https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

Localization in the mapping particle filter

Juan M. Guerrieri[1], Manuel A. Pulido[1, 2], Takemasa Miyoshi[3,4], Arata Amemiya[3,4], and Juan J. Ruiz[3,5,6]

[1]Department of Physics, FaCENA, Universidad Nacional del Nordeste, Corrientes, Argentina
[2]IMIT, CONICET, Argentina
[3]RIKEN Center for Computational Science, Kobe, Japan
[4]RIKEN Center for Interdisciplinary Theoretical and Mathematical Sciences, Kobe, Japan
[5]Centro de Investigaciones del Mar y la Atmósfera, CIMA/CONICET-UBA
[6]UMI-IFAECI/CNRS, Buenos Aires, Argentina

Correspondence: Juan M. Guerrieri (juanmguerrieri@comunidad.unne.edu.ar)

Abstract.

Data assimilation involves sequential inference in geophysical systems with nonlinear dynamics and observational operators. Particle filters are a promising approach for data assimilation because they are able to represent non-Gaussian densities. The mapping particle filter incorporates the Stein variational gradient descents to produce a particle flow that transforms state vectors from prior to posterior densities, aiming to minimize the Kullback-Leibler divergence. However, for applications in geophysical systems, challenges persist in high dimensions, where sample covariance underestimation leads to filter divergence. This work proposes two localization methods, one in which a local kernel function is defined and the particle flow is global. The second method, given a localization radius, physically partitions the state vector and performs local mappings at each grid point. Gaussian and Gaussian mixtures are evaluated as a prior density. The performance of the proposed Local Mapping Particle Filters (LMPFs) is assessed in synthetic experiments. Observations are produced with a two-scale Lorenz-96 system, while a single-scale Lorenz-96 is used as a surrogate model, introducing model error in the inference. The methods are evaluated with full and partial observations and with different linear and non-linear observational operators. The LMPFs with Gaussian mixtures perform similarly to Gaussian filters such as ETKF and LETKF in most cases, and in some scenarios, they provide competitive performance in terms of analysis accuracy.

1 Introduction

Particle filters have emerged as a valuable approach for addressing non-linear data assimilation challenges, particularly in the context of geophysical systems, with particular promise for improving short-term meteorological forecasting. This potential derives from the inherently non-Gaussian nature of convective instabilities—which dominate short-term weather patterns—and their rapid growth rates compared to synoptic-scale phenomena (Hohenegger and Schar, 2007). As model resolution increases and observation operators become more complex, including non-linear relationships with the model state, the challenge increases. Linear data assimilation techniques, such as four-dimensional variational and Kalman filter-based methods, encounter limitations when confronted with non-linearity. These methods assume a Gaussian prior probability density function for the state. Variational methods struggle under strong non-Gaussianity resulting in multimodal cost functions or when the obser-

https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

vational errors deviate from being Gaussian as well. Ensemble Kalman filters (EnKFs) also explicitly assume that the prior density function and the observation likelihood follow a Gaussian distribution. Notably, Ruiz et al. (2021) show that even when drastically reducing LETKF's assimilation window from 5 minutes to 30 seconds in 1km-resolution experiments, residual non-Gaussianity persists at 40% levels.

In contrast, particle filters, being non-parametric filters, offer distinct advantages in handling non-Gaussian error statistics (van Leeuwen et al., 2019). However, particle filters face challenges when dealing with high dimensionality, which is particularly prominent in geophysical applications characterized by a large number of variables. Addressing this issue has led to the development of various techniques, such as localization (Porterjoy, 2016; Robert and Künsch, 2017), tempering methods (Neal, 1996), or jittering (Cotter el al., 2020), aimed at mitigating the computational burden, instability and inaccuracy associated with high-dimensional problems.

This work is concerned with developing a localization scheme for the variational mapping particle filter (MPF) proposed by Pulido and van Leeuwen (2019). The MPF is a novel data assimilation approach that holds potential for non-linear applications in meteorology and oceanography. It is a sequential Monte Carlo algorithm that uses the Stein Variational Gradient Descent (SVGD) method, proposed by Liu and Wang (2016). In the MPF, state vectors, also known as particles, are propagated from the state predicted by the model (referred to as the background or forecast state) to states whose probability density function matches the posterior density, through a series of mappings. These gradient descent mappings aim to minimize the Kullback-Leibler divergence between the posterior density, which is obtained by applying Bayes' formula, and the sequence of intermediate densities.

The SVGD is a deterministic inference algorithm that converges in the limit of many particles (Del Moral, 2013), but it still faces the commonly referred problem known as 'the curse of dimensionality', for representing densities in high-dimensional spaces. This is a common problem in particle filters (Snyder et al., 2008). One of its manifestations is the underestimation of the sample covariance and the subsequent divergence of the filter. Zhuo et al. (2018) demonstrated that SVGD often collapses into the modes of the target distribution, and this drawback becomes more severe with higher dimensions. Additionally, Ba et al. (2022) have demonstrated that SVGD-based algorithms offer few convergence guarantees. This issue persists even when the number of particles (or ensemble members) is larger than the dimension of the state. Among these limited cases, convergence is achievable in the mean-field regime, which occurs when the number of particles tends to infinity. To address this lack of convergence problem, they proposed alternative formulations of SVGD to improve its convergence properties.

In the field of geophysical modeling, ensemble-based methods and particle filters are recognized as key frameworks for data assimilation. Both can incorporate localization methods to enhance their performance. Localization is a well-founded assumption considering that the state-dependent correlation between physical variables decreases with the distance between them. In the context of these frameworks, localization techniques serve the purpose of reducing the dimensionality of the assimilation process, ensuring accurate integration of observed data into the model state. For the EnKF, localization is typically achieved by adjusting the influence of observations and the prior error covariances based on their spatial proximity to the estimation point (Houtekamer and Mitchell, 2001; Hamill and Whitaker, 2001; Whitaker and Hamill, 2002). Developing and

https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

implementing these localization techniques within the EnKF and particle filters are critical for optimizing their effectiveness in real-world scenarios with spatio-temporal dynamics.

In particle filters, localization can be implemented in many ways (Farchi and Bocquet, 2018), including resampling-based approaches. For instance, Penny and Miyoshi (2016) proposed a local particle filter (LPF) that uses observation-space localization to compute independent analyses at each grid point. By applying deterministic resampling and smoothing the analysis weights across neighboring points, the LPF effectively mitigates particle degeneracy and enhances performance in highly non-linear and non-Gaussian scenarios. While this work demonstrates the advantages of resampling-based localization, alternative particle filter methods, such as flow-based approaches, avoid resampling and focus on continuous transformations. The standard Sequential Importance Resampling filter (SIR) (Doucet et al., 2001) preserves and statistically replicates only the particles near the observations, leading to sample impoverishment. This limitation motivates the development of flow-based particle filtering.

Hu and van Leeuwen (2021) proposed a matrix-valued kernel for the MPF and evaluated its performance with a 1000-variables Lorenz-96 system with 20 particles. They employed in the experiments three different observational operators and observed 25% of the state variables. A localization method was implemented with both the prior covariance matrix and the observational error covariance matrix. It consists of applying a decaying factor to the covariance matrices, which depends on the distance. The performance was compared with the LETKF's showing competitive results.

In this work, two localization schemes in the particle flow are introduced to reduce dimensionality and mitigate the problem of the curse of dimensionality in the MPF. These schemes are evaluated in the two-layer Lorenz-96 model using both total and partial observations and nonlinear observation operators.

2 Methodology

2.1 Mapping particle filter's review

The Mapping Particle Filter (MPF), introduced by Pulido and van Leeuwen (2019), is a non-parametric deterministic data assimilation method based on sample points, i.e., particles. It involves the transformation of the sample states from a prior density function to a posterior density by passing through intermediate states. These intermediate states are driven by an interacting particle flow designed to minimize the Kullback-Leibler divergence between a kernelized distribution of the sample states and the target posterior distribution. Other methodologies, such as those in Subrahmanya et al. (2025), explore particle flow dynamics governed by alternative mechanisms—including stochastic processes derived from the Fokker-Planck equation—to evolve particles and sample the posterior distribution.

In the MPF, based on a hidden Markov model, a state vector evolves over time using a dynamical model and is observed using an observational model simultaneously,

$$\mathbf{x}_k = \mathcal{M}(\mathbf{x}_{k-1}, \boldsymbol{\eta}_k), \tag{1}$$

$$\mathbf{y}_k = \mathcal{H}(\mathbf{x}_k, \boldsymbol{\nu}_k), \tag{2}$$

where $\mathbf{x}_k \in \mathbb{R}^{N_x}$ represents the state at time k, $\boldsymbol{\eta}_k$ denotes the random model error, $\mathbf{y}_k \in \mathbb{R}^{N_y}$ are the observations, $\mathcal{H}$ is the observation operator, and $\boldsymbol{\nu}_k$ represents the observational error. Here, a general framework is presented in which both model and observational errors can be non-additive.

The target density function of the particle flow corresponds to the posterior probability density using Bayes' formula during the assimilation stage,

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})}, \tag{3}$$

This probability density function delineates the analysis states by capturing the likelihood of the forecast given a particular set of observations and a specified prior density.

Consider a set of N_p particles $\{\mathbf{x}_{k-1}^{(1:N_p)}\}$ that sample the posterior density at time $k-1$. To obtain a state that matches the posterior density at time k, the MPF iteratively computes intermediate states from the prior to the target. The particles that sample the prior density at time k are states that undergo dynamical evolution from the particles that sample the posterior density at time $k-1$, denoted as $\{\mathbf{x}_k^{f(j)} = \mathbf{x}_{k,1}^{(j)} = \mathcal{M}(\mathbf{x}_{k-1}^{a(j)}, \boldsymbol{\eta}_k^{(j)})\}_{j=1}^{N_p}$, where the second subscript represents the pseudo-time of the mapping iteration. At each iteration, the particles are transformed by

$$\mathbf{x}_{k,i}^{(j)} = T(\mathbf{x}_{k,i-1}^{(j)}) = \mathbf{x}_{k,i-1}^{(j)} + \epsilon\,\mathbf{v}(\mathbf{x}_{k,i-1}^{(j)}), \tag{4}$$

where T represents the iteration mapping, $\mathbf{v}$ represents the velocity of the particle flow in pseudo-time, ϵ represents the step size of the mapping. It may be considered fixed or estimated adaptively by means of stochastic optimization algorithms (Kingma and Ba, 2014).

The velocity seeks to minimize the Kullback-Leibler divergence between the target posterior density function, and the density of the intermediate states. Therefore, the sample from the prior density is transformed towards a sample from the posterior density through a set of discrete transformations, which in the infinitesimal limit may be interpreted as a flow in the state space.

MPF is inspired by the Stein Variational Gradient Descent (SVGD, Liu and Wang (2016)) method, which is kernel-based. These methods are algorithms that rely on kernel functions to measure similarities between state vectors from different particles. The MPF selects a space of functions known as the unit ball of a reproducing kernel Hilbert space (RKHS), denoted as $\mathbb{F}$. The optimization task is to find $\mathbf{v} \in \mathbb{F}$ that indicates the steepest descent direction of the Kullback-Leibler Divergence D_{KL}.

By choosing an isotropic kernel K and given a set of particles $\{\mathbf{x}_{k,i-1}^{(1:N_p)}\}$ representing a sample of the intermediate density at pseudotime $i-1$, the gradient of the Monte Carlo integration of the KL divergence is computed as:

$$\mathbf{v}(\mathbf{x}) = \frac{1}{N_p}\sum_{j=1}^{N_p}\left[K(\mathbf{x}_{k,i-1}^{(j)},\mathbf{x})\left(\nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log p(\mathbf{x}_{k,i-1}^{(j)}) + \nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log K(\mathbf{x}_{k,i-1}^{(j)},\mathbf{x})\right)\right]. \tag{5}$$

The first term of (5), called the kernel-smoothed gradient of the posterior density, acts as a central force guiding the samples from an initial distribution density function towards the modes of the posterior density. The second term acts as the repulsive force and prevents the particles from collapsing into modes of the posterior.

Radial basis functions are used as kernels in this work,

$$K(\mathbf{x},\mathbf{x}') = e^{-\frac{1}{2}\|\mathbf{x}-\mathbf{x}'\|_{\Sigma}^{2}}, \tag{6}$$

where $\|\mathbf{x}-\mathbf{x}'\|_{\Sigma}^{2} = (\mathbf{x}-\mathbf{x}')^{\top}\Sigma^{-1}(\mathbf{x}-\mathbf{x}')$ denotes the square of the Mahalanobis distance and Σ is referred to as the kernel covariance matrix. This matrix needs to be specified at the beginning of the process. In this work, it is assumed to be proportional to the forecast covariance matrix, though other approaches for defining it are possible.

The gradient of the logarithm of the posterior density requires the analytical forms of the prior density and the likelihood function. In this work, observational errors are assumed to be additive and Gaussian, but the framework is general and other observational error distributions may be readily considered. The resulting gradient of the log posterior evaluated at $\mathbf{x}_{k,i-1}^{(j)}$ is:

$$\nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log p\left(\mathbf{x}_{k,i-1}^{(j)}\right) = \mathbf{H}^{\top}\mathbf{R}_{k}^{-1}\left(\mathbf{y}_{k} - \mathcal{H}\left(\mathbf{x}_{k,i-1}^{(j)}\right)\right) + \nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log p(\mathbf{x}_{k,i-1}^{(j)}|\mathbf{y}_{1:k-1}). \tag{7}$$

The first term is the observation likelihood function, in which $\mathcal{H}$ is the observational operator, $(\in \mathbb{R}^{N_y \times N_x})$, and $\mathbf{H}$ denotes the tangent linear observation operator, $\mathbf{H} = \frac{d\mathcal{H}}{d\mathbf{x}}(\mathbf{x})$, while $\mathbf{R}$ stands for the observational error covariance matrix $(\in \mathbb{R}^{N_y \times N_y})$. The second term in (7) is the gradient of the logarithm of the prior density.

In the case of a Gaussian prior density, the second term is reduced to

$$\nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log p(\mathbf{x}_{k}^{(j)}|\mathbf{y}_{1:k-1}) = -\mathbf{B}_{k}^{-1}\left(\mathbf{x}_{k,i-1}^{(j)} - \overline{\mathbf{x}}_{k,0}\right), \tag{8}$$

where $\mathbf{B}_k$ is the prior or background covariance matrix and $\overline{\mathbf{x}}_{k,0}$ is the prior mean. For sequential Monte Carlo, the prior density in (8) is given by the forecast density, so that $\mathbf{B}_k = \hat{\mathbf{P}}_{k}^{f}$.

Alternatively, if we assume the prior density is a Gaussian mixture based on the forecast particles, this term results in

$$\nabla_{\mathbf{x}_{k,i-1}^{(j)}} \log p(\mathbf{x}_{k}^{(j)}|\mathbf{y}_{1:k-1}) = -\mathbf{Q}_{k}^{-1}\left[\mathbf{x}_{k,i-1}^{(j)} - \frac{\sum_{m=1}^{N_p}\psi_{j,m}\mathcal{M}(\mathbf{x}_{k-1}^{(m)})}{\sum_{m=1}^{N_p}\psi_{j,m}}\right], \tag{9}$$

with $\psi_{j,m} = \exp\left[-\frac{1}{2}\left\|\mathbf{x}_{k,i-1}^{(j)} - \mathcal{M}(\mathbf{x}_{k-1}^{(m)})\right\|_{\mathbf{Q}_k}^{2}\right]$ and $\mathbf{Q}_k$ represents the covariance matrix of the Gaussian mixture.

2.2 Localization methods

The underlying assumption in the two developed localization methods is that on average, error correlations decay with the physical distance so that when the distance between two variables is larger than a given threshold known as the localization radius, the correlation is assumed to be negligible. The correlations of these far points are neglected so that it becomes feasible to produce an inference using only the points of the background state and the observations within the localization radius. This reduction in algorithmic complexity allows to reduce sampling noise and enhance the quality of the analysis for high-dimensional state spaces.

The α-localization algorithm assumes that the kernels are localized around each variable, so that distances between particles are low dimensional. Furthermore it uses a global banded prior covariance matrix, for instance, by localizing the sample covariance via the Hadamard product with a Gaspari-Cohn decaying matrix. The state updates are determined globally.

On the other hand, the β-localization algorithm assumes the full local variational mapping process is localized around each variable. In terms of the localization assumption, this method is similar to the localization in Hunt et al. (2007). For each variable, the optimization is conducted separately considering the observations and the prior state variables within the localization radius.

2.2.1 α-Localization

Given a variable x_l of the state $\mathbf{x}$, we assume that the variables from a neighborhood of x_l, $\mathcal{C}_l$, which are denoted as $\tilde{\mathbf{x}}_l = \{x_{l'}; l' \in \mathcal{C}_l\}$ are correlated with x_l while the rest of the state variables are independent,

$$p(x_l|\mathbf{x}) = p(x_l|\tilde{\mathbf{x}}_l). \tag{10}$$

For simplicity, we assume a single physical type of variable in the state space. In this approach, the local state $\tilde{x}^{(j)}_{k,i,l}$ is defined with four indices: k (time index), i (pseudo-time iteration index), l (space index) and j (particle index). To avoid overclutter, time and iteration indices are omitted. In a one dimensional space for a localization radius ℓ, the vector of neighbor variables is $\tilde{\mathbf{x}}_l = \{x_{l'}; l-\ell \leq l' \leq l+\ell\}$ with dimension $N_{\tilde{\mathbf{x}}} = 2\ell+1$.

The global update of variable x_l from Equation (5) is

$$v_l(\mathbf{x}) = \frac{1}{N_p}\sum_{j=1}^{N_p}\left[K(\mathbf{x}^{(j)},\mathbf{x})\left(\partial_{\tilde{\mathbf{x}}_l^{(j)}}\log p(\mathbf{x}^{(j)}) + \partial_{\tilde{\mathbf{x}}_l^{(j)}}\log K(\mathbf{x}^{(j)},\mathbf{x})\right)\right]. \tag{11}$$

Since the variables beyond the localization radius are assumed to be uncorrelated, we approximate (11) by considering a local kernel following Wang et al. (2018) in which only the variables within the localization radius around l are considered. This local kernel is denoted as $K_l(\tilde{\mathbf{x}}_l^{(j)}, \tilde{\mathbf{x}}_l)$.

The local kernel is specific to each grid point x_l. It calculates the Mahalanobis distance between particles using a state vector that is centered at x_l and includes only the neighboring points that fall within the defined localization radius.

The local kernel is defined with a radial basis function as the global one, but with a kernel covariance matrix given by a block of the global one, $\Lambda_l = \Gamma_l \circ \Sigma$, where the localization matrix Γ_l could be a block matrix around l with one's and all the rest as 0's (as in 12) or some decaying coefficient with the distance of the rest of the points to the l-th grid point (e.g. Gaspari and Cohn (1999) factor). The neighborhood variables $\tilde{\mathbf{x}}_l$ are the ones where Γ_l is not null.

$$(\Gamma_l)_{mn} = \begin{cases} 1 & \text{if} \quad l-\ell \leq m,n \leq l+\ell \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

The resulting local kernel is:

$$K_l(\tilde{\mathbf{x}}_l, \tilde{\mathbf{x}}_l') = e^{-\frac{1}{2}\|\tilde{\mathbf{x}}_l - \tilde{\mathbf{x}}_l'\|^2_{\Lambda_l}} \tag{13}$$

The crucial feature of the local kernel is that the Mahalanobis distance calculation only takes into account low-dimensional states. The local flow in the l variable is therefore approximated by

$$v_l(\tilde{\mathbf{x}}_l) \approx \frac{1}{N_p}\sum_{j=1}^{N_p}\left[K_l(\tilde{\mathbf{x}}_l^{(j)},\tilde{\mathbf{x}}_l)\left(\partial_{\tilde{\mathbf{x}}_l^{(j)}}\log p(\tilde{\mathbf{x}}_l^{(j)}) + \partial_{\tilde{\mathbf{x}}_l^{(j)}}\log K_l(\tilde{\mathbf{x}}_l^{(j)},\tilde{\mathbf{x}}_l)\right)\right] \tag{14}$$

The gradient of the posterior density will be calculated following (7) with these modifications:

- For the gradient of the likelihood term, it is globally calculated using the first term of (7), resulting in a matrix in $\mathbb{R}^{N_x \times N_p}$. The term used in (14) corresponds to the l-th row of that global matrix.
- For the prior density term, we calculate it according to (8) or (9) (depending on our hypothesis), but we use the localized vector $\tilde{\mathbf{x}}_l$ and apply the localized covariance matrix $\Gamma_l \circ \mathbf{B}_k$ or $\Gamma_l \circ \mathbf{Q}_k$. This approach is the same as in the local kernel calculation in (13)

Once the complete velocity vector is reconstructed with each component computed separately, the global states in the next pseudo-time are determined by equation 4. Therefore, covariance inversion and the mappings are global. In (1), a pseudocode of the α-localization algorithm is presented. In this case, the time index is omitted, while the pseudo-time, space, and particle indices are retained.

Algorithm 1 LMPF-α: Global update

```
Compute global $\Sigma$
# Number of pseudo time step iterations is denoted as $N_{it}$
for $i = 1$ to $N_{it}$ do
    for $l = 1$ to $N_x$ do
        $\tilde{\mathbf{x}}_{i,l}^{(j)} \leftarrow x_{i,m}^{(j)}$ , with $m = l-\ell,\dots,l+\ell$ and $j = 1,\dots,N_p$
        Compute $\Lambda_l^{-1}$
        Compute localized log posterior
        Compute $v_l^{(j)}\left(\tilde{\mathbf{x}}_{i,l}^{(1:N_p)}\right)$ as in (14)
        $x_{i+1,l}^{(j)} \leftarrow x_l^{(j)} + \epsilon v_l^{(j)}\left(\tilde{\mathbf{x}}_{i,l}^{(1:N_p)}\right)$
    end for
    # Global state updated at pseudo-time step $i+1$
end for
```

Algorithm 2 LMPF-β: Local update

```
for $l = 1$ to $N_x$ do
    $\tilde{\mathbf{x}}_l^{(j)} \leftarrow x_{1,m}^{(j)}$, with $m = l-\ell,\dots,l+\ell$ and $j = 1,\dots,N_p$
    for $i = 1$ to $N_{it}$ do
        Compute $\Sigma^{-1}$ in the local set $\{\tilde{\mathbf{x}}_{i,l}^{(1:N_p)}\}$ )
        Compute $\mathbf{v}_l^{(j)}\left(\tilde{\mathbf{x}}_{i,l}^{(1:N_p)}\right)$
        $\tilde{\mathbf{x}}_{i+1,l}^{(j)} \leftarrow \tilde{\mathbf{x}}_{i,l}^{(j)} + \epsilon \mathbf{v}_l^{(j)}\left(\tilde{\mathbf{x}}_{i,l}^{(1:N_p)}\right)$
    end for
    Retain center value from $\tilde{\mathbf{x}}_{N_{it},l}^{(j)}$
    # Local convergence at point $l$
end for
```

2.2.2 β-Localization

The β-localization involves a physical partitioning of the state space centred around each variable based on distance between variables. Subsequently, it leverages the same principles of the global MPF to each partition.

This methodology is based on Zhuo et al. (2018) in which the KL divergence is decomposed as,

$$D_{KL}\left(q||p\right) = D_{KL}\left(q(x_l|\mathbf{x}_{\neg l})q(\mathbf{x}_{\neg l})||p(x_l|\tilde{\mathbf{x}}_l)p(\mathbf{x}_{\neg l})\right) + D_{KL}\left(q(\mathbf{x}_{\neg l})||p(\mathbf{x}_{\neg l})\right) \tag{15}$$

where $\mathbf{x}_{\neg l}$ is composed by all the state variables except x_l. Therefore, we can solve a local minimization problem for x_l to find $q(x_l|\tilde{\mathbf{x}}_l)$ and by keeping fixed the rest, $q(\mathbf{x}_{\neg l})$. Although this optimization involves neighbor variables

This approach guarantees that the analysis is performed independently for each state variable, with no dependency on intermediate updates of other grid points. However, the neighborhood variables are considered to define the map for each state variable. This means that while the local analysis at a given grid point depends on nearby observations, the convergence at each point remains independent. This reminds the application of normalizing flows with transformations in each direction (Tabak and Turner, 2012). These local minimizations are iterated along l.

The β-localization algorithm consists of applying the global MPF to the neighborhood of x_l. For a given localization radius ℓ, we use the neighborhood vector as in the α-localization: $\tilde{\mathbf{x}}_l$. The local velocity is defined as the global velocity in equation 5, but calculated only over the localized state vector $\tilde{\mathbf{x}}_l$. Therefore, it considers a kernel as in equation 6 calculated in the physically partitioned state. A localized posterior density is also used, in which only the forecast states in the local domain $\tilde{\mathbf{x}}_{\mathbf{l}}{}^{f(j)} \in N_{\tilde{\mathbf{x}}_{\mathbf{l}}}$ are considered. Observations within the localization radius are selected. For that purpose we define $\mathcal{I}_l$ as the set of observation indices corresponding to the observations that are relevant to the localized state vector $\tilde{\mathbf{x}}_l$. Specifically, for a one-dimensional domain this is

$$\mathcal{I}_l = \{m \mid \text{the position of observation } y_m \text{ lies within the interval } [l-\ell, l+\ell]\}. \tag{16}$$

The localized observation vector $\tilde{\mathbf{y}}_l$ is then defined as the subset of observations whose indices belong to $\mathcal{I}_l$:

$$\tilde{\mathbf{y}}_l = \{y_m \mid m \in \mathcal{I}_l\}. \tag{17}$$

The blocks of $\tilde{\mathcal{H}}_l \in N_{\tilde{\mathbf{y}}_{\mathbf{l}}} \times N_{\tilde{\mathbf{x}}_{\mathbf{l}}}$ and $\mathbf{R}_l \in N_{\tilde{\mathbf{y}}_{\mathbf{l}}} \times N_{\tilde{\mathbf{y}}_{\mathbf{l}}}$ are also coherently selected,

$$\tilde{\mathcal{H}}_l = (\tilde{\mathcal{H}})_{mn} \text{ with } m \in \mathcal{I}_l, n = l-\ell, \ldots, l+\ell, \tag{18}$$

$$\mathbf{R}_l = (\mathbf{R})_{mn} \text{ with } m, n \in \mathcal{I}_l. \tag{19}$$

Thus, we are using observations from a subspace and their associated error covariance related to that subset of observations. Additionally, for large localization radii where distant spurious covariances might still occur, a length-decaying factor could be useful.

For each grid point in the domain, i.e. state variable, the following iterative transformation is applied:

$$\tilde{\mathbf{x}}^{(j)}_{i+1,l} = \tilde{\mathbf{x}}^{(j)}_{i,l} + \epsilon \mathbf{v}_l(\tilde{\mathbf{x}}^{(j)}_{i,l}) \tag{20}$$

The convergence is independent for each grid point. To obtain the global analysis vector, only the element at position l from this local analysis vector is kept. This process is repeated for every spatial point on the grid, i.e. variable of the state vector.

https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

The order of pseudo-time iterations and localized step iterations is reversed between the two methodologies. In the α approach, for each pseudo-time step, the entire domain is updated, resulting in a global state for each pseudo-time step. In contrast, in the β approach, for each local point in the domain, all pseudo-time steps are iterated independently before moving to the next state variable, leading to localized convergence without a global state. The exchange of iterations is easier to observe by looking at the algorithms of LMPF-α in 1 and LMPF-β in 2.

3 Numerical setup

The global MPF and the two variants of the localized MPF are assessed in experiments with synthetic observations. In these experiments, observations are generated based on a known dynamical model. The true state is the solution of the known model, referred to here as the nature model. In contrast, the forecast model is a surrogate for the nature model, so we consider the assimilation experiments in the presence of model error. The surrogate model is used to produce forecasts within the assimilation system, $\mathbf{x}_k^{f(j)} = \mathcal{M}(\mathbf{x}_{k-1}^{a(j)})$. After evolving the previous analysis ensemble states, $\mathbf{x}_{k-1}^{a(j)}$ with this surrogate model, the data assimilation step is conducted, and so on. This approach allows us to examine the assimilation scheme with a known true state in the presence of model errors. In these proof-of-concept experiments, the nature model is the two-scale Lorenz-96 system (Section 3.1), while the surrogate model is the one-scale Lorenz-96 (Section 3.2) so that the source of model errors is the lack of the explicit representation of small-scale dynamics. Both models are deterministic, with no explicit additive stochastic error terms.

3.1 Description of the true model

The nature model is defined by the two-scale Lorenz-96 system equations:

$$\begin{cases} \dot{X}_n = -X_{n-1}(X_{n-2} - X_{n+1}) - X_n + F - \frac{hc}{b}\sum_{j=J(n-1)+1}^{nJ} Y_j \quad n = 1, \cdots, N_{LS} \\ \dot{Y}_m = -cbY_{m+1}(Y_{m+2} - Y_{m-1}) - cY_m + \frac{hc}{b} X_{\text{int}[(m-1)/J]+1} \quad m = 1, \cdots, N_{SS} \end{cases} \tag{21}$$

within a cyclic domain, i.e, $X_{N_{LS}+1} = X_1$,$X_0 = X_{N_{LS}}$, and $X_{-1} = X_{N_{LS}-1}$; $Y_{N_{SS}+1} = Y_1$, $Y_{N_{SS}+2} = Y_2$ and $Y_0 = Y_{N_{SS}}$. The equations are solved using a fourth-order Runge-Kutta scheme. The parameters of the nature model are specified in Table 1. They correspond to the standard configuration of Lorenz 96 following Wilks (2005).

3.2 Description of the surrogate model

The forecast model employed in the data assimilation system is the corresponding single-scale Lorenz '96 system. This model exclusively replicates the large-scale equations so that the influence of the small-scale variables must be parameterized. As the true model, the equations are solved using a fourth-order Runge-Kutta scheme. The equations for the one-scale case are

$$\dot{X}_n = -X_{n-1}(X_{n-2} - X_{n+1}) - X_n + f(X_n) \quad n = 1, \cdots, N_{SU} \tag{22}$$

https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

where $f(X_n)$ the external forcing is defined as $f(X_n) = F + f^*(X_n)$ and consists of a linear parameterization of the effects of small-scale dynamics. The parameterization coefficients, F and f^*, are estimated using the methodology proposed by Pulido et al. (2016). The parameters of the forecast model are specified in Table 2.

Table 1. True model parameters.

Variable	Value	Variable Name
N_{LS}	40	Long-scale dimension
N_{SS}	1280	Small-scale dimension
J	32	N_{LS}/N_{SS}
F	26	External forcing
c	10	Time scale-ratio
b	10	Space scale-ratio
h	1	Coupling constant
dt	1.25×10^{-3}	Time integration step

Table 2. Surrogate model parameters.

Variable	Value	Variable Name
N_{SU}	40	Surrogate model dimension
$f(X_n)$	$26 + f^*(X_n)$	Forcing terms
$f^*(X_n)$	$0.73 \cdot X_n + 0.91$	Parameterized forcing
dt	5×10^{-3}	Time integration step

3.3 Experimental setup

3.3.1 Initial state and observations

To generate the synthetic observations, an initial true state $\mathbf{x}_0^t$ is obtained after integrating the nature model from random initial condition over a long period. The nature model is then evolved from this initial true state for $N_t = 2200$ cycle times. Observations are then generated from the large-scale part of the true states every 0.05 time units,

$$\mathbf{y}_k = \mathcal{H}\left(\mathbf{x}_k^t\Big|_{LS}\right) + \boldsymbol{\nu}_k, \tag{23}$$

where observational errors are unbiased with covariance $\mathbf{R}_k$, i.e. $\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ and $\mathbf{x}_k^t$ represents the evolution of the nature model $\mathbf{x}_k^t = \mathcal{M}^t(\mathbf{x}_{k-1}^t)$. The observation operator is assumed to be constant over time. We assume that the observational covariance matrix is also fixed, and diagonal, i.e.

$$\mathbf{R}_k = R \cdot \mathbf{I}_{N_y \times N_y} \tag{24}$$

Three different observational operators are used: A linear operator $\mathcal{H}$, where $\mathcal{H}(x) = x$ and $R = 0.5$. A square operator $\mathcal{H}$, where $\mathcal{H}(x) = x^2$ and $R = 0.5$. A logarithmic operator $\mathcal{H}$, where $\mathcal{H}(x) = \log(|x| + 1)$ and $R = 0.05$. The logarithmic operator $\log(|x| + 1)$ was chosen instead of $\log(|x|)$ because, for values of x close to zero, the observation operator may diverge and worsen the performance of the assimilation, making it necessary to apply a quality control routine. Also, following Kurosawa and Poterjoy (2021), a smaller observation error is used to avoid filter divergence.

Experiments for each observation operator were conducted with full observations (that is, $N_y = 40$) and with partial observations (that is, $N_y = 20$ with observations at every other grid point). In addition, each combination of observation operator and observation network was run with $N_p = 20$ and $N_p = 50$ particles.

To set the N_p initial states of the particles, we use randomly chosen times from a long simulation of the nature model. From these times, the large-scale components of N_p true states are selected. This selection is used to create the first ensemble, whose particles are independent of the initial true state.

3.3.2 Specifications of the MPF

As mentioned, a Gaussian radial basis function is used as the kernel (6), with its covariance matrix taken to be proportional to the forecast covariance estimated from the sample,

$$\Sigma = \gamma \cdot \hat{\mathbf{P}}^f = \frac{\gamma}{N_p - 1} \sum_{j=1}^{N_p} (\mathbf{x}^{f(j)} - \overline{\mathbf{x}}^f)(\mathbf{x}^{f(j)} - \overline{\mathbf{x}}^f)^\top \tag{25}$$

where γ is a bandwidth hyperparameter and $\overline{\mathbf{x}}^f$ denotes the sample mean of the forecasts across the particles. In this work, we tune this γ hyperparameter for the experiments using a brute-force search.

The step size of the mapping $\boldsymbol{\epsilon}$ is determined adaptively using the Adam optimization method (Kingma and Ba, 2014) with up to 500 iterations of pseudo-time in each cycle. For two experimental setups, the sensitivity of the localization radius and the resulting performance are evaluated in experiments to be shown from which we determine as default localization radius $\ell = 3$ for the rest of the experiments. As mentioned before, the experiments are conducted with 20 and 50 particles. In addition, a specific experiment was performed to evaluate the performance sensitivity to the particle number.

A non-Gaussian posterior density may be the result of a non-linear observation operator or a non-Gaussian prior density distribution resulting from non-linear forecasts. One of the objectives of this work is to evaluate the performance of the MPF in experiments with two prior density distributions: a Gaussian and a Gaussian mixture. In the Gaussian experiments, the resulting gradient of the logarithm of the prior density function is given by (8), in which we take $\mathbf{B} = \hat{\mathbf{P}}^f$. In the global and α-localization cases, this matrix is scaled by a Gaspari-Cohn decaying factor. However, in β-localization, scaling of the prior covariance matrix is not required for small localization radii and thus will not be applied.

In the Gaussian mixture experiments, we use the expression given in (9) for the density. The matrix $\mathbf{Q}_k$ is defined as $\mathbf{Q}_k = \xi \cdot \mathbf{P}^f$ where ξ is a bandwidth hyperparameter of the mixtures. Tuning this hyperparameter contributes to enhancing the performance of the MPF. The number of Gaussians corresponds to the number of particles.

In preliminary experiments, we found that a multiplicative or additive inflation factor is not required in the MPF even when applied over an extended period. In fact, adding an inflation factor degraded the performance of the filter.

4 Results

In each experiment, a comparison is made between the global MPF, both localization schemes, the Ensemble Transform Kalman Filter and the Local Ensemble Transform Kalman Filter (Hunt et al., 2007) with $\ell = 3$.

The Root Mean Square Error (RMSE) and the spread are the primary metrics used to compare the performance of each experiment. The time series consists of 2,200 cycles, with the initial 200 cycles designated as the spin-up period and excluded from the analysis. The temporal averages of the RMSE and spread are then calculated over the subsequent 2,000 cycles.

Before the experiments, we conducted a hyperparameter optimization. In the case of Kalman filters, this involves a multiplicative inflation factor that minimizes the RMSE of the analyses. For the MPF and its local versions, one of the key hyperparameters is the proportionality factor of the kernel sample covariance γ. For experiments assuming a Gaussian mixture, another hyperparameter is the width of the Gaussian mixtures, ξ. Thus, the optimization is performed in the 2D space defined by γ and ξ for the Gaussian mixture case. As an example of the hyperparameter tuning, Figure 1 shows the optimization of the global MPF with Gaussian mixture prior in a fully observed linear case using 20 particles. Optimization for the different variants of the MPF and the ensemble Kalman filters is performed for each particle size and observation network.

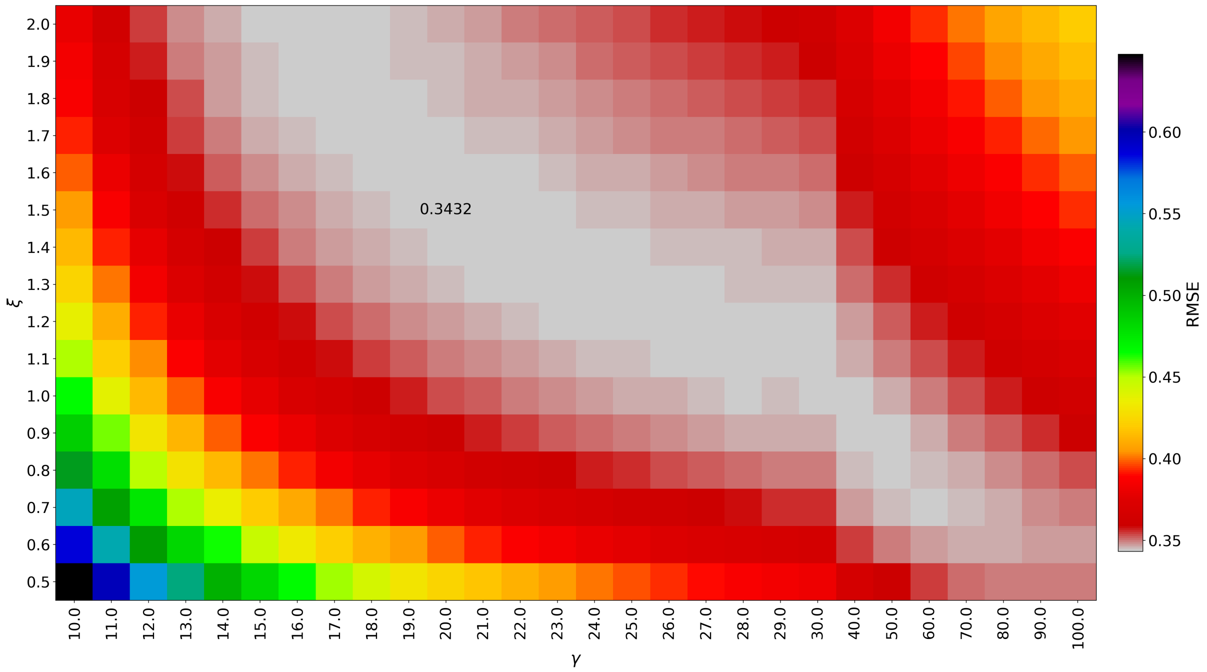
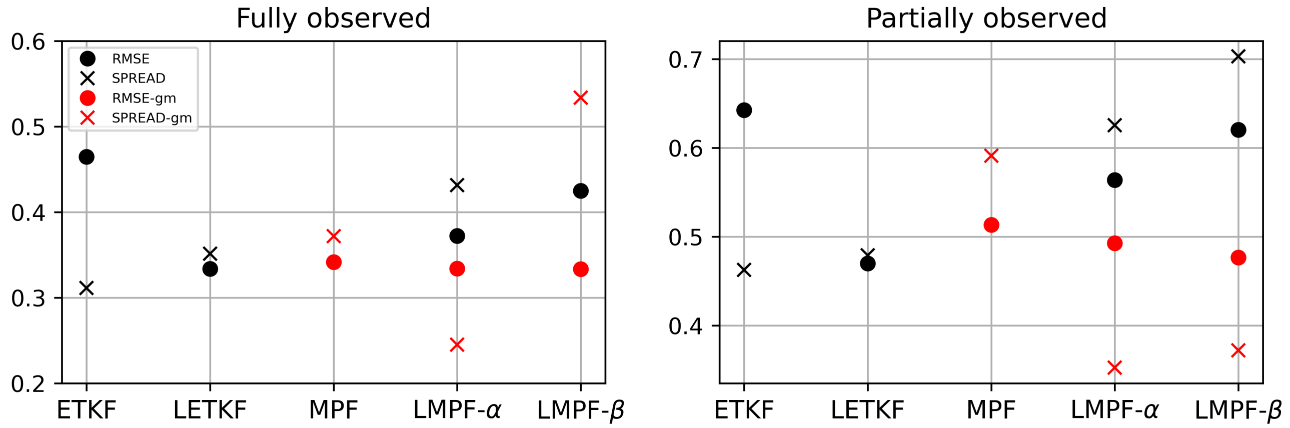


Figure 1. Time and variable averaged RMSE for the MPF experiment as a function of the bandwidth of the Gaussian mixtures ξ and the bandwidth of the kernel γ.

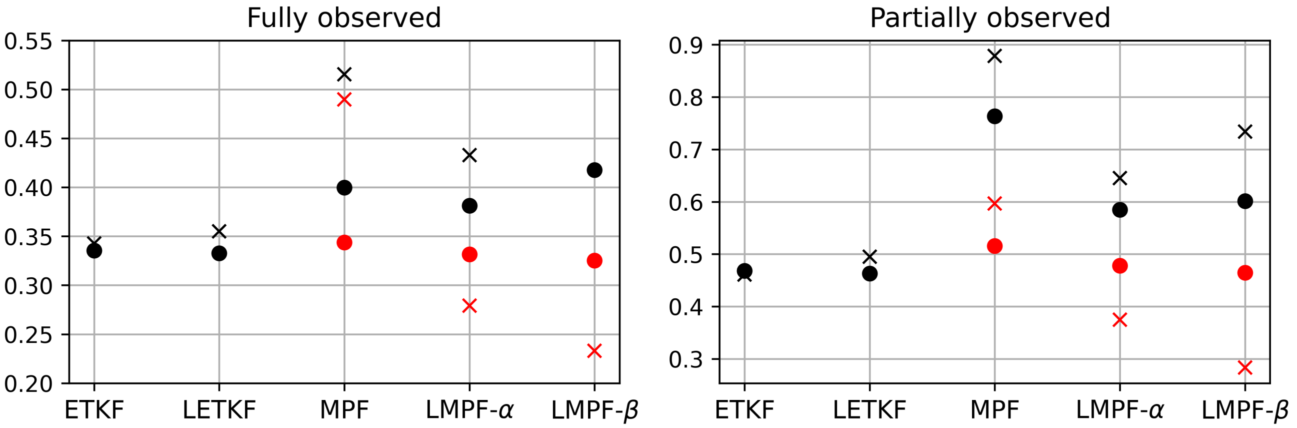
For the experiments shown in this work with the two-scale Lorenz-96 and its surrogate one-scale Lorenz-96 model, a model integration without data assimilation achieves an RMSE of 6.78 and a spread of 6.55. The RMSE of 6.78 represents the maximum error of the forecast model without assimilation, providing a top value for evaluating the impact of incorporating observational data in the assimilation process.

4.1 Linear observational operator

The first experiment evaluates the performance of the local mapping particle filters under a linear observation operator (Eq. 23). Figure 2 shows the results for the fully observed (left panels) and partially observed (right panels) scenarios, employing 20 (2a) and 50 particles (2b). Black dots and crosses represent Gaussian filters or particle filters that assume Gaussian priors. Red dots and crosses represent particle filters with Gaussian mixture priors.



(a) $N_p = 20$



(b) $N_p = 50$

Figure 2. RMSE and spread in the linear observation operator for 20 and 50 particles, under both fully and partially observed scenarios.

All MPF experiments exhibit better performance than ETKF for the 20-particle experiments with a full observed state (Figure 2a) except for the global MPF using a pure Gaussian prior PDF. This last case converges to an RMSE of 0.63 but with an extremely high spread value (5.9). On the other hand, when a Gaussian mixture prior density is utilized, represented by the red dots, all three MPF experiments demonstrate performances similar to LETKF.

Regarding the spread, MPF and LMPF with pure Gaussian priors tend to have large dispersion. In the case of LMPF-β, the spread is 0.92 and is not shown. However, this pattern changes in Gaussian mixture experiments, where the spread is much closer to the RMSE. The global case provides the spread that is closest to the RMSE. It is important to note that these spread results come from experiments using optimal hyperparameters in terms of RMSE.

Despite the linearity of the observational operator, the model dynamics is non-linear. Consequently, it is expected that Gaussian mixtures capture non-linearities more effectively compared to experiments utilizing pure Gaussian priors. This could explain the better performance of the Gaussian mixture experiments.

The right panel of Figure 2a shows results for partially observed experiments. In the Gaussian prior case, the global MPF fails to converge. The localized filters achieve a lower RMSE than ETKF, and perform similarly than LETKF.

As in the fully observed scenario, the Gaussian mixture experiments show a significant improvement across all MPFs. The resulting RMSE is comparable to that of LETKF.

Figure 2b presents the results for the 50-particle experiments. The performance relationships among the experiments are similar to the previous case, with the notable exception of ETKF, which shows the most significant improvement. In this case, the Gaussian-prior global MPF achieves convergence, although its RMSE remains higher than that of the Kalman filters. Once again, the Gaussian-prior LMPF-β exhibits a very high spread (0.86). Similarly, the Gaussian mixture experiments demonstrate a significant improvement in RMSE.

In the partially observed scenario, the ETKF demonstrates the most significant improvement, and the Gaussian-prior MPF successfully converges. Additionally, the spread of the Gaussian-mixture prior MPF is closer to the RMSE in the global case. The localized particle filters exhibit a similar behavior to that observed in the fully observed case.

We note that these experiments use a localization radius of $l = 3$ which is optimum for the ensemble Kalman filter. On the other hand, the localized variants of the MPF exhibit better performance for longer radii as is shown in Section 4.4.

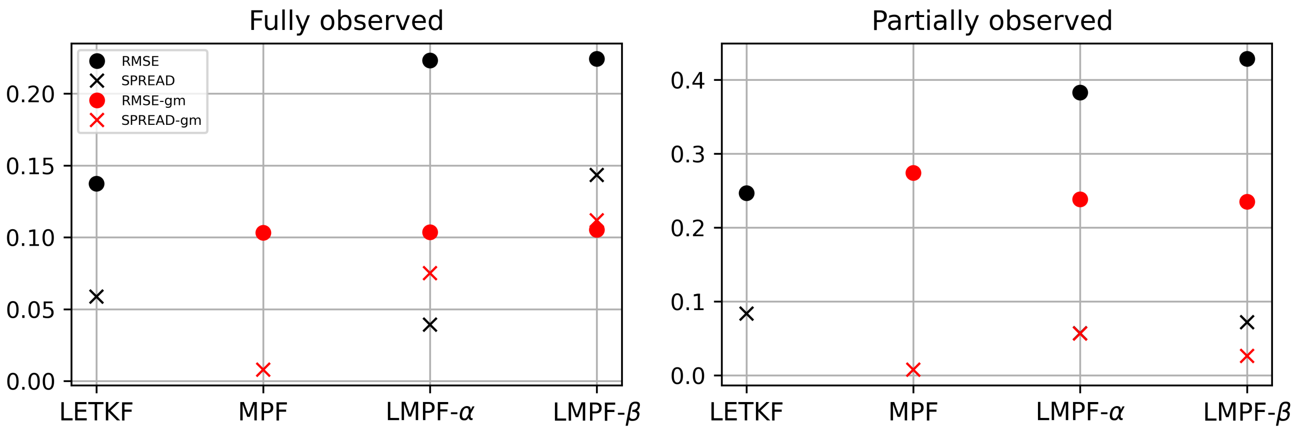
4.2 Square observational operator

A square observational operator presents a challenge for data assimilation schemes, as it treats negative and positive true states with the same absolute value as equivalent, so that the error distributions in the hidden state space is likely to be a bimodal distribution.

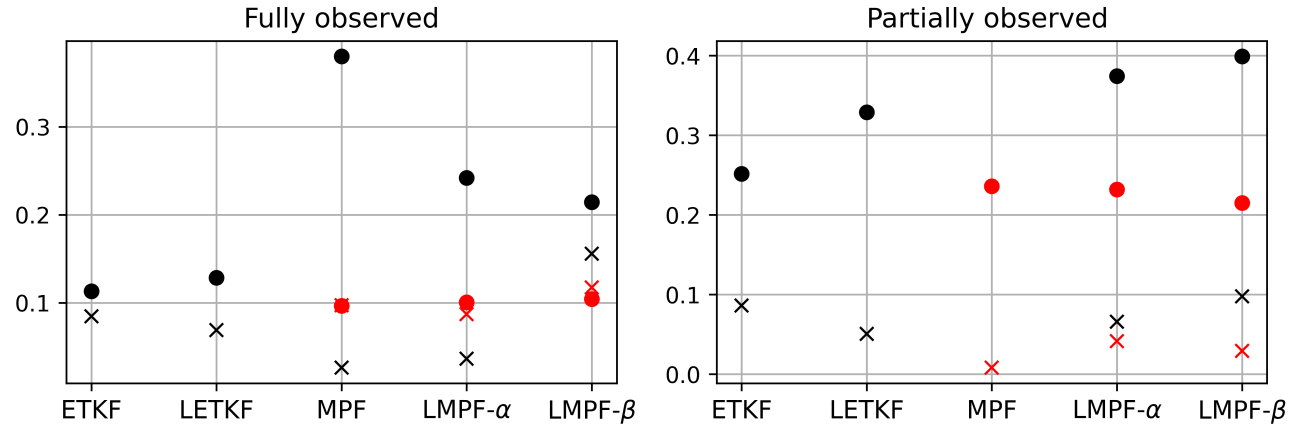
Figure 3a presents the square-$\mathcal{H}$ results for the 20-particle experiments. Overall, the RMSE values are smaller compared to the linear case. This difference is linked to the choice of model error variance. While $R = 0.5$ in both cases, the magnitude of nonlinear observations is typically much greater than that of linear observations, resulting in a relatively smaller error in the nonlinear case.

In the fully observed case, both the ETKF and the global MPF with a Gaussian prior failed to converge. However, the LETKF and Gaussian-prior LMPFs achieve good RMSE performance, though with a significant underestimation of the spread. In contrast, the three experiments employing Gaussian-mixture priors demonstrate very good performance, outperforming the

LETKF. Among these, only LMPF-β exhibits a good spread representation, while the others show low spreads. The impact of localization is pronounced in the ensemble Kalman filters (as seen in the ETKF vs. LETKF performance) but has only a minor effect on the Gaussian-mixture MPFs.



(a) $N_p = 20$



(b) $N_p = 50$

Figure 3. RMSE and spread for the experiments with a square observation operator for 20 (a) and 50 (b) particles, under both fully (left panels) and partially observed (right panels) scenarios.

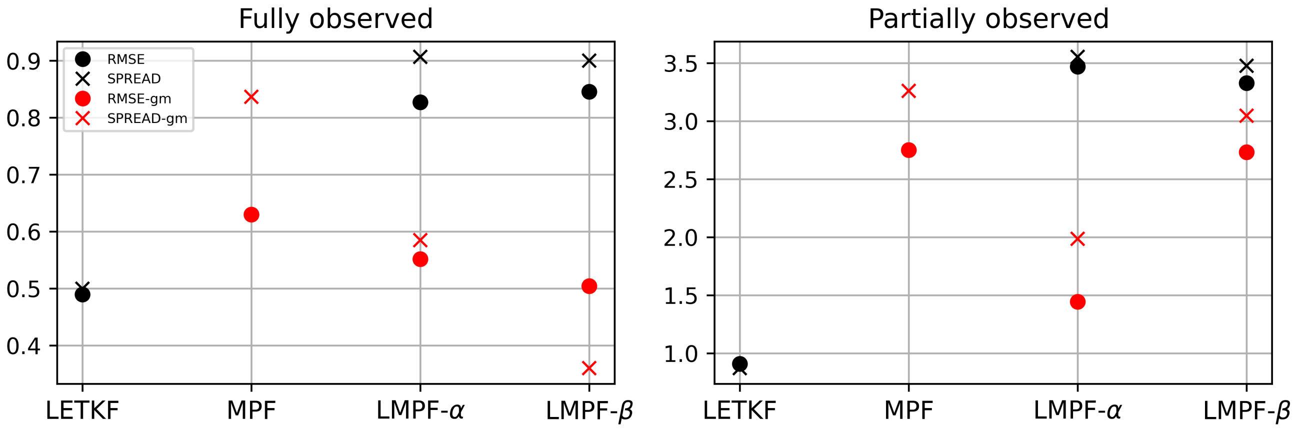
365 In the partially observed case, LETKF shows a similar RMSE than the Gaussian-mixture particle filters. Figure 3b presents the results for the 50-particle experiments. In the fully observed scenario, the Gaussian-prior MPF successfully converges, unlike in the 20-particle case, but with a high RMSE and a significant low spread. The ETKF demonstrates a notable improvement in accuracy, outperforming its localized version. A similar effect is observed in the Gaussian-mixture experiments, where the

MPF achieves slightly better accuracy than its localized counterparts. In this case, the Gaussian-mixture particle filters provide a higher spreads compared to the Gaussian-prior filters.

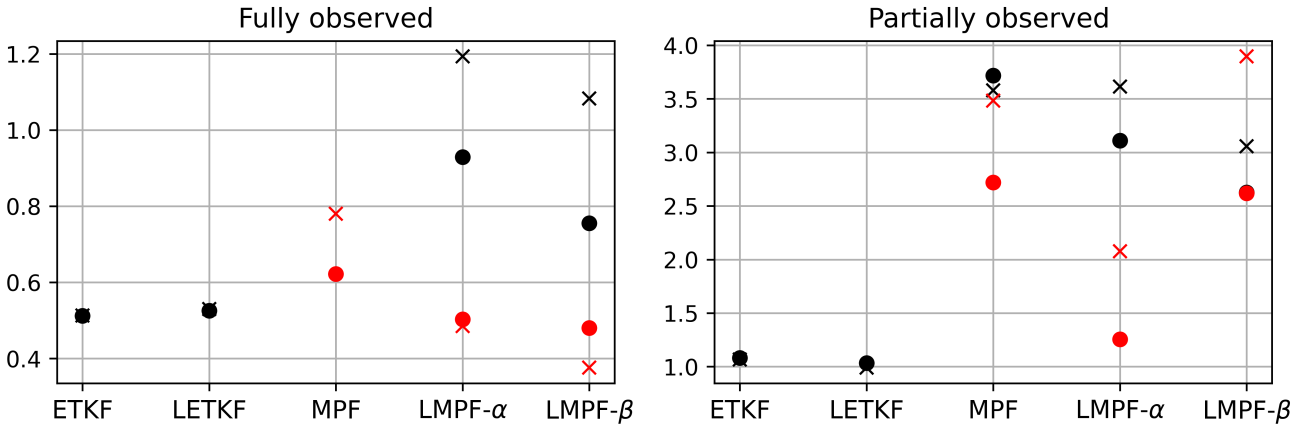
In the partially observed scenario, the ETKF achieves convergence with a lower RMSE than the LETKF. Once again, the Gaussian-mixture filters demonstrate the best performance, comparable to the ETKF. However, as in the 20-particle case, all filters significantly underestimate the spread.

4.3 Logarithmic observational operator

Figure 4 shows the performance of the filters in the logarithmic observation operator case, assessing a highly non-Gaussian regime.



(a) $N_p = 20$



(b) $N_p = 50$

Figure 4. RMSE and spread in the logarithm operator for 20 (a) and 50 (b) particles, under both fully and partially observed scenarios.

For the 20-particle experiments in Fig. 4a, both the ETKF and the Gaussian-prior global MPF diverge in fully and partially observed cases. However, the LETKF achieves excellent RMSE values with a closely matching spread. In contrast, the local particle filters with Gaussian priors exhibit the worst performance. Meanwhile, the Gaussian-mixture prior localized filters show good performance, comparable to the LETKF. In the case of partial observations, the Gaussian-mixture MPFs clearly outperform the Gaussian-prior particle filters. However, the LETKF shows the best overall performance.

For 50 particles (Figure 4b), the ETKF successfully converges in the fully observed case, showing performance comparable to that of the LETKF. In this scenario, the Gaussian-mixture particle filters also demonstrate competitive results.

In the partially observed scenario, ETKF and LETKF again show good results. However, the Gaussian-mixture filters struggle, with the exception of LMPF-α, which delivers performance comparable to LETKF.

4.4 Sensitivity to the localization radius

The performance of localized particle filters is assessed by varying the radius of localization. This study is made on the linear fully observed case, and on the logarithm and partially observed case, the most non-Gaussian scenario. The number of particles used is 20 and only Gaussian-mixture prior densities are used in the MPFs.

Figure 5 shows the results of the linear experiment. The LETKF achieves a minimum RMSE at a localization radius of $\ell = 3$. This is the main reason why we selected this localization radius to conduct all localized experiments.

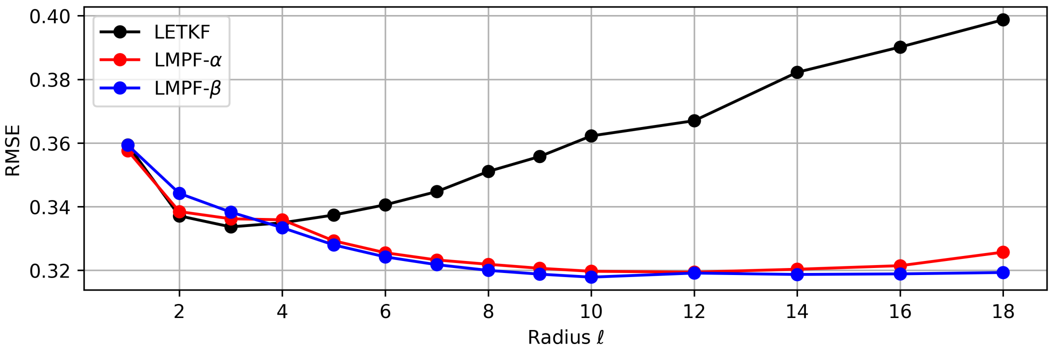


Figure 5. RMSE as a function of localization radius for the LMPFs and the LETKF for the linear and fully observed case with $N_p = 20$.

For radii smaller than 4, LMPFs exhibit an RMSE higher than that of the LETKF. However, for radii greater than 4, the LETKF degrades more rapidly than LMPFs. The LMPFs tend to converge to the same RMSE as the global MPF when a decaying Gaspari-Cohn matrix with a localization radius of 18 is used. This global case considers all distant covariances in the domain while applying a smooth decaying weight. As a result, every covariance, including the farthest ones, contributes to improving the estimation.

LMPF-α exhibits a behavior similar to the β-case but results in slightly higher RMSE values and reaches a minimum around $\ell = 12$. However the difference is small considering that the RMSE as a function of the localization radius is almost flat for that range of localization scales.

This result suggests that a localization scheme might not be necessary, as the best result corresponds to the global case. A possible explanation is that, for the 40-dimensional L96 dynamics, the number of positive Lyapunov exponents is smaller than the ensemble size used in these experiments (20 particles). This ensures sufficiently accurate covariance sampling.

To test this hypothesis, the same experiment was repeated with 10 particles, a number smaller than the required Lyapunov exponents. The results of this experiment are shown in Figure 6. The LETKF exhibits behavior similar to the 20-particle case. For the localized particle filters, a minimum value appears around $\ell = 9$. In the case of the LMPF-α, the filter does not converge for localization radii greater than 12.

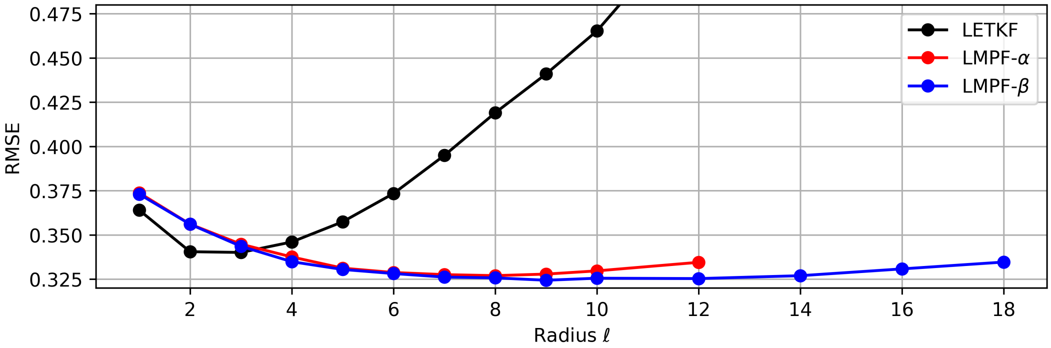


Figure 6. RMSE as a function of localization radius for the LMPFs and the LETKF for the linear and fully observed case with $N_p = 10$.

We remind that the experiments are under the presence of model error. This affects the optimal localization radius, in particular, the LETKF has a longer optimal localization radius for twin perfect-model experiments. In realistic applications, the presence of model errors is also expected to affect long-range correlations. The MPF appears to behave more robustly to this effect.

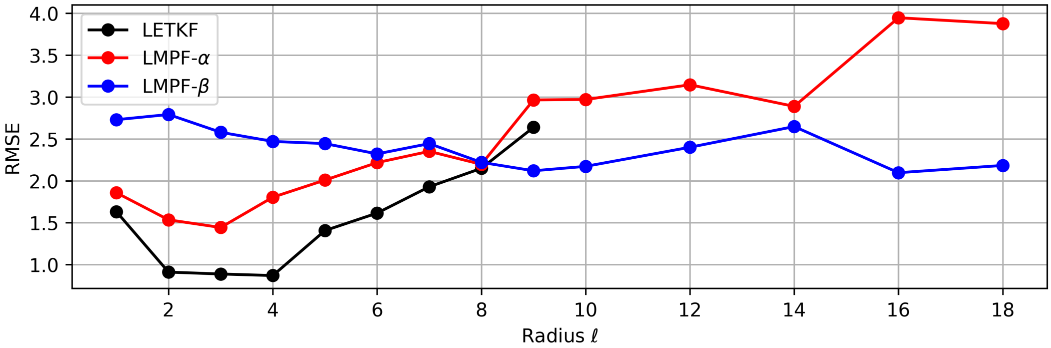


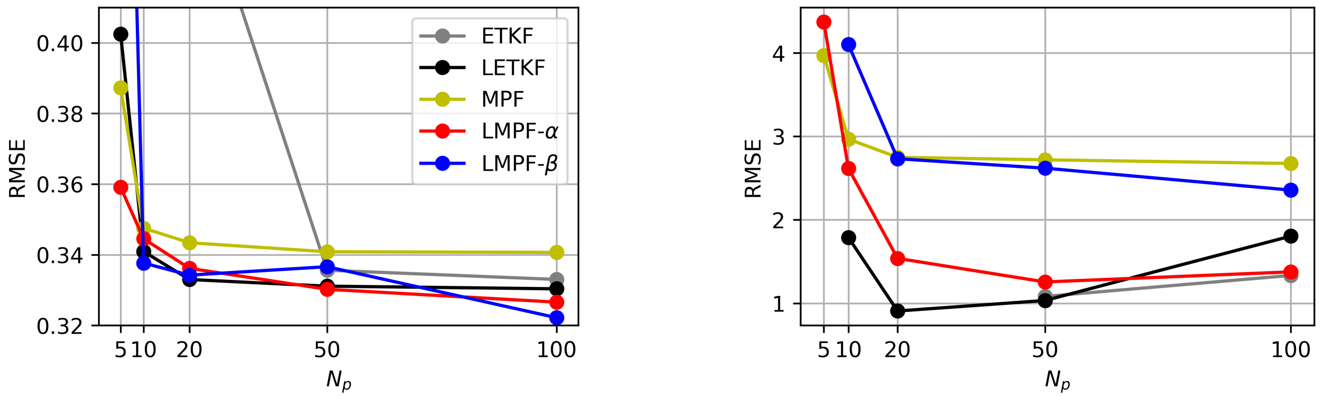
Figure 7. RMSE as a function of localization radius for the LMPFs and the LETKF for the logarithmic and partially observed case.

Figure 7 shows the performance of the filters for the logarithmic and partially observed experiments. In this scenario, the LETKF achieves a minimum RMSE around $\ell = 4$. The LMPF-β exhibits minimal values around $\ell = 9$ and shows steady behavior at higher radii. LMPF-α achieves a minimum value at $\ell = 2, 3$ but experiences degradation at larger radii.

4.5 Sensitivity to the particle number

415 The two extreme experimental setups—fully observed with a linear observation operator, and partially observed with a logarithmic observation operator—are used to evaluate the sensitivity of performance to the number of particles. As in the previous subsection, only Gaussian-mixture prior densities are considered. The localization radius is fixed at $\ell = 3$, and the experiments are conducted with particle numbers of 5, 10, 20, 50, and 100.

Global MPF and LMPF-α demonstrate very good performance for small particle numbers in the linear experiment (Figure 420 8a). For larger particle numbers, both localized particle filters achieve excellent performance, comparable to that of the LETKF.



(a) Fully observed linear case. RMSE: LMPF-β ($N_p = 5$) = 0.583, ETKF ($N_p < 20$) > 6.0, ETKF ($N_p = 20$) = 0.455.

(b) Partially observed logartihmic case.

Figure 8. RMSE as a function of particle number for extreme cases.

In contrast, the results for the partially observed logarithmic case (Figure 8b) are unexpected. For a small number of particles, only LMPF-α and global MPF converge, although with a high RMSE. At larger particle numbers, LMPF-α achieves convergence with an accuracy similar to that of the Kalman filters. The performance of LMPF-β is worse than that of the global MPF for small particle numbers, while for large particle numbers, its performance is only slightly better.

425 The performance of the LETKF in this non-Gaussian experiment deteriorates for ensembles of 50 and 100 particles. A plausible explanation is that certain ensemble members diverge and fail to return to the Lorenz attractor, an effect that is found in deterministic filters (Amezcua et al., 2012). LMPFs do not degrade in performance with increasing numbers of particles and appear to be unaffected by this issue.

4.6 Sensitivity to large model error

In all previous experiments, a linear parameterization of small-scale effects was used. This results in a relatively small model error. To evaluate a large model error scenario, we neglect the linear term of the parameterization f^* (Table 2) and only use the external forcing $f(X_n) = 26$ in the surrogate model. As said before, the RMSE without data assimilation for the parametrized surrogate model is 6.78 and the spread is 6.55. For this large model error environment, the RMSE is 6.86 and the spread reaches 9.01.

Both Kalman and particle filters are tested for the linear and logaritmic observation operators, using $\ell = 3$ for the local filters and $N_p = 20$ particles. Again, we use the localization radius that is optimal for the LETKF, and the hyperparameters γ and ξ are tuned for these cases. The results are displayed in Figure 9. In these large model error conditions, the ETKF did not converge.

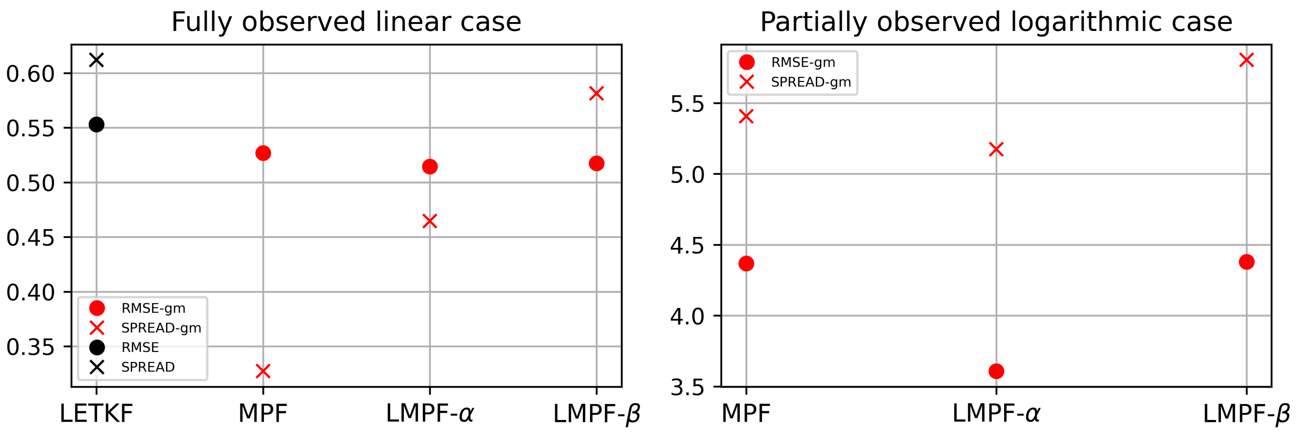


Figure 9. RMSE and spread for the large model error experiments using linear and fully observed cases and logarithm and partial observations with 20 particles and a localization radius of 3.

In the linear case, shown in Fig 9(a), the Gaussian-mixture particle filters have a better performance in terms of RMSE. Nevertheless, the spread of the global MPF is strongly subestimated. Localized particle filters show higher spreads.

In the logarithmic scenario, LETKF fails to converge. In contrast, the MPF and its localized variants handle model error more effectively and perform better in this challenging case (Fig 9b). However, all three filters exhibit rather high RMSE values. Among the tested methods, LMPF-α achieves the best performance in this setup.

5 Conclusions

In this work, two localization schemes for the mapping particle filter were proposed. Both schemes are based on the hypothesis that distant observations do not impact the analysis, but their approaches differ. LMPF-α first calculates a global kernel co-

https://doi.org/10.5194/egusphere-2025-2420
Preprint. Discussion started: 24 June 2025

variance matrix and inverts it. Then, it performs local transformations at each pseudo-time step to obtain a global intermediate state vector in each step. Therefore, convergence is achieved globally. On the other hand, LMPF-β applies the global algorithm in small regions, retaining the center value of each local analysis to obtain a smooth solution. Kernel covariance matrices are calculated in each small domain. Hence, each local analysis achieves convergence independently.

Both frameworks were tested in different setups and compared with the ETKF, LETKF, and the global MPF. The MPF requires the prior density function to be declared beforehand. In general, there is a clear positive impact when taking the prior probability density as a Gaussian mixture compared to a Gaussian prior density.

For both linear and non-linear operators, LMPF's improve estimation compared to their global version when a Gaussian prior is used and provide slightly better estimations when Gaussian mixtures are used. Furthermore, LMPF's provide better estimates compared to the ETKF and highly competitive performances against the LETKF.

In the linear case, LMPF's show very good estimations in terms of RMSE. In the squared case, Gaussian-mixture filters show very good estimations. Both Gaussian and non-Gaussian filters show poor spread representation, especially in partially observed scenarios. In the logarithmic case, Gaussian-mixture LMPF's provide competitive solutions against the LETKF. Again, the partially observed scenario degrades the performance of particle filters while Kalman filters are less affected. LETKF presents a very good performance in the logarithmic operator case under weak model error outperforming MPFs in that non-linear experiment.

When the number of particles varies, Gaussian-mixture MPF and LMPF-α show better estimates at low particle numbers. For the experiments with large model error the MPF and LMPF exhibit robust performances and successfully converge while ensemble based Kalman filters did not deal well with large model errors failing to converge in the logarithmic experiment. However, it is important to highlight that all these experiments required brute-force optimization of two hyperparameters in Gaussian mixtures experiments which is computationally expensive.

The structure of the kernel covariance matrix remains fixed during MPF iterations. However, as particles propagate from the prior forecast density to the posterior density, this matrix must evolve with pseudo time. Particularly for partially observed systems, incorporating the pseudo-time evolution of the kernel covariance matrix could significantly enhance method performance. This represents a interesting direction for future research.

The implementation of the particle filter for data assimilation in Lorenz 96 model experiments represents an essential first step in validating our newly developed methodology. Working with simplified models like Lorenz 96 provides a crucial foundation before advancing to more complex atmospheric forecast models, a progression that will be addressed in future research.

Code availability. All codes used in this study can be obtained from the corresponding author upon reasonable request.

Competing interests. At least one of the (co-)authors is a member of the editorial board of *Nonlinear Processes in Geophysics*.

Author contributions. JG and MP participated in the conception of the ideas and in the design of the experiments. JG conducted the experiments. All the authors reviewed the results. JG and MP wrote the draft, all the authors made corrections and comments to the subsequent versions of the manuscript and approved the final version of the paper.

Financial support. National Agency for the Promotion of Science and Technology of Argentina (ANPCYT, Grant No. PICT 2019/3095) This research has been partially supported by the PREVENIR project implemented by the Japan International Cooperation Agency and the Japan Science and Technology Agency under the Science and Technology Research Partnership for Sustainable Development Program.

References

Amezcua, J., Ide, K., Bishop, C. H. & Kalnay, E. (2012). 'Ensemble clustering in deterministic ensemble Kalman filters', *Tellus A: Dynamic Meteorology and Oceanography*, **64**(1),18039. doi:10.3402/tellusa.v64i0.18039

Ba, J., Erdogdu, M. A., Ghassemi, M., Sun, S., Suzuki, T., Wu, D. & Zhang, T. (2021) 'Understanding the variance collapse of SVGD in high dimensions', *International Conference on Learning Representations*.

Cotter, C. & Crisan, D. & Holm, D. & Pan, W. & Shevchenko, I. (2020). 'A Particle Filter for Stochastic Advection by Lie Transport: A Case Study for the Damped and Forced Incompressible Two-Dimensional Euler Equation'. *SIAM/ASA Journal on Uncertainty Quantification*, **8**, 1446-1492. 10.1137/19M1277606.

Del Moral, P. (2013) 'Mean field simulation for Monte Carlo integration', *Monographs on Statistics and Applied Probability*, **126**, 26.

Doucet, A., de Freitas, N. & Gordon, N. (2001) 'Sequential Monte Carlo Methods in Practice', *Springer*.

Farchi, A., & Bocquet, M. (2018). 'Review article: Comparison of local particle filters and new implementations'. *Nonlinear Processes in Geophysics*, **25**(4), 765–807. doi:10.5194/npg-25-765-2018

Gaspari, G. & Cohn, S. E. (1999) 'Construction of correlation functions in two and three dimensions', *Quarterly Journal of the Royal Meteorological Society*, **125**(554), 723–757. doi:10.1002/qj.49712555417

Hamill, T. M., Whitaker, J. S. & Snyder, C. (2001) 'Distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter', *Monthly Weather Review*, **129**(11), 2776–2790.

Hohenegger, C., & Schar, C. (2007). 'Atmospheric Predictability at Synoptic Versus Cloud-Resolving Scales'. *Bulletin of the American Meteorological Society*, **88**(11), 1783–1794. doi:10.1175/bams-88-11-1783

Houtekamer, P. L. & Mitchell, H. L. (2001) 'A sequential ensemble Kalman filter for atmospheric data assimilation', *Monthly Weather Review*, **129**(1), 123–137.

Hu, C. & van Leeuwen, P. J. (2021) 'A particle flow filter for high-dimensional system applications', *Quarterly Journal of the Royal Meteorological Society*, **147**(737), 2352–2374. doi:10.1002/qj.4028

Hunt, B. R., Kostelich, E. J. & Szunyogh, I. (2007) 'Efficient data assimilation for spatiotemporal chaos: A local ensemble transform Kalman filter', *Physica D: Nonlinear Phenomena*, **230**(1-2), 112–126. doi:10.1016/j.physd.2006.11.008

Kingma, D. P. & Ba, J. (2014) 'Adam: A method for stochastic optimization', *International Conference for Learning Representations*, arXiv preprint arXiv:1412.6980.

Kurosawa K. & Poterjoy, J. (2021) 'Data Assimilation Challenges Posed by Nonlinear Operators: A Comparative Study of Ensemble and Variational Filters and Smoothers', *Monthly Weather Review*, **149**(7), 2369–2389, https://doi.org/10.1175/MWR-D-20-0368.1.

Liu, Q. & Wang, D., 2016. 'Stein Variational Gradient Descent: A General Purpose Bayesian Inference Algorithm', *Advances in neural information processing systems*, **29**.

Neal, R. M. (1996) 'Sampling from multimodal distributions using tempered transitions', *Statistics and Computing*, **6**(4), 353–366. doi:10.1007/bf00143556

Penny, S. G. & Miyoshi, T. (2016) ‘A local particle filter for high-dimensional geophysical systems’, *Nonlin. Processes Geophys.*, **23**, 391–405. https://doi.org/10.5194/npg-23-391-2016.

Poterjoy, J. (2016) ‘A localized particle filter for high-dimensional nonlinear systems’, *Monthly Weather Review*, **144**(1), 59–76. doi:10.1175/mwr-d-15-0163.1

Pulido, M., Scheffler, G., Ruiz, J. J., Lucini, M. M. & Tandeo, P. (2016). ‘Estimation of the functional form of subgrid-scale parametrizations using ensemble-based data assimilation: a simple model experiment’. *Quarterly Journal of the Royal Meteorological Society*, **142**, 2974–2984. doi:10.1002/qj.2879

Pulido, M. & van Leeuwen, P. J. (2019) ‘Sequential Monte Carlo with kernel embedded mappings: The mapping particle filter’, *Journal of Computational Physics*, **396**, 400–415. doi:10.1016/j.jcp.2019.06.060

Robert, S. & Künsch, H. R. (2017) ‘Localizing the Ensemble Kalman Particle Filter’, *Tellus A: Dynamic Meteorology and Oceanography*, **69**(1), 1282016. doi:10.1080/16000870.2017.1282

Ruiz, J.J., Lien, G., Kondo, K., Otsuka, S. & Miyoshi, T. (2021) ‘Reduced non-Gaussianity by 30s rapid update in convective-scale numerical weather prediction’, *Nonlin. Processes Geophys.*, **28**, 615–626.

Snyder, C., Bengtsson, T., Bickel, P. & Anderson, J. (2008) ‘Obstacles to high-dimensional particle filtering’, *Monthly Weather Review*, **136**(12), 4629–4640. doi:10.1175/2008mwr2529.1

Subrahmanya, A. N., Popov, A. A. & Sandu, A. (2025) ‘Ensemble variational Fokker-Planck methods for data assimilation’, *Journal of Computational Physics*, **523**, 113681. https://doi.org/10.1016/j.jcp.2024.113681

Tabak, E. G. & Turner, C. V. (2012) ‘A Family of Nonparametric Density Estimation Algorithms’, *Communications on Pure and Applied Mathematics*, **66**(2), 145–164. doi:10.1002/cpa.21423

van Leeuwen, P. J., Künsch, H.R., Nerger, L., Potthast, R. & Reich, S. (2019) ‘Particle filters for high-dimensional geoscience applications: A review’, *Quarterly Journal of the Royal Meteorological Society*, **145**(723), 2335–2365. doi:10.1002/qj.3551

Wang, D., Zeng, Z. & Liu, Q. (2018) ‘Stein variational message passing for continuous graphical models’, *International Conference on Machine Learning*, 5219–5227.

Wilks, D. S. (2005) ‘Effects of stochastic parameterizations in the Lorenz 96 system’, *Quarterly Journal of the Royal Meteorological Society*, **131**, 389-407. doi: 10.1256/qj.04.03

Whitaker, J. S. & Hamill, T. M. (2002) ‘Ensemble data assimilation without perturbed observations’, *Monthly weather review*, **130**(7), 1913–1924.

Zhuo, J., Liu, C., Shi, J., Zhu, J., Chen, N. & Zhang, B. (2018) ‘Message passing Stein variational gradient descent’, *Proceedings of the 35th International Conference on Machine Learning*, **80**, 6018–6027.