

Ensemble Data Assimilation for Particle-based Methods

Marius Duvillard¹, Loïc Giraldi¹, and Olivier Le Maître²

¹CEA, DES, IRESNE, DEC, SESC, LMCP, Cadarache, F-13108
Saint-Paul-Lez-Durance, France

²CNRS, Inria, Centre de Mathématiques Appliquées, Ecole Polytechnique,
IPP, Route de Saclay, 91128, Palaiseau Cedex, France

Abstract

This study presents a novel approach for integrating data assimilation techniques into particle-based simulations using the Ensemble Kalman Filter. If data assimilation methods have been applied to Eulerian simulations for a long, they have never been properly used in the context of a Lagrangian solution discretization. Two specific methodologies are introduced to complete the analysis. The first one is based on the use of an intermediary Eulerian transformation combining a projection and a remeshing process. The second is a purely Lagrangian scheme that is applicable when remeshing is not adapted. These methods are evaluated using a one-dimensional advection-diffusion model with periodic boundaries. Subsequently, assimilation schemes are applied to a non-linear two-dimensional incompressible flow problem, solved via the Vortex-In-Cell method. In the one-dimensional scenario, the performance of these filters is benchmarked against a grid-based assimilation filter. In the two-dimensional case, the study demonstrates the feasibility of applying these methods in more intricate scenarios.

Keywords: Particle-based Method, Lagrangian Simulation, Data Assimilation, EnKF, Ensemble Methods, Vortex Methods.

Contents

1	Introduction	3
2	Background	5
2.1	Data assimilation	5
2.1.1	Data assimilation setting	5

2.1.2	Bayesian filtering	6
2.1.3	Ensemble Kalman Filter	6
2.2	Particle-based methods	7
2.2.1	Particle discretization	7
2.3	Particle-based function manipulations	8
2.3.1	Approximation operator	9
2.3.2	Regression operator	9
2.3.3	Remeshing operator	10
3	Methods	11
3.1	Remesh-EnKF	12
3.2	Particles-EnKF	13
4	1D density advection-diffusion problem	15
4.1	Description of the problem	15
4.2	Assimilation parameters and ensemble generation	16
4.2.1	Ensemble distribution	16
4.2.2	Error definition	16
4.2.3	Numerical parameters	17
4.3	Results	17
5	2D vortex-in-cell problem	21
5.1	Description of the method	21
5.2	Lamb-Chaplygin dipole and simulation parameters	21
5.3	Assimilation parameters and ensemble generation	23
5.3.1	Ensemble distribution	23
5.3.2	Error definition	24
5.3.3	Numerical parameters	24
5.4	Results	25
5.4.1	Error through time	25
5.4.2	Error with respect to assimilation parameters	26
5.4.3	Error with respect to simulation parameters	27
6	Conclusion	30
A	Stochastic Ensemble Kalman Filter	31
B	Moment conservation of particle discretization	32
C	Parameters	33
C.1	One dimension problem	33
C.2	Two dimension problem	33

1 Introduction

Numerical simulation enables the assessment of complex real-world systems, for instance, to facilitate the optimization of complex systems and perform risk analysis, all while reducing experimental costs. Thanks to the increasing computational resources, they help in understanding and designing processes, particularly in the mechanical field. The solid and fluid mechanics historically leaned on grid-based or mesh-based methods. These techniques necessitate the use of structured meshes. The shift towards meshless methods offers significant promises for complex physics or large deformations (moving interfaces, material disintegration, or distortion) to avoid computing complex geometries.

Meshless methods, specifically particle-based methods, describe geometry as a collection of particles that move with the deformation flow in a Lagrangian fashion. Each particle transports material properties and internal variables. Particles can discretize a continuum medium and are associated with a kernel to reconstruct continuous fields and differential operators. In this article, we will mainly focus on the Vortex Method [9, 20], which discretizes the vorticity field and solves the incompressible fluid flow equation with the vorticity-stream function formulation.

The computed solution may contain errors that need to be understood, quantified, and minimized. If observations are available, integrating this information can lead to a more accurate estimation of the simulation state. In this context, data assimilation techniques offer an optimal way to combine various sources of information, resulting in a more precise estimation of the system state. Integrating model predictions and observational data has been widely applied in disciplines such as meteorology, oceanography, hydrology, and geosciences [4].

In the domain of data assimilation, two prominent families of approaches have emerged: variational and stochastic methods. Variational approaches [26] focus on minimizing a cost function that measures the misfit between model predictions and observations, seeking the optimal system state. The most common formulations derive from 3D-Var, 4D-Var [30].

On the other hand, stochastic approaches go beyond mere state estimation; they delve into the quantification of uncertainty associated with the estimated states. Uncertainty quantification is a critical aspect, especially in dynamic and uncertain systems, where acknowledging and characterizing becomes paramount for reliable decision-making and model improvement. In this case, the estimate is sequentially updated based on previous and current observations. The assimilation process is performed through a Bayesian framework with a forecast and an analysis step. The Kalman filter [17] is an example of a sequential formulation considering a linear model and Gaussian distribution assumptions. However, more advanced filters have been introduced to be adapted to nonlinear and arbitrary distributions. One of the most popular Bayesian filters is undoubtedly the Ensemble Kalman Filter, introduced by Evensen [14], primarily because of its adaptability to high-dimensional problems with any evolution model and its remarkable resilience to deviations from the initial Gaussian assumptions. It consists of approximating the probability distribution of a state thanks to an ensemble of simulations called particles or members.

This paper introduces new approaches to applying ensemble data assimilation techniques to meshless simulations that discretize a continuum domain. The hypothesis considers several members of different particle distributions. The Ensemble Kalman Filter (EnKF) has been extensively employed for Eulerian discretization frameworks. However, its application in the Lagrangian approach presents unique challenges. These issues primarily revolve around defining a unified state representation across all ensemble members and effectively updating this state during the analysis phase.

When particle-based methods discretize a field on a continuum discretization, the particles are point entities and thus allow a certain flexibility to the update. Operations such as agglomeration, splitting, or resampling are utilized to update particle configurations, primarily to mitigate issues like distortion, excessive deformation or to manage particle count [31, 11].

Nevertheless, the crux of the challenge lies in the inherent disparity in discretization across different ensemble members. The first solution is to consider a reference discretization for all members. In fixed-grid methodologies with Multi-Resolution Analysis (MRA) and moving mesh scenarios, the state definition on varied grids with assimilation is managed through projection and interpolation techniques to establish a reference grid for state updates [27, 6]. The selection of the reference and updated grids provides a spectrum of implementation possibilities. Furthermore, Siripatana et al. [27] highlight that the EnKF correction is contingent solely upon the predictions and observations, thereby rendering it independent of the state definition.

Another solution consists of defining the state with the union of the particles, considering the position and associated intensities of each particle [12]. Complex filters have been developed to estimate correctly the posterior discretization with a nonlinear observation model or with a deficient number of sensors of preasure [18]. However, these methods grapple with scenarios involving markedly divergent particle discretizations or highly variant model flows. In this general case, using a particle state for all particles for the update is unfeasible. Indeed, the update implies a linear combination of all members, leading to an exponential increase of particles. On the other hand, the state could be associated with the spatial field defined in a functional space. The updated fields could be evaluated on the entire domain. Finally, using approximation or regression, a new particle discretization could be approximated. These final modifications have already been introduced in the Vortex Method to better approximate the vorticity field by changing the particle intensities regroup under the label Meshless Rezoning Methods in [20]. It mainly involved iterative methods [2], triangulation [25] or Radial Basis Function (RBF) interpolation [1, 28]. RBF offers to introduce new particles or introduce penalization to regularize optimization problems.

Based on those different tools, we propose two novel EnKF-based filters. First, the Remesh-EnKF uses an Eulerian intermediate update. This way, the analysis is defined on an Eulerian framework, and the method is reduced to the previous development of data assimilation. The projection step is something familiar for Particle-In-Cell methods [29, 7].

Nevertheless, it involves remesh the discretization entirely on a regular grid of particles. This method is based on the regridding of the particle discretization as described by [11]. Then, in a case where the particle discretization would be preserved, the Particle-EnKF is introduced. In this case, the analyzed field is approximated on the previous particle discretization of each members. The particle positions are unchanged; only the strengths are modified by regression or approximation of the analyzed solution. In the next part, background on sequential filtering and EnKF algorithm will be introduced 2.1, then on particle-based methods 2.2. Then, the two categories of method will be described in section 3. Afterward, those filters will be compared with a grid-based filter in a 1D Advection-Diffusion problem in section 4, and an incompressible viscous flow is solved using a Vortex Method 5 where the filters are quantitatively analyses.

2 Background

2.1 Data assimilation

Data assimilation could be generally formulated with a probabilistic framework. It allows us to rigorously deal with measurement and model error in order to not only deduce an estimate of the real state but also associate uncertainty. Thus, state and observation are modeled as random variables. A filtering approach is then applied to estimate the current state based on past observations sequentially.

The goal is to establish the recurrence in probability distributions that, through Bayesian estimation, will enable us to estimate the current state and predict the future state.

2.1.1 Data assimilation setting

A hidden Markov chain is used to model this recurrence. We position ourselves within a finite-dimensional context. The forecast and observation are introduced, such as for $k \geq 0$,

$$\begin{cases} \mathbf{x}_{k+1} = \mathcal{M}_{k+1}(\mathbf{x}_k) + \boldsymbol{\eta}_{k+1}, \\ \mathbf{y}_{k+1} = \mathcal{H}_{k+1}(\mathbf{x}_{k+1}) + \boldsymbol{\varepsilon}_{k+1}, \end{cases}$$

where \mathcal{M}_{k+1} is the model operator describing the time evolution of the state from time k to time $k+1$, and \mathcal{H}_k is the observation operator. The term $\mathbf{x}_k \in \mathbb{R}^n$ is the vector state at time k and $\mathbf{y}_k \in \mathbb{R}^m$ the observation vector, $\boldsymbol{\eta}_k$ is the model error that accounts for error in the numerical model and the errors due to discretization, and $\boldsymbol{\varepsilon}_k$ is the observation error which combine measurement error and representativeness error. We assume that $\boldsymbol{\eta}_k$, $\boldsymbol{\varepsilon}_k$ are random variables following Gaussian distributions with zero mean and covariance matrices \mathbf{Q}_k and \mathbf{R}_k respectively. Finally, we assume that the observation and the model errors are independent though the time and that initial error on \mathbf{x}_0 , $\boldsymbol{\varepsilon}_k$ and $\boldsymbol{\eta}_k$ are mutually independent. Let $\mathcal{D}_k = \{\mathbf{y}_l\}_{l=1}^k$ represent the accumulated data up to time k . Thus, \mathbf{x}_{k+1} and \mathcal{D}_k are conditionally independent with respect to \mathbf{x}_k , as well as \mathbf{y}_{k+1} and \mathbf{x}_{k+1} , leading to simplifications in the next paragraph.

2.1.2 Bayesian filtering

The filtering problem consists of assessing the current state of the signal by utilizing data observation up to the present moment. Filtering involves the determination of $p(\mathbf{x}_k | \mathcal{D}_k)$, the probability density function associated with the probability measure on the random variable $\mathbf{x}_k | \mathcal{D}_k$. Specifically, filtering focuses on the sequential updating of this probability density function as the index k is incremented. The state density is initialized by the a priori density of the initial state p_{x_0} . Then, for all $k \geq 0$, probability distributions are propagated. The forecast step is obtained through the law of total probability

$$p(\mathbf{x}_{k+1} | \mathcal{D}_k) = \mathbb{E}_{\mathbf{x}_k} [p(\mathbf{x}_{k+1} | \mathbf{x}_k, \mathcal{D}_k)] = \mathbb{E}_{\mathbf{x}_k} [p(\mathbf{x}_{k+1} | \mathbf{x}_k)].$$

The a priori law of the $k + 1$ observations can be obtained again through the law of total probability

$$p(\mathbf{y}_{k+1} | \mathcal{D}_k) = \mathbb{E}_{\mathbf{x}_{k+1}} [p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1}) | \mathcal{D}_k].$$

After the $k + 1$ observation of the random variable \mathbf{y}_{k+1} , the analysis step determines the a posteriori law of the state using Bayes law

$$p(\mathbf{x}_{k+1} | \mathcal{D}_{k+1}) = p(\mathbf{x}_{k+1} | \mathbf{y}_{k+1}, \mathcal{D}_k) = \frac{p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1}, \mathcal{D}_k)p(\mathbf{x}_{k+1} | \mathcal{D}_k)}{p(\mathbf{y}_{k+1} | \mathcal{D}_k)}.$$

Due to the independence hypothesis the formula reduced to

$$p(\mathbf{x}_{k+1} | \mathcal{D}_{k+1}) = \frac{p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1})p(\mathbf{x}_{k+1})}{p(\mathbf{y}_{k+1})} \propto p(\mathbf{y}_{k+1} | \mathbf{x}_{k+1})p(\mathbf{x}_{k+1}).$$

This finally lead to a mapping from the prior $p(\mathbf{x}_{k+1} | \mathcal{D}_k)$ to the posterior $p(\mathbf{x}_{k+1} | \mathcal{D}_{k+1})$. We remove the time subscript k in the rest of the section for simplicity and present the forecast and analysis step for a time increment.

2.1.3 Ensemble Kalman Filter

The Kalman filter [17] is a Bayesian filter that, in addition to the previously mentioned assumptions, requires \mathcal{M}_k and \mathcal{H}_k to be linear operators. In this case, the posterior distribution of the state is still Gaussian, so only the mean and the variance are updated. The Kalman estimator is thus a recursive version of the Minimum Mean Square Error applied to the Gaussian Linear model.

The ensemble Kalman Filter (EnKF) is a data assimilation method adapted to high dimensional non-linear problems introduced by Evensen [14]. The formulation uses an ensemble of discrete samples based on the assumptions of a multivariate Gaussian distribution, such as the Kalman filter. We present the stochastic EnKF, where the observations are perturbed to account for observation errors.

Assuming we have an ensemble of N states $\{\mathbf{x}_i\}_{i=1}^N$, we forecast the ensemble by propagating each state with the dynamic model and obtain a forecast ensemble. The two first moments of the error are given by

$$\begin{aligned}\bar{\mathbf{x}}^f &= \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i^f, \\ \mathbf{P}^f &= \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i^f - \bar{\mathbf{x}}^f)(\mathbf{x}_i^f - \bar{\mathbf{x}}^f)^T,\end{aligned}$$

where $\bar{\mathbf{x}}^f$ and \mathbf{P}^f are respectively the empirical estimates of the mean and the covariance matrix of the state distribution obtained from the ensemble members.

Likewise, the mean and covariance of the observation $\left\{ \mathcal{H}(\mathbf{x}_i^f) \right\}_{i=1}^N$ could be estimated as well as the covariance between state and observation.

We develop the general formulation of the EnKF filter in the Appendix A.

Finally, our formulation of EnKF takes advantage of a correction of the state defined in the member space. We define \mathbf{F} , the correction matrix that gives the update in terms of linear combinations of the forward states

$$\mathbf{X}^a = \mathbf{X}^f + \mathbf{X}^f \mathbf{F}, \quad (1)$$

where the matrix \mathbf{F} only depends on the ensemble members through the predicted observations ensemble $\left\{ \mathcal{H}(\mathbf{x}_i^f) \right\}_{i=1}^N$, the observation \mathbf{y} and the associate perturbations $\{\boldsymbol{\varepsilon}_i\}_{i=1}^N$. This independence is made possible by the linearization of the observation operator and the low-rank approach. The same property is verified for the ensemble transform Kalman filter [?]. Indeed, the filter is explicitly defined in the perturbation space.

2.2 Particle-based methods

We consider particle methods for solving continuous problems in fluid or solid mechanics. The Lagrangian methods decompose the domain on a set $\mathcal{P} = \{\mathbf{z}_p, \mathbf{U}_p\}_{p=1}^{n_p}$, where $\mathbf{z}_p \in \Omega$, the spatial coordinates a particle p and $\mathbf{U}_p \in \mathbb{R}^n$ its intensity.

We focus our work on methods that discretize a solution \mathbf{u} on a continuous domain $\Omega \subset \mathbb{R}^d$ with Ω the spatial domain. This includes methods like Smoothed particle hydrodynamics (SPH) [16, 19], or the Vortex Method (VM) [9] and is extended to other methods like the Material Point Method (MPM) [29].

2.2.1 Particle discretization

Any smooth field \mathbf{u} on Ω could be written

$$\mathbf{u}(\mathbf{z}) = \int_{\Omega} \mathbf{u}(\mathbf{z}') \delta(\mathbf{z}' - \mathbf{z}) d\mathbf{z}',$$

with δ the Dirac distribution.

A kernel function ϕ_ε is introduced to obtain an average estimate $\langle \mathbf{u} \rangle$ of \mathbf{u} such that

$$\langle \mathbf{u}(\mathbf{z}) \rangle = \int_{\Omega} \mathbf{u}(\mathbf{z}') \phi_{\varepsilon}(\mathbf{z} - \mathbf{z}') d\mathbf{z},$$

where ε is the smoothing length. The smooth kernel should at least respect the following properties

$$\begin{aligned} \int_{\Omega} \phi_{\varepsilon}(\mathbf{z}) d\mathbf{z} &= 1, \\ \phi_{\varepsilon}(\mathbf{z}) &\rightarrow \delta(\mathbf{z}), \quad \varepsilon \rightarrow 0, \\ \phi_{\varepsilon}(\mathbf{z}) &\in C^k, \quad k \geq 1, \end{aligned}$$

where the two first properties are remanent properties of the Dirac distribution and the last is a differentiability requirement.

The average function $\langle \mathbf{u} \rangle$ is then used to approximate the original function.

Finally, the original domain Ω is subdivided with N_p subdomain Ω_p associated with a Lagrangian particle in the location $\mathbf{z}_p \in \Omega_p$. We denote by V_p the volume of Ω_p . This discretization is then used to approximate the average function such that

$$u(\mathbf{z}) \approx \sum_{p \in \mathcal{P}} U_p \phi_{\varepsilon}(\mathbf{z} - \mathbf{z}_p). \quad (2)$$

Thus, any function defined on a particle discretization is defined by a particle discretization \mathcal{P} and a smoothing kernel ϕ_{ε} . Based on this discretization, the differential operator could be derived through this formulation.

Several kernels have been used depending on the method. Theoretically, it could be the Gaussian kernel function

$$\phi_g(\mathbf{r}) = \frac{1}{(\pi \varepsilon^2)^{d/2}} \exp(-\|\mathbf{r}\|^2 / \varepsilon^2).$$

This kernel is infinitely differentiable but defined on non-compact support. In practice, we use a cutoff to remove negligible value for a large distance from a particle.

2.3 Particle-based function manipulations

Based on particle discretization, we present several particle manipulations that will be used in our methods. Initially, those manipulations are either dedicated to improving the quality of the approximation, avoiding high distortion by creating a new particle discretization or projecting the solution on an Eulerian configuration. The different operators will be used in the assimilation process in order to update the particle solution of each member in Section 3.

2.3.1 Approximation operator

The first category of manipulations aims to improve the approximation of the field by modifying particle strength. A first approximation could be to use the particle approximation to reevaluate the particle intensities like in Equation 2 such as

$$\mathbf{U}_p = \int_{\Omega_p} \mathbf{u}(z) dz = \mathbf{u}(\mathbf{z}_p) V_p,$$

where \mathbf{z}_p is the particle location.

This approximation is easily computable but does not ensure the conservation of all the moments of the field. A better approximation could be obtained using the iterative Beale's formula [2], which corrected circulation values in order to recover the vorticity field at the particle locations.

2.3.2 Regression operator

Based on regression methods, the new intensities of the particles defined $\mathbf{U} = [\mathbf{U}_1, \dots, \mathbf{U}_p]^T$ could be obtained by minimizing the quadratic error. Assume that we have some vector $\mathbf{u} = [\mathbf{u}_1(z_1), \dots, \mathbf{u}_p(\mathbf{z}_p)]^T$ of the continuous field evaluations. The particle approximation could be computed on each particle position \mathbf{z}_p given

$$\mathbf{u} \simeq \Phi \mathbf{U},$$

where $\Phi_{ij} = \phi_\varepsilon(z_i - z_j)$.

Finding the new intensities \mathbf{U}^* corresponds to solving the previous system in the least square sense. It corresponds to the classical problem of finding the minimizer of the following quadratic function

$$\mathbf{U}^* = \arg \min_{\mathbf{U}} \|\mathbf{u} - \Phi \mathbf{U}\|_2^2.$$

In this case, the solution is $\mathbf{U}^* = (\Phi^T \Phi)^{-1} \Phi \mathbf{u}$. This problem may be ill-posed, particularly in the case of a large set of non-well-distributed particles. We choose to regularize the solution by introducing a penalization term. The Ridge regression introduces a penalization on of the form $\lambda \|\mathbf{U}\|_2^2$, where λ is a penalization coefficient, such as the new problem is

$$\mathbf{U}_{\text{ridge}}^* = \arg \min_{\mathbf{U}} \|\mathbf{u} - \Phi \mathbf{U}\|_2^2 + \lambda \|\mathbf{U}\|_2^2,$$

given the following solution $\mathbf{U}_{\text{ridge}}^* = (\Phi^T \Phi + \lambda \mathbf{I})^{-1} \Phi \mathbf{u}$.

2.3.3 Remeshing operator

A second type of manipulation is based this time on a complete projection of the solution on a new regular grid of particles [9, 11]. This method allows us to switch from a Lagrangian discretization \mathcal{P} to an Eulerian one Λ and then switch back to an entirely new regular particles discretization \mathcal{P}' that conserves as much as possible the moment of the particle discretization.

In our methodology, we propose a two-step approach. First, we execute an assignment step ((a)) to transfer the particle discretization to the grid discretization. Subsequently, an interpolation step ((b)) is performed to yield a new set of regularly spaced particles.

Our analysis pertains to the one-dimensional spatial scenario, where $\Omega \in \mathbb{R}$. The extension to the n -dimensional case can be achieved through the tensorization of the one-dimensional approach.

(a) Assignment on an Eulerian grid

We denote by z_I and z_p respectively the grid and the old particle locations. The new particles are defined on a grid of n_g elements with regular spacing $\ell_I = 2d_p$ where d_p is the characteristic size of the particles. We define the particle intensities as U_p and the nodal field values as u_I . By using some shape function W , the assignment step from particles to each node $I \in \Lambda$ can be written as

$$u_I = \frac{1}{V_I} \sum_{p \in \mathcal{P}} U_p W \left(\frac{z_I - z_p}{\ell_I} \right).$$

Where W determines a redistribution of the intensity on the grid, the new discretization can then be used to approximate the field u_p , defined by the particle discretization by interpolation given.

$$u_p(z) \approx u_g(z) = \sum_{I \in \Lambda} u_I W \left(\frac{z - z_I}{\ell_I} \right) \quad \forall z \in \Omega.$$

(b) Interpolation on a new regular particle discretization

A new set of particles is defined at the quarter of each cell such that the new position is defined at $z_{p'} = d_p/2 + i d_p$, $i = 0, \dots, 2n_g$. The value of the field is then interpolated at that new location and multiplied with the volume of the particle $U_{p'} = u_g(z_{p'})V_{p'}$ in order to give a new particle approximation of the field.

$$u_g(z) \approx u_{p'}(z) = \sum_{p' \in \mathcal{P}'} u_g(z_{p'}) V_{p'},$$

The combination of these two steps can initially be utilized to generate a new undistorted particle distribution. The shape function W determines the type and quality of the transfer. The method effectiveness is evaluated by assessing the conservation of the first moments of the particle distributions, as detailed in the Appendix B.

For W , one may employ the piecewise linear interpolation function, which ensures the conservation of moment 0. For higher moment conservation, the B-spline function provides a smoothing function for higher order.

However, while higher-order B-splines improve the smoothness of the solution, their accuracy is limited to the second order, allowing only exact interpolation of linear functions.

Monaghan [21] proposes a systematic approach to enhance accuracy and maintain smoothness through extrapolation. The concept involves constructing a new shape function based on a cutoff and its radial derivative. For $m = 4$, the cubic B-spline is improved by the following new interpolating kernel

$$M'_4(z) = \begin{cases} 1 - \frac{5}{2}z^2 + \frac{3}{2}|z|^3 & 0 \leq |z| \leq 1 \\ \frac{1}{2}(2 - |z|)^2(1 - |z|) & 1 \leq |z| \leq 2 \\ 0 & 2 \leq |z|. \end{cases}$$

The drawback of this method is that it does not ensure positivity. Therefore, we opt to utilize the M'_4 kernel for our implementation.

Finally, in multidimensional space, the redistribution kernel W can be obtained as the product of the one-dimensional kernel applied to each coordinate, as follows

$$\begin{aligned} U_p &= \sum_{I \in \Lambda} U_I W(\mathbf{z}_p - \mathbf{z}_I, \ell_I) \\ &= \sum_{I \in \Lambda} U_I \prod_{i=1}^d W_{1D}\left(\frac{\mathbf{z}_{I,i} - \mathbf{z}_{p,i}}{\ell_I}\right) \end{aligned}$$

3 Methods

This section outlines the development of ensemble data assimilation techniques tailored for particle-based simulations. While the forward step aligns with the traditional Ensemble Kalman Filter, the primary challenge lies in the update during the analysis step. In order to be state independent, we define the analysis thanks to the equation (1). This step benefits from an observation matrix-free implementation, rendering the analysis independent of state discretization. Hence, the correction matrix in the analysis \mathbf{F} relies solely on the observation \mathbf{y} , predictive observations $\{\mathbf{h}_i\}_{i=1}^{N_{\text{ens}}}$, and noise samples $\{\boldsymbol{\epsilon}_i\}_{i=1}^{N_{\text{ens}}}$. The analysis fields are obtained at any spacial coordinates thanks to the particle approximation of each member field \mathbf{u}_i^f such as for all $\mathbf{z} \in \Omega$

$$\mathbf{u}_i^a(\mathbf{z}) = \mathbf{u}_i^f(\mathbf{z}) + \sum_j \mathbf{F}_{ji} \mathbf{u}_j^f(\mathbf{z}) \quad i = 1, \dots, N$$

Thus, solutions are also described on a particle discretization $\mathcal{P}_i^a = \bigcup_k \mathcal{P}_k^f$. Such as

$$\mathbf{u}_i^a(\mathbf{z}) = \sum_p \mathbf{U}_{ip}^f \phi_\varepsilon(\mathbf{z} - \mathbf{z}_{ip}) + \sum_j \mathbf{F}_{ij} \sum_p \mathbf{U}_{jp}^f \phi_\varepsilon(\mathbf{z} - \mathbf{z}_{jp}) \quad i = 1, \dots, N \quad (3)$$

However, each assimilation introduce an exponential growth of the number of particle. It could be easily evaluate taking N_{parts} the sum of particles over the ensemble. The first assimilation will create new members of N_{parts} particles. The next assimilation multiply this value by N_{ens} leading to the exponential growth of particles.

To overcome this effect, we introduce two distinct EnKF-adapted filters

- The Remesh-EnKF filter (3.1) employs an intermediate Eulerian discretization of the field, consistent across all members. Consequently, classical EnKF can be applied. Then a regular grid of particles is set to allow a constant number of particles.
- The Particles-EnKF filter (3.2) executes data assimilation with (3). Analyzed fields are approximated on each forward member discretization.

The choice of filter depends on the context, particularly regarding the feasibility of a remeshing process.

3.1 Remesh-EnKF

The first method consists by defining a scheme that combine an intermediate projection on a grid, to perform the assimilation, with a remeshing process to generate a new set of particles. The global scheme is build to conserve as much as possible the property of the particle discretization of the members.

The assimilation is performed with the following step:

- The members are forward, given new particle set $\mathcal{P}_i^f = \{(\mathbf{z}_{ip}^f, \mathbf{U}_{ip}^f)\}_{ip=1}^{N_{ip}}$,
- The associate field is project on a regular grid of n_g elements of characteristic length $\ell_{iI} = 2dp$. Using the assignment operator, we obtain for each node $iI \in \Lambda_i$

$$\mathbf{u}_{iI}^f = \frac{1}{V_{iI}} \sum_{ip \in \mathcal{P}_i^f} \mathbf{U}_{ip}^f \mathbf{W} \left(\frac{\mathbf{z}_{iI} - \mathbf{z}_{ip}^f}{\ell_{iI}} \right)$$

- Based on this new discretization, an Eulerian-based data assimilation could be apply on the nodal state values \mathbf{u}_{iI}^f such as the analysis state \mathbf{u}_{iI}^a is

$$\mathbf{u}_{iI}^a = \mathbf{u}_{iI}^f + \sum_{j=1}^{N_{\text{ens}}} \mathbf{F}_{ji} \mathbf{u}_{jI}^f,$$

- A new regular particle discretization is initialized. Two particles by directions are placed inside each cell of the grid. The new particle intensities could be evaluate thanks to the interpolation operator, such as for $ip' \in \mathcal{P}_i^a$

$$U_{ip'}^a = \sum_{iI \in \Lambda} \mathbf{u}_{iI}^a \left(\frac{\mathbf{z}_{iI} - \mathbf{z}_{ip'}}{\ell_{iI}} \right).$$

The Remesh-Filter update scheme is sum-up in the algorithm 1.

Algorithm 1: Remesh Filter analysis update

Data: $\mathbf{G} \in \mathbb{R}^{n_g \times d}$, $\mathbf{z}^a \in \mathbb{R}^{2n_g \times d}$; // grid discretization
Data: $\mathbf{R} \in \mathbb{R}^m$; // observation covariance
Input: $\mathcal{P}_i^f = \{(\mathbf{z}_{ip}^f, \mathbf{U}_{ip}^f)\}_{ip=1}^{N_{ip}}$, $i = 1, \dots, N_{\text{ens}}$; // forward discretizations
Input: $\mathbf{Y}_f \in \mathbb{R}^{m \times N_{\text{ens}}}$; // the associate observation anomalies
Input: $\mathbf{D} \in \mathbb{R}^{m \times N_{\text{ens}}}$; // the perturbed observations
 $\mathbf{F} = \mathbf{Y}_f^T (\mathbf{Y}_f \mathbf{Y}_f^T + \mathbf{R})^{-1} (\mathbf{D} - \mathcal{Y})$; // correction matrix
foreach $i = 1, \dots, N_{\text{ens}}$ **do**
 $\mathbf{u}[:, i] = \text{Projection}(\mathcal{P}_i^f, \mathbf{G})$
 $\mathbf{u} = \mathbf{u} + \mathbf{uF}$; // analysis update
foreach $i = 1, \dots, N_{\text{ens}}$ **do**
 $\mathbf{z}_{ip}^a, \mathbf{U}_{ip}^a = \text{Assign}(\mathbf{u}[:, i])$
return $\mathcal{P}_i^a = \{\mathbf{z}_{ip}^a, \mathbf{U}_{ip}^a\}_{ip=1}^{N_a}$, $i = 1, \dots, N_{\text{ens}}$; // analyse discretizations

3.2 Particles-EnKF

The goal of the Particles-EnKF formulation is to define the analysis on the particle discretization as much as possible. In this scheme, we keep the particle positions after the forward is unchanged. The change concern the particle intensities. This way, the Lagrangian representation of the solution at the end of the forward step is kept the same as much as possible.

The fields, defined in equation (3), are approximated on the previous discretization such that $Z_i^a = Z_i^f = \{\mathbf{z}\}_{i=1}^{N_i}$ to avoid exponential growth of the number of particles.

By this way, the analyzed field is approximated by $\hat{\mathbf{u}}_i^a$ such as

$$\hat{\mathbf{u}}_i^a(\mathbf{z}) \simeq \sum_p \mathbf{U}_{ip}^a \varphi_{ip}(\mathbf{z})$$

where \mathbf{U}_{ip}^a have been determined by approximation or regression.

However, because this regression is only performed on the support, the current forecast discretization Z_i^f additional particles could be introduced at the support border to allow a better estimate.

The Part-EnKF algorithm is expressed in Algo 2

Algorithm 2: Part-EnKF Filter analysis update

Data: $\mathbf{R} \in \mathbb{R}^m$; // observation covariance
Input: $\mathcal{P}_i^f = \{(\mathbf{z}_{ip}^f, \mathbf{U}_{ip}^f)\}_{ip=1}^{N_{ip}}$, $i = 1, \dots, N_{\text{ens}}$; // forward discretizations
Input: $\mathbf{Y}_f \in \mathbb{R}^{m \times N_{\text{ens}}}$; // the associate observation anomalies
Input: $\mathbf{D} \in \mathbb{R}^{m \times N_{\text{ens}}}$; // the perturbed observations
 $\mathbf{F} = \mathbf{Y}_f^T (\mathbf{Y}_f \mathbf{Y}_f^T + \mathbf{R})^{-1} (\mathbf{D} - \mathcal{Y})$; // correction matrix
foreach $i = 1, \dots, N_{\text{ens}}$ **do**
 $\mathbf{u}_{ip}^a = \text{AnalysisFieldValues}(\mathcal{P}_i^f, \mathbf{F})$; // evaluate the analysis field
 $\mathbf{U}_{ip}^a = \text{Approx}(\mathbf{u}_{ip}^a)$; // approximate the analysis field
return $\mathcal{P}_i^a = \{\mathbf{z}_{ip}^f, \mathbf{U}_{ip}^a\}_{ip=1}^{N_{ip}}$, $i = 1, \dots, N_{\text{ens}}$; // analyse discretizations

4 1D density advection-diffusion problem

4.1 Description of the problem

An initial exploration is conducted on a one-dimensional application to assess the filter performance. We define the following one-dimensional 2π -periodic convection-diffusion problem such as

$$\frac{\partial u}{\partial t}(z, t) + v \frac{\partial u}{\partial z}(z, t) = D \frac{\partial^2 u}{\partial z^2}(z, t),$$

with z the spatial coordinate, v a constant velocity and D a constant diffusion coefficient. For the following application, the reference solution will use $v = 1.0$ and $D = 0.05$ as parameters. We define the 2π -periodic heat kernel in one dimension, such as

$$\phi(u, s) = \sum_{k=-\infty}^{\infty} \frac{1}{\sqrt{4\pi s}} \exp\left(-\frac{(u - 2\pi k)^2}{4s}\right).$$

Considering an initial condition characterized by a Gaussian shape expressed as $u^{gt}(z, 0) = \phi(z - z_0, Dt_0)$, where $z_0 = 0.02$, $t_0 = \frac{\sigma_0^2}{2D}$, and $\sigma_0^2 = 0.5$, we derive the comprehensive analytical solution utilizing the Green equation solution

$$u^{gt}(z, t) = \phi(z - vt - z_0, D(t + t_0)).$$

The analytical solution is succinctly described as a Gaussian function, characterized by a mean that moves at the advection velocity and a standard deviation proportional to t and D . This solution is visually depicted in Figure 1 across various assimilation time frames.

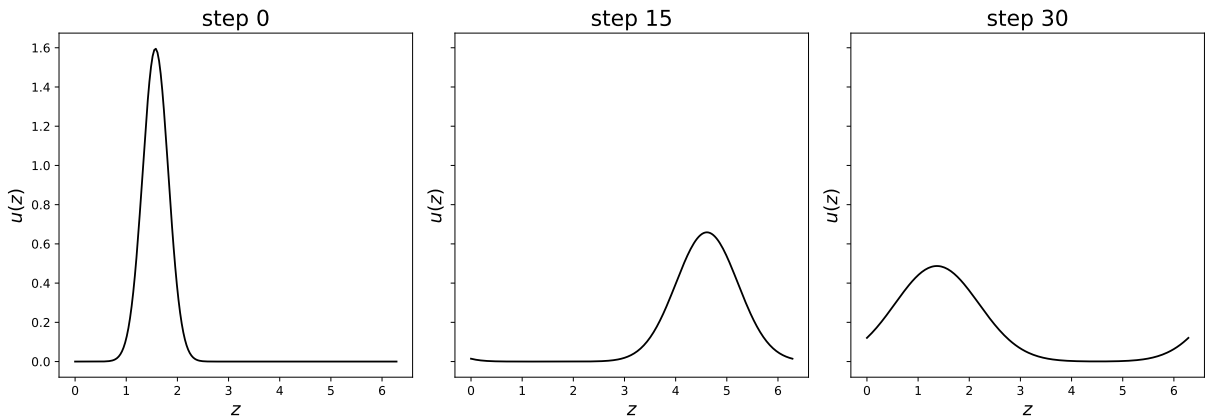


Figure 1: The analytical solution of the convection-diffusion problem evolves over time, with the final snapshot revealing a complete spatial period.

Following a Lagrangian perspective by tracking a fluid particle of position z_p and intensity U_p , the equation becomes

$$\frac{dz_p}{dt} = v(z_p, t), \quad \frac{dU_p}{dt} = D \frac{d^2 U_p}{dz^2}$$

To solve the convection-diffusion scheme, we employ the two steps of the viscous splitting algorithm. The advection is taken into account by updating the position of the particle with an Euler explicit scheme. On the other hand, we use a redistribution method called the Particle Strength Exchange Method (PSE) [13, 10] to approximate the laplacian term $\frac{d^2 U_p}{dz^2}$.

$$D \frac{d^2 U_p}{dz^2} = D \frac{d^2 u_p}{dz^2} V_p \approx D \varepsilon^{-d} \int [u(z) - u(y)] \phi_\varepsilon(y - z) V_p dz,$$

It deals with the particle approximation by a redistribution of the particle intensities in their previous locations, such as

$$\frac{dU_p}{dt} = D \varepsilon^{-d} \sum_q (V_p U_q - V_q U_p) \phi_\varepsilon(z_q - z_p),$$

where V_p the volume of the particle p and d the dimension of Ω . For further details on the computation, please refer to [10].

For the periodic boundary problem described in section 4, we define an equivalent kernel function $\phi_\varepsilon = \phi_g^P = \sum_{n=-\infty}^{+\infty} \phi_g(r - 2\pi n)$.

The particle-based model employs a discretization of $N_{part} = 100$ particles with a size of $h = \frac{L}{N_{part}}$ and a smoothing length of $\varepsilon = 1.3h$. For the sake of comparison, we solve the convection-diffusion equation with an explicit central finite difference scheme discretized on a regular grid with $N_{grid} = 100$ nodes.

4.2 Assimilation parameters and ensemble generation

4.2.1 Ensemble distribution

All filters undergo testing on an identical initial prior ensemble of size $N = 25$ members, characterized by Gaussian shapes that are shifted and scaled. The mean of the function and its standard deviation are sampled. The total mass is set equal to 1. Parameters of velocity v and the diffusion D are also sampled. The different distributions are defined in Appendix 1. Moreover, the parameters samples and initial state are illustrated in Figure 2. The observational data is subject to additive noise, denoted as $\eta \sim \mathcal{N}(0, \sigma_y \mathbf{I})$, where $\sigma_y = 0.05$ and \mathbf{I} represents the identity matrix.

4.2.2 Error definition

We define the error as the following relative ratio

$$e_{L_2} = \frac{\left[\frac{1}{N_{\text{ens}}} \sum_{i=1}^{N_{\text{ens}}} \int_{\Omega} (u_i(z) - u^{gt}(z))^2 dz \right]^{1/2}}{\|u^{gt}\|_{L_2}} \quad (4)$$

where u_i denote the i -th member of the ensemble and $\|u\|_{L_2}$ denote the L_2 norm of u^{gt} . The L_2 norm is computed using a quadrature over a regular grid of an ensemble of cells \mathcal{C} such as for any $f \in L_2$

$$\|f\|_{L_2} = \int_{\Omega} f^2 d\Omega \approx \sum_{c \in \mathcal{C}} f(z) V_c$$

where z_c is the center of the cell c and V_c the volume of the cell. The grid is still the same for all the simulations.

4.2.3 Numerical parameters

We conduct $N_{\text{assim}} = 30$ assimilation steps at evenly spaced intervals until the final time $t_f = 2\frac{L}{v}$. During each assimilation step, the field u^{gt} is observed at six regularly spaced positions x_{obs} .

In the particle-based simulation, fields are discretized using regularly spaced particles that are shifted. Intensity values are obtained by fitting an interpolation operator like in Section 2.3.1 to the particle intensity. The parameter $\varepsilon_{\text{mass}}$ is introduced as a cutoff for particle selection, allowing for the definition of varying numbers of particles for each simulation. The particle support poses challenges in the Part-EnKF as described in Section 3.2.

Simultaneously, a standard Ensemble Kalman Filter (EnKF) update is applied to the nodal variables to construct the reference filter Grid-EnKF, which uses a grid-based model. For grid-based simulation, the fields of each member are interpolated at the node locations. In this way, the ensemble generated is still the same for the sake of comparison.

4.3 Results

We compare the different filters on the assimilation of the state. We first compare the Grid-EnKF, the Remesh EnKF, and two Part-ENKF filters with 100 and 60 particles. The parameter sample is still unchanged. We take unknown parameters into account as model uncertainties. The two filters outlined in the Method section 3 and the Eulerian filter are compared with the reference filter based on a grid discretization.

In figure 4, we appreciate different assimilation steps for the Remesh-EnKF filter.

The result is quite similar for all the different filters except the Part-EnKF with 60 particles. The primary issue arises from the regression on non-overlapping support, where the regression struggles to fit the analysis solution defined on a more considerable space support. This leads to heightened variability, particularly in the tail of the distribution. Addressing this common challenge in RBF Regression [15] involves increasing the Ridge penalization coefficient, a parameter we choose through cross-validation Ridge regression. Even with a more stable regression, it remains a projection of the analysis solution onto the forecast

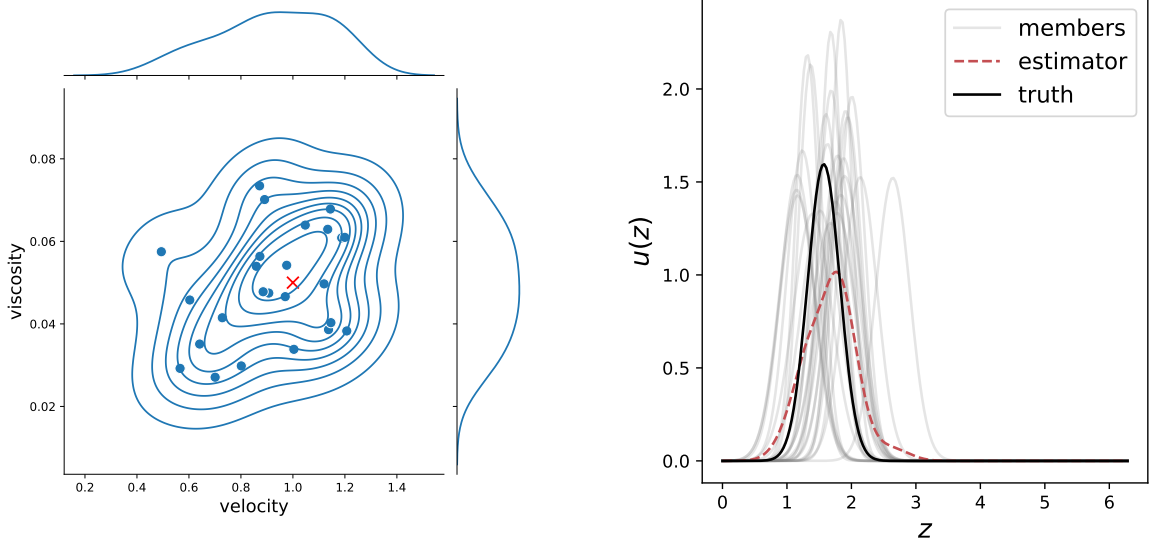


Figure 2: On the left the initial parameters sample, v in abscissa and D in ordinate. On the right is the initial ensemble state.

support. It is imperative to increase the number of particles to achieve a better approximation of the analysis solution using the particle approximation operator in Section 2.3.1 or the regression operator in Section 2.3.2.

We validate this assumption by varying the initial support of particles. Quantitatively, as observed in Figure 5, the error estimate and dispersion decrease with an increase in the number of particles. At the level of 70 particles support, a threshold is reached. Moreover, qualitatively examining the snapshot on the right reveals that the solution closely aligns with the reference.

However, adding particles to a more complex solution is challenging. Indeed, good spacing between particles and particle density has to be preserved. In this case, we advise defining

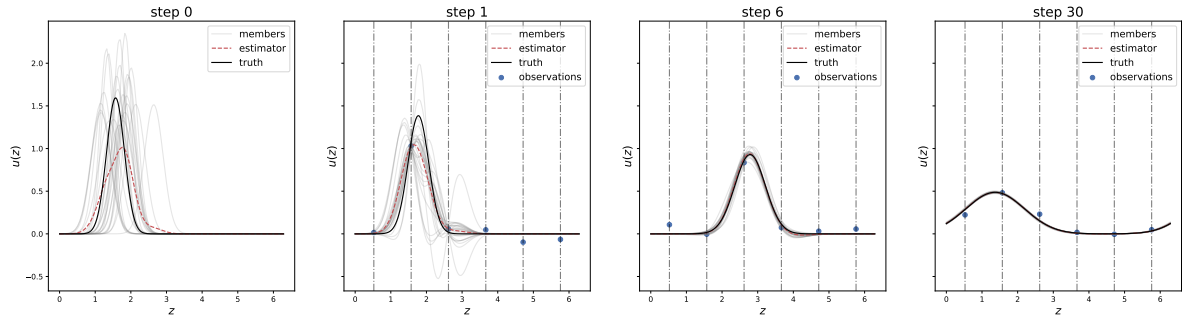


Figure 3: Data assimilation over assimilation step for the Remesh-EnKF filter.

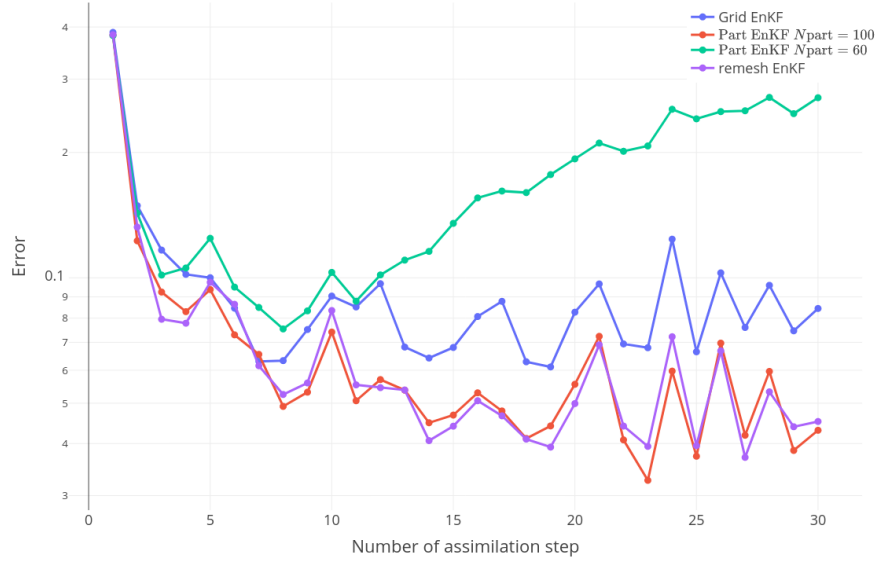


Figure 4: State error with respect to assimilation time step.

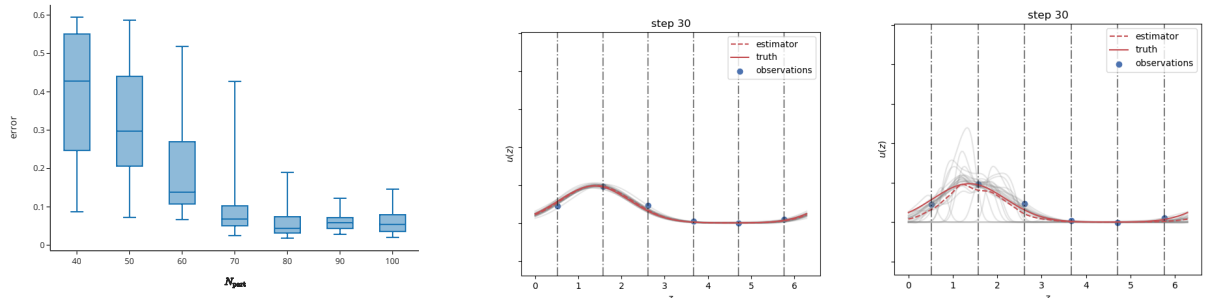


Figure 5: Left: Error with respect to particle support size, Middle: Final step for a support of 100 particles, Right: Final step for a support of 60 particles.

criteria for the reconstruction' error. Instead of adding particles, we advise generating a new, regularly spaced grid of particles to reconstruct the solution.

In conclusion, this example underscores the Remesh-EnKF filter capability to yield results comparable to the classical EnKF applied to a grid model. Additionally, it highlights the Part-EnKF capability in assimilating on a particle discretization while also emphasizing the importance of addressing spatial discrepancies between members, which can pose challenges in solution reconstruction. The computation of solution error reconstruction provides a straightforward criterion for remeshing a member and applying the analysis solution approximation.

5 2D vortex-in-cell problem

5.1 Description of the method

In this section, we apply the Vortex Method in a two-dimensional scenario, as outlined by Cottet et al. [9]. The Vortex Method is a Lagrangian approach utilizing a particle ensemble to discretize the vorticity field, allowing for the solution of the Navier-Stokes equation for viscous incompressible flow. The method is grounded in the vorticity-velocity formulation of the Euler equation, where $\boldsymbol{\omega} = \nabla \times \mathbf{v}$ satisfies

$$\begin{aligned}\frac{\partial \boldsymbol{\omega}}{\partial t} + (\mathbf{v} \cdot \nabla) \boldsymbol{\omega} - \nu \Delta \boldsymbol{\omega} &= 0, \\ \nabla \cdot \mathbf{v} &= 0,\end{aligned}$$

where ω denotes vorticity, \mathbf{v} represents velocity, and ν stands for viscosity.

In the context of 2D flow, vorticity is perpendicular to the flow plane, forming a scalar field denoted as ω . In Cartesian coordinates, it is expressed as $\omega = \frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y}$. The vorticity field is discretized using a collection of discrete vortices, each characterized by a position \mathbf{z}_p , an associated kernel ϕ_ε , and a circulation Γ_p . For all points \mathbf{z} within the domain Ω , the vorticity is expressed as

$$\omega(\mathbf{z}) = \sum_{i=1}^{N_p} \Gamma_p \phi_\varepsilon(\mathbf{z} - \mathbf{z}_p).$$

To address the Navier-Stokes equation, we employ a viscous splitting scheme, following the methodology outlined in [10], acknowledging the predominance of the convection term over viscosity. We use the Vortex-In-Cell algorithm [8, 3], coupled with an FFT solver to compute the advection velocity. The subsequent steps involve assigning particle vorticity values to the grid using a particle-to-grid formula and computing the velocity field by solving the Poisson equation on the grid verified by the stream function. Finally, the velocity is interpolated back onto the particles using the grid-to-particles formula. A Runge-Kutta 3 time-stepping scheme is employed to update the particle positions through a time integration scheme. The final phase involves solving the heat equation and updating the particle intensities thanks to the PSE method previously described in Section 4.

5.2 Lamb-Chaplygin dipole and simulation parameters

We define the reference as the advection of the Lamb-Chaplygin dipole inside a close domain with stress-free walls. Lamb-Chaplygin dipole is a popular choice for numerical studies [22]. The model represents a specific steady, inviscid dipolar vortex flow and offers a non-trivial solution to the two-dimensional Euler equations. The dipole is characterized by a translation velocity U , a mean position \mathbf{z}_0 , a radius R , and an orientation α .

The dipole vorticity field ω could be expressed as

$$\omega(r) = \begin{cases} \frac{-2kUJ_1(kr)}{J_0(kR)} \sin \alpha & \text{for } r < R, \\ 0 & \text{otherwise,} \end{cases}$$

where (r, α) represent the polar coordinates in the dipole reference frame. Here, J_0 and J_1 denote the zeroth and first-order Bessel functions of the first kind, respectively, and k is determined such that kR corresponds to the first non-trivial zero of the first Bessel function. The dipole vorticity field is depicted in Figure 6.

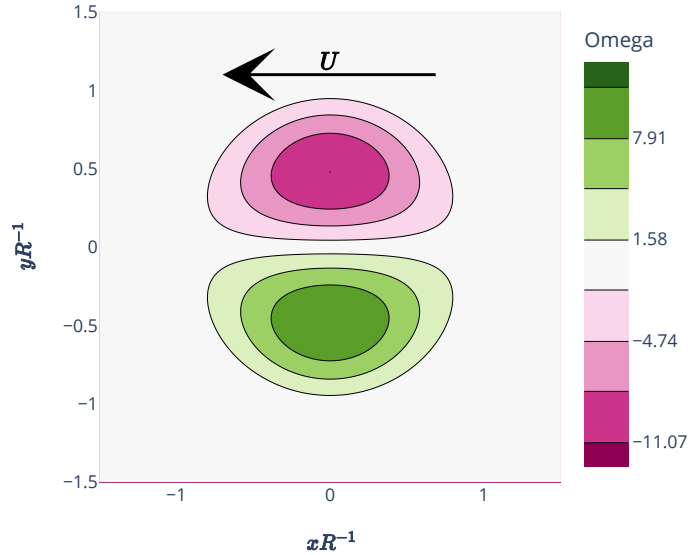


Figure 6: The Lamb-Chaplygin dipole vorticity field on a normalized space.

The dipole is positioned at the center of a box with dimensions $[0, \pi] \times [0, \pi]$, featuring an orientation of $\frac{7\pi}{8}$ rad., a radius of 0.5 meters, and a velocity U of 0.25 m.s^{-1} . The complete reference setting is listed in Table 2.

The boundary box features stress-free walls, meaning fluid cannot pass through them. The velocity perpendicular to the walls is zero, while tangential velocity remains undetermined. When a vortex, such as a dipole, reaches this boundary, it walks along the wall, sensing its reflection and interacting with it.

Because this problem does not have an explicit solution on a closed domain, we simulate the ground truth with the vortex method for a fined discretization and fixed set of parameters also described in Table 2. The trajectory of the ground truth is illustrated in Figure 7 on a regularly spaced grid.

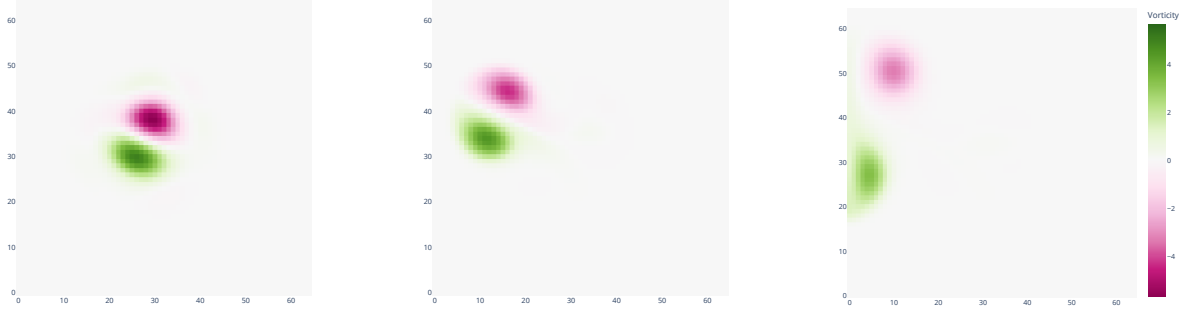


Figure 7: Trajectory of the ground truth. The vorticity is represented on a regularly spaced grid. For $t = [1, 5, 10]s$.

Several parameters in the simulation influence the particle distribution and can lead to different results. The first one is the particle size defined by d_p . Another significant parameter is ε_ω , associated with the remeshing process occurring either during the forecast (to prevent high distortion of the particle distribution) or during the Remesh-EnKF filter. ε_ω serves as a threshold, determining whether a particle is retained after the remeshing process based on the condition $V_p \Gamma_p > \varepsilon_\omega$. The impact of this parameter is illustrated for one member after the first forward in Figure 8.

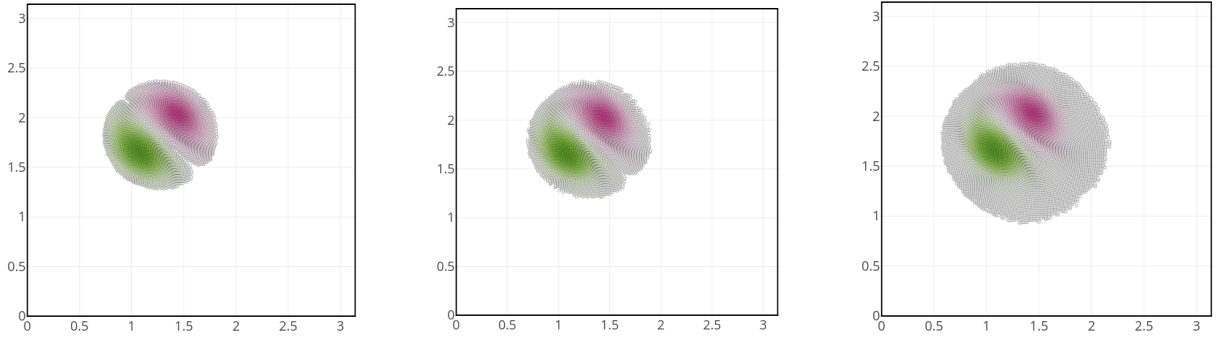


Figure 8: Effect of the parameter ε_ω on the particle discretization of the solution for one member. From left to right, results for $\varepsilon_\omega = 0.1, 0.01$, and $1.e^{-6}$.

For the following paragraphs, if the value is not explicitly changed, we use the nominal parameters described in Table 3 for the simulation.

5.3 Assimilation parameters and ensemble generation

5.3.1 Ensemble distribution

An ensemble of 32 members is created by sampling distributions over the dipole parameters. We sample the radius R , the prescribed velocity U , the orientation α , and the

barycenter \mathbf{z}_{mean} . Additionally, the model viscosity ν is also sampled. All the distributions are summarized in Table 4. The first six members are plotted in Figure 9.

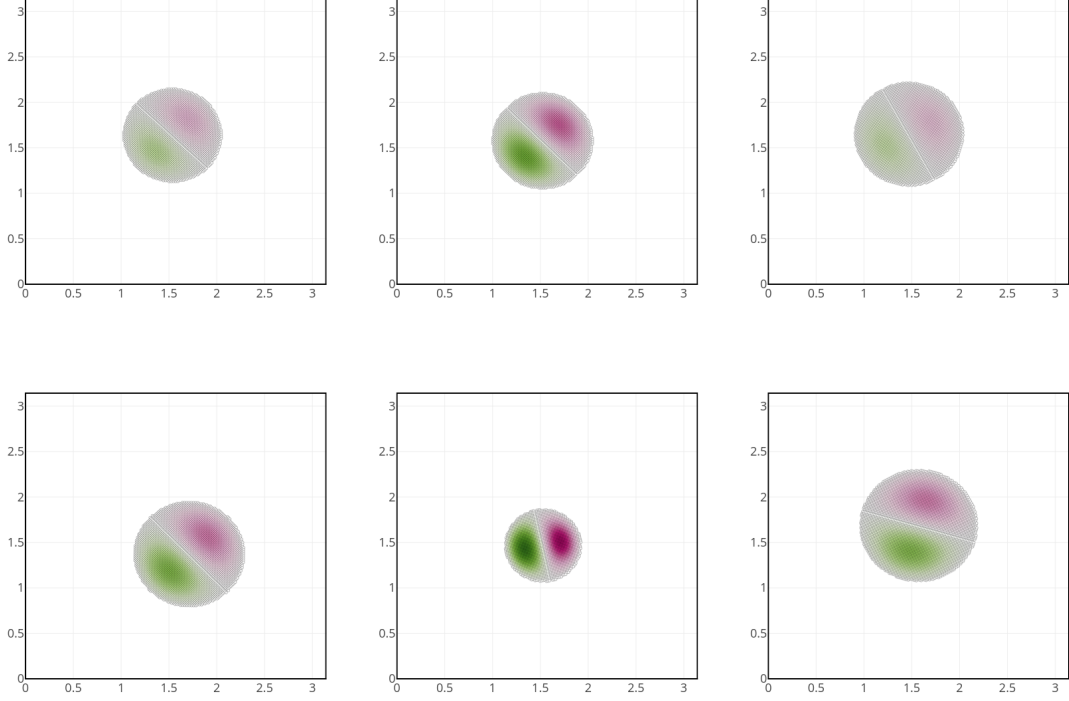


Figure 9: Six samples from the initial ensemble.

The initial vorticity field is first discretized on a regular grid of particles with a characteristic length d_p , where each particle receives the circulation $\Gamma_p = \omega(\mathbf{z}_p)V_p$ and $V_p = d_p^2$ represents the volume of the particle.

5.3.2 Error definition

We use an absolute error absolute L_2 -error defined as $\frac{1}{N_{\text{ens}}} \sum_{i=1}^{N_{\text{ens}}} \int_{\Omega} (\omega_i(\mathbf{z}) - \omega^{gt}(\mathbf{z}))^2 d\mathbf{z}$. We also use the member errors to evaluate the dispersion of the error estimate.

5.3.3 Numerical parameters

The assimilation frequency is defined by the assimilation step dt_a . The simulation is performed over a duration of t_f . All simulation parameters are summarized in Table 3. Observations are collected on a regular grid of size N_{obs} , measuring both components of the velocity. The observations follow a normal distribution $\mathcal{N}(0, \sigma_{\text{obs}}^2 \mathbf{I})$, indicating an ensemble of independent measurements, each characterized by a standard distribution of σ_{obs} . An example of observed velocity with and without noise is illustrated in Figure 10.

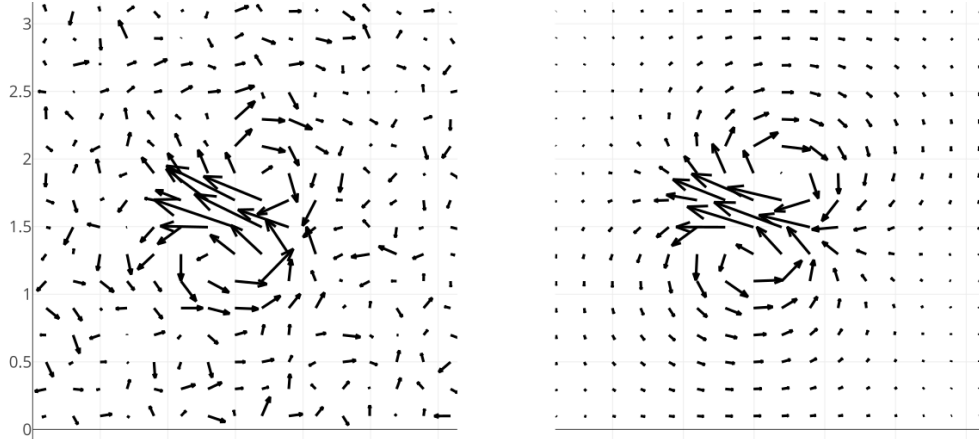


Figure 10: Observed and reference velocity fields. The error on each component is a sample from a centered normal distribution with the nominal value $\sigma_{\text{obs}} = 0.05$.

5.4 Results

5.4.1 Error through time

We start by analyzing the assimilation error over time. Figure 11 illustrates the error throughout the assimilation process for the nominal set of assimilation parameters, demonstrating comparable results for both filters. At each assimilation step, the error decreases and prevents the solution from diverging elsewhere.

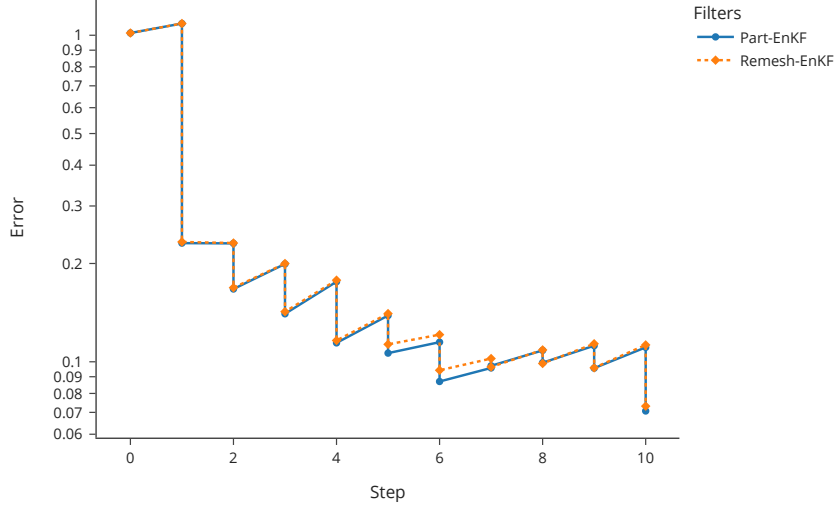


Figure 11: error curves through assimilation steps. Left: L_2 -error of the field, Right: Error for the viscosity parameter. With Part-EnKF in blue and Remesh-EnKF in red.

5.4.2 Error with respect to assimilation parameters

We also assess the performances of the different filters by evaluating the convergence of the error with respect to the assimilation parameters.

We observe the convergence rate concerning data assimilation parameters: The observation precision, which is $1/\sigma_{\text{obs}}^2$, the number of observations N_{obs} , the number of assimilation step N_{assim} .

Figure 12a illustrates a decreasing error bias and variances with respect to observation precision similarly for both filters. What is striking in the same Figure 12b but in the log-scale is the regular convergence rate for both filters with respect to the observation precision. The order of convergence is about 0.68 for Part-EnKF and 0.75 for Remesh-EnKF.

In Figure 13a, the reduction of error is still prominent and shows a reduction of variance as the number of observations increases. In the log-scale Figure 13b, the error decrease also at a constant rate for both filters. We notice a more substantial order of 1.8 for the Remesh-EnKF compared to an order of 1.4 for Part-EnKF.

Finally, we analyze the error convergence with respect to the number of observations. Observation locations increase regularly on both axes. In Figure 14a, the error estimate and variances decrease. For a relatively small number of observations, the two filters offer similar results when, in contrast, the Remesh-EnKF has relatively better results when the number of observations is increased. Moreover, the convergence rate seems to change around 200 observation points, as illustrated in the log-scale Figure 14b. Nevertheless, it illustrates adequate performances for both filters.

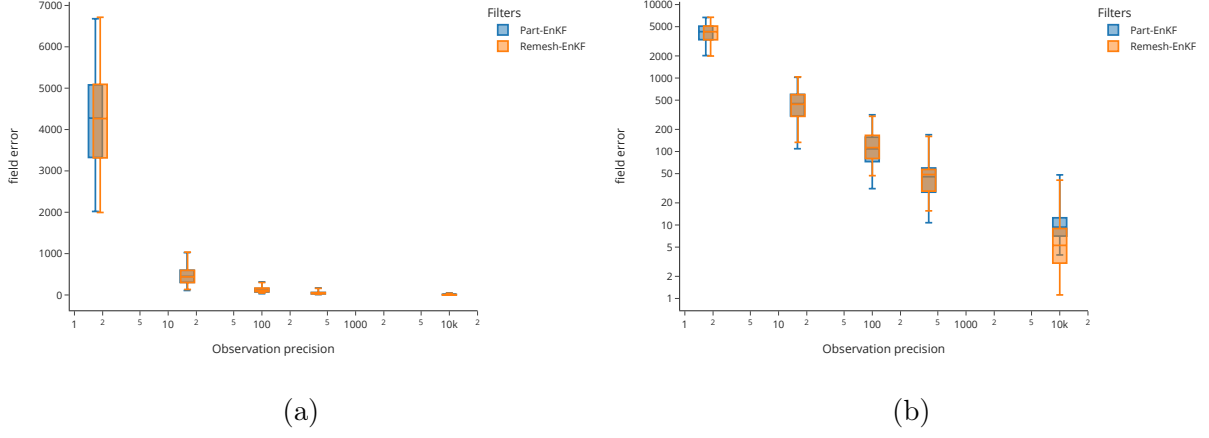


Figure 12: Box plots of the state error w.r.t. $1/\sigma_{\text{obs}}^2$.

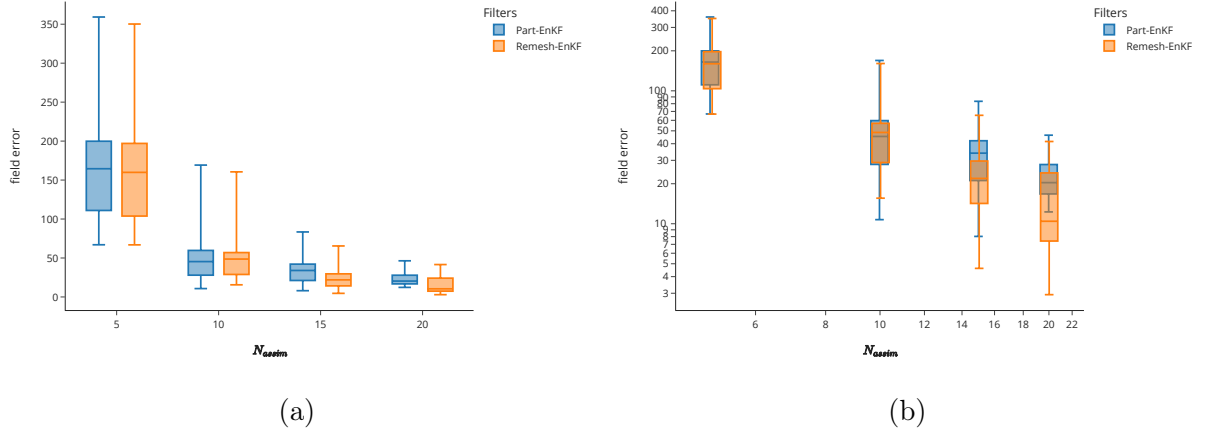
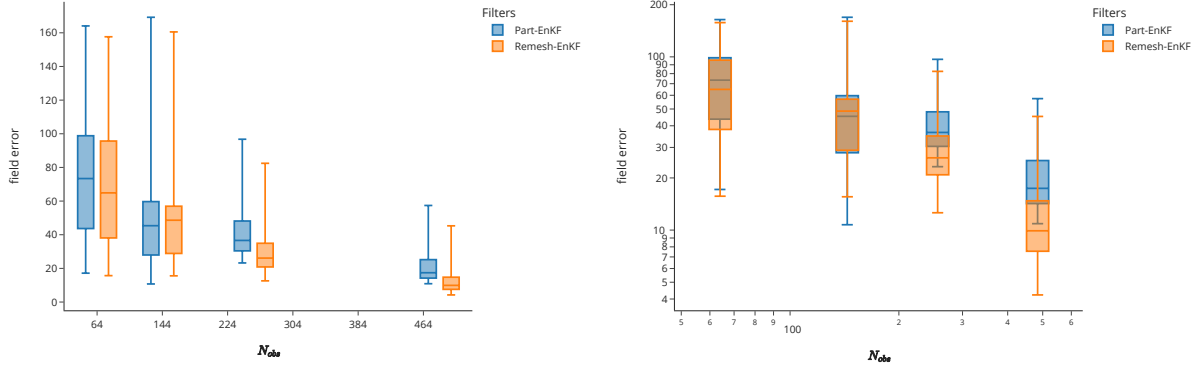


Figure 13: Box plots of the state error w.r.t. N_{assim} .

5.4.3 Error with respect to simulation parameters

To better understand the differences, let us now turn to the evolution of the error with respect to particle discretization parameters. For the Part-EnKF, remember that each member has its own particle discretization that flows according to the dipole direction and velocity. Each analyzed member's solutions are then respectively projected on their member discretization. However, this scheme could introduce different sources of error. First, due to particle irregularity in the particle distribution, severe approximation was introduced, which led to errors between the analyzed and the approximated solution. Even more seriously, certain parts of the solution may vanish as no particle in the support can interpolate it. This effect could be appreciated on several samples of the ensemble where the analysis is projected on a non-conforming particle discretization. For instance, we analyzed



(a) Box plots of the state error w.r.t. N_{obs} . (b) Box plots of the state error w.r.t. N_{obs} .

the first assimilation step of one member for the different filters. If the analyzed field is known over the space domain, we observed in Figure 15 that the Remesh-Filter is only able to interpolate the entire solution. The Part-EnKF is not entirely able to interpolate the solution with the forecast member discretization. Moreover, some distortions observed in the particle distribution are not in line with the analyzed field flow. These remarks are more critical when the forecast step is longer, leading to high errors, or when the size of the support is lower. Moreover, the approximation of Section 2.3.2 will introduce the approximation error function of the size of the particles. For instance, the particle will not conserve the total circulation due to a quadrature error, which is the opposite case for the Remesh-EnKF filter.

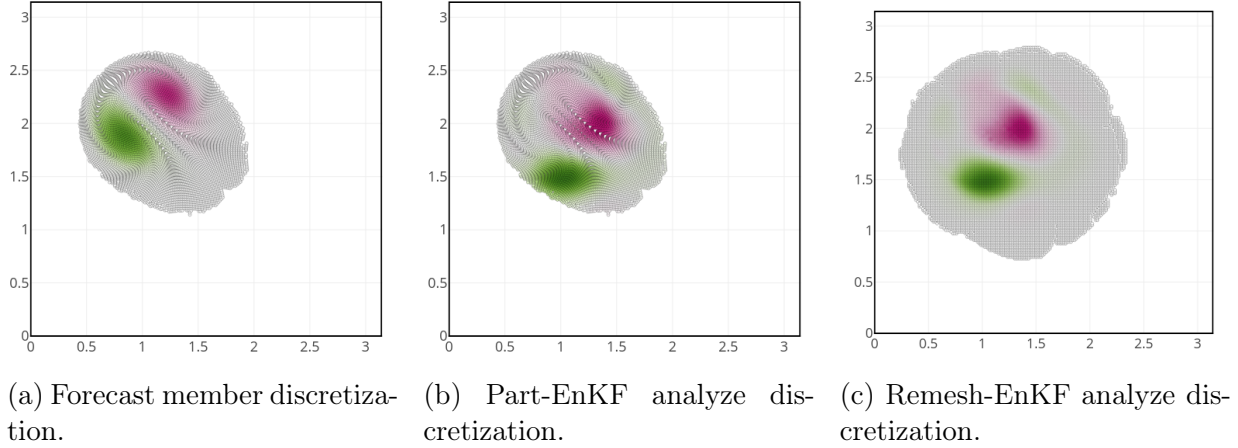
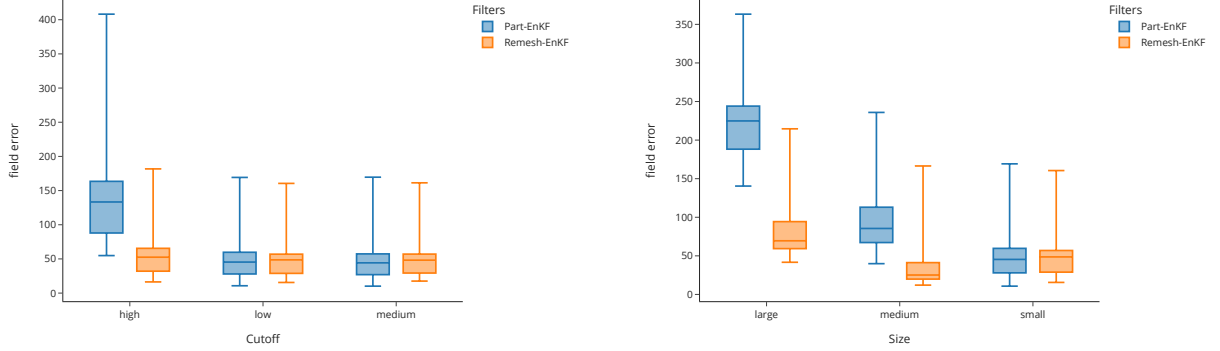


Figure 15: Assimilation of one member with a forecast discretization unadapted to the analyses solution. The forecast discretization used by the Part-EnKF does not always support approximation for the analyses and introduces discretization errors.

To evaluate the effect of the size of the support, we varied the value of ε_ω . We have seen in Figure 8 that this parameter affects the number of particles and, thus, the size of the

support. In Figure 16a, we observe high disparities between the two filters. However, the error stabilizes rapidly by decreasing the threshold. These findings suggest an impact on the threshold and, thus, the particle support for the Part-EnKF. By contrast, Remesh-EnKF is less sensible. As a consequence, it only thresholds low values in the analyzed solutions.



(a) state error w.r.t. ε_ω . The High, low, and medium cutoff correspond respectively to $\varepsilon_\omega = 0.1, 1.e^{-6}$ and 0.01 .

(b) state error w.r.t. d_p . The large, medium, and low sizes correspond to $d_p = 0.0327, 0.0245$, and 0.0123 .

Figure 16: Error box plots for simulation parameters. The effect of ε_ω on the error is particularly observed on high value. The Part-EnKF error is strongly linked to d_p through the particle approximation error.

In terms of particle size d_p , we evaluate its effect on the assimilation. In this section, we have used the regression operator defined in Section 2.3.2 for Part-EnKF. This choice has been motivated to accelerate the update process. in Figure 16b. We observe that the error for Part-EnKF increases with d_p . In fact, it increases proportionally as for the approximation error. This high error confirmed, in that case, the high effect of the particle approximation for Part-EnKF. In contrast, the Remesh-EnKF error is relatively low, taking advantage of a high-order projection interpolation scheme. Using a regression operator to approximate the analyzed solution should alleviate this effect provided other particle discretization considerations (distortion, support size) as illustrated in Part 4 and the choice of an adequate penalty coefficient to succeed in approximating the solution. This discussion pointed out the high dependency of the Part-EnKF on particle discretization. The particle discretization of a member may be too far from the solution support of particles. This opens the question of the choice or modification of the particle discretization. As suggested in Part 4, an error estimation could be introduced to choose between the different filters. On the other hand, other approaches could be to select the member with the maximum likelihood estimate to approximate the solution. This proposition has to be evaluated because it could considerably reduce the variance of the ensemble. Finally, an alignment of particles might be introduced to better fit the analyzed solution.

6 Conclusion

In this study, we introduce a novel framework that combines a sequential ensemble data assimilation approach with particle-based models. Specifically, we have developed two novel Ensemble Kalman Filter schemes dedicated to meshfree simulations. These formulations depend either on updating the particle quantities or on remeshing the particle discretization on a common ensemble grid.

We proposed two update strategies relying on an update matrix that does not directly depend on the state discretization. The first strategy, Remesh-EnKF, is based on the projection of every member onto a new common discretization of particles. The second strategy, Part-EnKF, evaluates the analysis field at particle locations to update the particle quantities.

The different classes have been initially tested on a one-dimensional example and compared with an Eulerian representation of the solution. The results demonstrate comparable performance across the various filters, with the exception of certain configurations of the Part-EnKF. It has been demonstrated that provided the support of the particles is consistent with the analysis solution, the filters yield similar results. However, in cases where the support deviates from the analysis field, members diverge. Increasing the support for the solution is necessary.

A two-dimensional case has also been tested, particularly to assess nonlinear advection schemes with various configurations. We observe good agreement among the different filters. However, this time, the particle approximation is predominant in the Part-EnKF. Both strategies offer several derivations. Remesh-EnKF is mainly dependent on the redistribution kernel to obtain the new regularly spaced particle set. Part-EnKF could be extended because it is highly flexible when defining the new set of particles for each member. This tuning is essential, as we have highlighted the issue of the non-conforming support of the forward particle position with the analysis.

Several methodologies to introduce new particles or change the previous ones could be explored, particularly at the edge of the distribution. For the sake of simplicity, we advocate generating a new set of particles with a regular spacing for any member encountering difficulties in accurately reconstructing the analysis field. Another alternative is to update the positions of the particles in addition to the intensities instead of only changing the intensities. In this way, the error due to a misfit of alignment or approximation could be avoided. These type of adaptation could be derived from optimal transport scheme [5] or correction of the background ensemble alignment with the observation [23, 24].

A Stochastic Ensemble Kalman Filter

We define the matrix of states and the matrix of anomalies $\mathbf{Z}^f = [\mathbf{x}^1, \dots, \mathbf{x}^N]$, \mathbf{A} whose columns are the member states and the normalized anomalies.

$$\mathbf{A} = \frac{1}{\sqrt{N-1}}(\mathbf{Z}^f - \bar{\mathbf{x}}^f \mathbf{1}^T),$$

where $\mathbf{1} \in \mathbb{R}^N$ is a vector of one.

Respectively the matrix of observation and observation anomalies are $\mathcal{Y}^f = [\mathcal{H}(\mathbf{x}_1^f), \dots, \mathcal{H}(\mathbf{x}_N^f)]$, \mathbf{Y} where columns are

$$\mathbf{Y} = \frac{1}{\sqrt{N-1}}(\mathcal{Y} - \bar{\mathbf{y}} \mathbf{1}^T) \quad \text{with} \quad \bar{\mathbf{y}} = \frac{1}{N} \sum_{j=1}^N \mathcal{H}(\mathbf{x}_j^f).$$

The ensemble defines the covariance between states and observations $\mathbf{P}\mathbf{H}^T$, the covariance between observations $\mathbf{P}\mathbf{H}^T$, and $\tilde{\mathbf{K}}$

$$\begin{aligned} \mathbf{P}\mathbf{H}^T &\simeq \frac{1}{N-1} \sum_{i=1}^N (\mathbf{x}_i^f - \bar{\mathbf{x}}^f) [\mathcal{H}_k(\mathbf{x}_i^f) - \bar{\mathbf{y}}]^T = \mathbf{A}\mathbf{Y}^T, \\ \mathbf{H}\mathbf{P}\mathbf{H}^T &\simeq \frac{1}{N-1} \sum_{i=1}^N [\mathcal{H}_k(\mathbf{x}_i^f) - \bar{\mathbf{y}}] [\mathcal{H}_k(\mathbf{x}_i^f) - \bar{\mathbf{y}}]^T = \mathbf{Y}\mathbf{Y}^T, \\ \tilde{\mathbf{K}} &= \mathbf{P}\mathbf{H}^T(\mathbf{H}\mathbf{P}\mathbf{H}^T + \mathbf{R})^{-1} = \mathbf{A}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \mathbf{R})^{-1}. \end{aligned}$$

This observation matrix-free implementation rely on the secant method approximation $\mathcal{H}(\mathbf{x}_i^f - \bar{\mathbf{x}}^f) \approx \mathcal{H}(\mathbf{z}_i^f) - \bar{\mathbf{y}}$. The forecast is then update to a posterior ensemble $\{\mathbf{x}_i^a\}_{i=1}^N$ such as

$$\mathbf{Z}^a = \mathbf{Z}^f + \tilde{\mathbf{K}}(\mathbf{D} - \mathcal{Y}), \quad (5)$$

where $[\mathbf{D}]_i = \mathbf{y} + \boldsymbol{\varepsilon}_i$ is the perturbed observation with $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$, $\tilde{\mathbf{K}}$ the ensemble Kalman gain matrix and $(\mathbf{D} - \mathcal{Y})$ the innovation term. The forecast step is then applied to the analyzed ensemble until the following observation. Based on this formulation, we can deduce a correction formula only based on the member's predictions and observations. We can rewrite the classical update formula using the previous anomaly matrices.

$$\mathbf{Z}^a = \mathbf{Z}^f + \mathbf{A}\mathbf{Y}^T(\mathbf{Y}\mathbf{Y}^T + \mathbf{R})^{-1}(\mathbf{D} - \mathcal{Y})$$

We reformulate the correction term by remarking that $\mathbf{1}^T \mathbf{Y}^T = \mathbf{0}$. We define \mathbf{F} , the correction matrix that gives the update in terms of linear combinations of the forward states

$$\mathbf{Z}^a a = \mathbf{Z}^f + \mathbf{Z}^f \mathbf{F}, \quad \mathbf{F} = \frac{1}{\sqrt{N-1}} \mathbf{Y}^T (\mathbf{Y} \mathbf{Y}^T + \mathbf{R})^{-1} (\mathbf{D} - \mathcal{Y}).$$

using the Sherman-Morrison-Woodbury formula we obtain

$$\mathbf{F} = \frac{1}{\sqrt{N-1}} (\mathbf{I}_N + \mathbf{Y}^T \mathbf{R}^{-1} \mathbf{Y})^{-1} \mathbf{Y}^T \mathbf{R}^{-1} (\mathbf{D} - \mathcal{Y}).$$

B Moment conservation of particle discretization

The m -th moment of a particle distribution is defined as the quantity $\sum_p z_p^\alpha \mathbf{U}_p$. First, we see that the partition of unity is required

$$\sum_{I \in \Lambda} W \left(\frac{z - z_I}{\ell_I} \right) = 1, \quad z \in \Omega \quad (6)$$

due to the final particle arrangement \mathcal{P}' on a grid of size d_p , it leads to the following property

$$\sum_{p' \in \mathcal{P}'} W \left(\frac{z - z_{p'}}{\ell_I} \right) = \frac{V_I}{V'_p}, \quad z \in \Omega. \quad (7)$$

Attention should be focused on the border. Extending the domain with "ghost" particles or nodes allows for verification of properties within Ω .

This property is the necessary condition for the conservation of the first moment. Primarily for the assignment

$$\begin{aligned} \sum_{I \in \Lambda} \mathbf{u}_I V_I &= \sum_{p \in \Lambda} \mathbf{U}_p W \left(\frac{z_I - z_p}{\ell_I} \right) \\ &= \sum_{p \in \mathcal{P}} \mathbf{U}_p \sum_{I \in \Lambda} W \left(\frac{z_I - z_p}{\ell_I} \right) = \sum_{p \in \mathcal{P}} \mathbf{U}_p. \end{aligned}$$

using the property (6). Secondary, for the interpolation process

$$\begin{aligned} \sum_{p' \in \mathcal{P}'} \mathbf{U}_{p'} &= \sum_{p' \in \mathcal{P}'} \mathbf{u}_g(z_{p'}) V_{p'} = \sum_{p' \in \mathcal{P}'} V_{p'} \sum_{I \in \Lambda} \mathbf{u}_I W \left(\frac{z_{p'} - z_I}{\ell_I} \right) \\ &= \sum_{I \in \Lambda} \mathbf{u}_I V_{p'} \sum_{p' \in \mathcal{P}'} W \left(\frac{z_{p'} - z_I}{\ell_I} \right) \\ &= \sum_{I \in \Lambda} \frac{V_I}{V'_p} V_{p'} \mathbf{u}_I = \sum_{I \in \Lambda} \mathbf{u}_I V_I = \sum_{p \in \mathcal{P}} \mathbf{U}_p \end{aligned} \quad ,$$

using equation (7).

It can be shown moreover that if for $1 \leq |\alpha| \leq m - 1$, W satisfies,

$$\sum_{I \in \Lambda} (z - z_I)^\alpha W \left(\frac{z - z_I}{\ell_I} \right) = 0, \quad (8)$$

The regridding procedure will be ordered at m . Equivalently, the previous equality lead, for $0 \leq |\alpha| \leq m - 1$, to

$$\sum_{I \in \Lambda} z_I^\alpha W \left(\frac{z_p - z_I}{\ell_I} \right) = z^\alpha,$$

obtained by developing $(z - z_q)^\alpha$ and using a recurrence on previous orders. This means that the interpolation is exact for polynomials of degrees less or equal to $m - 1$ or that the moment of order $m - 1$ is conserved.

C Parameters

C.1 One dimension problem

Table 1: Ensemble generation variables

Variables	Distributions
Gaussian mean	$Z_m \sim \mathcal{N}(\pi/2 + 0.6, 0.5)$
Gaussian standard deviation	$S_m \sim \mathcal{U}(0.8, 1.2)$
velocity	$v \sim \mathcal{N}(0.9, 1.2)$
diffusion	$D \sim \mathcal{U}(0.02, 0.08)$

C.2 Two dimension problem

Table 2: Reference parameters

Parameters	Values
reference viscosity	$v_{\text{ref}} = 0.001$
reference orientation	$\theta_{\text{ref}} = \frac{7\pi}{8} (\text{rad.})$
barycenter position	$\mathbf{z}_{\text{ref}} = \left[\frac{\pi}{2}, \frac{\pi}{2} \right]^T$
translation velocity	$U_{\text{ref}} = 0.25$

Table 3: Nominal assimilation and simulation parameters

Parameters	Values
time step	$dt = 0.005$
final time	$t_f = 10$
std. observation	$\sigma_{obs} = 5.0e^{-2}$
vorticity threshold	$\varepsilon_\omega = 1.0e^{-4}$
particle characteristic length	$dp = \frac{pi}{256} \approx 0.01227$
smoothing length	$h = 2.0dp$
number of assimilation	$N_{\text{assim}} = 10$
ensemble size	$N_{\text{ens}} = 32$
number of observation	$N_{\text{obs}} = 12^2 = 144$
grid discretization	$N_{\text{grid}} = 65^2 = 4225$
number of remeshing by forecast	$N_{\text{remesh}} = 2$

Table 4: Ensemble generation variables

Variables	Distributions
radius	$R \sim \mathcal{N}(1.0, 0.05^2)$
orientation	$\theta \sim \mathcal{U}\left(\pi, \frac{\pi}{2}\right)$ (rad.)
barycenter	$z_{\text{mean},x} \sim \mathcal{N}\left(\frac{\pi}{2}, 0.1^2\right), \quad z_{\text{mean},y} \sim \mathcal{N}\left(\frac{\pi}{2}, 0.1^2\right)$
velocity	$U \sim \mathcal{U}(0, 0.5^2)$
viscosity	$v \sim \mathcal{N}(0.0015, 0.0005^2)$

References

- [1] L. A. Barba. *Vortex Method for Computing High-Reynolds Number Flows: Increased Accuracy with a Fully Mesh-Less Formulation*. PhD thesis, California Institute of Technology, May 2004.
- [2] J. T. Beale. On the Accuracy of Vortex Methods at Large Times. In Bjorn Engquist, Andrew Majda, and Mitchell Luskin, editors, *Computational Fluid Dynamics and Reacting Gas Flows*, The IMA Volumes in Mathematics and Its Applications, pages 19–32, New York, NY, 1988. Springer.
- [3] C. K. Birdsall and D. Fuss. Clouds-in-clouds, clouds-in-cells physics for many-body plasma simulation. *Journal of Computational Physics*, 3(4):494–511, 1969.
- [4] M. Bocquet. Introduction to the principles and methods of data assimilation in the geosciences. page 89, 2014.
- [5] Marc Bocquet, Pierre Vanderbeeken, Alban Farchi, Joffrey Dumont Le Brazidec, and Yelva Roustan. *Bridging classical data assimilation and optimal transport*. December 2023.
- [6] B. Bonan, N. Nichols, M. Baines, and D. Partridge. Data assimilation for moving mesh methods with an application to ice sheet modelling. *Nonlinear Processes in Geophysics*, 24:515–534, September 2017.
- [7] J. U. Brackbill, D. B. Kothe, and H. M. Ruppel. Flip: A low-dissipation, particle-in-cell method for fluid flow. *Computer Physics Communications*, 48(1):25–38, January 1988.
- [8] I. P. Christiansen. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics*, 13(3):363–379, 1973.
- [9] G.-H Cottet and P. Koumoutsakos. *Vortex methods - theory and practice*. March 2000.
- [10] G.-H. Cottet and S. Mas-Gallic. A particle method to solve the Navier-Stokes system. *Numerische Mathematik*, 57(1):805–827, December 1990.
- [11] G.-H. Cottet, M.-L. Ould Salihi, and M. El Hamroui. Multi-purpose regridding in vortex methods. *ESAIM: Proceedings*, 7:94–103, 1999.
- [12] D. Darakananda, A. F. De Castro da Silva, T. Colonius, and J. Eldredge. Data-assimilated low-order vortex modeling of separated flows. *Physical Review Fluids*, November 2018.
- [13] P. Degond and S. Mas-Gallic. The weighted particle method for convection-diffusion equations. part 1: The case of an isotropic viscosity. *Mathematics of Computation*, 53(188):485–507, 1989.

- [14] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, 1994.
- [15] B. Fornberg and N. Flyer. Solving pdes with radial basis functions. *Acta Numerica*, 24:215–258, 2015.
- [16] R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181(3):375–389, 12 1977.
- [17] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960.
- [18] M. Le Provost, R. Baptista, Y. Marzouk, and J. Eldredge. *A low-rank nonlinear ensemble filter for vortex models of aerodynamic flows*. January 2021.
- [19] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82:1013–1024, 1977.
- [20] C. Mimeau and I. Mortazavi. A Review of Vortex Methods and Their Applications: From Creation to Recent Advances. *Fluids*, 6(2):68, February 2021.
- [21] J. J. Monaghan. Extrapolating B splines for interpolation. *Journal of Computational Physics*, 60(2):253–262, September 1985.
- [22] P. Orlandi. Vortex dipole rebound from a wall. *Physics of Fluids A: Fluid Dynamics*, 2(8):1429–1436, August 1990.
- [23] Sai Ravela, Kerry Emanuel, and Dennis McLaughlin. Data assimilation by field alignment. *Physica D: Nonlinear Phenomena*, 230(1-2):127–145, June 2007.
- [24] W. Steven Rosenthal, Shankar Venkataramani, Arthur J. Mariano, and Juan M. Restrepo. Displacement data assimilation. *Journal of Computational Physics*, 330:594–614, February 2017.
- [25] G. Russo and J. A. Strain. Fast triangulated vortex methods for the 2d euler equations. *Journal of Computational Physics*, 111:291–323, 1994.
- [26] Y. Sasaki. An ObJective Analysis Based on the Variational Method. *Journal of the Meteorological Society of Japan. Ser. II*, 36(3):77–88, 1958.
- [27] A. Siripatana, L. Giraldi, O. P. Le Maître, O. M. Knio, and I. Hoteit. Combining ensemble Kalman filter and multiresolution analysis for efficient assimilation into adaptive mesh models. *Computational Geosciences*, 23(6):1259–1276, December 2019.
- [28] P. Sperotto, S. Pieraccini, and M. A. Mendez. A meshless method to compute pressure fields from image velocimetry. *Measurement Science and Technology*, 33, 2021.

- [29] D. Sulsky, Z. Chenb, and H.-L. Schreyer. A particle method for history-dependent materials. page 18, 1994.
- [30] O. Talagrand. Assimilation of observations, an introduction. *Journal of the Meteorological Society of Japan*, 75(1B):191–209, 1997.
- [31] Yonghao Y., B. Smith, C. Batty, C. Zheng, and E. Grinspun. Continuum Foam: A Material Point Method for Shear-Dependent Flows. *ACM Transactions on Graphics*, 34(5):1–20, November 2015.

./biblio