Ensemble Data Assimilation for Meshless Methods

Marius Duvillard[1,2], Loïc Giraldi[1], and Olivier Le Maître[3]

[1]CEA, DES, IRESNE, DEC, SESC, LMCP, Cadarache, F-13108 Saint-Paul-Lez-Durance, France
[3]CNRS, Inria, Centre de Mathématiques Appliquées, Ecole Polytechnique, IPP, Route de Saclay, 91128, Palaiseau Cedex, France
[2]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 meshless simulations using the Ensemble Kalman Filter. If data assimilation methods have been apply on Eulerian simulations for long, there 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 combine a remeshing process to reduce to previous scheme. The second is a purely Lagrangian scheme useful when remeshing is not adapted. These methods are evaluated using a one-dimensional advection-diffusion model with periodic boundaries. Subsequently, assimilation schere are applied to a more complex two-dimensional inviscid 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: Meshless Methods, Particle-based Method, Data Assimilation, EnKF, Ensemble Methods, Vortex Methods.

Contents

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 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 particularly focused on the Vortex Method [7, 18], which discretize the vorticity field and solve 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's 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 [22] focus on optimizing a cost function that measures the misfit between model predictions and observations, seeking the most plausible estimate of the system state. The most commonly formulations derive from 3D-VAR, 4D-VAR [26]. On the other hand, stochastic approaches go beyond mere state estimation; they delve into the quantification of uncertainty associated with the estimated states. This is a critical aspect, especially in dynamic and uncertain systems, where acknowledging and characterizing uncertainty becomes paramount for reliable decision-making and model improvement. This is particularly crucial in dynamic and uncertain systems, where acknowledging and characterizing uncertainty becomes paramount for reliable decision-making and model improvement. In this approach, 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 [15] 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 [12], 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 in approximating the probability distribution of a state thanks to an ensemble of simulations called particles or members.

The goal of this paper is to introduce new approaches to apply 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. Common 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 [27, 9].
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 [23, 5]. The selection of the reference and updated grids provides a spectrum of implementation possibilities. Furthermore, Siripatana et al. [23] elucidate 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. [10]. Complex filters have been developed to estimate correctly the posterior discretization based on a nonlinear observation model or a deficient number of pressure sensors [16]. However, these methods grapple with scenarios involving markedly divergent particle discretizations or highly variant model flows. In this general case, using a particle state using 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 modification 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 [18]. It mainly involved iterative methods [2], triangulation [21] or Radial Basis Function (RBF) interpolation [1, 24]. RBF offer to introduce new particles or introduce penalization to regularize optimization problems.

Based on those different formulations, we propose two novel EnKF-based filters. First, the Remesh-EnKF uses an Eulerian intermediate update. This way, the analysis is define on an Eulerian framework and the method reduced to previous developement of data assimilation. Nevertheless, it involves to remesh completely the discretization on a regular grid of particle. This method is based on the regridding of the particle discretization as described by [9]. 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 analysed 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 formulate with a probabilistic framework. It allows 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, including future observations.

2.1.1 Data assimilation setting

This recurrence is modeled by the use of a hidden Markov chain model. We position ourselves within a finite-dimensional context. The state $\boldsymbol{x}_k \in \mathbb{R}^d, \boldsymbol{y}_k \in \mathbb{R}^m$ where d and m are the state and observation dimension. The forecast and observation are introduced such as for $k \geq 0$,

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

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 $\boldsymbol{x}_k \in \mathbb{R}^n$ is the vector state at time k and $\boldsymbol{y}_k \in \mathbb{R}^m$ 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 $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$ respectively. Finally, we assume that the observation and the model errors are independent though the time and that initial error on $\boldsymbol{x}_0$, $\boldsymbol{\varepsilon}_k$ and $\boldsymbol{\eta}_k$ are mutually independent.Let $\mathcal{D}_k = \{\boldsymbol{y}_l\}_{l=1}^k$ represent the accumulated data up to time k. Thus, $\boldsymbol{x}_{k+1}$ and $\mathcal{D}_k$ are conditionally independent with respect to $\boldsymbol{x}_k$, and $\boldsymbol{y}_{k+1}$ and $\boldsymbol{x}_{k+1}$ are conditionally independent with respect with $\boldsymbol{x}_k$, leading to simplifications in the next paragraph.

2.1.2 Bayesian filtering

The filtering problem consist to assess the current state of the signal by utilizing data observation up to the present moment. Filtering involves the determination of $p_{\boldsymbol{x}_k|\mathcal{D}_k}$, the probability density function associated with the probability measure on the random variable $\boldsymbol{x}_k|\mathcal{D}_k$. 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

$$p(\boldsymbol{x}_{k+1} \mid \mathcal{D}_k) = \mathbb{E}_{\boldsymbol{x}_k}\left[p(\boldsymbol{x}_{k+1} \mid \mathcal{D}_k, \boldsymbol{x}_k) \mid \mathcal{D}_k\right] = \mathbb{E}_{\boldsymbol{x}_k}\left[p(\boldsymbol{x}_{k+1} \mid \boldsymbol{x}_k) \mid \mathcal{D}_k\right].$$

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

$$p(\boldsymbol{y}_{k+1} \mid \mathcal{D}_k) = \mathbb{E}_{\boldsymbol{x}_{k+1}}\left[p(\boldsymbol{y}_{k+1} \mid \boldsymbol{x}_{k+1}) \mid \mathcal{D}_k\right].$$

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

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

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

2.1.3 Ensemble Kalman Filter

The Kalman filter [15] 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 [12]. The formulation uses an ensemble of discrete samples based on the assumptions of a multivariate Gaussian distribution, as for the Kalman filter. We present the stochastic EnKF, where the observations are perturbed to account for observation errors and to introduce stochasticity into the assimilation process, allowing for a more realistic representation of uncertainties and avoiding filter divergence issues.
Assuming we have an ensemble of N states $\{\boldsymbol{x}^i\}_{i=1}^N$, we could 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

$$\begin{aligned}\overline{\boldsymbol{x}}_f &= \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_f^i,\\ \boldsymbol{P} &= \frac{1}{N-1}\sum_{i=1}^{N}(\boldsymbol{x}_f^i-\overline{\boldsymbol{x}}_f)(\boldsymbol{x}_f^i-\overline{\boldsymbol{x}}_f)^T,\end{aligned}$$

where $\overline{\boldsymbol{x}}_f$ and $\boldsymbol{P}$ are the empirical estimates of the mean and covariance matrix of the state distribution obtained from the ensemble members.
Respectively, the mean and covariance of the observation $\left\{\mathcal{H}(\boldsymbol{x}_f^1)\right\}_{i=1}^N$ could be estimate 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 advantages of a correction of the state define in the member space. We define $\boldsymbol{F}$, the correction matrix that gives the update in terms of linear combinations of the forward states

$$\boldsymbol{X}_a = \boldsymbol{X}_f + \boldsymbol{X}_f\boldsymbol{F}, \tag{1}$$

where the matrix $\boldsymbol{F}$ only depends on the ensemble members through the predicted observations ensemble $\left\{\mathcal{H}(\boldsymbol{x}_f^1)\right\}_{i=1}$, the observation $\boldsymbol{y}$ and the associate perturbations $\{\boldsymbol{\varepsilon}^i\}_{i=1}$.

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 $\mathcal{P}$ of particles that follow the dynamic of the problem. Each particle of the set $\mathcal{P}$ carries both quantities $\boldsymbol{U}_p$ and the spatial coordinates $\boldsymbol{z}_p$.
We will focus our work on methods that discretize a solution a solution u on a continuous domain $\Omega \subset \mathbb{R}^d$ with Ω the spatial domain. This includes methods like Smoothed particle hydrodynamics (SPH) [14, 17] and the Vortex Method (VM) [7] and is extended to other methods like the Material Point Method (MPM) [25].

2.2.1 Particle discretization

Let $\Omega \in \mathbb{R}^d$ be our domain, where d is the space dimension. Any smooth field $\boldsymbol{u}$ on Ω could be written

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

with δ the Dirac delta distribution.
A kernel function ϕ_ε is introduced to obtain an average estimate $\langle\boldsymbol{u}\rangle$ of $\boldsymbol{u}$ such that

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

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

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

where the two first properties are remanent properties of the Dirac distribution and the last is a differentiability requirement.
The average function $\langle \boldsymbol{u} \rangle$ 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 $z_p \in \Omega_p$. We denote by V_p the volume of Ω_p. This discretization is then used to approximate the average function such that

$$\begin{aligned}
\langle \boldsymbol{u}(\boldsymbol{z}) \rangle &= \sum_{p\in\mathcal{P}} \int_{\Omega_p} \boldsymbol{u}(\boldsymbol{z}')\phi_\varepsilon(\boldsymbol{z}-\boldsymbol{z}')d\boldsymbol{z}' &(2)\\
&\approx \sum_{p\in\mathcal{P}} \boldsymbol{u}(\boldsymbol{z}_p)V_p\phi_\varepsilon(\boldsymbol{z}-\boldsymbol{z}_p) &(3)\\
&\approx \sum_{p\in\mathcal{P}} \boldsymbol{U}_p\phi_\varepsilon(\boldsymbol{z}-\boldsymbol{z}_p). &(4)
\end{aligned}$$

Thus, any function defined on a particle discretization is defined by an ensemble of particle location $\boldsymbol{z}_p$ associated with a particle value $\boldsymbol{U}_p = \boldsymbol{u}(z_p)V_p$ and a smooth 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

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

.
This kernel is infinitely differentiable but defined on non-compact support. In practice, we use a cut-off to remove negligible value for large distance from a particle.
Other kernels, based on B-Spline functions to work on a compact support. Those functions are also positive which is a requirement for some field like the density.

2.3 Particle-based function manipulations

Based on particle discretization, we present several particle manipulation that will used in our methods. Originally, those manipulation are either dedicate to improve the quality

of the approximation, avoid high distorsion by creating a new particle discretization or to project the solution on a Eulerian configuration. The different operators will be used in the assimilation process in order to update particle solution of each members 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

$$\boldsymbol{U}_p = \int_{\Omega_p} \boldsymbol{u}(z) d\boldsymbol{z} = \boldsymbol{u}(\boldsymbol{z}_p) V_p,$$

where $\boldsymbol{z}_p$ is the particle location.
This approximation is easily computable but do not ensure the conservation of the all the moment of the field. A better approximation could be obtain using the iterative Beale's formula [2] which corrected circulation values in order to recover the vorticity field at the particle locations which is closed to the next paragraph.

2.3.2 Regression operator

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

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

where $\boldsymbol{\Phi}_{ij} = \phi_\varepsilon(z_i - z_j)$.
Finding the new intensities U_p correspond to solving the previous system in the least square sense. It corresponds to the classical problem to find the minimizer of the following quadratic function

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

In this case, the solution is $\boldsymbol{U} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi u}$. Because the problem may be ill-posed, particularly in the case of large set of non well-distributed particles, the solution is regularized by introducing a penalization term. The Ridge regression we used introduced a penalization on of the form $\Lambda \|\boldsymbol{U}\|_2^2$, where Λ is a penalization coefficient, such as the new problem is

$$\boldsymbol{U}_{p,\text{ridge}} = \arg\min_{\boldsymbol{U}} \|\boldsymbol{u} - \boldsymbol{\Phi U}\|_2^2 + \Lambda \|\boldsymbol{U}\|_2^2 ,$$

given the following solution $\boldsymbol{U}_{p,\text{ridge}} = (\boldsymbol{\Phi}^T \boldsymbol{\Phi} + \Lambda \boldsymbol{I})^{-1} \boldsymbol{\Phi u}$.

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 [7, 9]. This method allow us to switch from an Lagrangian discretization $\mathcal{P}$ to an Eulerian one Λ, and then switch back to a completely new regular particles discretization $\mathcal{P}'$ that conserve as much as possible the moment of the particle discretization.
In our methodology, we propose a two-step approach. Initially, we execute an assignment step ((a)) to transition the particle discretization field into a grid discretization field. Subsequently, an interpolation step ((b)) is performed to yield a new set of regularly spaced particles. We demonstrate that the combination of these two operations preserves the moments of the particle distribution contingent upon the choice of the interpolation shape function W associated with the grid.
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 $\boldsymbol{U}_p$ and the nodal field values as $\boldsymbol{u}_I$. By using some shape function W, the assignment step from particles to each node $I \in \Lambda$ can be written as

$$\boldsymbol{u}_I = \frac{1}{V_I} \sum_{p \in \mathcal{P}} \boldsymbol{U}_p W \left(\frac{z_I - z_p}{\ell_I} \right).$$

Where W determine a redistribution of the intensity on the grid. The new discretization can then be used to approximate the field $\boldsymbol{u}_p$, defined by the particle discretization, by interpolation such that

$$\boldsymbol{u}_p(z) \approx \boldsymbol{u}_g(z) = \sum_{I \in \Lambda} \boldsymbol{u}_I W \left(\frac{z - z_I}{\ell_I} \right) \quad \forall z \in \Omega.$$

(b) *Interpolation on a new regular particle discretization*

A new set of particles is define at the quarter of each cells such that the new position are define at $z_{p'} = d_p/2 + i\ dp, \quad i = 0, \ldots, 2n_g$. The value of the field is then interpolate at that new location and multiply with the volume of the particle $\boldsymbol{U}_{p'} = \boldsymbol{u}_g(z_{p'})V_{p'}$ in order to give a new particle approximation of the field

$$\boldsymbol{u}_g(z) \approx \boldsymbol{u}_{p'}(z) = \sum_{p' \in \mathcal{P}'} \boldsymbol{u}_g(z_{p'})V_p,$$

.

The combination of these two steps can initially be utilized to generate a new undistorted particle distribution. The quality of the remeshing or regridding process depends on the shape function W, which serves as a redistribution kernel. The shape function W determines the type and quality of the transfer. The method's effectiveness is evaluated by assessing the conservation of the first moments of the particle distributions, as detailed in the Appendix B.
For the shape function 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 second order, allowing only exact interpolation of linear functions.
Monaghan [19] 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

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

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 independant, 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 $\boldsymbol{F}$ relies solely on the observation $\boldsymbol{y}$, predictive observations $\{\boldsymbol{h}_i\}_{i=1}^{N\mathrm{ens}}$, and noise samples $\{\boldsymbol{\varepsilon}_i\}_{i=1}^{N\mathrm{ens}}$. The analysis

fields are obtained at any spacial coordinates thanks to the particle approximation of each member field $\boldsymbol{u}_i^f$ such as for all $\boldsymbol{z} \in \Omega$

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

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

$$\boldsymbol{u}_i^a(\boldsymbol{z}) = \sum_p \boldsymbol{U}_{ip}^f \phi_\varepsilon(\boldsymbol{z} - \boldsymbol{z}_{ip}) + \sum_j \boldsymbol{F}_{ij} \sum_p \boldsymbol{U}_{jp}^f \phi_\varepsilon(\boldsymbol{z} - \boldsymbol{z}_{jp}) \quad i = 1, \dots, N \tag{5}$$

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 a allow a constant number of particles.
- The Particles-EnKF filter (3.2) executes data assimilation with (5). Analysed 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 $\mathcal{P}_i^f = \{(\boldsymbol{z}_{ip}^f, \boldsymbol{U}_{ip}^f)\}_{ip=1}^{N_{ip}}$,
- The associate field is project on a regular grid of n_g elements of characteristic length $2dp$. Using the assigment operator, we obtain for each node $iI \in \Lambda_i$

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

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

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

- 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 $ip \in \mathcal{P}_i^a$

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

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

Algorithm 2: Remesh Filter analysis update

Data: $\boldsymbol{G} \in \mathbb{R}^{n_g \times d}, \boldsymbol{z}^a \in \mathbb{R}^{2n_g \times d}$; `// grid discretization`
Data: $\boldsymbol{R} \in \mathbb{R}^m$; `// observation covariance`
Input: $\mathcal{P}_i^f = \{(\boldsymbol{z}_{ip}^f, \boldsymbol{U}_{ip}^f)\}_{ip=1}^{N_{ip}}, \quad i = 1, \dots, N_{\text{ens}}$; `// forward discretizations`
Input: $\boldsymbol{Y}_f \in \mathbb{R}^{m \times N_{\text{ens}}}$; `// the associate observation anomalies`
Input: $\boldsymbol{D} \in \mathbb{R}^{m \times N_{\text{ens}}}$; `// the perturbed observations`
$\boldsymbol{F} = \boldsymbol{Y}_f^T (\boldsymbol{Y}_f \boldsymbol{Y}_f^T + \boldsymbol{R})^{-1} (\boldsymbol{D} - \boldsymbol{Y})$; `// correction matrix`
foreach $i = 1, \dots, N_{ens}$ **do**
  $\boldsymbol{u}[:, i] =$ `Projection`$(\mathcal{P}_i^f, \boldsymbol{G})$
$\boldsymbol{u} = \boldsymbol{u} + \boldsymbol{u}\boldsymbol{F}$; `// analysis update`
foreach $i = 1, \dots, N_{ens}$ **do**
  $\boldsymbol{z}_{ip}^a, \boldsymbol{U}_{ip}^a =$ `Assign`$(\boldsymbol{u}[:, i])$
return $\mathcal{P}_i^a = \{\boldsymbol{z}_{ip}^a, U_{ip}^a\}_{ip=1}^{N_a}, \quad i = 1, \dots, N_{ens}$; `// 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 (5), are approximated on the previous discretization such that $Z_i^a = Z_i^f$ to avoid exponential growth of the number of particles.

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

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

where $\boldsymbol{U}_{ip}^a$ 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 1

Algorithm 3: Part-EnKF Filter analysis update

1
Data: $\boldsymbol{R} \in \mathbb{R}^m$; `// observation covariance`
Input: $\mathcal{P}_i^f = \{(\boldsymbol{z}_{ip}^f, \boldsymbol{U}_{ip}^f)\}_{ip=1}^{N_{ip}}, \quad i = 1, \ldots, N_{\text{ens}}$; `// forward discretizations`
Input: $\boldsymbol{Y}_f \in \mathbb{R}^{m \times N_{\text{ens}}}$; `// the associate observation anomalies`
Input: $\boldsymbol{D} \in \mathbb{R}^{m \times N_{\text{ens}}}$; `// the perturbed observations`
2 $\boldsymbol{F} = \boldsymbol{Y}_f^T (\boldsymbol{Y}_f \boldsymbol{Y}_f^T + \boldsymbol{R})^{-1} (\boldsymbol{D} - \boldsymbol{Y})$; `// correction matrix`
3 **foreach** $i = 1, \ldots, N_{ens}$ **do**
4 $\quad \boldsymbol{U}_{ip}^a =$`Approx(`$\mathcal{P}_i^f, \boldsymbol{F}$`)` ; `// approximate the analysis fields`
5 **return** $\mathcal{P}_i^a = \{\boldsymbol{z}_{ip}^f, U_{ip}^a\}_{ip=1}^{N_a}, \quad i = 1, \ldots, N_{ens}$; `// 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

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

where $L = 2\pi$ the periodic length
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.

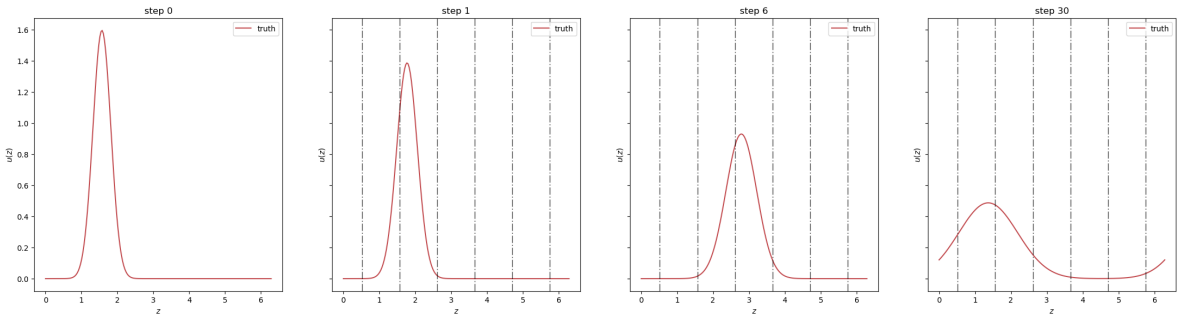


Figure 1: The analytical solution of the convection-diffusion problem evolves over time, with the final snapshot revealing a complete spatial period.

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

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

For solving the convection-diffusion scheme, we employ the two steps of the viscous splitting algorithm outlined in Section 5. Initially, we address the equation for non-diffusive advection

$$\frac{dz_p}{dt} = v(z_p, t), \quad \frac{dU_p}{dt} = 0$$

and subsequently, we apply the diffusion equation

$$\frac{dz_p}{dt} = 0, \quad \frac{dU_p}{dt} = D\frac{d^2U_p}{dz^2}$$

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) [11, 8] to approximate the laplacian term $\frac{d^2U_p}{dz^2}$.
Such as, the intensities U_p are updated using the following formula

$$U_p = U_p + \varepsilon^{-1} \sum_q (U_q - U_p)\phi_g^P(z_q - z_p).$$

with ε the smoothing length of the Gaussian periodic kernel define as $\phi_g^P = \sum_{n=-\infty}^{+\infty} \phi_g(r - nL)$.
For the periodic boundary problem described in section 4, we define an equivalent kernel function $\phi_g^P = \sum_{n=-\infty}^{+\infty} \phi_g(r - nL)$,
. All the kernel properties are still verified on a single period.
Our particle-based model employs a discretization of $N_{part} = 100$ particles with a size of $h = \frac{L}{N_{\text{part}}}$ and a smoothing length of $\varepsilon = 1.3h$. For the sake of comparison, we solve the convection diffusion equation with a explicit (?) central finite difference scheme discretized on a regular grid with $N_{grid} = 100$ nodes.

4.2 Assimilation parameters and ensemble generation

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}. The observational data is subject to additive noise, denoted as $\eta \sim \mathcal{N}(0, \sigma_y \boldsymbol{I}_5)$, where $\sigma_y = 0.05$ and $\boldsymbol{I}_5$ represents the identity matrix.
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 ensemble is drawn from $Z_m \sim \mathcal{N}(\pi/2 + 0.6, 0.5)$, while the standard deviation is drawn from $S_m \sim \mathcal{U}(0.8, 1.2)$. 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 differentiation in particle support poses challenges during the interpolation phase in the Part-EnKF. Velocity v is sampled from $v \sim \mathcal{N}(0.9, 1.2)$, and diffusion D is sampled from $D \sim \mathcal{U}(0.02, 0.08)$. The sample and initial state are illustrated in Figure 2. Simultaneously, a standard Ensemble Kalman Filter (EnKF) update is applied to the nodal variables to construct the reference filter Grid-EnKF which use a grid-based model.

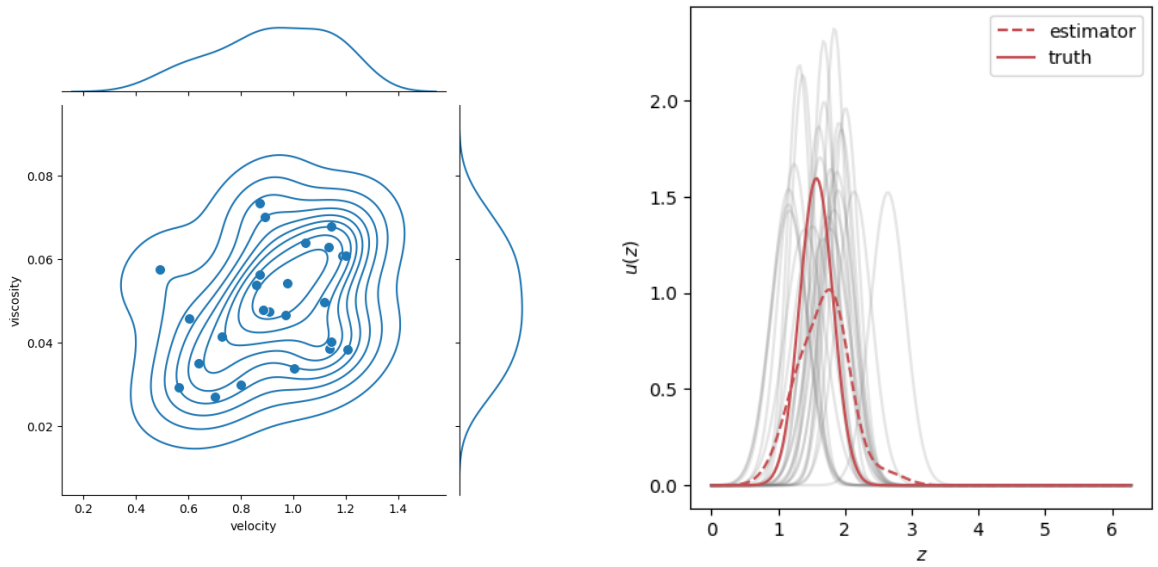


Figure 2: On the left the initial parameters sample, v in abscissa and D in ordinate. On the right is the initial ensemble state.

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. Finally, we define the relative L_2 error as

$$e_{L_2} = \frac{\frac{1}{N_{\text{ens}}} \sum_{i=1}^{N_{\text{ens}}} \left(\int_\Omega \left(u^{gt}(z) - u_i(z) \right)^2 dz \right)^{1/2}}{\|u\|_{L_2}} \tag{6}$$

where u_i denote the i-th member of the ensemble and $\|u\|_{L_2}$ denote the L_2 norm of u^{gt}. We compute the parameter error with a norm-2 as $e_\theta = \frac{\mathbb{E}(\|\theta - \theta_{EnKF}\|_2^2)^{1/2}}{\|\theta_{EnKF}\|_2}$.

4.3 Results

We begin the comparison of the different filters by refraining from calibrating the parameters of the model. The filters exclusively update the state, treating the unknown parameters as a source of uncertainty for the model. 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 3, we appreciate a similar agreement for all the filters, except for Particle-EnKF, with a reduced number of particles in the support.

Figure 3: State error with respect to assimilation time step.

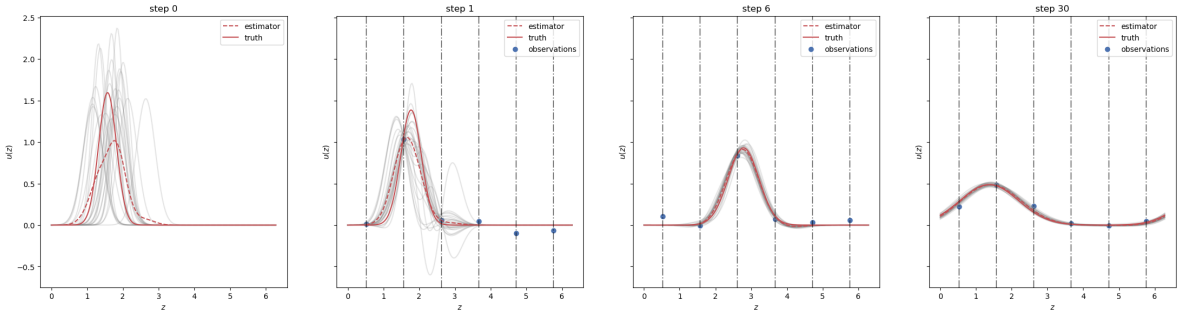


Figure 4: Data assimilation comparison over different filters. From top to bottom, with Grid Filter, RemeshEnKF, PartEnKF with $N_{part} = 100$, and PartEnKF with $N_{part} = 60$.

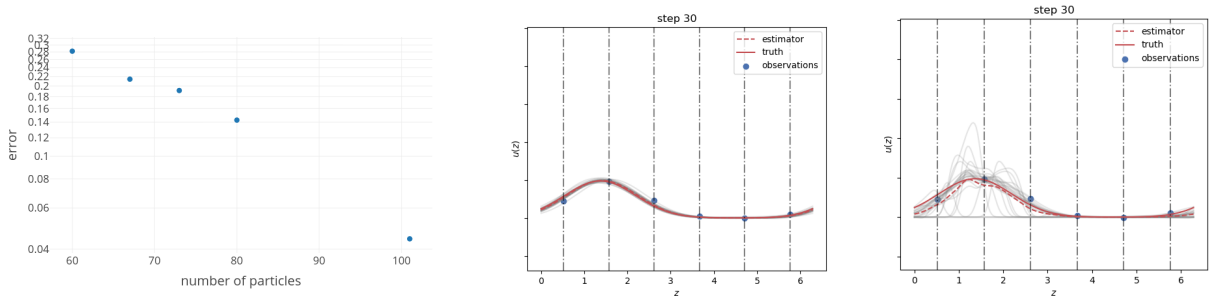


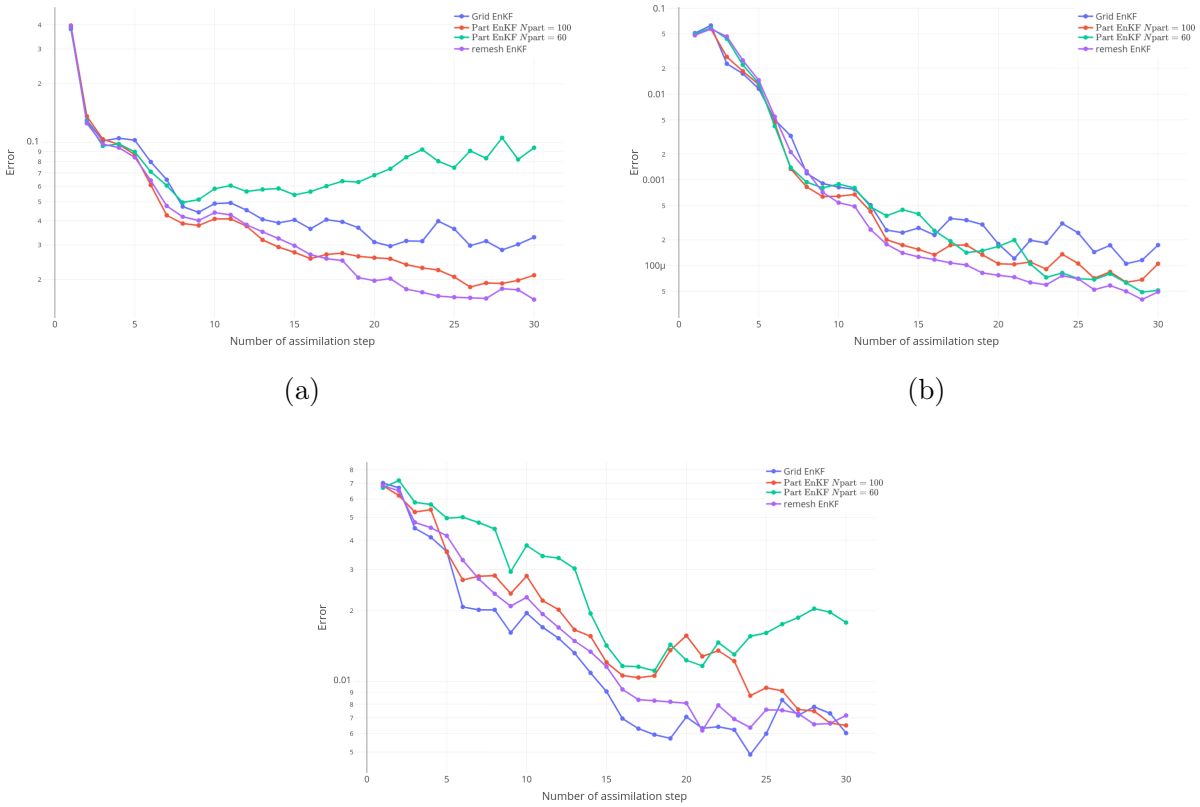
Figure 5: Left: Error with respect to particle support size, Middle: Final step for a support of 100 particles, Right: Final step for a support of 60 particles.

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 [13] 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 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 decreases with an increase in the number of particles. Moreover, qualitatively examining the snapshot on the right reveals that the solution closely aligns with the reference.
However, Adding particles in a more complex solution is a challenging task. Indeed, a good spacing between particles and the density of particles has to be preserved. In this case, we advise defining criteria for the error of the reconstruction. 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's 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.

4.4 Expanded state

In the case of an expanded state, we extend the update to parameters by introducing the variable $\theta = (v, D)$ in the state definition. This approach allows the calibration of these quantities. Figure 7 displays the marginal distribution of parameters over time obtained with Remesh-EnKF. The velocity and diffusion exhibit well-estimated values, indicated by

unbiased means and a reduction in variability.



(a) (b)

(c)

Figure 6: In a, b, c respectively state, velocity, and viscosity errors with respect to the assimilation time step.

Quantitatively, we also observe a reduction of error of the estimation of the state in 6a, the estimate of the velocity 6b, and the estimate of the diffusion 6c.
Indeed, even though the state estimation was already well accurate in Figure 3, the state estimation further improves here, thanks to the calibration of the model parameters. Updating the parameters allows for adjusting the model to the observed data, thereby enhancing the accuracy of the state estimation. This underscores the significance of parameter calibration in achieving more precise results in the context of data assimilation. It is noteworthy that even in the worst-case scenario of the Part-EnKF, the estimation of parameters remains accurate, indicating a good understanding of the dynamic behavior of the problem.

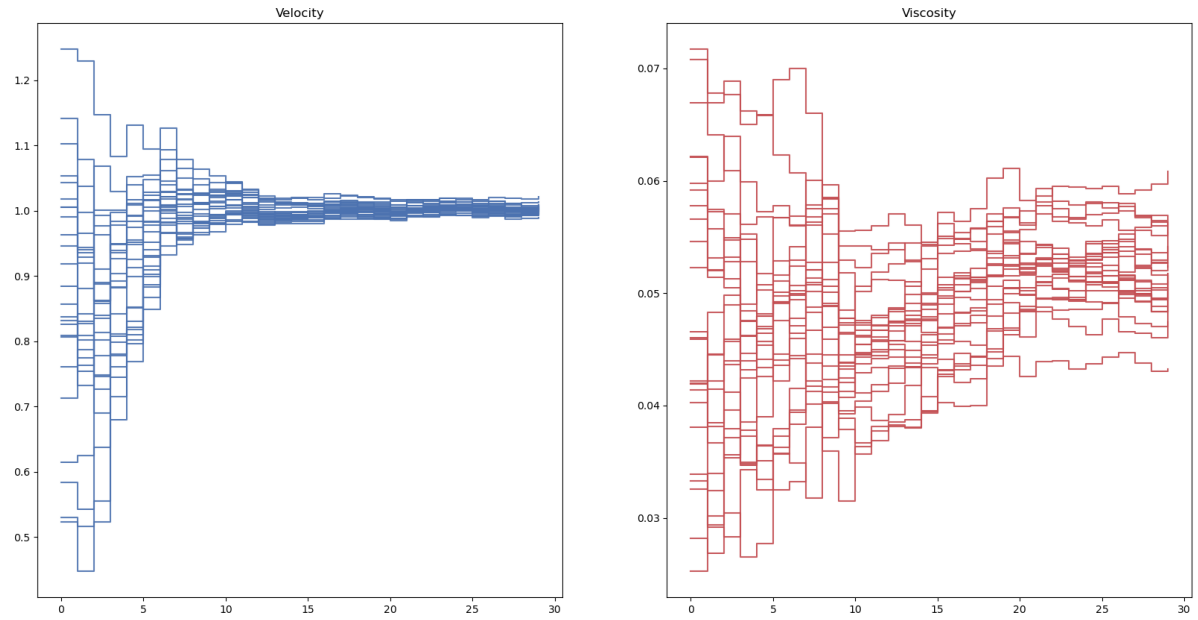


Figure 7: Marginal distribution of velocity (left) and viscosity (right) over time (Remesh-EnKF).

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. [7]. 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 $\boldsymbol{\omega} = \nabla \times \boldsymbol{v}$ satisfies

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

where ω denotes vorticity, $\boldsymbol{v}$ 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 $\boldsymbol{z}_p$, an associated kernel ϕ_ε, and a circulation Γ_p. For all points $\boldsymbol{z}$ within the domain Ω, the vorticity is expressed as

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

To address the Navier-Stokes equation, we employ a viscous splitting scheme, following the methodology outlined in [8], acknowledging the predominance of the convection term over viscosity. The initial phase involves solving the advection component for each particle in a Lagrangian context

$$\frac{d\boldsymbol{z}_p}{dt} = \boldsymbol{v}(\boldsymbol{z}_p), \quad \frac{\omega_p}{dt} = 0.$$

The particle trajectory depends on the local velocity field obtained using the Biot-Savart law. The resolution is based on the Vortex-In-Cell algorithm [6, 3], coupled with an FFT solver to enhance computation performance.
The subsequent steps involve assigning particle vorticity values to the grid using a particle-to-grid formula, 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.
In conclusion, a Runge-Kutta 3 time-stepping scheme is employed to update the particle positions through a time integration scheme. The second phase involves solving the heat equation.

$$\frac{\mathrm{d}\boldsymbol{z}_p}{\mathrm{d}t} = 0, \quad \frac{\mathrm{d}\omega_p}{\mathrm{d}t} = \nu \Delta\omega(\boldsymbol{z}_p),$$

by employing an integral equation that approximates $\Delta\omega$

$$\Delta\omega \approx \varepsilon^{-2} \int [\omega(\boldsymbol{z}) - \omega(\boldsymbol{y})] \phi_\varepsilon(\boldsymbol{y} - \boldsymbol{z}) d\boldsymbol{z},$$

This leads, with the particle approximation, to a redistribution of the particles' intensities in their previous locations, such as

$$\frac{d\Gamma_p}{dt} = \nu \varepsilon^{-2} \sum_q (G_q - G_p) \phi_\varepsilon(\boldsymbol{z}_q - \boldsymbol{z}_p).$$

For further details on the computation, please refer to [8]. Consequently, our vortex model equation is solely dependent on the fluid's viscosity as a model parameter.

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 [20]. 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 $\boldsymbol{z_0}$, a radius R, and an orientation α. This

particular test case is designed to assess the impact of a relatively small domain of the particle support, which is crucial for the Part-EnKF filter.
The dipole vorticity field ω could be expressed as

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

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 8.

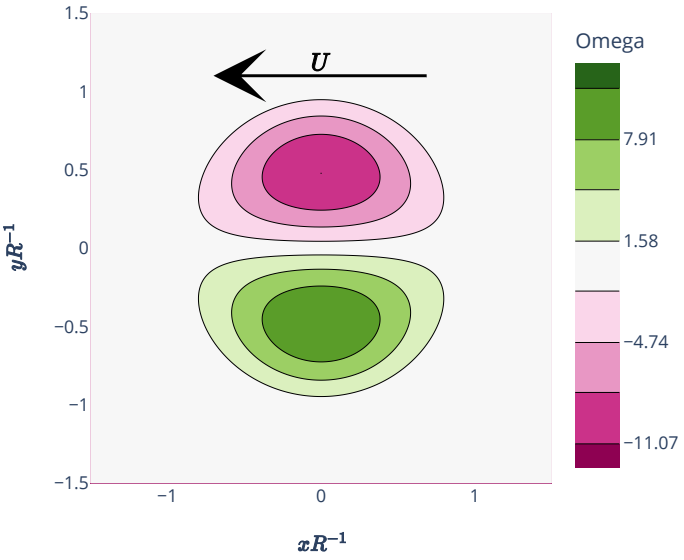


Figure 8: The Lamb-Chaplygin dipole vorticity field on a normalize space.

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 1.
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

described also in Table 1. The trajectory of the ground truth is illustrate in the Figure 9 on a regularly spaced grid.

Figure 9: 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 10.

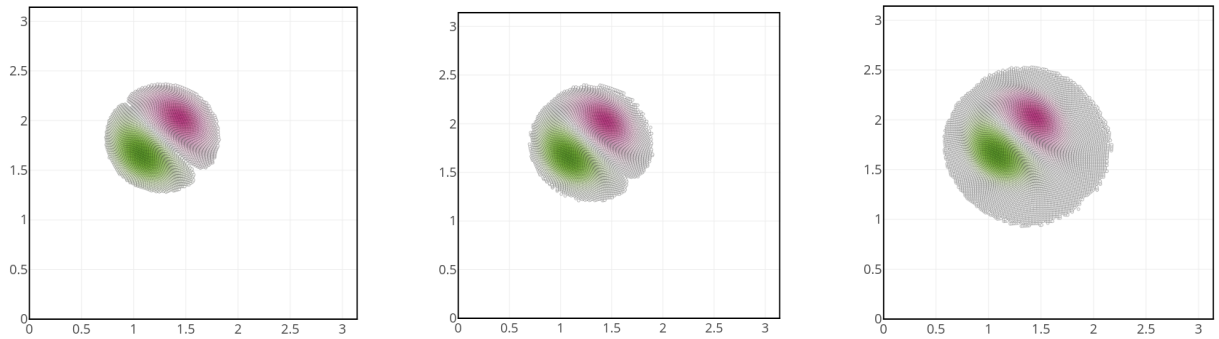


Figure 10: Effect of the parameter ε_ω on the particle discretization of the solution for one member. From left to right, results for $\varepsilon_\omega = 0.1, 0.01$, and $1.e^{-6}$.

For the next paragraphs, if the value is not explicitly changed, we use the nominal parameters described in Table 2 for the simulation.

5.3 Assimilation parameters and ensemble generation

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 $\boldsymbol{z}_{\text{mean}}$. Additionally, the model viscosity ν is also sampled. All the distributions are summarized in Table 3. The first six members are plotted in Figure 11.

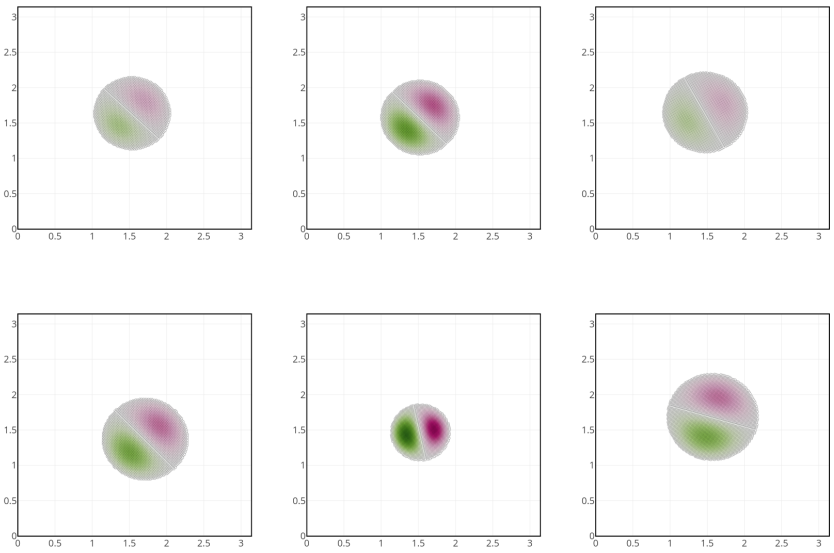


Figure 11: Six samples from the initial ensemble.

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 $\Gamma_p = \omega(\boldsymbol{z}_p)V_p$, and $V_p = d_p^2$ represents the volume of the particle. The 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 2.

Observations are collected on a regular grid of size N_{obs}, measuring both components of the velocity. The observations follow a normal distribution $\mathcal{N}(0, \sigma_{obs}^2 \boldsymbol{I})$, 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 12.

We will employ the same metric, the relative L_2-error, similar to the one-dimensional application in 6 as well as the absolute L_2-error. The L_2 norm is computed using a quadrature over a regular grid of an ensemble of cells $\mathcal{C}$ such as for any $f \in L_2$

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

where $\boldsymbol{z}_c$ 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.

The Part-EnKF uses here the approximation over the forward particle support. It has been seen to yield relatively good results with efficient time computation. This choice

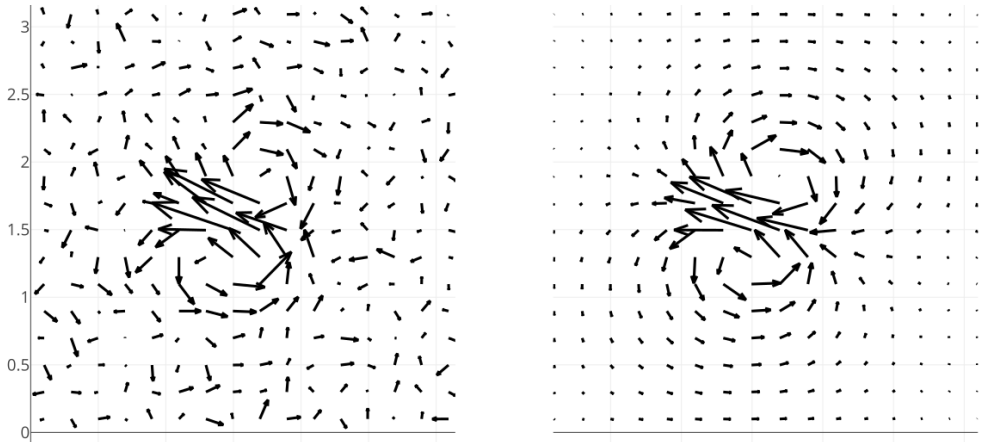


Figure 12: Observed and reference velocity fields. The error on each component is sample from a centred normal distribution with the nominal value $\sigma_{\text{obs}} = 0.05$.

introduces a dependence on the particle discretization. Similar conclusions would be drawn in the following study if a regression operator had been used to approximate the analyzed solution.

5.4 Results

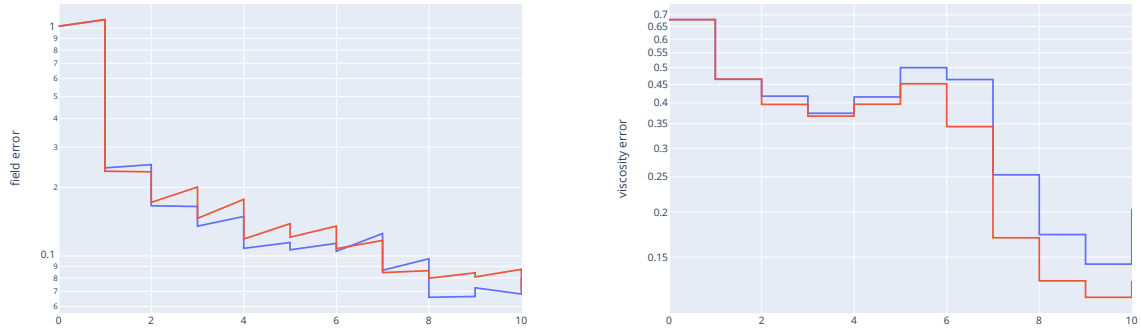


Figure 13: Error curves through assimilation steps. Left: L_2-error of the field, Right: Error for the viscosity parameter. With Part-EnKF in blue and Remesh-EnKF in red.

We start by analyzing the assimilation error over time. Figure 13 illustrates the relative error throughout the assimilation process for the nominal set of parameters, demonstrating comparable results for both filters, as well as for the assimilation of state and model viscosity. Additionally, Figure 14 depicts the evolution of the ensemble distribution for the viscosity parameters. In both cases, the variance and the bias decrease over time. However, the Remesh-EnKF has higher variances compared to the Part-EnKF, particularly at the middle and end stages of assimilation. Notably, the filters underestimate the viscosity in the initial assimilation steps. It could be primarily due to the dissipation introduced in the first estimate, which is subsequently offset thanks to the parameter. After subsequent assimilation steps, the parameter finally stabilizes. Nevertheless, the reference always falls within the range of the distribution.
We assess the performances of the different filters by evaluating the convergence of the error with respect to the assimilation and the simulation parameters. Additionally, we added a filter called Part-Grid-EnKF. This last filter combined the approximation process of the Part-EnKF but on a regularly spaced particle discretization like the Remesh-EnKF filter. This filter will allow us to better show if the difference between the filters is due to the Part-EnKF discretization or the particle approximation.
First, we observe the convergence rate concerning data assimilation parameters: the observation precision, which is $1/\sigma^2_{\text{obs}}$, the number of observations N_{obs}, the number of assimilation step N_{assim}. The results are regrouped in Figure 15. We first observed that the convergence is more substantial and more regular for the Remesh-EnKF. The convergence is the same at low precision, but for high precision, the error decreases slower for the Part-EnKF and Part-Grid-EnKF. In fact, the observed high reduction of variances and the high bias suggest a collapse of the state of the ensemble to a suboptimal solution. Nevertheless, the viscosity errors are all comparable. This observation leads to the conclusion that the approximation step bounds the error of the state. As concern the convergence for other

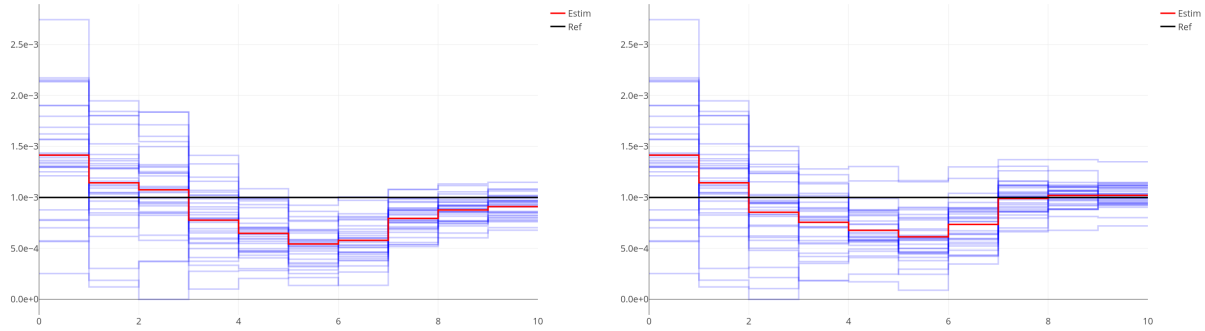
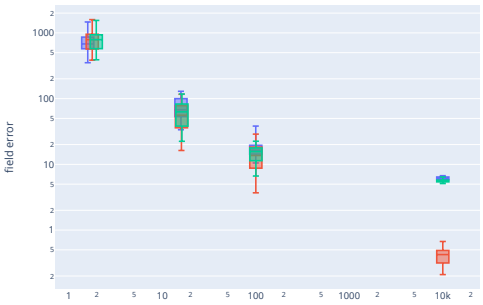


Figure 14: Evolution of the viscosity ensemble through assimilation. Left: for the Part-EnKF, Right: for the Remesh-EnKF. In blue the ensemble values, in red the estimate and black the reference.

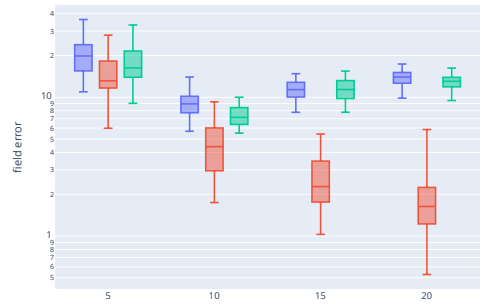
assimilation parameters N_{obs}, σ_{obs}, the conclusion is quite similar. The field error decreases with N_{obs}, σ_{obs} for the Remesh-EnKF. The result is more mitigated for the Part-EnKF. Again, it confirms that the Part-EnKF fails to converge as successfully as the Remesh-EnKF. Finally, we note that the converge with respect to N_{obs}, σ_{obs} are less regular even for the Remesh-EnKF. This observation is probably mainly due to the fact that the measures and assimilation steps are regularly spaced over space and time. In fact, the number of measures over the dipole at time t is low, and the relative position could have a relatively high impact on the assimilation.

To better understand the differences, we also evaluate 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, it introduced severe approximation that 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 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 16 that the Remesh-Filter and Part-Grid-EnKF are 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

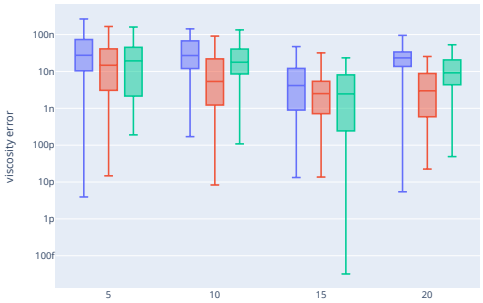


state error w.r.t. $1/\sigma^2_{\text{obs}}$

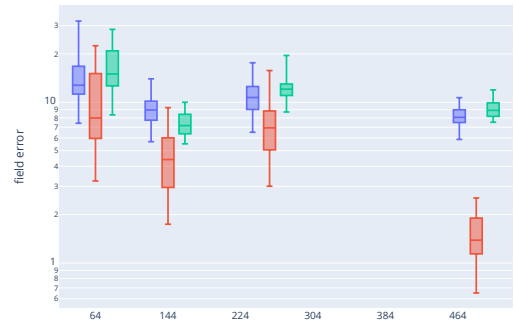
viscosity error w.r.t $1/\sigma^2_{\text{obs}}$



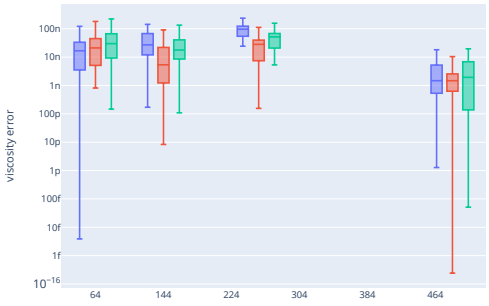
state error w.r.t. N_{assim}



viscosity error w.r.t. N_{assim}



state error w.r.t. N_{obs}



viscosity error w.r.t N_{obs}

Figure 15: Error curves for different assimilation parameters. Left: The normalize L_2 error for the vorticity field. Right: The normalize error on viscosity.

the Remesh-EnKF filter.

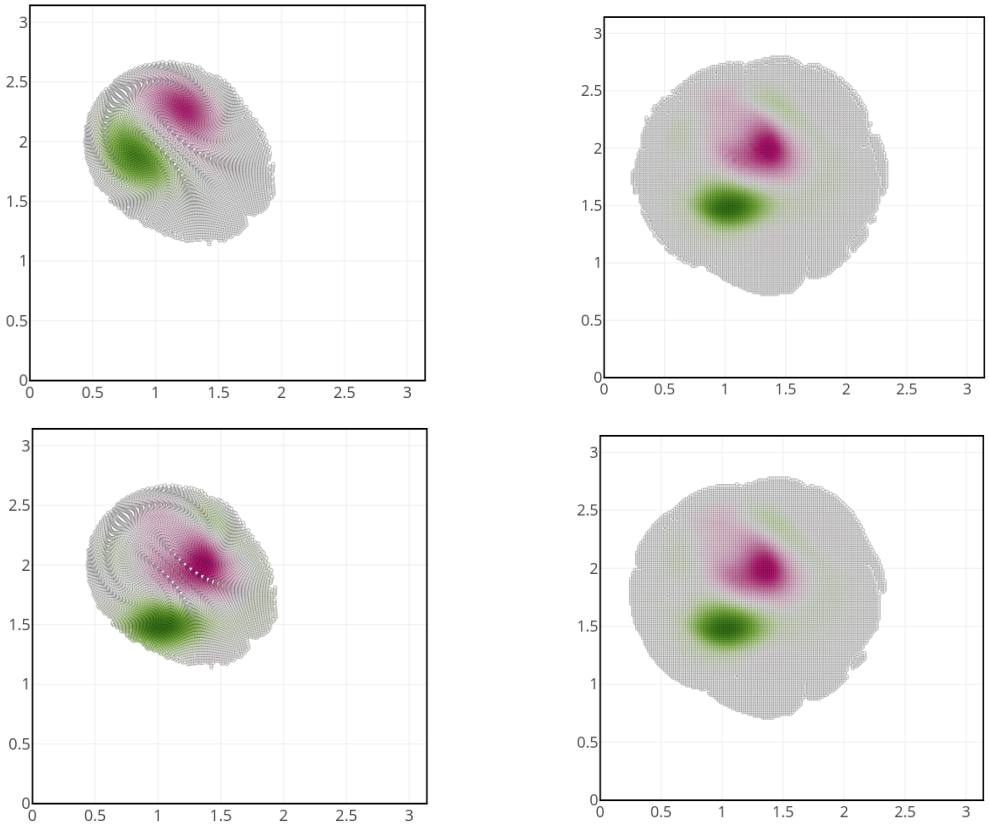
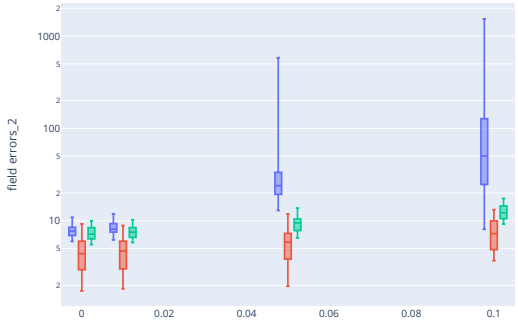


Figure 16: Assimilation of one member with a forecast discretization unadapted to the analyses solution. From left to right and up to down: the forecast member, the analyses member with the Remesh-EnKF filter, the analyses member with the Part-EnKF filter and finally the analyses member with the Part-Grid-EnKF. The forecast discretization use by the Part-EnKF is not always a well support of approximation for the analyses and introduce discretization errors.

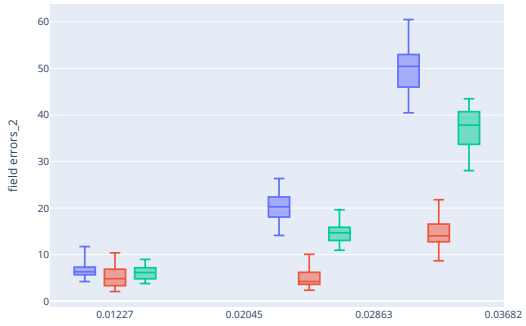
To evaluate the effect of the size of the support, we varied the value of ϵ_ω. We have seen in Figure 10 that this parameter affects the number of particles and, thus, the size of the support. We observe relatively bad results for very high ϵ_ω because the solution can not be approximated over all the dipole and leads to a significant difference with Remesh-EnKF/Part-Grid-EnKF and variance inside the ensemble. However, the error stabilizes rapidly by decreasing the threshold. A relatively low effect is also observed on the Remesh-EnKF due to truncation errors at the border of the dipole solution.
Thanks to the Part-Grid-EnKF, we could better decompose the error. On the one hand, the difference between the Part-EnKF and the Part-Grid-EnKF is linked to the support and the distortion of the particle distribution. On the other hand, the remaining difference between Part-Grid-EnKF is due to the particle approximation error. This hypothesis is confirmed by analyzing the error with respect to the particle size d_p in Figure 17. We observe that

the error for Part-EnKF and Part-Grid-EnKF increases proportionally with d_p as for the approximation error. The same order of magnitude for the different filters is observed is observed for relatively small particle sizes. This confirmed the high effect of the particle approximation in this case. 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.

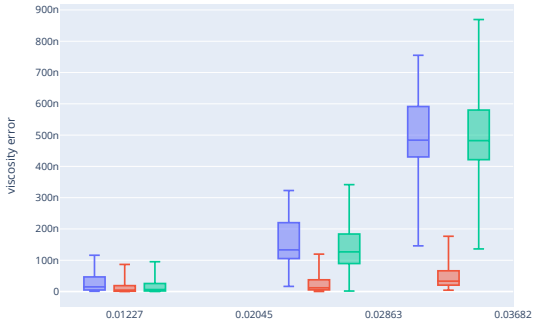


state error w.r.t. ε_ω

viscosity error w.r.t. ε_ω



state error w.r.t. d_p



viscosity error w.r.t. d_p

Figure 17: Error curves 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.

This discussion pointed out the high dependency of the Part-EnKF on particle discretization. As it could be understood, the particle discretization of a member could be too far from the solution support of particles. This opens the question of the choice of 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 proposi-

tion has to be evaluated because it could considerably reduce the variance of the ensemble. Moreover, we note that the particle approximation could rapidly have a high effect, as seen in this Part.

A Stochastic Ensemble Kalman Filter

We define the matrix of states and the matrix of anomalies $\boldsymbol{X}_f = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^N]$, $\boldsymbol{A}_f$ whose columns are the member states and the normalized anomalies.

$$\boldsymbol{A}_f = \frac{1}{\sqrt{N-1}}(\boldsymbol{X}_f - \overline{\boldsymbol{x}}_f \mathbf{1}^T),$$

where$\mathbf{1} \in \mathbb{R}^N$ is a vector of one.
Respectively the matrix of observation and observation anomalies are $\mathcal{Y}_f = [\mathcal{H}(\boldsymbol{x}_f^1), \ldots, \mathcal{H}(\boldsymbol{x}_f^N)]$, $\boldsymbol{Y}_f$ where columns are

$$\boldsymbol{Y}_f = \frac{1}{\sqrt{N-1}}\left(\mathcal{Y}_f - \overline{\boldsymbol{y}}_f \mathbf{1}^T\right) \quad \text{with} \quad \overline{\boldsymbol{y}}_f = \frac{1}{N}\sum_{j=1}^{N} \mathcal{H}(\boldsymbol{x}_f^j).$$

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

$$\begin{aligned}
\boldsymbol{P}\boldsymbol{H}^T &= \frac{1}{N-1}\sum_{i=1}^{N} (\boldsymbol{x}_f^i - \overline{\boldsymbol{x}}_f)^T \left[\mathcal{H}_k(\boldsymbol{x}_f^i) - \overline{\boldsymbol{y}}_f\right]^T = \boldsymbol{A}_f \boldsymbol{Y}_f^T, \\
\boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^T &= \frac{1}{N-1}\sum_{i=1}^{N} \left[\mathcal{H}_k(\boldsymbol{x}_f^i) - \overline{\boldsymbol{y}}_f\right] \left[\mathcal{H}_k(\boldsymbol{x}_f^i) - \overline{\boldsymbol{y}}_f\right]^T = \boldsymbol{Y}_f \boldsymbol{Y}_f^T, \\
\tilde{\boldsymbol{K}} &= \boldsymbol{P}\boldsymbol{H}^T(\boldsymbol{H}\boldsymbol{P}\boldsymbol{H}^T + \boldsymbol{R})^{-1} = \boldsymbol{A}_f \boldsymbol{Y}_f^T(\boldsymbol{Y}_f \boldsymbol{Y}_f^T + \boldsymbol{R})^{-1}.
\end{aligned}$$

This observation matrix-free implementation rely on the secant method approximation $\mathcal{H}(\boldsymbol{x}_f^i - \overline{\boldsymbol{x}}_f) \approx \mathcal{H}(\boldsymbol{x}_f^i) - \overline{\boldsymbol{y}}_f$. The forecast is then update to a posterior ensemble $[\boldsymbol{x}_a^i]_{i=1}^N$ such as

$$\boldsymbol{X}_a = \boldsymbol{X}_f + \tilde{\boldsymbol{K}}(\boldsymbol{D} - \boldsymbol{Y}), \tag{7}$$

where $[\boldsymbol{D}]^i = \boldsymbol{y} + \boldsymbol{\varepsilon}^i$ is the perturbed observation with $\boldsymbol{\varepsilon}^i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{R})$, $\tilde{\boldsymbol{K}}$ the ensemble Kalman gain matrix and $(\boldsymbol{D} - \boldsymbol{Y})$ the innovation term. The forecast step is then applied to the analyzed ensemble until the next 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.

$$\boldsymbol{X}_a = \boldsymbol{X}_f + \boldsymbol{A}_f \boldsymbol{Y}_f^T(\boldsymbol{Y}_f \boldsymbol{Y}_f^T + \boldsymbol{R})^{-1}(\boldsymbol{D} - \boldsymbol{Y})$$

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

$$\boldsymbol{X}_a = \boldsymbol{X}_f + \boldsymbol{X}_f \boldsymbol{F}, \quad \boldsymbol{F} = \boldsymbol{Y}_f^T (\boldsymbol{Y}_f \boldsymbol{Y}_f^T + \boldsymbol{R})^{-1} (\boldsymbol{D} - \boldsymbol{Y}). \tag{8}$$

.

B Moment conservation of particle discretization

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

$$\sum_{I \in \Lambda} W\left(\frac{z - z_I}{\ell_I}\right) = 1, \quad z \in \Omega \tag{9}$$

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

$$\sum_{p' \in \mathcal{P}'} W\left(\frac{z - z_{p'}}{\ell_I}\right) = \frac{V_I}{V_p'}, \quad z \in \Omega. \tag{10}$$

.
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. Primarly for the assigment (a)

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

using the property (9). Secondary, for the interpolation process (b)

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

using equation (10).

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

$$\sum_{I \in \Lambda} (\boldsymbol{z} - \boldsymbol{z}_I)^{\alpha} W\left(\frac{\boldsymbol{z} - \boldsymbol{z}_I}{\ell_I}\right) = 0, \tag{11}$$

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

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

obtained by developing $(\boldsymbol{z} - \boldsymbol{z}_q)^{\alpha}$ 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

Table 1: Reference parameters

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

Table 2: Nominal assimilation and simulation parameters

Parameters	Values
time step	$dt = 0.005$
final time	$tf = 10$
std. observation	$\sigma_{obs} = 5.0e^{-2}$
vorticity threshold	$\varepsilon_{\omega} = 1.0e^{-4}$
particle characteristic length	$dp = \frac{2\pi}{256} \approx 0.01227$
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 3: 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] 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.

[6] I. P. Christiansen. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics*, 13(3):363–379, 1973.

[7] G.-H Cottet and P. Koumoutsakos. *Vortex methods - theory and practice.* March 2000.

[8] G.-H. Cottet and S. Mas-Gallic. A particle method to solve the Navier-Stokes system. *Numerische Mathematik*, 57(1):805–827, December 1990.

[9] G.-H. Cottet, M.-L. Ould Salihi, and M. El Hamroui. Multi-purpose regridding in vortex methods. *ESAIM: Proceedings*, 7:94–103, 1999.

[10] 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.

[11] 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.

[12] 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.

[13] B. Fornberg and N. Flyer. Solving pdes with radial basis functions. *Acta Numerica*, 24:215–258, 2015.

[14] 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.

[15] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960.

[16] M. Le Provost, R. Baptista, Y. Marzouk, and J. Eldredge. *A low-rank nonlinear ensemble filter for vortex models of aerodynamic flows*. January 2021.

[17] L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82:1013–1024, 1977.

[18] C. Mimeau and I. Mortazavi. A Review of Vortex Methods and Their Applications: From Creation to Recent Advances. *Fluids*, 6(2):68, February 2021.

[19] J. J. Monaghan. Extrapolating B splines for interpolation. *Journal of Computational Physics*, 60(2):253–262, September 1985.

[20] P. Orlandi. Vortex dipole rebound from a wall. *Physics of Fluids A: Fluid Dynamics*, 2(8):1429–1436, August 1990.

[21] G. Russo and J. A. Strain. Fast triangulated vortex methods for the 2d euler equations. *Journal of Computational Physics*, 111:291–323, 1994.

[22] Y. Sasaki. An ObJective Analysis Based on the Variational Method. *Journal of the Meteorological Society of Japan. Ser. II*, 36(3):77–88, 1958.

[23] 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.

[24] P. Sperotto, S. Pieraccini, and M. A. Mendez. A meshless method to compute pressure fields from image velocimetry. *Measurement Science and Technology*, 33, 2021.

[25] D. Sulsky, Z. Chenb, and H.-L. Schreyer. A particle method for history-dependent materials. page 18, 1994.

[26] O. Talagrand. Assimilation of observations, an introduction. *Journal of the Meteorological Society of Japan*, 75(1B):191–209, 1997.

[27] 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.