\mathrm{T}}\mathbf{Q}^b_{\boldsymbol{\theta}^\star}\mathbf{x}$ is the negative log-likelihood of the true states given the estimated precision matrix $\mathbf{Q}^b_{\boldsymbol{\theta}^\star}$, thus ensuring consistency between the actual ground truth and the SPDE parameters.

Using $\mathcal{L}_1$ will lead to satisfactory reconstructions without any constraints on the SPDE parameters. The single use of $\mathcal{L}_2$ should lead to satisfactory results if the analytical solver, i.e. the inversion of the linear system or the gradient-based minimization of the variational cost, were used. But because the solver is trained, it also needs to be constrained by an appropriate loss function for the reconstruction, meaning $\mathcal{L}_1$. The best solution is then to create a mixed loss function, combination of $\mathcal{L}_2$ to estimate at best the SPDE parameters and optimize the prior model, and $\mathcal{L}_1$ to satisfy the reconstruction criteria and optimize the solver.

The log-determinant of the precision matrix $\log|\mathbf{Q}^b_{\boldsymbol{\theta}^\star}|$ is usually difficult to handle when computing $\mathcal{L}_2$. Hopefully, based on the particular structure and the notations already introduced for the spatio-temporal precision matrix $\mathbf{Q}^b_{\boldsymbol{\theta}}$ in Eq.(B2), it writes, see e.g. Clarotto et al. (2022):

$$
\begin{aligned}
\log|\mathbf{Q}^b_{\boldsymbol{\theta}^\star}| &= \log|\mathbf{P}_0^{-1}| + \log|\mathbf{S}_1^{-1}| + \cdots + \log|\mathbf{S}_L^{-1}| \\
&= \log|\mathbf{P}_0^{-1}| + \sum_{i=1}^{L} \log\left(|\mathbf{L}_i\mathbf{L}_i^{\mathrm{T}}|\right) \\
&= \log|\mathbf{P}_0^{-1}| + 2\sum_{i=1}^{L}\sum_{j=1}^{m} \log \mathbf{L}_i(j,j)
\end{aligned}
\tag{20}
$$

where $\mathbf{L}_i$ denotes here the Cholesky decomposition of $\mathbf{S}_k^{-1}$. In case of unsupervised learning, the same strategy may apply but $\mathcal{L}_2$ will be the likelihood of the observations given the estimated SPDE parameters since in this case, the true states would not be available during the training process.

(ii) Two-step learning schemes. In the initial version of 4DVarNet schemes, the output $\mathbf{x}^\star$ provided by the neural formulation is deterministic and can be seen as the posterior mean of the state given the observations. Within this SPDE-based parametrization of the prior $\mathbf{x}^b$, we also aim at providing the distribution of the prior as a GP process. Though, for realistic geophysical fields, even the prior cannot be considered as a zero-mean Gaussian field and some non-linearities have to be accounted for in its deterministic mean $\mathbf{x}^b$. To solve for this specific case, we involve a 2-step learning process in which the deterministic mean $\mathbf{x}^b$ is first estimated by a 4DVarNet scheme applied on coarser resolution than the actual observations. Second, the modified 4DVarNet scheme is involved to estimate jointly the posterior mean $\mathbf{x}^\star$ together with the SPDE parametrization $\boldsymbol{\theta}^\star$ of the prior $\mathbf{x} \sim \mathcal{N}\left(\mathbf{x}^b, \mathbf{Q}^b_{\boldsymbol{\theta}^\star}\right)$. This two-steps procedure also enables to simplify the SPDE training scheme: the first guess is estimated based on the entire set of observations along the state sequence, while the SPDE parametrization is estimated only on a reduced window of length 5, centered on the targeted time of interest. This considerably reduces the size of precision matrix $\mathbf{Q}^b_{\boldsymbol{\theta}}$, making the algorithm scalable for any application. Fig. 4 shows a schematic overview of this two-steps learning scheme with illustrations coming from realistic Sea Surface Heigh datasets provided in Application 2, Section 4. In the end, this will lead to a stochastic version of the neural variational scheme based on the prior GP distribution in which we can sample ensemble members.

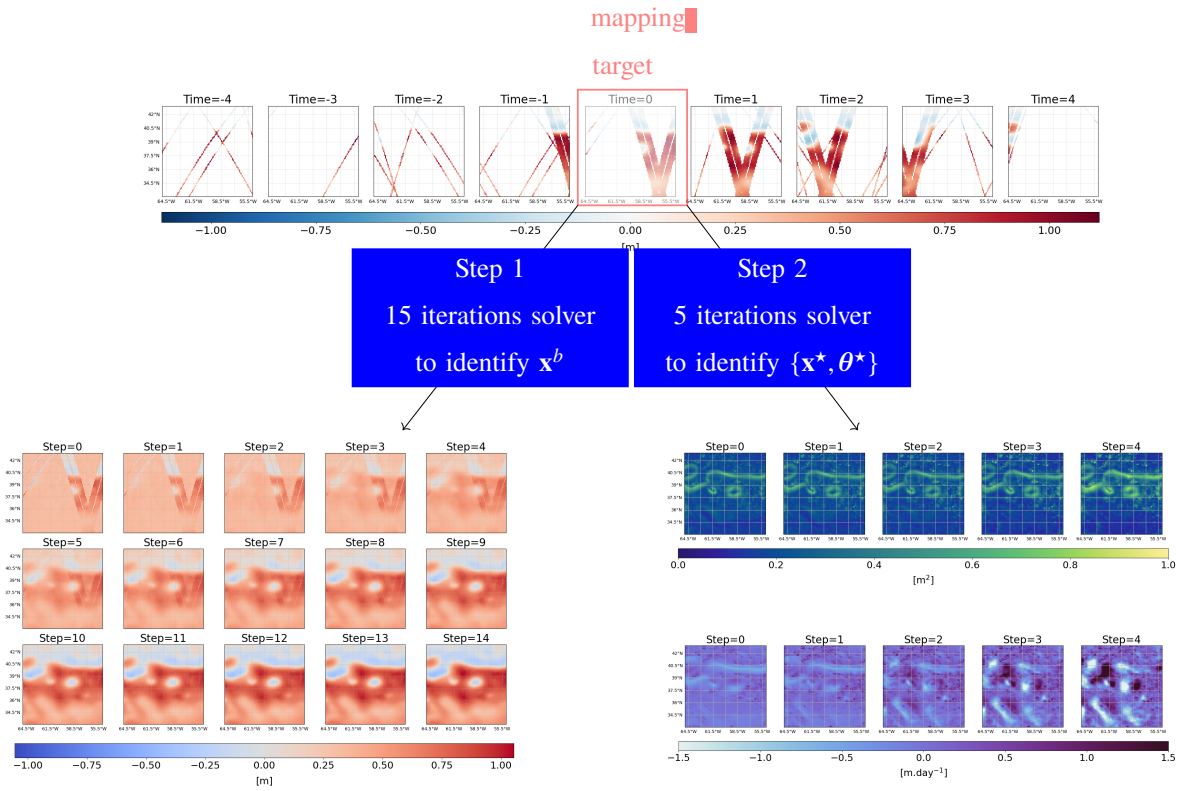


FIG. 4: Two-step adaptation of the 4DVarNet scheme: on the top panel, an example of 4 nadirs+SWOT observations along the 9-day data assimilation window, see Application 2 in Section 4. The target reconstruction day is at the center of the DAW. On the bottom-left panel is displayed the 15 iterations of the first 4DVarNet solver to retrieve the prior mean $\mathbf{x}^b$: at the beginning, the non-observed parts of the domain are filled with the global mean of the training dataset (0 when normalized). On the bottom-right panel is displayed the 5 iterations of the modified 4DVarNet solver with augmented state formalism to retrieve both $\mathbf{x}^\star$ and $\boldsymbol{\theta}^\star$: at the beginning, the parameters in $\boldsymbol{\theta}$ are initialized with partial derivatives of $\mathbf{x}^b$, see Section 3.c

(iii) Complementary PyTorch developments. Regarding the implementation of our model, we use Pytorch (Paszke et al. 2017) whose sparse linear algebra is not providing yet a sparse Cholesky algorithm and a sparse solver of linear systems, which is critical, especially when computing the likelihood $\mathcal{L}_2$ in the training loss function. As a consequence, despite the theoretical tools have been fully detailed in the previous sections, we had to implement new functionalities based on scipy sparse linear algebra (Virtanen et al. 2020) to store the precision matrices in an efficient way and compute the inner variational cost, see again Eq.8. In particular, we provide a PyTorch extension of sparse cholesky matrices, see Appendix c, based on the CHOLMOD supernodal Cholesky factorization (Chen et al. 2008) and draw from Seeger et al. (2019) to provide the

backward pass of the sparse Cholesky decomposition.

e. UQ scheme

Because the PDE is stochastic, it provides an easy way to generate a set of N gaussian prior simulations $\mathbf{x}_i$, $i = 1, \cdots, N$:

$$\mathbf{x}_i = \mathbf{x}^b + \mathbf{L}^b_{\boldsymbol{\theta}} \mathbf{z}_i$$

where $\mathbf{L}^b_{\boldsymbol{\theta}}$ stands for the Cholesky decomposition of $\mathbf{Q}^b_{\boldsymbol{\theta}}$ and $\mathbf{z}_i$ is a white noise. From a geostatistical point of view, this can be seen as SPDE-based spatio-temporal non conditional simulations of state $\mathbf{x}$, meaning that we produce surrogate simulations sharing the same physical properties than the true states. These simulations are then conditioned by the neural solver given the observations available $\mathbf{y}$, see Fig. 5. To do so, we draw from traditional geostatistics to realize SPDE-based spatio-temporal non conditional simulations with a kriging-based conditioning (Wackernagel 2003). Except that we replace the kriging algorithm by our neural approach, which has to be seen as a generic interpolation tool here:

$$\mathbf{x}^\star_i(\mathbf{s}, t) = \mathbf{x}^\star(\mathbf{s}, t) + (\mathbf{x}_i(\mathbf{s}, t) - \widehat{\mathbf{x}}_i(\mathbf{s}, t)) \tag{21}$$

where $\mathbf{x}^\star$ denotes the neural-based interpolation, $\mathbf{x}_i$ is one SPDE non-conditional simulation of the process $\mathbf{x}$ based on the parameters $\boldsymbol{\theta}^\star$ and $\widehat{\mathbf{x}}_i$ is the neural reconstruction of this non-conditional simulation, using as pseudo-observations a subsampling of $\mathbf{x}_i$ based on the actual data locations. Because $\mathrm{E}[\mathbf{x}_i - \widehat{\mathbf{x}}_i] = 0$, the resulting simulation is well conditioned by the observations at data locations.

Running an ensemble of N conditional simulations gives an approximation of the probability distribution function $p_{\mathbf{x}|\mathbf{y}}$ of state $\mathbf{x}^\star = \mathbf{x}|\mathbf{y} = \{\mathbf{x}_0|\mathbf{y}, \cdots, \mathbf{x}_L|\mathbf{y}\}$. The ensemble mean $\overline{\mathbf{x}^\star_i}$ will be $\mathbf{x}^\star$ in

the limits of $N \longrightarrow +\infty$:

$$\frac{1}{N}\sum_i \mathbf{x}_i^\star(\mathbf{s},t) \xrightarrow[N\to+\infty]{} \mathbf{x}^\star(\mathbf{s},t)$$

Such an approach has already been successfully tested in Beauchamp et al. (2023a) when using analog operator strategy (Tandeo et al. 2015) to draw non-conditional simulation in the prior distribution.

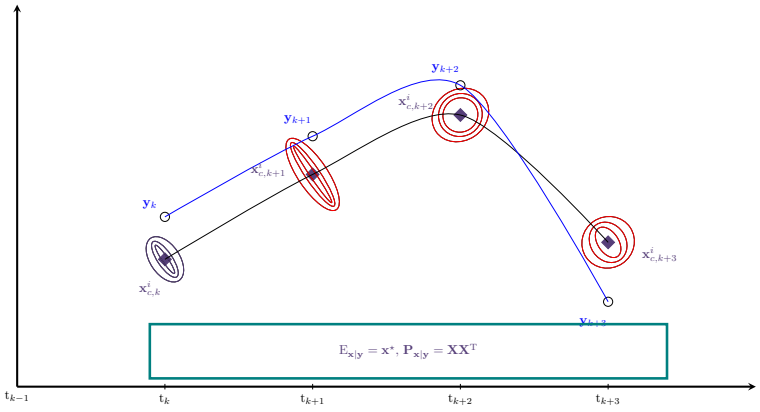


FIG. 5: Ensemble-based neural variational scheme with SPDE-based GP prior: the assimilation is not sequential. The inversion scheme embeds the global precision matrix of the state sequence $\{\mathbf{x}_0, \cdots, \mathbf{x}_{Ldt}\}$ in the inner variational cost to minimize. Ensemble members are generated from the Gaussian prior surrogate SPDE model, then conditioned by the neural solver so that the posterior pdf is no longer Gaussian. The posterior pdf is empirically ensemble-based computed: $\mathbf{P}_{\mathbf{x}|\mathbf{y}} = \mathbf{X}\mathbf{X}^\mathrm{T}$ with $\mathbf{X} = (1/\sqrt{N-1})\left[\mathbf{x}_i^\star - \mathbf{x}^\star\right]$

Let note that in this formulation, the idea is to run SPDE-based conditional simulation of the prior Matérn field $X(\mathbf{s},t)$. One ensemble member is obtained by running one space-time simulation, and two space-time neural reconstructions along the data assimilation window, then combined through Eq. (21). As a consequence, there is no sequential assimilation. Though, in the idea, such a conditioning is exactly similar to the one produced by EnKF simulations: in geostatistical terms we can interpret the forecast step of the EnKF as being unconditional simulations at time $t-1$ generating N realizations of a non-stationary random function (SPDE-based here) for time t. Both prior mean and covariance matrix are then computed directly on this set of unconditional realizations, before the analysis step, i.e. their conditiong with the observations. The key advantage of the EnKF is its flow-dependency that propagates the uncertainties at each

time step with the evolution model, while the classic EnOI method generally used by geoscientists, see e.g. Asch et al. (2016); Counillon and Bertino (2009) replaces the flow-dependent EnKF error covariance matrix by a stationary matrix calculated from an historical ensemble. This is an important difference with our approach: while the neural architecture can be seen as a way to learn neural ensemble-based optimal interpolation models and solvers, our GP prior encoded by its SPDE precision matrix, built according to the FDM scheme with varying parameters over space and time, still allows for flow dependency, based on the learning of the SPDE parametrization given the input observations.

Such a strategy also shares many similarities with the so-called conditional generative models in deep learning, see (Goodfellow et al. 2014; Kingma and Welling 2022; Dinh et al. 2017; Ho et al. 2020). Instead of using neural networks to learn how to simulate in the prior distribution, we use here a generic class of advection-diffusion SPDE which we see as a first step towards physically-sounded generatives models in learning-based methods. Fig. 6 displays an example of such non-conditional simulations for the term $\mathbf{x} - \mathbf{x}^b$, again for Application 2 on realistic SSH datasets over some energetic area along the Gulf Stream, see Section 4, i.e. the anomaly between the true state sequence and the deterministic mean $\mathbf{x}^b$, retrieved from a preliminary 4DVarNet coarse resolution scheme. We can appreciate how both the generic class of SPDE selected here, together with the training of its parameter, leads to realistic anomalies with a clear increase of the variance along the main meander of the Gulf Stream, due to a correct distribution of SPDE parameter $\tau^\star(\mathbf{s}, t)$. For a detailed analysis of the parameter estimation, please report to Section 4.

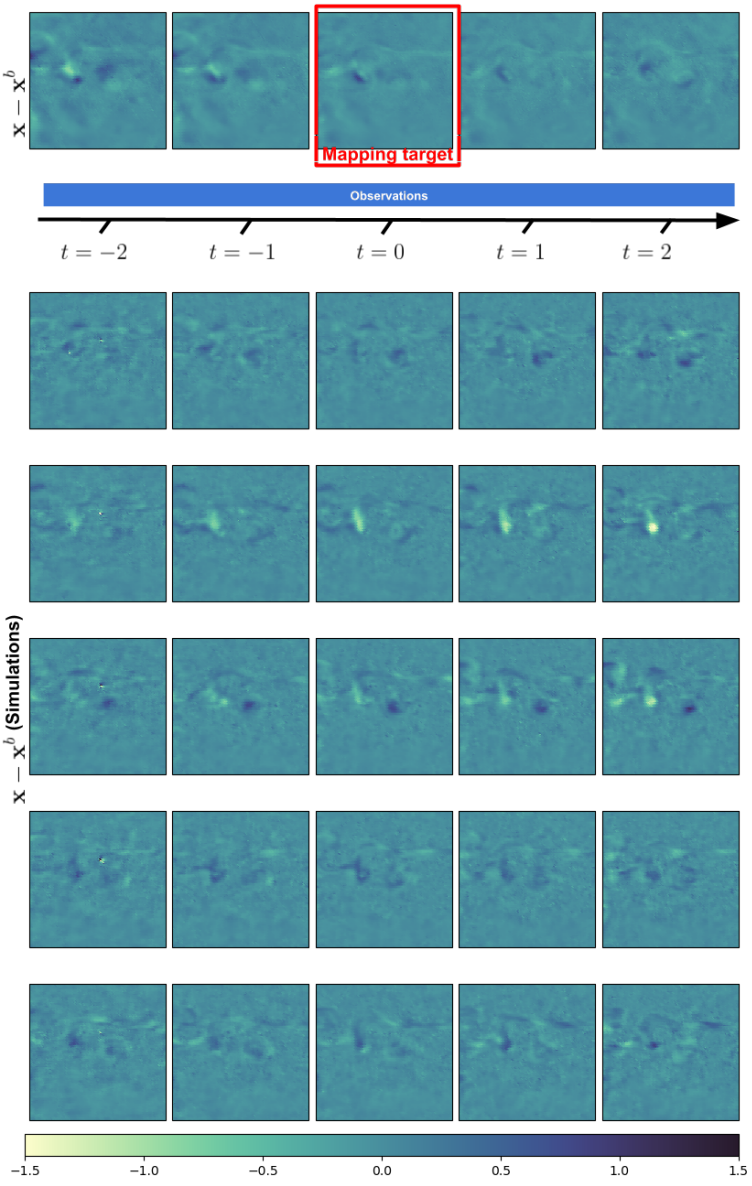


FIG. 6: Top panel: an example of the Ground Truth anomaly between $\mathbf{x} - \mathbf{x}^b$; Bottom panel: five simulations of the same anomaly based on the SPDE parametrization $\boldsymbol{\theta}$ learned after training

4. Results

In this Section, we provide two applications of this work:

- The first example relies on a spatio-temporal GP simulation driven by a non-stationary spatial diffusion tensor. Because the dynamical process is linear, the best reconstruction is provided

by the optimal interpolation using the SPDE parameters used in the simulation to fill in the precision matrix.

- The second example uses an Observation System Simulation Experiment (OSSE) of the Sea Surface Height (SSH) along the Gulf Stream. We will use the SPDE-based prior as a surrogate model along the data assimilation window to provide ensemble members of the posterior distribution.

a. Diffusion-based non stationary GP

Dataset. In this first application, we simulate 500 states of a GP driven by the following diffusion SPDE:

$$\left\{\kappa^2 - \nabla \cdot \mathbf{H}(\mathbf{s})\nabla\right\}^{\alpha/2} \mathbf{x}(\mathbf{s},t) = \tau \mathbf{z}(\mathbf{s},t) \tag{22}$$

The regularity parameter $\kappa = 0.33$ is fixed over space and time. To ensure the GP to be smooth enough, we use a value of $\alpha = 4$. Such a formulation enables to generate GPs driven by local anisotropies in space leading to non stationary spatio-temporal fields with eddy patterns. The diffusion tensor $\mathbf{H}$ is a 2-dimensional diffusion tensor generated by drawing from the spatial statistics literature, see e.g. (Fuglstad et al. 2015a). We introduce a generic decomposition of $\mathbf{H}(\mathbf{s},t)$ through the equation:

$$\mathbf{H} = \gamma \mathbf{I}_2 + \beta \mathbf{v}(\mathbf{s})^{\mathrm{T}} \mathbf{v}(\mathbf{s})$$

with $\gamma = 1$, $\beta = 25$ and $\mathbf{v}(\mathbf{s}) = (v_1(\mathbf{s}), v_2(\mathbf{s}))^{\mathrm{T}}$ using a periodic formulation of its two vector fields components, see Section b. We use the Finite Difference Method in space coupled with an implicit Euler scheme in time to solve the equation. Let $\mathcal{D} = [0,100] \times [0,100]$ be the square spatial domain of simulation and $\mathcal{T} = [0,500]$ the temporal domain. Both spatial and temporal domains are discretized so that the simulation is made on a uniform Cartesian grid consisting of points (x_i, y_j, t_k) where x_i=$i\Delta x$, y_j=$j\Delta j$, t_k=$k\Delta t$ with Δx, Δy and Δt all set to 1.

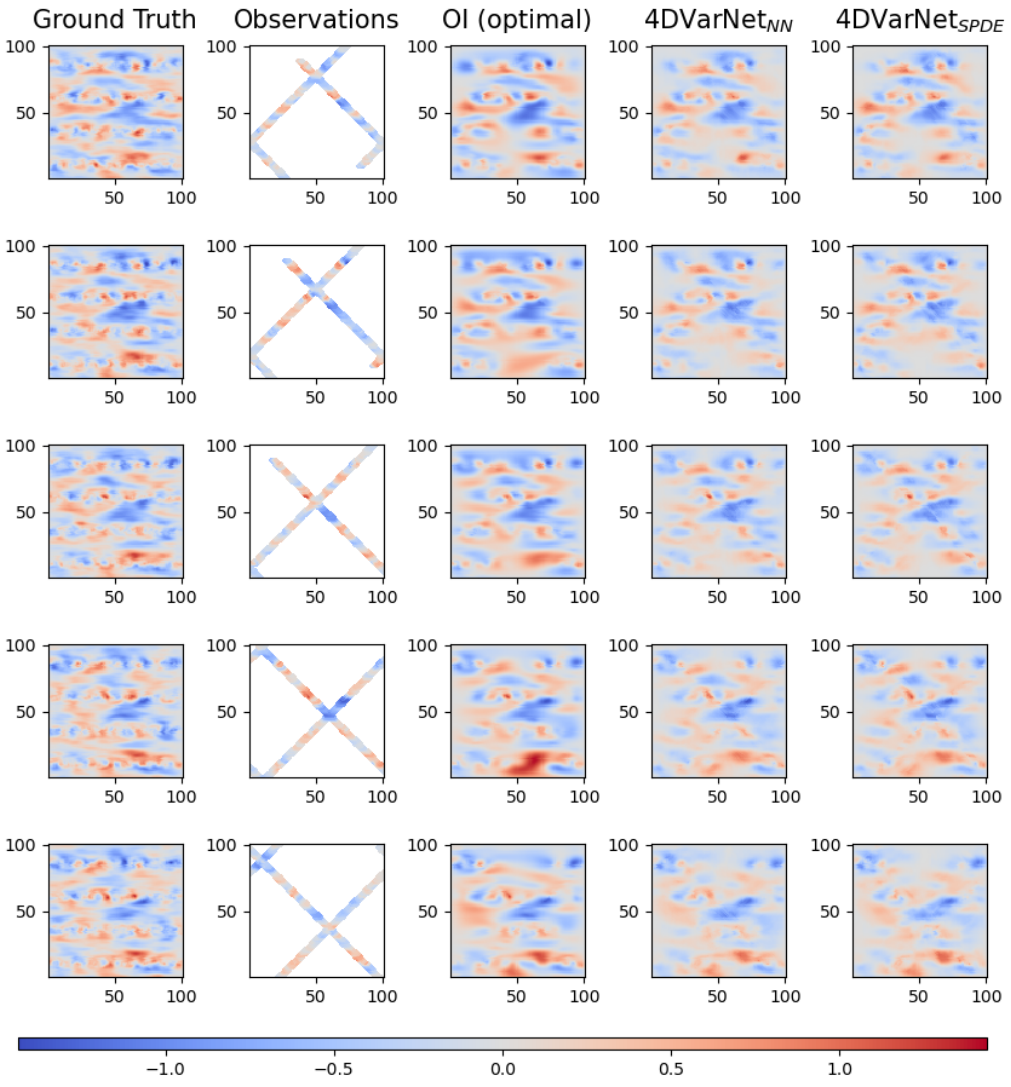


FIG. 7: From left to right: Ground Truth, Pseudo-observations, Optimal Interpolation, neural variational scheme with UNet-based and SPDE-based prior operator Φ. A data assimilation window of length 5 is used

Training setting. We train the neural architectures for both UNet-based and SPDE-based priors with Adam optimizer on 50 epochs. The training period goes from timestep 100 to 400. During the training procedure, we select the best model according to metrics computed over the validation period from timestep 30 to 80. Overall, the set of metrics is computed on a test period going from timestep 400 to 500. No further improvements in the training losses are seen when training the

model longer. We use a data assimilation window of length 5 and generate pseudo-observations from the ground truth, inspired by orbiting satellites tracks around the earth, see Application 2 in Section b.

Results. Fig. 7 displays the results obtained by Optimal Interpolation and the neural implicit solver with both UNet and SPDE-based parametrization of the prior. Neural-based reconstructions are optimized at the center of the assimilation window ($T = 2$), which is why the performance may be affected for other leadtimes. There is no significant differences between the two prior formulations used in the neural scheme, which overall retrieve the main patterns of the OI. Some artefacts may appear due to the observation term in the inner variational cost that tends to lead the solution towards the observation in its close neighbourhood. Improvements may be expected when adding regularization terms in the training loss to counteract such effects, see e.g. Beauchamp et al. (2023b). Regarding the derived framework proposed in this work, one of the question was: is it possible to retrieve interpretable SPDE parametrizations from the joint learning setting? Two configurations were considered: when using as initial condition for the parametrization a gradient-based information of the accumulated alongtrack observations, i.e. $\mathbf{H}_{11} = \nabla_{\overrightarrow{x}}\mathbf{y}$ and $\mathbf{H}_{22} = \nabla_{\overrightarrow{y}}\mathbf{y}$; or an isotropic initialization, i.e. $\mathbf{H} = \mathbf{I}$. The first configuration enables to identify patterns in zonal and meridional components of the true diffusion tensor and leads to an optimal parametrization $\boldsymbol{\theta}^\star$ very close to the true diffusion tensor. While leading to different SPDE parametrizations $\boldsymbol{\theta}^\star$, the interpolation metrics are similar in the end for the two initializations. In addition, using an isotropic initial condition is more general (see the next realistic SSH application) and also retrieves in the end the main zonal flow directions encoded by $\mathbf{H}_{22}$, while the meriodional and periodic structures of $\mathbf{H}_{11}$ and $\mathbf{H}_{12}$ are partly seen as well.

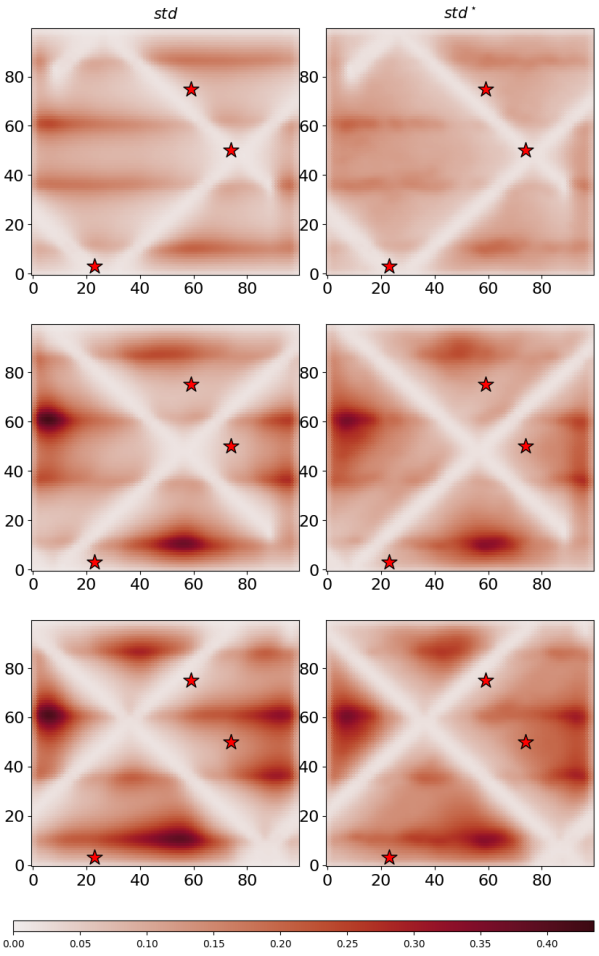


FIG. 8: True (left) and estimated (right) posterior standard deviations at the beginning, center and end of the assimilation window $time = 0, 2, 4$

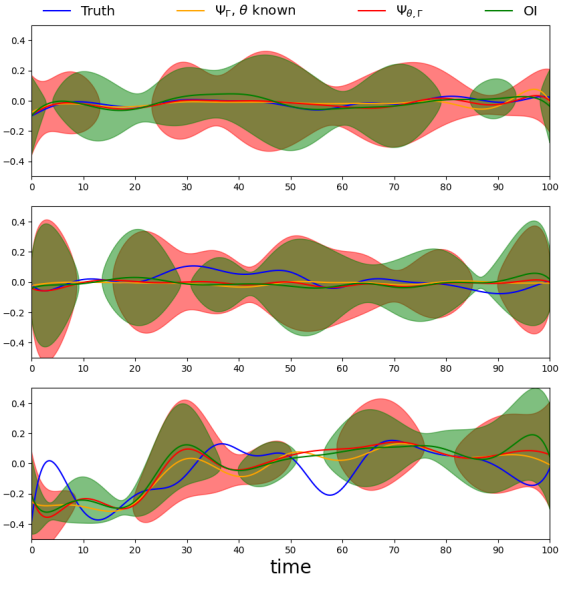


FIG. 9: Ground truth, OI and its posterior variance (blue), neural scheme with SPDE parametrization $\boldsymbol{\theta}$ known (orange) and with inference of $\boldsymbol{\theta}$ (red) for the same three points identified in Fig. 8 along the 100 time step test period

Because we know the process is Gaussian, we know that $\mathbf{Q}(\mathbf{x}|\mathbf{y}) = \mathbf{Q}^b + \mathbf{H}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{H}$, see Section 2, to compute the closed form of the posterior pdf. Looking at the estimated posterior variance obtained starting from isotropic initial condition of the parameters, see Fig. 8, we understand that even if the initial set of parameters is not retrieved, they still remain interpretable. Visually, a simulation produced by this set of SPDE parameters is also consistent and propose a spatio-temporal diffusion process close to the original one. It also highlights the dependency of the OI uncertainty quantification to the sampling scheme, especially here where no noise is added to the partial set of observations. On Fig. 9 is also shown the time series on three different locations (red stars) $\in \mathcal{D}$, see again Fig. 8, of the GT (blue line) along the test period (100 time steps), the OI and corresponding standard deviation (green line), and the neural scheme estimations when the SPDE parametrization is known (orange line) or estimated (red line with associated uncertainty). Both neural scheme configurations are generally close which validates the capability of this framework to estimate jointly both state and prior parametrization. Because the neural scheme is optimized on the global MSE, its solution may deviate more or less significantly from the OI depending of the position we are looking at in domain $\mathcal{D}$.

Last, Fig. 10 provides the scatterplot of the global MSE w.r.t the OI variational cost:

$$\mathcal{J}_{OI}(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}^\star) = ||\mathbf{y} - \mathbf{H}\mathbf{x}||^2 + \mathbf{x}^{\mathrm{T}}\mathbf{Q}^b_{\boldsymbol{\theta}^\star}\mathbf{x}$$

throughout the iteration process after training of the neural schemes. For SPDE-based prior, the initial parametrization $\boldsymbol{\theta}^{(0)}$ relates to an isotropic GP process. LSTM-based iterative solvers are all consistent with the optimal solution in terms of MSE. When the latter is used as training loss, 20 iterations is enough to reach satisfactory performance. Using the same loss for inner variational cost and outer training loss (not shown here), see Beauchamp et al. (2022), would require more iterations to converge. Also, constraining the prior to follow the same SPDE simulation ensures to also jointly minimize the OI variational cost (asymptotic convergence of the red line to the yellow star on Fig. 10) which is not the case when looking for an optimal solution within the bi-level neural optimization of prior and solver (blue line) that may lead to deviate from the original variational cost to minimize. Let note that by construction, the analytical OI solution is optimal regarding the OI variational cost: it is unbiased with minimal variance. In other words, at a given

spatio-temporal location $(\mathbf{s},t)$, its variance (which is the local MSE) is minimal. In our case, we compute the global MSE over the entire domain $\mathcal{D}\times[t_k-2,t_k+2]$ w.r.t the true state because we have only one single realization to compute this metrics. This is why the global MSE (see $\mathcal{L}_1$) of the OI may be outperformed by learning-based methods.

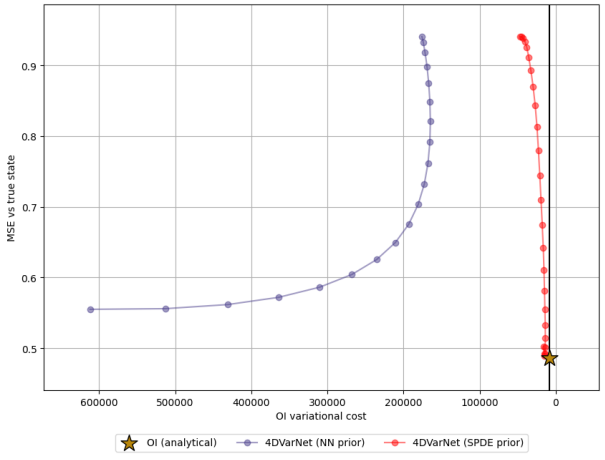


FIG. 10: Optimal Interpolation derived variational cost vs Mean Squared Error (MSE) loss (a) for the gradient-based descent of the variational cost, the classical implementation of the neural scheme and its SPDE-based prior formulation. For the analytical Optimal Interpolation solution (the yellow star), there is no iterations, then a single point is displayed.

b. Realistic SSH datasets

Dataset. In this application on Sea Surface Height (SSH) spatio-temporal fields, we focus on a small part of the GULFSTREAM, see Fig. 11, mainly driven by energetic mesoscale dynamics, to illustrate how our framework may help to solve for the oversmoothing of the state-of-the-art Optimal interpolation (OI) and how the SPDE formulation of the prior is a consistent linearization of the dynamics in the data assimilation window that helps to generate ensemble members in the posterior distribution. We use an Observation System Simulation Experiment (OSSE) with the NEMO (Nucleus for European Modeling of the Ocean) model NATL60 high resolution basin-scale configuration Molines (2018). Based on this one-year long simulation, we generate pseudo along-track nadir data for the current capabilities of the observation system (Ballarotta et al. 2019) and pseudo wide-swath SWOT data in the context of the upcoming SWOT mission (Metref et al. 2020), with additional observation errors (Dufau et al. 2016; Esteban-Fernandez 2014; Gaultier and Ubelmann 2010). The two types of observations may be merged, see Fig. 11, to produce a

one-year long daily datasets of partial and noisy observations of the idealized Ground Truth (GT). Last the DUACS operational system (CMEMS/C3S Copernicus program) provides the Optimal Interpolation baseline (Taburet et al. 2019) as daily gridded (0.25°x0.25°) products

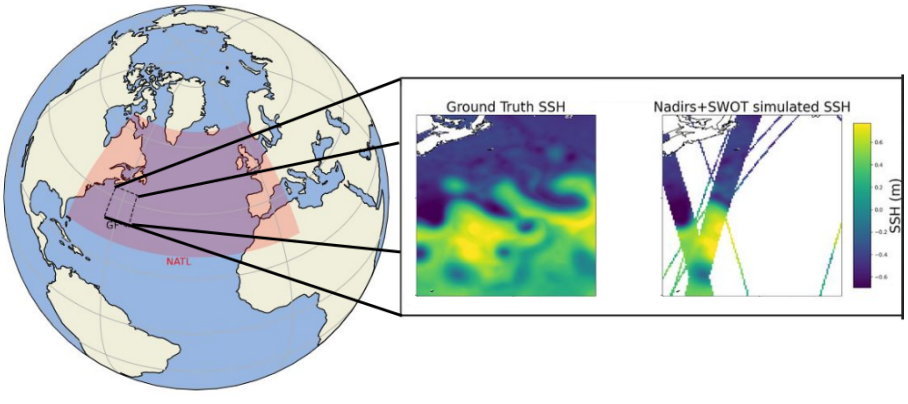


FIG. 11: NATL60 and GULFSTREAM domain and zoom-in picture for one day Ground Truth and accumulated along-track nadir + wide-swath SWOT SSH pseudo-observations

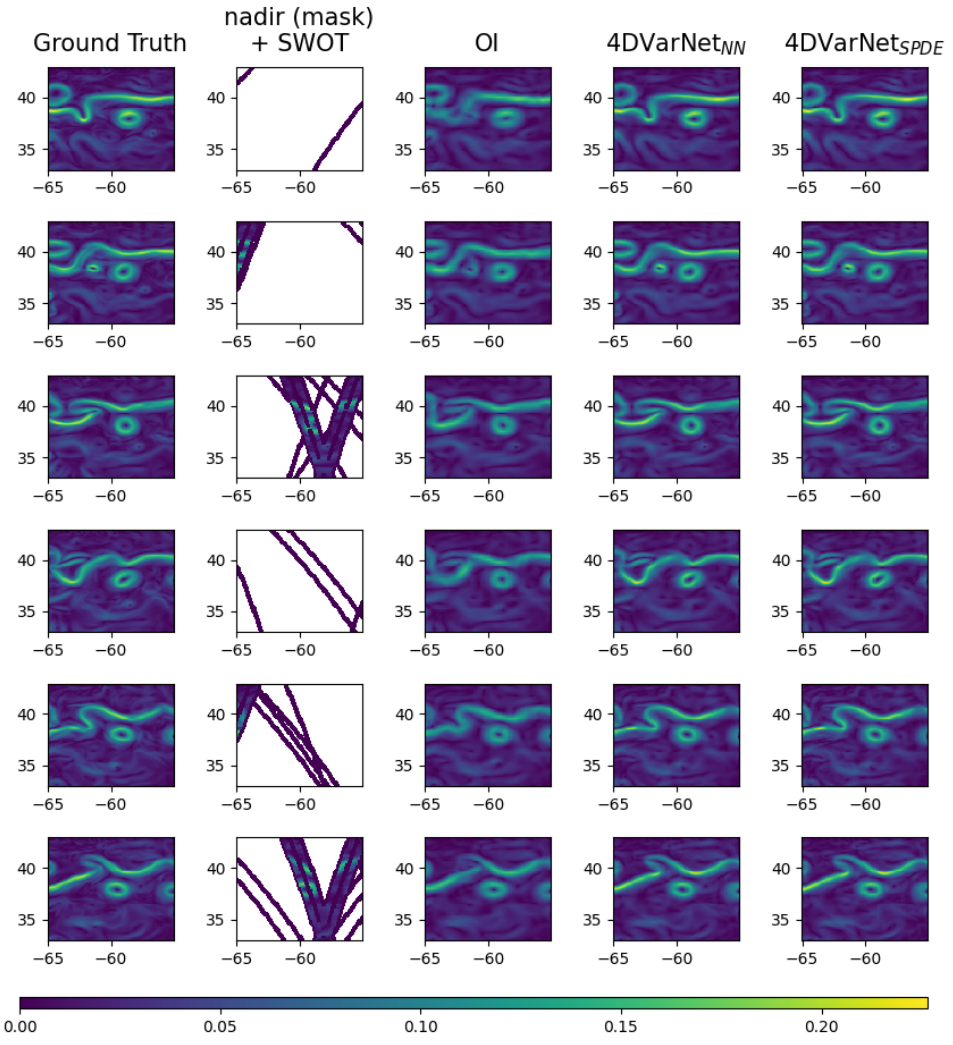


FIG. 12: From left to right: SSH Gradient Ground Truth, Pseudo-observations (nadir mask and SWOT Gradient field), DUACS Optimal Interpolation, and neural variational scheme results obtained with UNet-based and SPDE-based prior operator Φ. A data assimilation window of length 5 is used

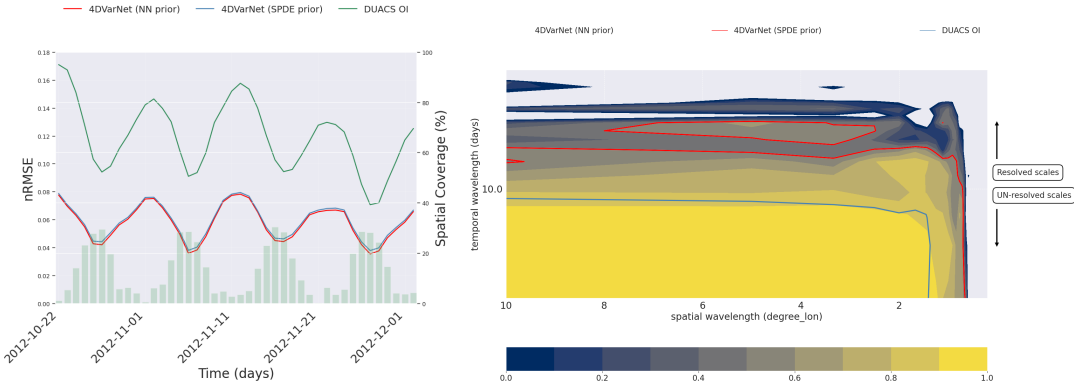
Training setting. All the datasets are downscaled from the original resolution of 1/60° to 1/10°. For the training, the dataset spans from mid-February 2013 to October 2013, while the validation period refers to January 2013. All methods are tested on the test period from October 22, 2012 to December 2, 2012. We still use Adam optimizer with 1100 epochs. Regarding the metrics, we use the ocean data challenge [1] strategy looking at RMSE-score, and both spatial and temporel

[1] https://github.com/ocean-data-challenges/2020a_SSH_mapping_NATL60

minimal scales resolved. The reader may refer to (Beauchamp et al. 2023b) for more details.

Results. Because we aim at assessing how the use of SPDE priors in the neural architecture is relevant, we used a Gaussian prior formulation $\mathbf{x} \sim \mathcal{N}\left(\mathbf{x}^b, \mathbf{Q}_{\boldsymbol{\theta}}^b\right)$ where the mean $\mathbf{x}^b$ is provided as a first guess by a preliminary 4DVarNet run. Doing so, the SPDE parametrization $\boldsymbol{\theta}$ is focused on the reconstruction and surrogate parametrization of the small scales not catched by this deterministic mean. Then, we can reduce the data assimilation window used to estimate the prior mean $\mathbf{x}^b$, to a reasonable length $L = 5$ here, so that the storage of multiple (sparse) precision matrices throughout the computational graph remains possible, see Section 3.d. Moving to longer time windows including mesoscale-related autocorrelations (more than 10 days) in this SPDE framework would lead to similar results, see e.g. Febvre et al. (2022) but would require moving to matrix-free formulations, with potential existing solutions, see e.g. Pereira et al. (2022).

Last point on this experimental configuration: because the pseudo-observations are subsampled from hourly simulations but we target daily reconstructions, they are noisy due to representativity errors between the two temporal resolutions. This is not currently adressed by our framework where the observation term in the minimization cost, see Eq. 8, is only the L2-norm of the innovations. But it might be easily considered, either by using a known observation error covariance matrix $\mathbf{R}$ or by learning one of its possible parametrization as an additional feature of the neural scheme.



(a) nRMSE (b) Spectrum

FIG. 13: For DUACS Optimal Interpolation, neural solvers with UNet-based and SPDE-based prior, (a) provides their temporal performance, i.e. nRMSE time series along the BOOST-SWOT DC evaluation period ; and (b) displays their spectral performance, i.e. the PSD-based score is used to evaluate the spatio-temporal scales resolved in the GULFSTREAM domain (yellow area)

As in Example 7.1, we provide in Fig. 12 the reconstructions, as the SSH gradients, obtained from DUACS OI baseline, and both neural solver formulations, the one using as prior a UNet-based parametrization, see e.g. (Beauchamp et al. 2023c) and our SPDE-based formulation. As expected and already seen in previous related studies (Beauchamp et al. 2023b,a; Fablet et al. 2021), the neural schemes improve the baseline by retrieving the dynamics along the main meander of the Gulf Stream and additional small energetic eddies. Again, there is no significant differences between the two neural formulations, which was again expected because the aim of the SPDE formulation is not to improve the mean state estimation obtained when using a neural prior operator, which is even more general, but to provide a a stochastic framework for interpretability and uncertainty quantification. This is supported by Figs. 13a and 13b respectively showing the normalized RMSE and the space-time spectrum along the test period. The periodic improvements of the score are due to the SWOT sampling that does not provide informations every day on this Gulf Stream domain. Overall, the nRMSE is in average improved by 60% when using the neural architecture. For the spectrum, minimal spatio-temporal scales λ_x and λ_t also improve resp. by 30% and 60%. From both figures and scores provided in Table 1, we can see that the UNet formulation of prior Φ leads to a small improvement in the reconstruction, which was expected because this is the only task optimized by the pure neural formulation. Introducing the SPDE formulation leads to optimize both reconstruction and likelihood of the parameters, which is more difficult. Though, the reconstruction performed by the latter is satisfactory enough and very close to the original solution proposed in Fablet et al. (2021). It also competes with other state-of-the-art method available in the ocean data challenges 2020a, among which DUACS OI (Taburet et al. 2019), MIOST (Multi-scale OI) Ardhuin et al. (2020) or a 4DVar scheme based on a QG dynamical model Le Guillou (2022). While DUACS OI has minimal spatial and temporal resolution of 1.22° and 11.15 days, 4DVarNet reaches 0.62° and 4.35 days, which reaches similar order of performance that combining a 4DVar with a QG model.

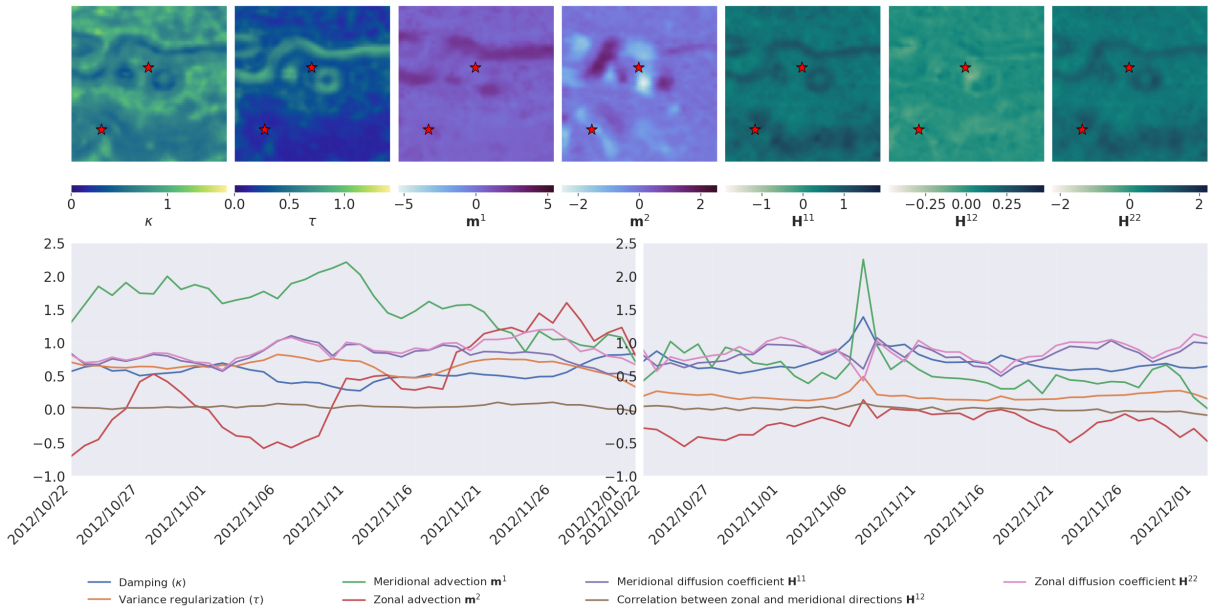


FIG. 14: Parameter estimation of the SPDE prior: in the neural scheme along the 42 days test period and every 6 days. Top, from left to right: $\boldsymbol{\tau}$, $\mathbf{m}^1$, $\mathbf{m}^2$ (advection fields), $\mathbf{H}^{1,1}$, $\mathbf{H}^{1,2}$ and $\mathbf{H}^{2,2}$ (diffusion tensor) estimated on the first day of the test period. Bottom: time series of these parameters along the test period on the two locations identified by the red stars. The first one (left) is located along the Gulf Stream meander and the second one (right) is in the less energetic left-lower part of the domain.

In this very general setup, where the equation-based dataset provide a supervised learning setting on state $\mathbf{x}$, but not on the SPDE parameters $\boldsymbol{\theta}$, two main questions are raised. On the first one: does the parameters retrieved by the iterative solver are interpretable? If considering Fig. 14 that shows parameters $\boldsymbol{\tau}$, $\mathbf{m}_1$, $\mathbf{m}_2$ (advection fields), $\mathbf{H}_{11}$, $\mathbf{H}_{12}$ and $\mathbf{H}_{22}$ (diffusion tensor) on the first day of the test period, they seem consistent with the SSH field $\mathbf{x}$ that partially encodes the SPDE parametrization, which also opens avenue for state-dependent parameters. We also show the time series along the 42 days of the test period for these parameters on two locations of the Gulf Stream domain (red stars): a first one right in the Gulf Stream meander (left time series) and a second one in the left-lower part of the domain, with less variability (right time series). Interestingly, in less energetic areas the parameters are almost perfectly correlated while in the Gulf Stream, they might behave differently. Playing with both damping and variance regularization parameters provide a flexible way to handle complex GP priors with both low and high marginal variances for a given time. This is a key aspect here because the range of possible values attributed to the anomaly generally differs according to the spatio-temporal dynamics of the SSH: it is high along

the main meander of the Gulf Stream and eddies not catched up by the OI, and lower elsewhere. This is also supported by generating non-conditional simulations based on these parameters and comparing it to the ground truth anomaly field, see again Fig. 6. The spatio-temporal fields are clearly consistent with the original simulation which also makes the link between our approach and generative modeling. The average of a large number of simulations would be a zero state, while its covariance relates to the model error matrix used in equation-based DA to perturb the members in ensemble methods.

	μ(RMSE)	σ(RMSE)	λ_x (degree)	λ_t (days)
OI	0.92	0.02	1.22	11.06
MIOST	0.94	0.01	1.18	10.33
4DVar-QG	0.96	0.01	0.66	4.65
4DVarNet (NN prior)	0.96	0.01	0.62	4.35
4DVarNet (SPDE prior)	0.96	0.01	0.64	5.03

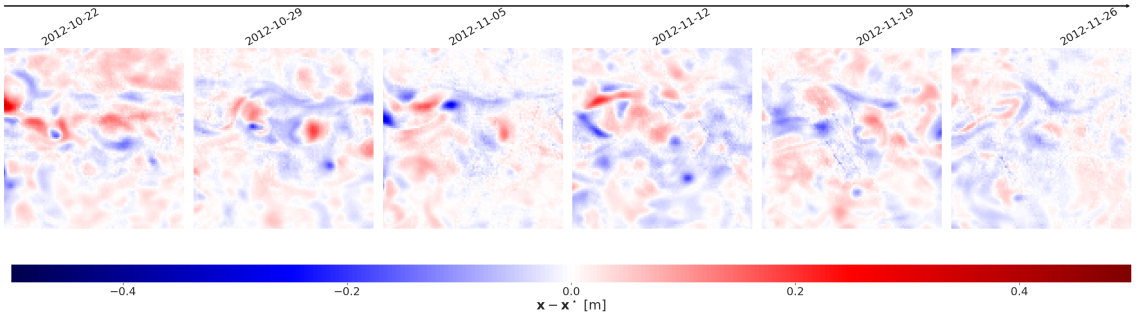
TABLE 1: Evaluation of the performance between OI, MIOST, 4DVar-QG, 4DVarNet schemes with UNet and SPDE-based prior parametrization. The OSSE involves 1 SWOT + 4 nadirs as pseudo-observations.

The second question is: does this approach, SPDE-based generation of members, followed by their conditioning with observations, is efficient to estimate the posterior pdf $p_{\mathbf{x}|\mathbf{y}}$? Fig. 15 shows in a) the reconstruction error $\mathbf{x}-\mathbf{x}^\star$ for six days along the test period and in b) the empirical posterior standard deviations computed from 200 members. In c), we also provide pointwise Continuous Ranked Probability Score (CRPS) maps to assess the accurary of the ensemble-based predictions. Given the observations $\mathbf{y}_{ijk}$, we compute the empirical CDF of the stochastic process X at a given spatio-temporal location $\{i,j,k\}$ as $F_{ijk}(z) = \mathbf{P}\left[X_{ijk} \leq z\right]$:

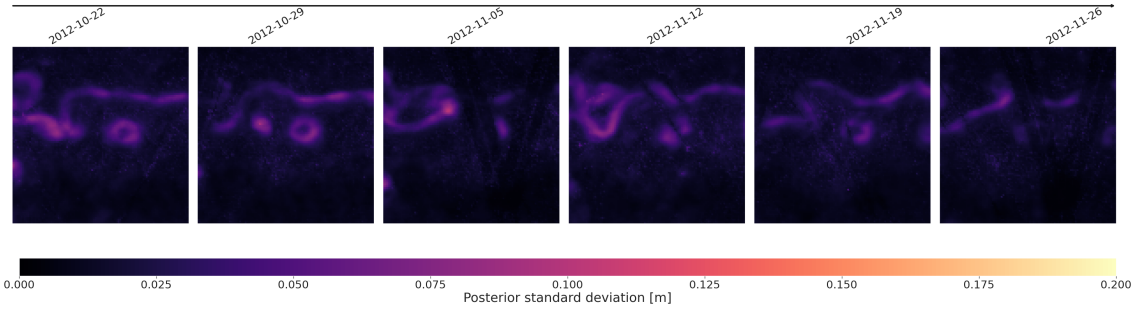
$$CRPS(F_{ijk}, \mathbf{y}_{ijk}) = \int_{-\infty}^{+\infty} \left(F(z) - \mathbf{1}_{z-\mathbf{y}_{ijk}}\right)^2 dz$$

Looking at the estimated standard deviation produced by the ensemble of neural variational reconstructions, they are not as dependent of the observations as an OI scheme would be, which validates the flow-dependency discussed in Section e. We can still see the observation mask as blurry areas

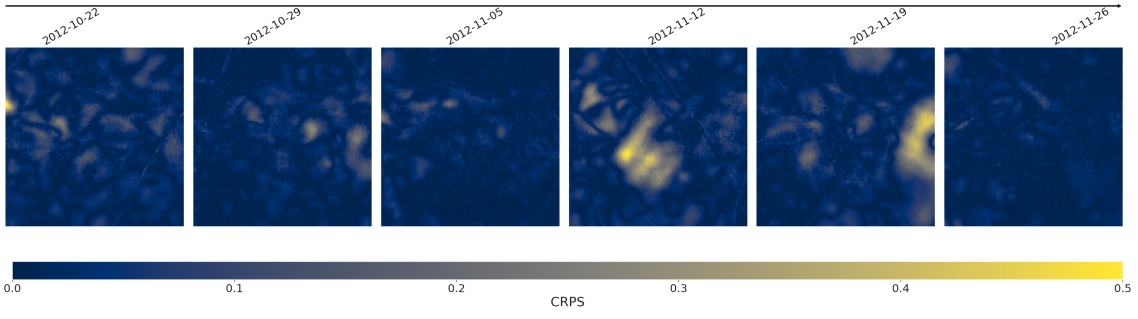
but the standard deviations are rather increasing and continuous along the main meander of the Gulf Stream. The CRPS is often close to zero, which indicates the ensemble of recontruction is wholly accurate with a realistic posterior standard deviation. The highest CRPS values observed are about 0.4 which remains reasonable, and can be explained by high reconstruction errors outside the main Gulf Stream meander that are not correctly handled by the ensemble.



(a) $\mathbf{x} - \mathbf{x}^{\star}$



(b) posterior standard deviation



(c) continuous ranked probability score (CRPS) maps

FIG. 15: For six days along the OSSE test period: a) corresponding reconstruction error at the center of the assimilation window, b) ensemble-based posterior standard deviations and c) Continuous Ranked Probability Score

5. Conclusion

We explore a new neural architecture to tackle the reconstruction inverse problem of a dynamical process from partial and potentially noisy observations. We provide a joint end-to-end learning scheme of both stochastic prior models and solvers. The idea is to optimize in the same time the state and the stochastic parametrization of the prior so that we minimize the mean squared error between the reconstruction and the true states, while we are also able to provide uncertainty quantification of the mean state reconstruction, either analytically in the gaussian case, or ensemble-derived in the more general configuration.

In our work, we draw from recent advances in geostatistical simulations of SPDE-based Gaussian Processes to provide a flexible trainable prior embedded in a neural architecture backboned on variational data assimilation. The SPDE parameters are added as latent variables in an augmented state the trainable solver has to reconstruct. A bi-level optimization scheme is used to optimize in the same time:

- the inner variational cost derived from OI-based formulations, which depends on state $\mathbf{x}$, observations $\mathbf{y}$ and SPDE parameters $\boldsymbol{\theta}$ and,
- the outer training loss function of the neural architecture, which drives the optimization of the LSTM-based residual solver parameters $\boldsymbol{\omega}$ leading to the reconstruction of the augmented state.

The first application of the framework on a diffusion-based GP showed that it reaches the same performance, in terms of MSE w.r.t the ground truth, observed when using a neural-based prior in the classic implementation fo the neural scheme (Fablet et al. 2020), which asymptotically converges towards the optimal solution. In addition, the posterior variance ot the mean state derived from the SPDE parameters is close to the true variance of the Optimal Interpolation, which demonstrates the potentiality of the proposed scheme to handle uncertainty quantifications. Indeed, if we showed that retrieving the original set of SPDE parameters might be difficult, the local minimum found after the training process leads to a parametrization with high likelihood, similar diffusion-based spatial patterns and spatio-temporal covariances structures. Though not always physically explainable, the SPDE prior formulation still helps to interprete the dynamical

process in terms of statistical properties.

In a more general setup, we also present an application on Sea Surface Height dynamics based on Observation System Simulation Experiment (OSSE), for which the ground truth is given by a state-of-the-art ocean model and pseudo-observations are generated by a realistic satellite subsampling of the ground truth. In this case, the process is not linear and the gaussian framework does not apply. Then, the idea is to use the GP linear SPDE formulation as a surrogate model to linearize the prior dynamics along the data assimilation window. It provides an efficient way for fast sampling of a huge set of members in the prior distribution within a few minutes. Based on the neural solver, able to handle non-linear and non-gaussian dynamics based on its supervised training, the conditioning of these simulations leads to the estimation of the posterior distribution. The preliminary conclusions made on the GP experiment holds for the reconstruction of the mean state, i.e. no significant differences observed when using a trainable neural prior. The key aspect of the framework is here revealed by the SPDE parametrization, which is fully non-stationary in space and time, and allows for online estimation after training for any new set of input observations, in contrast with other approaches in most of the spatial statistics literature which requires offline new parameter inference. We showed that the prior parametrization is both statistically consistent with the original ground truth used in the training and physically sounded with similar patterns observed in the simulations. Comparing the posterior pdf retrieved from the ensemble members, higher variances are observed in areas of high reconstruction errors which is a good indicator of the framework capability to quantify realistic uncertainties. Additional applications would be necessary on real datasets to complement these preliminary conclusions.

Regarding the potential extensions of this methodology, it is important to understand that such an SPDE parametrization of the prior term is a way of going to generative modelling, as it is called in the machine learning community. In our case, the SPDE is linear and already provide an efficient way to produce fast, large and realistic ensembles. Promising avenues would be to draw from the existing link between diffusion models and SDE to enrich our framework, provide stochastic non-linear priors and see if and how it helps to improve the results obtained in this work. Also, we made the choice to model the prior with a linear SPDE before conditioning it with our neural solver. A direct use of the SPDE formulation for the posterior would have been possible,

but restrictive to GP reconstructions which is in most cases rather limited. Back to the non-linear neural diffusion operators, an other way of addressing the problem would be to directly optimize the sampling in the posterior pdf.

Acknowledgments. This work has been supported by the LEFE program (LEFE MANU project IA-OAC), CNES (grant OSTST DUACS-HR), and ANR Projects Melody and OceaniX. This research has been supported by GENCI- IDRIS (grant no. 2020-101030).

Data availability statement. The open-source 4DVarNet with SPDE priors version of the code is available at https://github.com/CIA-Oceanix/4dvarnet-starter/tree/maximebeauchamp. The datasets are shared through the ocean data challenge 2020a also available on GitHub `https://github.com/ocean-data-challenges/2020a\_SSH\_mapping\_NATL60,lastaccess:2022`

APPENDIX A

Finite difference schemes for SPDE

In this work, we propose to use the Euler implicit scheme as a discretization method for the stochastic PDEs. We need to define the discretized version of several differential operators on the 2D regular grid, namely:

a. Discretization of the spatial differential operators

$$
\begin{aligned}
\Delta \mathbf{x}_{i,j} &= \frac{\partial^2 \mathbf{x}}{\partial x^2}_{i,j} + \frac{\partial^2 \mathbf{x}}{\partial y^2}_{i,j} \\
&= \frac{\partial}{\partial x}\left(\frac{\partial \mathbf{x}}{\partial x}\right)_{i,j} + \frac{\partial}{\partial y}\left(\frac{\partial \mathbf{x}}{\partial y}\right)_{i,j} \\
&= \frac{\partial}{\partial x}\left(\frac{\mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j}}{2dx}\right) + \frac{\partial}{\partial y}\left(\frac{\mathbf{x}_{i,j+1} - \mathbf{x}_{i,j-1}}{2dy}\right) \\
&= \left(\frac{\mathbf{x}_{i+1,j}}{dx^2} - \frac{\mathbf{x}_{i,j}}{dx^2}\right) - \left(\frac{\mathbf{x}_{i,j}}{dx^2} - \frac{\mathbf{x}_{i-1,j}}{dx^2}\right) \\
&+ \left(\frac{\mathbf{x}_{i,j+1}}{dy^2} - \frac{\mathbf{x}_{i,j}}{dy^2}\right) - \left(\frac{\mathbf{x}_{i,j}}{dy^2} - \frac{\mathbf{x}_{i,j-1}}{dy^2}\right) \\
&= \frac{\mathbf{x}_{i+1,j} - 2\mathbf{x}_{i,j} + \mathbf{x}_{i-1,j}}{dx^2} + \frac{\mathbf{x}_{i,j+1} - 2\mathbf{x}_{i,j} + \mathbf{x}_{i,j-1}}{dy^2}
\end{aligned}
\tag{A1a}
$$

$$
\begin{aligned}
\nabla\mathbf{H}\cdot\nabla\mathbf{x}_{i,j} &= \left(\mathbf{H}^{1,1}\frac{\partial}{\partial x^2}+\mathbf{H}^{1,2}\frac{\partial}{\partial x\partial y}+\mathbf{H}^{2,1}\frac{\partial}{\partial x\partial y}+\mathbf{H}^{2,2}\frac{\partial}{\partial y^2}\right)\mathbf{x}_{i,j} \\
&= \mathbf{H}^{1,1}\frac{\mathbf{x}_{i+1,j}-2\mathbf{x}_{i,j}+\mathbf{x}_{i-1,j}}{dx^2} \\
&+ \mathbf{H}^{2,2}\frac{\mathbf{x}_{i,j+1}-2\mathbf{x}_{i,j}+\mathbf{x}_{i,j-1}}{dy^2} \\
&+ \mathbf{H}^{1,2}\frac{\mathbf{x}_{i+1,j+1}-\mathbf{x}_{i+1,j-1}-\mathbf{x}_{i-1,j+1}+\mathbf{x}_{i-1,j-1}}{4dxdy} \\
&+ \mathbf{H}^{2,1}\frac{\mathbf{x}_{i+1,j+1}-\mathbf{x}_{i-1,j+1}-\mathbf{x}_{i+1,j-1}+\mathbf{x}_{i-1,j-1}}{4dxdy} \\
&= \mathbf{H}^{1,1}\frac{\mathbf{x}_{i+1,j}-2\mathbf{x}_{i,j}+\mathbf{x}_{i-1,j}}{dx^2} \\
&+ \mathbf{H}^{2,2}\frac{\mathbf{x}_{i,j+1}-2\mathbf{x}_{i,j}+\mathbf{x}_{i,j-1}}{dy^2} \\
&+ \mathbf{H}^{1,2}\frac{\mathbf{x}_{i+1,j+1}-\mathbf{x}_{i+1,j-1}-\mathbf{x}_{i-1,j+1}+\mathbf{x}_{i-1,j-1}}{2dxdy}
\end{aligned}
\tag{A1b}
$$

$$
\begin{aligned}
\mathbf{m}\cdot\nabla\mathbf{x}_{i,j} &= \mathbf{m^1}\frac{\partial}{\partial x}\mathbf{x}_{i,j}+\mathbf{m^2}\frac{\partial}{\partial y}\mathbf{x}_{i,j} \\
&= \mathbf{m^1}\frac{\mathbf{x}_{i+1,j}-\mathbf{x}_{i-1,j}}{2dx}+\mathbf{m^2}\frac{\mathbf{x}_{i,j+1}-\mathbf{x}_{i,j-1}}{2dy}
\end{aligned}
\tag{A1c}
$$

b. Upwind schemes for advection-dominated SPDE

With such an advection-diffusion framework detailed in Section b, it is known that the solution to the space-centered scheme does not oscillate only when the Peclet number is lower than 2 and the Courant–Friedrichs–Lewy condition (CFL) condition $Cr = dt\left(\mathbf{m}^1/dx+\mathbf{m}^2/dy\right) \leq 1$ is satisfied (Lewy et al. 1928; Price et al. 1966). For unsatisfied Peclet conditions, damped oscillations occur with nonreal eigenvalues (Finlayson 1992; Price et al. 1966), while in the limiting case of pure advection $\mathbf{H} \to \mathbf{0}$, such a scheme would be unconditionally unstable (Finlayson 1992; Strikwerda 1989). Because the velocity fiels is allowed to vary in space and time, the CFL number is different at each discrete space-time location (i, j, t). A necessary condition for convergence is that the CFL condition be satisfied at each point location, the velocity and diffusion parameters being unknown,

we have to choose the timestep dt small enough so that the maximum CFL number (observed in space at each time step) satisfies the CFL condition. Again, because the velocity field is trained, it may happen that the CFL condition is not satisfied if the latter is more and more dominant during the training. A simple way of counteracting this problem is to use an activation function on the two velocity components by clipping their maximum value.

One way of addressing this problem of stabilities in FDM when the advection term is predominant over the diffusion relates to the class of upwind schemes (UFDM). It is used to numerically simulate more properly the direction of propagation of the state in a flow field. The first order upwind FDM uses a one-sided finite difference in the upstream direction to approximate the advection term in the transport SPDE. The spatial accuracy of the first-order upwind scheme can be improved by choosing a more accurate finite difference stencil for the approximation of spatial derivative. Let note that UFDM scheme for SPDE eliminates the nonphysical oscillations in the space-centered scheme and generate stable solutions even for very complicated flows.

1) First-order upwind scheme (UFDM1)

Instead of using centered differences:

$$\left(\frac{\partial \mathbf{x}}{\partial x}\right)_{i,j} = \frac{\mathbf{x}_{i+1,j} - \mathbf{x}_{i-1,j}}{2dx} \tag{A2a}$$

$$\left(\frac{\partial \mathbf{x}}{\partial y}\right)_{i,j} = \frac{\mathbf{x}_{i,j+1} - \mathbf{x}_{i,j-1}}{2dy} \tag{A2b}$$

We use the one-sided upwind differences :

$$\begin{cases} \left(\frac{\partial \mathbf{x}}{\partial x}\right)_{i,j} = \frac{\mathbf{x}_{i,j} - \mathbf{x}_{i-1,j}}{dx} \text{ if } \mathbf{m}^1_{i,j} > 0 \\ \left(\frac{\partial \mathbf{x}}{\partial x}\right)_{i,j} = \frac{\mathbf{x}_{i+1,j} - \mathbf{x}_{i,j}}{dx} \text{ if } \mathbf{m}^1_{i,j} < 0 \end{cases} \tag{A3a}$$

and

$$\begin{cases} \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j} = \dfrac{\mathbf{x}_{i,j} - \mathbf{x}_{i,j-1}}{dy} \text{ if } \mathbf{m}^2_{i,j} < 0 \\ \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j} = \dfrac{\mathbf{x}_{i,j+1} - \mathbf{x}_{i,j}}{dy} \text{ if } \mathbf{m}^2_{i,j} < 0 \end{cases} \tag{A3b}$$

2) Third-order upwind scheme (UFDM3)

It can be shown, see e.g. (E. Ewing and Wang 2001) that the UFDM scheme is actually a second-order approximation of the SPDE with a modified diffusion term. Along this line, it comes with the family of methods that may introduce excessive numerical diffusion in the solution with large gradients. Thus, we use a third order upwind scheme for the approximation of spatial derivatives with four points instead of two, with only a reduced increase in the degree of sparsity of the precision matrix. This scheme is less diffusive compared to the second-order accurate scheme. It comes with four points instead of two for the approximation, with only a reduced increase in the degree of sparsity of the discretized differential operator.

It can be expressed as follows:

$$\begin{cases} \left(\dfrac{\partial \mathbf{x}}{\partial x}\right)_{i,j} = \dfrac{2\mathbf{x}_{i+1,j} + 3\mathbf{x}_{i,j} - 6\mathbf{x}_{i-1,j} + \mathbf{x}_{i-2,j}}{6dx} \text{ if } \mathbf{m}^1_{i,j} > 0 \\ \left(\dfrac{\partial \mathbf{x}}{\partial x}\right)_{i,j} = \dfrac{-\mathbf{x}_{i+2,j} + 6\mathbf{x}_{i+1,j} - 3\mathbf{x}_{i,j} - 2\mathbf{x}_{i-1,j}}{6dx} \text{ if } \mathbf{m}^1_{i,j} < 0 \end{cases} \tag{A4a}$$

and

$$\begin{cases} \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j} = \dfrac{2\mathbf{x}_{i,j+1} + 3\mathbf{x}_{i,j} - 6\mathbf{x}_{i,j-1} + \mathbf{x}_{i,j-2}}{6dy} \text{ if } \mathbf{m}^2_{i,j} > 0 \\ \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j} = \dfrac{-\mathbf{x}_{i,j+2} + 6\mathbf{x}_{i,j+1} - 3\mathbf{x}_{i,j} - 2\mathbf{x}_{i,j-1}}{6dy} \text{ if } \mathbf{m}^2_{i,j} < 0 \end{cases} \tag{A4b}$$

c. Discretization of the spatio-temporal SPDE

Based on the discretization of the spatial operators using upwind finite difference schemes for the advection term and centered difference schemes for the diffusion term, we involve an implicit

Euler scheme to solve the advection-diffusion SPDE:

$$\frac{\partial \mathbf{x}}{\partial t} + \left\{\boldsymbol{\kappa}^2(\mathbf{s},t) + \mathbf{m}(\mathbf{s},t)\cdot\nabla - \nabla\cdot\mathbf{H}(\mathbf{s},t)\nabla\right\}^{\alpha/2}\mathbf{x}(\mathbf{s},t) = \boldsymbol{\tau}(\mathbf{s},t)\mathbf{z}(\mathbf{s},t)$$

Because the upwind advection schemes is not symmetric and does not involve the same neighbours in the difference scheme approximation according to the predominant flow direction, we introduce the following generic notations.

$$\begin{cases} \mathbf{m}_{i,j}^{1,t,-} = \left(\dfrac{\partial \mathbf{x}}{\partial x}\right)_{i,j}^{t} & \text{if } \mathbf{m}_{i,j}^{1,t} > 0 \\ \mathbf{m}_{i,j}^{1,t,+} = \left(\dfrac{\partial \mathbf{x}}{\partial x}\right)_{i,j}^{t} & \text{if } \mathbf{m}_{i,j}^{1,t} < 0 \end{cases} \quad \text{and} \quad \begin{cases} \mathbf{m}_{i,j}^{2,t,-} = \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j}^{t} & \text{if } \mathbf{m}_{i,j}^{2,t} > 0 \\ \mathbf{m}_{i,j}^{2,t,+} = \left(\dfrac{\partial \mathbf{x}}{\partial y}\right)_{i,j}^{t} & \text{if } \mathbf{m}_{i,j}^{2,t} < 0 \end{cases}$$

and by denoting $\mathbf{a}_{i,j}^{1,t,+} = \max(\mathbf{m}_{i,j}^{1,t}, 0)$, $\mathbf{a}_{i,j}^{1,t,-} = \min(\mathbf{m}_{i,j}^{1,t}, 0)$, $\mathbf{a}_{i,j}^{2,t,+} = \max(\mathbf{m}_{i,j}^{2,t}, 0)$, $\mathbf{a}_{i,j}^{2,t,-} = \min(\mathbf{m}_{i,j}^{2,t}, 0)$, the resulting UFDM, whatever the order of the scheme, can be written in its compact form as:

$$\begin{aligned} \mathbf{x}_{i,j}^{t+1} = \mathbf{x}_{i,j}^{t} + dt\Bigg[& \kappa_{i,j}^{t}\mathbf{x}_{i,j}^{t} + \left(\mathbf{a}_{i,j}^{1,t,+}\mathbf{m}_{i,j}^{1,t,-} + \mathbf{a}_{i,j}^{1,t,-}\mathbf{m}_{i,j}^{1,t,+}\right) + \left(\mathbf{a}_{i,j}^{2,t,+}\mathbf{m}_{i,j}^{2,t,-} + \mathbf{a}_{i,j}^{2,t,-}\mathbf{m}_{i,j}^{2,t,+}\right) \\ & + \mathbf{H}_{i,j}^{1,1,t}\frac{\mathbf{x}_{i+1,j}^{t} - 2\mathbf{x}_{i,j}^{t} + \mathbf{x}_{i-1,j}^{t}}{dx^2} + \mathbf{H}_{i,j}^{2,2,t}\frac{\mathbf{x}_{i,j+1}^{t} - 2\mathbf{x}_{i,j}^{t} + \mathbf{x}_{i,j-1}^{t}}{dy^2} \\ & + \mathbf{H}_{i,j}^{1,2,t}\frac{\mathbf{x}_{i+1,j+1}^{t} - \mathbf{x}_{i+1,j-1}^{t} - \mathbf{x}_{i-1,j+1}^{t} + \mathbf{x}_{i-1,j-1}^{t}}{2dxdy} + \tau_{i,j}^{t}\mathbf{z}_{i,j}\Bigg] \end{aligned}$$

Using again notation $i = \lfloor k/N_x \rfloor$ and $j = k \bmod N_x$, operator $\mathbf{A}_t$ associated to UFDM1 finally writes:

$$\mathbf{A}_{k,l}(t) = \begin{cases} \mathbf{H}_{i,j}^{1,2,t}/2dxdy \text{ if } l = k \pm (N_x+1) \\ \mathbf{H}_{i,j}^{1,2,t}/2dxdy \text{ if } l = k \pm (N_x-1) \\ -\mathbf{H}_{i,j}^{1,1,t}/dx^2 + \left(\mathbf{a}_{i,j}^{1,t,+} + \mathbf{a}_{i,j}^{1,t,-}\right)/dx \text{ if } l = k-1 \\ -\mathbf{H}_{i,j}^{1,1,t}/dx^2 + \left(\mathbf{a}_{i,j}^{1,t,+} + \mathbf{a}_{i,j}^{1,t,-}\right)/dx \text{ if } l = k+1 \\ -\mathbf{H}_{i,j}^{2,2,t}/dy^2 + \left(\mathbf{a}_{i,j}^{2,t,+} + \mathbf{a}_{i,j}^{1,t,-}\right)/dy \text{ if } l = k-N_x \\ -\mathbf{H}_{i,j}^{2,2,t}/dy^2 + \left(\mathbf{a}_{i,j}^{2,t,+} + \mathbf{a}_{i,j}^{1,t,-}\right)/dy \text{ if } l = k+N_x \\ \left(\kappa_{i,j}^t\right)^2 + 2(\mathbf{H}_{i,j}^{1,1,t}/dx^2 + \mathbf{H}_{i,j}^{2,2,t}/dy^2) \\ + \left(\mathbf{a}_{i,j}^{1,t,+} + \mathbf{a}_{i,j}^{1,t,-}\right)/dx + \left(\mathbf{a}_{i,j}^{2,t,+} + \mathbf{a}_{i,j}^{2,t,-}\right)/dy \text{ if } k = l \\ 0 \text{ otherwise} \end{cases} \tag{A5}$$

The same operator associated to UFDM3 writes:

$$\mathbf{A}_{k,l}(t) = \begin{cases} \mathbf{H}_{i,j}^{1,2,t}/2dxdy & \text{if } l = k \pm (N_x+1) \\ \mathbf{H}_{i,j}^{1,2,t}/2dxdy & \text{if } l = k \pm (N_x-1) \\ \mathbf{a}_{i,j}^{1,t,+}/6dx & \text{if } l = k-2 \\ -\mathbf{H}_{i,j}^{1,1,t}/dx^2 - \left(6\mathbf{a}_{i,j}^{1,t,+} + 2\mathbf{a}_{i,j}^{1,t,-}\right)/6dx & \text{if } l = k-1 \\ -\mathbf{H}_{i,j}^{1,1,t}/dx^2 + \left(2\mathbf{a}_{i,j}^{1,t,+} + 6\mathbf{a}_{i,j}^{1,t,-}\right)/6dx & \text{if } l = k+1 \\ -\mathbf{a}_{i,j}^{1,t,-}/6dx & \text{if } l = k+2 \\ \mathbf{a}_{i,j}^{2,t,+}/6dy & \text{if } l = k-2Nx \\ -\mathbf{H}_{i,j}^{2,2,t}/dy^2 - \left(6\mathbf{a}_{i,j}^{2,t,+} + 2\mathbf{a}_{i,j}^{2,t,-}\right)/6dy & \text{if } l = k-N_x \\ -\mathbf{H}_{i,j}^{2,2,t}/dy^2 + \left(2\mathbf{a}_{i,j}^{2,t,+} + 6\mathbf{a}_{i,j}^{2,t,-}\right)/6dy & \text{if } l = k+N_x \\ -\mathbf{a}_{i,j}^{2,t,-}/6dy & \text{if } l = k+2Nx \\ \left(\kappa_{i,j}^t\right)^2 + 2(\mathbf{H}_{i,j}^{1,1,t}/dx^2 + \mathbf{H}_{i,j}^{2,2,t}/dy^2) & \\ +3\left(\mathbf{a}_{i,j}^{1,t,+} - \mathbf{a}_{i,j}^{1,t,-}\right)/6dx & \\ +3\left(\mathbf{a}_{i,j}^{2,t,+} - \mathbf{a}_{i,j}^{2,t,-}\right)/6dy & \text{if } k = l \end{cases} \tag{A6}$$

and 0 otherwise.

APPENDIX B

SPDE-based precision matrix

Let define the spatio-temporal SPDE prior $\mathbf{x} = \{\mathbf{x}_0, \cdots, \mathbf{x}_{Ldt}\}$. From now on, $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{P_0})$ denotes the initial state and $\mathbf{Q}_0 = \mathbf{P_0}^{-1}$ is always taken as the precision matrix obtained after a given stabilization run, i.e. the evolution of the dynamical system over N timesteps using as stationary parameters the initial parametrization $\boldsymbol{\theta}_0$ of the SPDE at time $t = 0$. We can rewrite :

$$\{\mathbf{x}_0, \cdots, \mathbf{x}_{Ldt}\} = \mathbf{M}_G \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{z} \end{bmatrix}$$

with $\mathbf{z} = [\mathbf{z}_1, \ldots, \mathbf{z}_t]^{\mathrm{T}}$ and

$$\mathbf{M}_G = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_1 & \mathbf{T}_1 & 0 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_2\mathbf{M}_1 & \mathbf{M}_2\mathbf{T}_1 & \mathbf{T}_2 & 0 & 0 & \ldots & 0 \\ \mathbf{M}_3\mathbf{M}_2\mathbf{M}_1 & \mathbf{M}_3\mathbf{M}_2\mathbf{T}_1 & \mathbf{M}_3\mathbf{T}_2 & \mathbf{T}_3 & 0 & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \mathbf{T}_L \end{bmatrix}$$

Despite its apparent complexity, $\mathbf{M}_G$ has a particular structure which allows to easily compute its inverse:

$$\mathbf{M}_G^{-1} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 & \ldots & 0 \\ -\mathbf{T}_1^{-1}\mathbf{M}_1 & \mathbf{T}_1^{-1} & 0 & 0 & 0 & \ldots & 0 \\ 0 & -\mathbf{T}_2^{-1}\mathbf{M}_2 & \mathbf{T}_2^{-1} & 0 & 0 & \ldots & 0 \\ 0 & 0 & -\mathbf{T}_3^{-1}\mathbf{M}_3 & \mathbf{T}_3^{-1} & 0 & \ldots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & -\mathbf{T}_L^{-1}\mathbf{M}_L & \mathbf{T}_L^{-1} \end{bmatrix}$$

All this information is embedded in $\mathbf{Q}^b$ (Eq. B2), which is the inverse of the prior covariance matrix $\mathbf{B}$:

$$\mathbf{Q}^b = \mathbf{B}^{-1} = \begin{bmatrix} \mathbf{P}_0 & \mathbf{P}_{0,1} & \dots & \dots & \mathbf{P}_{0,L} \\ \mathbf{P}_{1,0} & \mathbf{P}_1 & \dots & \dots & \mathbf{P}_{1,L} \\ \vdots & \mathbf{P}_{2,1} & \mathbf{P}_2 & \dots & \mathbf{P}_{2,L} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \mathbf{P}_{L-1,0} & \dots & \dots & \mathbf{P}_{L-1} & \mathbf{P}_{L-1,L} \\ \mathbf{P}_{L,0} & \mathbf{P}_{L,1} & \dots & \mathbf{P}_{L,L-1} & \mathbf{P}_L \end{bmatrix}^{-1} = \mathbf{M}_G^{-1\mathrm{T}} \begin{bmatrix} \mathbf{P}_0^{-1} & 0 & \dots & 0 \\ 0 & \mathbf{I} & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{I} \end{bmatrix} \mathbf{M}_G^{-1} \tag{B1}$$

By denoting $\mathbf{S}_k = \mathbf{T}_k \mathbf{T}_k^{\mathrm{T}}$, we have

$$\mathbf{Q}^b = \begin{bmatrix} \mathbf{P}_0^{-1}+\mathbf{M}_1^{\mathrm{T}}\mathbf{S}_1^{-1}\mathbf{M}_1 & -\mathbf{M}_1^{\mathrm{T}}\mathbf{S}_1^{-1} & 0 & 0 & 0 & \dots & 0 \\ -\mathbf{S}_1^{-1}\mathbf{M}_1 & \mathbf{S}_1^{-1}+\mathbf{M}_2^{\mathrm{T}}\mathbf{S}_2^{-1}\mathbf{M}_2 & -\mathbf{M}_2^{\mathrm{T}}\mathbf{S}_2^{-1} & 0 & 0 & \dots & 0 \\ 0 & -\mathbf{S}_2^{-1}\mathbf{M}_2 & \mathbf{S}_2^{-1}+\mathbf{M}_3^{\mathrm{T}}\mathbf{S}_3^{-1}\mathbf{M}_3 & -\mathbf{M}_3^{\mathrm{T}}\mathbf{S}_3^{-1} & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\mathbf{S}_{L-1}^{-1}\mathbf{M}_{L-1} & \mathbf{S}_{L-1}^{-1}+\mathbf{M}_L^{\mathrm{T}}\mathbf{S}_L^{-1}\mathbf{M}_L & -\mathbf{M}_L^{\mathrm{T}}\mathbf{S}_L^{-1} \\ 0 & \ddots & \ddots & \ddots & 0 & -\mathbf{S}_L^{-1}\mathbf{M}_L & \mathbf{S}_L^{-1} \end{bmatrix} \tag{B2}$$

Because of the formulation of $\mathbf{M}_t$ and $\mathbf{T}_t$, the precision matrix $\mathbf{Q}^b$ with the FDM scheme also writes:

$$\mathbf{Q}^b = \frac{1}{dt} \begin{bmatrix} \mathbf{P}_0^{-1}+\widetilde{\mathbf{Q}}_{s,1} & -\widetilde{\mathbf{Q}}_{s,1}\mathbf{M}_1^{-1} & 0 & 0 & 0 & \dots & 0 \\ -\left(\mathbf{M}_1^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,1} & \mathbf{M}_1^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,1}\mathbf{M}_1+\widetilde{\mathbf{Q}}_{s,2} & -\widetilde{\mathbf{Q}}_{s,2}\mathbf{M}_2^{-1} & 0 & 0 & \dots & 0 \\ 0 & -\left(\mathbf{M}_2^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,2} & \mathbf{M}_2^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,2}\mathbf{M}_2+\widetilde{\mathbf{Q}}_{s,3} & -\widetilde{\mathbf{Q}}_{s,3}\mathbf{M}_3^{-1} & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -\left(\mathbf{M}_{L-1}^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,L-1} & \mathbf{M}_L^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,L-1}\mathbf{M}_{L-1}+\widetilde{\mathbf{Q}}_{s,L} & -\widetilde{\mathbf{Q}}_{s,L}\mathbf{M}_L^{-1} \\ 0 & \ddots & \ddots & \ddots & 0 & -\left(\mathbf{M}_L^{\mathrm{T}}\right)^{-1}\widetilde{\mathbf{Q}}_{s,L} & \mathbf{M}_L^{\mathrm{T}}\widetilde{\mathbf{Q}}_{s,L}\mathbf{M}_L \end{bmatrix} \tag{B3}$$

where $\widetilde{\mathbf{Q}}_{s,t}$ is the precision matrix of the colored noise weighted by the non-uniform regularization variance τ_t.

APPENDIX C

PyTorch implementation of sparse linear algebra

Currently, two pieces of codes are missing in the PyTorch sparse linear algebra to achieve a fully sparse implementation of our algorithm: the automatic differentiation tools for

- solving sparse linear systems
- running sparse Cholesky decomposition

First, regarding the implementation of the backward pass for solving linear systems, we start by writing the forward pass of this system of equation:

$$x = \mathbf{A}^{-1}\mathbf{b}$$

where $\mathbf{A}$ denotes a $2D$ square matrix and $\mathbf{b}$ a one-dimensional vector.
We need to provide the gradients wrt both $\mathbf{A}$ and $\mathbf{b}$:

$$\begin{aligned}\frac{\partial L}{\partial \mathbf{b}} &= \frac{\partial L}{\partial x_i}\frac{\partial x_i}{\partial \mathbf{b}_j} = \frac{\partial L}{\partial x_i}\frac{\partial}{\partial \mathbf{b}_k}(\mathbf{A}^{-1}_{ij}\mathbf{b}_j) = \frac{\partial L}{\partial x_i}\mathbf{A}^{-1}_{ij}\frac{\partial \mathbf{b}_j}{\partial \mathbf{b}_k} \\ &= \frac{\partial L}{\partial x_i}\mathbf{A}^{-1}_{ij}\delta_{jk} = \frac{\partial L}{\partial x_i}\mathbf{A}^{-1}_{ik} = \left(\mathbf{A}^{-1}\right)^T\frac{\partial L}{\partial x} \\ &= \text{solve}\left(\mathbf{A}^{\mathrm{T}},\ \frac{\partial L}{\partial x}\right)\end{aligned}$$

and

$$
\begin{aligned}
\frac{\partial L}{\partial \mathbf{A}} &= \frac{\partial L}{\partial x_i}\frac{\partial x_i}{\partial \mathbf{A}_{mn}} = \frac{\partial L}{\partial x_i}\frac{\partial}{\partial \mathbf{A}_{mn}}(\mathbf{A}_{ij}^{-1}\mathbf{b}_j) \\
&= -\frac{\partial L}{\partial x_i}\mathbf{A}_{ij}^{-1}\frac{\partial \mathbf{A}_{jk}}{\partial \mathbf{A}_{mn}}\mathbf{A}_{kl}^{-1}\mathbf{b}_l \\
&= -\frac{\partial L}{\partial x_i}\mathbf{A}_{ij}^{-1}\delta_{jm}\delta_{kn}\mathbf{A}_{kl}^{-1}\mathbf{b}_l \\
&= -\frac{\partial L}{\partial x_i}\mathbf{A}_{im}^{-1}\mathbf{A}_{nl}^{-1}\mathbf{b}_l \\
&= -\left(\left(\mathbf{A}^{-1}\right)^T\frac{\partial L}{\partial x}\right)\otimes\left(\mathbf{A}^{-1}\mathbf{b}\right) \\
&= -\frac{\partial L}{\partial \mathbf{b}}\otimes x
\end{aligned}
$$

where we used the einstein summation convention during the derivations, as well as the following identity:

$$
\frac{\partial(\mathbf{A}^{-1})}{\partial p} = -\mathbf{A}^{-1}\frac{\partial \mathbf{A}}{\partial p}\mathbf{A}^{-1}.
$$

These two expressions $\frac{\partial L}{\partial \mathbf{b}}$ and $\frac{\partial L}{\partial \mathbf{A}}$ are easily implemented in PyTorch based on sparse representations.

Second, the backward pass for a sparse Cholesky decomposition may be found for instance in Seeger et al. (2019): given a symmetric, positive definite matrix $\mathbf{A}$, its Cholesky factor $\mathbf{L}$ is lower triangular with positive diagonal, such that the forward pass of the Cholesky decomposition is defined as $\mathbf{A} = \mathbf{L}\mathbf{L}^{\mathrm{T}}$. Given the output gradient $\overline{\mathbf{L}}$ and the Cholesky factor $\mathbf{L}$, the backward pass compute the input gradient $\overline{\mathbf{A}}$ defined as :

$$
\overline{\mathbf{A}} = \frac{1}{2}\mathbf{L}^{-\mathrm{T}}ltu(\mathbf{L}^{\mathrm{T}}\overline{\mathbf{L}})\mathbf{L}^{-1} \tag{C1}
$$

where $ltu(\mathbf{X})$ generates a symmetric matrix by copying the lower triangle to the upper triangle. In our work, such an operation is useful when optimizing the likelihood in the outer training cost function of the neural scheme, see Eq. 20, which involves the determinant of sparse matrices Cholesky decomposition.

References

M. Andrychowicz, M. Denil, S. Gomez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. De Freitas. Learning to learn by gradient descent by gradient descent. In *Advances in neural information processing systems*, pages 3981–3989, 2016.

F. Ardhuin, C. Ubelmann, G. Dibarboure, L. Gaultier, A. Ponte, M. Ballarotta, and Y. Faugère. Reconstructing ocean surface current combining altimetry and future spaceborne doppler data. *Earth and Space Science Open Archive*, page 22, 2020. doi: 10.1002/essoar.10505014.1. URL https://doi.org/10.1002/essoar.10505014.1.

L. Arras, J. Arjona-Medina, M. Widrich, G. Montavon, M. Gillhofer, K.-R. Müller, S. Hochreiter, and W. Samek. Explaining and interpreting lstms. *Lecture Notes in Computer Science*, pages 211–238, 2019. ISSN 1611-3349. doi: 10.1007/978-3-030-28954-6_11. URL http://dx.doi.org/10.1007/978-3-030-28954-6_11.

M. Asch, M. Bocquet, and M. Nodet. *Data Assimilation*. Fundamentals of Algorithms. Society for Industrial and Applied Mathematics, Dec. 2016. ISBN 978-1-61197-453-9. doi: 10.1137/1.9781611974546. URL https://doi.org/10.1137/1.9781611974546.

M. Ballarotta, C. Ubelmann, M.-I. Pujol, G. Taburet, F. Fournier, J.-F. Legeais, Y. Faugère, A. Delepoulle, D. Chelton, G. Dibarboure, and N. Picot. On the resolutions of ocean altimetry maps. *Ocean Science*, 15(4):1091–1109, 2019. doi: 10.5194/os-15-1091-2019. URL https://www.ocean-sci.net/15/1091/2019/.

M. Beauchamp, J. Thompson, H. Georgenthum, Q. Febvre, and R. Fablet. Learning neural optimal interpolation models and solvers, 2022. URL https://arxiv.org/abs/2211.07209.

M. Beauchamp, Q. Febvre, and R. Fablet. Ensemble-based 4dvarnet uncertacinty quantification for the reconstruction of sea surface height dynamics. *Environmental Data Science*, 2:e18, 2023a. doi: 10.1017/eds.2023.19.

M. Beauchamp, Q. Febvre, H. Georgenthum, and R. Fablet. 4dvarnet-ssh: end-to-end learning of variational interpolation schemes for nadir and wide-swath satellite altimetry. *Geoscientific Model Development*, 16(8):2119–2147, 2023b. doi: 10.5194/gmd-16-2119-2023. URL https://gmd.copernicus.org/articles/16/2119/2023/.

M. Beauchamp, Q. Febvre, J. Thompson, H. Georgenthum, and R. Fablet. Learning neural optimal interpolation models and solvers. In J. Mikyška, C. de Mulatier, M. Paszynski, V. V. Krzhizhanovskaya, J. J. Dongarra, and P. M. Sloot, editors, *Computational Science – ICCS 2023*, pages 367–381, Cham, 2023c. Springer Nature Switzerland. ISBN 978-3-031-36027-5.

D. Bolin and J. Wallin. Spatially adaptive covariance tapering. *Spatial Statistics*, 18:163–178, 2016. ISSN 2211-6753. doi: https://doi.org/10.1016/j.spasta.2016.03.003. URL https://www.sciencedirect.com/science/article/pii/S2211675316000245. Spatial Statistics Avignon: Emerging Patterns.

P. Boudier, A. Fillion, S. Gratton, S. Gürol, and S. Zhang. Data assimilation networks. *Journal of Advances in Modeling Earth Systems*, 15(4):e2022MS003353, 2023. doi: https://doi.org/10.1029/2022MS003353. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003353. e2022MS003353 2022MS003353.

J. Brajard, A. Carrassi, M. Bocquet, and L. Bertino. Combining data assimilation and machine learning to infer unresolved scale parametrization, 2021. URL https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2020.0086.

A. Carrassi, M. Bocquet, L. Bertino, and G. Evensen. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change*, 9(5):e535, 2018. doi: https://doi.org/10.1002/wcc.535. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/wcc.535.

Y. Chen, T. Davis, W. Hager, and S. Rajamanickam. Algorithm 887: Cholmod, supernodal sparse cholesky factorization and update/downdate. *ACM Trans. Math. Softw.*, 35, 01 2008.

S. Cheng and M. Qiu. Observation error covariance specification in dynamical systems for data assimilation using recurrent neural networks. *Neural Computing and Applications*, 34(16): 13149–13167, Aug 2022. ISSN 1433-3058. doi: 10.1007/s00521-021-06739-4. URL https://doi.org/10.1007/s00521-021-06739-4.

S. Cheng, C. Quilodran-Casas, S. Ouala, A. Farchi, C. Liu, P. Tandeo, R. Fablet, D. Lucor, B. Iooss, J. Brajard, D. Xiao, T. Janjic, W. Ding, Y. Guo, A. Carrassi, M. Bocquet, and R. Arcucci. Machine

learning with data assimilation and uncertainty quantification for dynamical systems: a review, 2023.

J. Chilès and P. Delfiner. *Geostatistics : modeling spatial uncertainty*. Wiley, New-York, second edition, 2012.

J.-P. Chilès and N. Desassis. *Fifty Years of Kriging*, pages 589–612. Springer International Publishing, Cham, 2018. ISBN 978-3-319-78999-6. doi: 10.1007/978-3-319-78999-6_29. URL https://doi.org/10.1007/978-3-319-78999-6_29.

L. Clarotto, D. Allard, T. Romary, and N. Desassis. The spde approach for spatio-temporal datasets with advection and diffusion, 2022. URL https://arxiv.org/abs/2208.14015.

F. Counillon and L. Bertino. Ensemble optimal interpolation: multivariate properties in the gulf of mexico. *Tellus A*, 61(2):296–308, 2009. doi: https://doi.org/10.1111/j.1600-0870.2008.00383.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1600-0870.2008.00383.x.

N. Cressie and C. Wikle. *Statistics for Spatio-Temporal Data*. John Wiley & Sons, 2015. ISBN 978-1-119-24304-5.

L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real nvp, 2017.

A. Doury, S. Somot, S. Gadat, A. Ribes, and L. Corre. Regional climate model emulator based on deep learning: concept and first evaluation of a novel hybrid downscaling approach. *Climate Dynamics*, 60(5):1751–1779, Mar 2023. ISSN 1432-0894. doi: 10.1007/s00382-022-06343-9. URL https://doi.org/10.1007/s00382-022-06343-9.

C. Dufau, M. Orsztynowicz, G. Dibarboure, R. Morrow, and P.-Y. Le Traon. Mesoscale resolution capability of altimetry: Present and future. *Journal of Geophysical Research: Oceans*, 121(7): 4910–4927, 2016. doi: 10.1002/2015JC010904. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1002/2015JC010904.

R. E. Ewing and H. Wang. A summary of numerical methods for time-dependent advection-dominated partial differential equations. *Journal of Computational and Applied Mathematics*, 128(1):423–445, 2001. ISSN 0377-0427. doi: https://doi.org/10.1016/S0377-0427(00)00522-7.

URL https://www.sciencedirect.com/science/article/pii/S0377042700005227. Numerical Analysis 2000. Vol. VII: Partial Differential Equations.

D. Esteban-Fernandez. Swot project mission performance and error budget document. Technical report, JPL, NASA, 2014.

G. Evensen. *Data Assimilation*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 9783642037108 9783642037115. URL http://link.springer.com/10.1007/978-3-642-03711-5.

G. Evensen, , F. C. Vossepoel, and P. J. van Leeuwen. *Data Assimilation Fundamentals*. Springer Textbooks in Earth Sciences, Geography and Environment (STEGE). Springer Cham, 2022. ISBN 978-3-030-96708-6. doi: 10.1007/978-3-030-96709-3. URL https://doi.org/10.1007/978-3-030-96709-3.

R. Fablet, L. Drumetz, and F. Rousseau. Joint learning of variational representations and solvers for inverse problems with partially-observed data. *arXiv:2006.03653 [physics]*, 2020. URL https://arxiv.org/abs/2006.03653. arXiv:2006.03653.

R. Fablet, B. Chapron, L. Drumetz, E. Mémin, O. Pannekoucke, and F. Rousseau. Learning variational data assimilation models and solvers. *Journal of Advances in Modeling Earth Systems*, 13(10):e2021MS002572, 2021. doi: https://doi.org/10.1029/2021MS002572. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2021MS002572. e2021MS002572 2021MS002572.

Q. Febvre, R. Fablet, J. L. Sommer, and C. Ubelmann. Joint calibration and mapping of satellite altimetry data using trainable variational models. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1536–1540, 2022. doi: 10.1109/ICASSP43922.2022.9746889.

B. A. Finlayson. *Numerical Methods for Problems with Moving Fronts*. Ravenna Park Publishing, Seattle, 1992.

G.-A. Fuglstad, F. Lindgren, D. Simpson, and H. Rue. Exploring a new class of non-stationary spatial gaussian random fields with varying local anisotropy. *Statistica Sinica*, 25(1):115–133, 2015a. ISSN 1017-0405. doi: 10.5705/ss.2013.106w.

G.-A. Fuglstad, D. Simpson, F. Lindgren, and H. Rue. Does non-stationary spatial data always require non-stationary random fields? *Spatial Statistics*, 14:505–531, 2015b. ISSN 2211-6753. doi: https://doi.org/10.1016/j.spasta.2015.10.001. URL https://www.sciencedirect.com/science/article/pii/S2211675315000780.

G.-A. Fuglstad, D. Simpson, F. Lindgren, and H. Rue. Does non-stationary spatial data always require non-stationary random fields? *Spatial Statistics*, 14:505 – 531, 2015c. ISSN 2211-6753. doi: https://doi.org/10.1016/j.spasta.2015.10.001. URL http://www.sciencedirect.com/science/article/pii/S2211675315000780.

R. Furrer, M. G. Genton, and D. Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15(3):502–523, 2006. ISSN 10618600. URL http://www.jstor.org/stable/27594195.

L. Gaultier and C. Ubelmann. Swot simulator documentation. Technical report, JPL, NASA, 2010.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

A. Grigorievskiy, N. D. Lawrence, and S. Särkkä. Parallelizable sparse inverse formulation gaussian processes (spingp). *2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2016.

J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, 2020.

Y. Huang, Y. Zhang, and J. A. Chambers. A novel kullback–leibler divergence minimization-based adaptive student's t-filter. *IEEE Transactions on Signal Processing*, 67(20):5417–5432, 2019. doi: 10.1109/TSP.2019.2939079.

D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2022.

K. Kurosawa and J. Poterjoy. A statistical hypothesis testing strategy for adaptively blending particle filters and ensemble kalman filters for data assimilation. *Monthly Weather Review*, 151

(1):105 – 125, 2023. doi: 10.1175/MWR-D-22-0108.1. URL https://journals.ametsoc.org/view/journals/mwre/151/1/MWR-D-22-0108.1.xml.

F. Le Guillou. *Cartographie dynamique de la topographie de l'océan de surface par assimilation de données altimétriques*. Theses, Université Grenoble Alpes [2020-....], May 2022. URL https://theses.hal.science/tel-03775828.

F. Le Guillou, L. Gaultier, M. Ballarotta, S. Metref, C. Ubelmann, E. Cosme, and M.-H. Rio. Regional mapping of energetic short mesoscale ocean dynamics from altimetry: performances from real observations. *EGUsphere*, 2023:1–17, 2023. doi: 10.5194/egusphere-2023-509. URL https://egusphere.copernicus.org/preprints/2023/egusphere-2023-509/.

H. Lewy, K. Friedrichs, and R. Courant. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100:32–74, 1928. URL http://eudml.org/doc/159283.

X. Li, B. Wu, X. Zhu, and H. Yang. Consecutively missing seismic data interpolation based on coordinate attention unet. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2022. doi: 10.1109/LGRS.2021.3128511.

F. Lindgren, H. Rue, and J. Lindström. An explicit link between gaussian fields and gaussian markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011. doi: 10.1111/j.1467-9868.2011.00777.x. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9868.2011.00777.x.

S. A. Martin, G. E. Manucharyan, and P. Klein. Synthesizing sea surface temperature and satellite altimetry observations using deep learning improves the accuracy and resolution of gridded sea surface height anomalies. *Journal of Advances in Modeling Earth Systems*, 15(5):e2022MS003589, 2023. doi: https://doi.org/10.1029/2022MS003589. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2022MS003589. e2022MS003589 2022MS003589.

L. Menut, B. Bessagnet, R. Briant, A. Cholakian, F. Couvidat, S. Mailler, R. Pennel, G. Siour, P. Tuccella, S. Turquety, and M. Valari. The chimere v2020r1 online chemistry-transport model.

Geoscientific Model Development, 14(11):6781–6811, 2021. doi: 10.5194/gmd-14-6781-2021. URL https://gmd.copernicus.org/articles/14/6781/2021/.

S. Metref, E. Cosme, F. Le Guillou, J. Le Sommer, J.-M. Brankart, and J. Verron. Wide-swath altimetric satellite data assimilation with correlated-error reduction. *Frontiers in Marine Science*, 6:822, 2020. ISSN 2296-7745. doi: 10.3389/fmars.2019.00822. URL https://www.frontiersin.org/article/10.3389/fmars.2019.00822.

J.-M. Molines. meom-configurations/NATL60-CJM165: NATL60 code used for CJM165 experiment, Mar. 2018. URL https://zenodo.org/record/1210116\#.XDpmc89Khp8.

T. Nakamura, K. Fukami, K. Hasegawa, Y. Nabae, and K. Fukagata. Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow. *Physics of Fluids*, 33(2), Feb. 2021. ISSN 1070-6631. doi: 10.1063/5.0039845. Publisher Copyright: © 2021 Author(s).

A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. 2017.

M. Pereira, N. Desassis, and D. Allard. Geostatistics for large datasets on riemannian manifolds: A matrix-free approach. *Journal of Data Science*, 20(4):512–532, 2022. ISSN 1680-743X. doi: 10.6339/22-JDS1075.

H. S. Price, R. S. Varga, and J. E. Warren. Applications of oscillation matrices to diequations. *J. Math. Phys*, 45, 1966.

F. Rozet and G. Louppe. Score-based data assimilation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 40521–40541. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/7f7fa581cc8a1970a4332920cdf87395-Paper-Conference.pdf.

J. J. Ruiz, M. Pulido, and T. Miyoshi. Estimating model parameters with ensemble-based data assimilation: A review. *Journal of the Meteorological Society of Japan. Ser. II*, 91(2):79–99, 2013.

M. A. Sacco, J. J. Ruiz, M. Pulido, and P. Tandeo. Evaluation of machine learning techniques for forecast uncertainty quantification. *Quarterly Journal of the Royal Meteorological Society*, 148(749):3470–3490, 2022. doi: https://doi.org/10.1002/qj.4362. URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.4362.

M. A. Sacco, M. Pulido, J. J. Ruiz, and P. Tandeo. On-line machine-learning forecast uncertainty estimation for sequential data assimilation. *Quarterly Journal of the Royal Meteorological Society*, n/a(n/a), 2024. doi: https://doi.org/10.1002/qj.4743. URL https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.4743.

S. Särkka and J. Hartikainen. Infinite-dimensional kalman filtering approach to spatio-temporal gaussian process regression. In N. D. Lawrence and M. Girolami, editors, *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, volume 22 of *Proceedings of Machine Learning Research*, pages 993–1001, La Palma, Canary Islands, 21–23 Apr 2012. PMLR. URL https://proceedings.mlr.press/v22/sarkka12.html.

S. Särkka, A. Solin, and J. Hartikainen. Spatiotemporal learning via infinite-dimensional bayesian filtering and smoothing: A look at gaussian process regression through kalman filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013. doi: 10.1109/MSP.2013.2246292.

M. Seeger, A. Hetzel, Z. Dai, E. Meissner, and N. D. Lawrence. Auto-differentiating linear algebra, 2019.

Z. Shi, X. Xu, X. Liu, J. Chen, and M.-H. Yang. Video frame interpolation transformer, 2022. URL https://arxiv.org/abs/2111.13817.

F. Sigrist, H. R. Künsch, and W. A. Stahel. Stochastic partial differential equation based modelling of large space–time data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77(1):3–33, 2015. doi: https://doi.org/10.1111/rssb.12061. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12061.

H. Song, I. Hoteit, B. D. Cornuelle, X. Luo, and A. C. Subramanian. An adjoint-based adaptive ensemble kalman filter. *Monthly Weather Review*, 141(10):3343 – 3359, 2013. doi: 10.1175/MWR-D-12-00244.1. URL https://journals.ametsoc.org/view/journals/mwre/141/10/mwr-d-12-00244.1.xml.

J. C. Strikwerda. *Finite Difference Schemes and Partial Differential Equations (Wadsworth & Brooks/Cole Mathematics Series)*. Wadsworth & BrooksCole Advanced Books & Software, Pacific Grove, 1989.

Z. Su, J. Wang, P. Klein, A. F. Thompson, and D. Menemenlis. Ocean submesoscales as a key component of the global heat budget. *Nature Communications*, 9(1):775, Feb 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-02983-w. URL https://doi.org/10.1038/s41467-018-02983-w.

G. Taburet, A. Sanchez-Roman, M. Ballarotta, M.-I. Pujol, J.-F. Legeais, F. Fournier, Y. Faugere, and G. Dibarboure. DUACS DT2018: 25 years of reprocessed sea level altimetry products. 15 (5):1207–1224, 2019. ISSN 1812-0784. doi: https://doi.org/10.5194/os-15-1207-2019. URL https://www.ocean-sci.net/15/1207/2019/. Publisher: Copernicus GmbH.

P. Tandeo, P. Ailliot, J. Ruiz, A. Hannart, B. Chapron, A. Cuzol, V. Monbet, R. Easton, and R. Fablet. Combining Analog Method and Ensemble Data Assimilation: Application to the Lorenz-63 Chaotic System. In V. Lakshmanan, E. Gilleland, A. McGovern, and M. Tingley, editors, *Machine Learning and Data Mining Approaches to Climate Science*, pages 3–12. Springer, 2015.

P. Tandeo, P. Ailliot, M. Bocquet, A. Carrassi, T. Miyoshi, M. Pulido, and Y. Zhen. A review of innovation-based methods to jointly estimate model and observation error covariance matrices in ensemble data assimilation. *Monthly Weather Review*, 148(10):3973 – 3994, 2020. doi: 10.1175/MWR-D-19-0240.1. URL https://journals.ametsoc.org/view/journals/mwre/148/10/mwrD190240.xml.

J. Tong, L. Xie, W. Yang, and K. Zhang. Probabilistic decomposition transformer for time series forecasting, 2022.

P.-Y. Traon, F. Nadal, and N. Ducet. An improved mapping method of multisatellite altimeter data. *Journal of Atmospheric and Oceanic Technology*, 15 (2):522–534, 1998. doi: 10.1175/1520-0426(1998)015<0522:AIMMOM>2.0.CO;2. URL https://journals.ametsoc.org/view/journals/atot/15/2/1520-0426_1998_015_0522_aimmom_2_0_co_2.xml.

C. Ubelmann, P. Klein, and L.-L. Fu. Dynamic Interpolation of Sea Surface Height and Potential Applications for Future High-Resolution Altimetry Mapping. *Journal of Atmospheric and Oceanic Technology*, 32(1):177–184, Oct. 2014. ISSN 0739-0572. doi: 10.1175/JTECH-D-14-00152.1. URL http://journals.ametsoc.org/doi/abs/10.1175/JTECH-D-14-00152.1.

C. Ubelmann, P. Klein, and L.-L. Fu. Dynamic interpolation of sea surface height and potential applications for future high-resolution altimetry mapping. *Journal of Atmospheric and Oceanic Technology*, 32(1):177 – 184, 2015. doi: 10.1175/JTECH-D-14-00152.1. URL https://journals.ametsoc.org/view/journals/atot/32/1/jtech-d-14-00152_1.xml.

C. Ubelmann, G. Dibarboure, L. Gaultier, A. Ponte, F. Ardhuin, M. Ballarotta, and Y. Faugère. Reconstructing ocean surface current combining altimetry and future spaceborne doppler data. *Journal of Geophysical Research: Oceans*, 126(3):e2020JC016560, 2021. doi: https://doi.org/10.1029/2020JC016560. URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2020JC016560. e2020JC016560 2020JC016560.

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

H. Wackernagel. *Multivariate Geostatistics. An Introduction with Applications*. Springer-Verlag Berlin Heidelberg, New York, 2003. doi: 10.1007/978-3-662-05294-5. ISBN 978-3-540-44142-7.

P. Whittle. On stationary processes in the plane. *Biometrika*, 41(3-4):434–449, 1953. doi: 10.1093/biomet/41.3-4.434. URL http://biomet.oxfordjournals.org/content/41/3-4/434.short.

X. Zhang, M. A. Nawaz, X. Zhao, and A. Curtis. Chapter two - an introduction to variational inference in geophysical inverse problems. In C. Schmelzbach, editor, *Inversion of Geophysical*

Data, volume 62 of *Advances in Geophysics*, pages 73–140. Elsevier, 2021. doi: https://doi.org/10.1016/bs.agph.2021.06.003. URL https://www.sciencedirect.com/science/article/pii/S0065268721000030.

Y. Zhen, P. Tandeo, S. Leroux, S. Metref, T. Penduff, and J. L. Sommer. An adaptive optimal interpolation based on analog forecasting: Application to ssh in the gulf of mexico. *Journal of Atmospheric and Oceanic Technology*, 37(9):1697 – 1711, 2020. doi: 10.1175/JTECH-D-20-0001.1. URL https://journals.ametsoc.org/view/journals/atot/37/9/jtechD200001.xml.