

# Ensemble Data Assimilation for Meshless Methods

Marius Duvillard<sup>1,2</sup>, Loïc Giraldi<sup>1</sup>, and Olivier Le Maître<sup>3</sup>

<sup>1</sup>CEA, DES, IRESNE, DEC, SESC, LMCP, Cadarache, F-13108

Saint-Paul-Lez-Durance, France

<sup>3</sup>CNRS, Inria, Centre de Mathématiques Appliquées, Ecole Polytechnique,  
IPP, Route de Saclay, 91128, Palaiseau Cedex, France

<sup>2</sup>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 (EnKF). Combining EnKF with meshless methods facilitates an efficient estimation across various problems utilizing a Lagrangian solution discretization. Two specific methodologies have been introduced to either complete the analysis on a new grid of particles or on the forecast particle discretization of each member. These methods were initially evaluated using a one-dimensional advection-diffusion model with periodic boundaries. Subsequently, they were applied to a more complex two-dimensional inviscid flow problem, solved via the Vortex-In-Cell (VIC) method. In the one-dimensional scenario, the performance of these filters was benchmarked against a grid-based assimilation filter. In the two-dimensional case, the study demonstrated the feasibility of applying these methods in more intricate scenarios.

**Keywords:** Meshless Methods, Particle-based Method, Data Assimilation, EnKF, Ensemble Methods, Vortex Methods.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Background</b>	<b>5</b>
2.1	Data assimilation . . . . .	5
2.1.1	Data assimilation setting . . . . .	6
2.1.2	Bayesian filtering . . . . .	6

2.1.3	Ensemble Kalman Filter . . . . .	7
2.1.4	Expanded State Model . . . . .	9
2.2	Particle-based Methods . . . . .	9
2.2.1	Particle discretization . . . . .	9
2.2.2	Exemple of kernel functions . . . . .	10
2.3	Particle-based function manipulations . . . . .	12
2.3.1	Approximation operator . . . . .	12
2.3.2	Regression operator . . . . .	12
2.3.3	Remeshing operator . . . . .	13
<b>3</b>	<b>Methods</b>	<b>15</b>
3.1	Remesh-EnKF . . . . .	16
3.2	Particles-EnKF . . . . .	16
3.3	Complexity . . . . .	17
<b>4</b>	<b>Applications</b>	<b>19</b>
4.1	1D density advection-diffusion problem . . . . .	19
4.1.1	Description of the problem . . . . .	19
4.1.2	Assimilation parameters and ensemble generation . . . . .	20
4.1.3	Results . . . . .	21
4.1.4	Expanded state . . . . .	24
4.2	2D vortex-in-cell problem . . . . .	25
4.2.1	Description of the method . . . . .	25
4.2.2	Lamb-Chaplygin dipole and simulation parameters . . . . .	27
4.2.3	Assimilation parameters and ensemble generation . . . . .	29
4.2.4	Results . . . . .	31
<b>5</b>	<b>Conclusion</b>	<b>37</b>
<b>A</b>	<b>Parameters</b>	<b>38</b>

# 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 are a tool to understand and design processes, particularly in the mechanical field. The conventional approach to numerical analysis has historically leaned on grid-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. On one hand, particles can represent a discrete medium. Particles are individual entities with kinematic properties that interact locally and balance multi-body equilibrium. The Discrete Element Method (DEM), first introduced by Condall and Strack [14], has gained much popularity in modeling granular materials. On the other hand, particles can discretize a continuum medium and are associated with shape functions to reconstruct continuous fields and differential operators. The Smoothed Particle Hydrodynamic (SPH), independently introduced by Gingold, Monaghan, and Lucy [21, 26], is one of the first continuum particle-based methods. It associates a kernel to each particle to approximate the continuous fields and the derivative operator to solve the strong form of the equilibrium equation. It has first been applied to stellar models but also to fluid dynamics. The Material Point Method (MPM) introduced by Sulsky [35] is another particle-based that is part of the Particle-In-Cell family, like Fluid Implicit Particle [8] (FLIP) introduced an auxiliary grid to project, solve, and interpolate back the solution on the particles.

The computed solution involves errors that must be understood, quantified, and reduced. Uncertainty is a fundamental aspect of scientific inquiry and modeling. It often arises when our knowledge is limited or incomplete. This uncertainty can manifest in various forms, such as the ambiguity surrounding the value of a model parameter, the vagueness regarding initial conditions, or the uncertainty in setting boundary conditions or external forces. Moreover, if numerical models usually bring essential physical principles, they involve some simplifications. The numerical error appears due to the algorithm and discretization. Besides, it extends to the uncertainty associated with forthcoming experimental measurements calibrating numerical models.

Data assimilation is a crucial methodology in scientific research, especially in complex and chaotic systems. Its fundamental purpose lies in combining different sources of information to obtain a better estimate of the system state. Integrating model predictions and observational data has found extensive application in disciplines such as meteorology, oceanography, hydrology, and geosciences [5].

In the domain of data assimilation, two prominent families of approaches have emerged: variational and stochastic methods. Variational approaches 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 used formulations are 3D-VAR, 4D-VAR, and incremental 4D-VAR. 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 [24] 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 [18] mainly due to its adaptability to high dimensional problems with any evolution model. 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 particle-based simulation 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 primarily revolve around defining a unified state representation across all ensemble members and effectively updating this state during the analysis phase.

Meshless methods are more or less sensible to these issues. For particle-based methods like the Discrete Element Method [14] (DEM), the update phase is challenging due to interpenetration issues. For the classic soft-sphere approach introduced, the interaction is dependent on the geometry of the particle. Moving one particle to another leads to a complex global nonlinear optimization problem. In [9], an EnKF algorithm has been applied to a DEM simulation to study the Sea Ice flow. However, some simplifications have been introduced, like a new parametrization to reduce the number of particles, and changing particle positions have mild stability implications.

In contrast, particle-based methods in continuum discretization treat particles as point entities, thereby circumventing interpenetration issues. 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 [36, 12]. 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 [33, 7]. The selection of the reference and updated grids provides a spectrum of implementation possibilities. Furthermore, Siripatana et al. [33] 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. Darakananda et al. [15]. Complex filters have been developed to estimate correctly the posterior discretization based on a nonlinear observation model or a deficient number of pressure sensors [25]. 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 parti-

cles 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 types of methods 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 [27]. It mainly involved iterative methods [3], triangulation [32] or Radial Basis Function (RBF) interpolation [1, 34]. The last ones offer to easily introduce new particles or introduce penalization to regularize optimization problems.

Based on those different formulations, we propose two types of adaptation of the EnKF filter. First, the Remesh-EnKF uses a new reference particle discretization. This way, the state could be updated, and the number of particles is controlled. This first method is based on the regridding of the particle discretization as described by [12] on which the classical EnKF analysis could be performed. Then, in a case where the particle discretization would be preserved, the Particle-EnKF is introduced. In this case, the analyzed field is approximated with the previous particle discretization. The particle's positions are unchanged; only the strengths are modified by regression. 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 4.1, and an incompressible viscous flow is solved using a Vortex Method 4.2 where the filters are quantitatively analyses.

## 2 Background

In this introductory section, we will delve into two key aspects. Firstly, in Section 2.1, we will explore sequential data assimilation methods, focusing on the Ensemble Kalman Filter and the intricacies of calibration. Subsequently, in Section 2.2, our attention will shift to particle methods for addressing problems in continuous media. We will explain both particle discretization and essential techniques for filter development in Section 3.

### 2.1 Data assimilation

One type of data assimilation are Bayesian approaches. This family of methods introduced a probabilistic framework in order 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 simplified by the use of a hidden Markov chain model. We position ourselves within a finite-dimensional context. The state  $\mathbf{x} \in \mathbb{R}^d$ ,  $\mathbf{y} \in \mathbb{R}^m$  where  $d$  and  $m$  are the state and observation dimension. The forecast and observation are introduced such as  $\forall k \geq 0$ ,

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

where  $\mathcal{M}_{k,k+1}$  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  $\mathbf{x}_k$ , and  $\mathbf{y}_{k+1}$  and  $\mathbf{x}_{k+1}$  are conditionally independent with respect with  $\mathbf{x}_k$ , 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  $p_{\mathbf{x}_k|\mathcal{D}_k}$ , 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  $p_{\mathbf{x}_0}$ . Then, for all  $k \geq 0$ , probability distributions are propagated. The forecast step is obtained through the law of total probability

$$\begin{aligned} p_{\mathbf{x}_{k+1}|\mathcal{D}_k}(\mathbf{x}) &= \int p_{\mathbf{x}_{k+1}|\mathcal{D}_k, \mathbf{x}_k=\mathbf{x}'}(\mathbf{x}) p_{\mathbf{x}_k|\mathcal{D}_k}(\mathbf{x}') d\mathbf{x}', \\ &= \int p_{\mathbf{x}_{k+1}|\mathbf{x}_k=\mathbf{x}'}(\mathbf{x}) p_{\mathbf{x}_k|\mathcal{D}_k}(\mathbf{x}') d\mathbf{x}'. \end{aligned}$$

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

$$p_{\mathbf{Y}_{k+1}|\mathcal{D}_k}(\mathbf{y}) = \int p_{\mathbf{Y}_{k+1}|\mathbf{x}_{k+1}=\mathbf{x}}(\mathbf{y}) p_{\mathbf{x}_{k+1}|\mathcal{D}_k}(\mathbf{x}), d\mathbf{x}.$$

After the  $k + 1$  observation  $\mathbf{y}_{k+1}$ , the analysis step determines the a posteriori law of the state using Bayes law applied after measuring  $\mathbf{Y}_n$

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

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 one time increment.

### 2.1.3 Ensemble Kalman Filter

The Kalman filter [24] is the Bayesian filter that use, moreover the previous hypothesis, that  $\mathcal{M}_k$  and  $\mathcal{H}_k$  are linear operators. In this case, the posterior distribution of the state is still Gaussian, so only the mean and the variance are transmit. 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 [18]. The formulation uses an ensemble of discrete samples based on the assumptions of a multivariate Gaussian distribution, as for the Kalman filter. EnKF can be seen as a hybrid method between the Kalman filter and the particle filter. The forecast is performed in the same way as in the particle filter. Still, the analysis step is computed in Kalman's fashion with sample model error covariance. 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  $\{\mathbf{x}^i\}_{i=1}^N$ , 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

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

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

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

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

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

Respectively the matrix of observation and observation anomalies are  $\mathcal{Y}_f = [\mathcal{H}(\mathbf{x}_f^1), \dots, \mathcal{H}(\mathbf{x}_f^N)]$ ,  $\mathbf{Y}_f$  where columns are

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

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

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

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

$$\mathbf{X}_a = \mathbf{X}_f + \tilde{\mathbf{K}}(\mathbf{D} - \mathbf{Y}), \quad (1)$$

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  $(\mathbf{D} - \mathbf{Y})$  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.

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

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

$$\mathbf{X}_a = \mathbf{X}_f + \mathbf{X}_f \mathbf{F}, \quad \mathbf{F} = \mathbf{Y}_f^T (\mathbf{Y}_f \mathbf{Y}_f^T + \mathbf{R})^{-1} (\mathbf{D} - \mathbf{Y}). \quad (2)$$

where the matrix  $\mathbf{F}$  only depends on the ensemble members through the predicted observations ensemble  $\mathbf{Y}_f$  and the observation.

Consequently, the analysis could be entirely defined thanks to the ensemble predictions and the observation. Section 10.2 of the book [19] introduced various forms of the EnKF update where equation 10.2 is equivalent to 2.



### 2.1.4 Expanded State Model

The Bayesian calibration of model parameters is possible by defining an expanded state. Suppose we parametrize the dynamic operator  $\mathcal{M}(\mathbf{x}; \boldsymbol{\theta})$ . The parameter vector  $\boldsymbol{\theta} \in \mathbb{R}^q$  is then appended to the model state vector, and the model state is forecast with the parameter. The calibration is thus performed online, allowing a more accurate forecast prediction. The parameter is supposed to be constant in this article, meaning that the evolution model is simply identity, such as

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k.$$

The expanded state system model is then

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

where

$$\hat{\mathbf{x}}_k = \begin{pmatrix} \mathbf{x}_k \\ \boldsymbol{\theta}_k \end{pmatrix} \in \mathbb{R}^{n+q},$$

is the expanded state vector, with  $q$  the dimension of the parameter vector, and

The analysis step is then applies to  $\hat{\mathbf{x}}_k$ . This expression provide a derivative-free implementation of an inverse problem [22].

## 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  $\mathcal{P}$  brings quantities and the spatial coordinates  $\mathbf{z}_p$ .

We will focus our work on methods that discretize a solution on a continuous domain that can be defined with field  $\mathbf{u} : \Omega \in \mathbb{R}^d \rightarrow \mathbb{R}^n$  with  $\Omega$  the spatial domain,  $d$  is the space dimension and  $n$  the dimension of the solution. This includes methods like Smoothed particle hydrodynamics (SPH) [21, 26] and the Vortex Method (VM) [11] and is extended to other methods like the Material Point Method (MPM) [35].

### 2.2.1 Particle discretization

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

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

with  $\delta$  the Dirac delta distribution.

A kernel function  $\phi_\varepsilon$  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  $\varepsilon$  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  $\Omega$  is subdivided with  $N_p$  subdomain  $\Omega_p$  associated with a lagrangian particle in the location  $\mathbf{z}_p \in \Omega_p$ . We denote by  $V_p$  the volume of  $\Omega_p$ . This discretization is then used to approximate the average function such that

$$\begin{aligned} \langle \mathbf{u}(\mathbf{z}) \rangle &= \sum_p \int_{\Omega_p} \mathbf{u}(\mathbf{z}') \phi_\varepsilon(\mathbf{z} - \mathbf{z}') d\mathbf{z}' \\ &\approx \sum_p \mathbf{u}(\mathbf{z}_p) V_p \phi_\varepsilon(\mathbf{z} - \mathbf{z}_p) \\ &\approx \sum_p \mathbf{U}_p \phi_\varepsilon(\mathbf{z} - \mathbf{z}_p). \end{aligned}$$

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

Based on this discretization, the differential operator could be derived through this formulation.

### 2.2.2 Exemple of kernel functions

Several kernels have been used depending on the method. The original formulation of MPM did not use a substitute kernel and wrote the density such as

$$\mathbf{u}(\mathbf{z}) = \sum_p \mathbf{U}_p \phi_\varepsilon(\mathbf{z} - \mathbf{z}_p)$$

And the resolution is based on a projection on a background grid associated with some shape function [35].

The GIMP method is a different formulation that uses the Heaviside function [2] and thus associates a volume around each particle

$$M_1(r) = \frac{\alpha}{\varepsilon} \begin{cases} 1; & r \leq \varepsilon \\ 0; & \text{otherwise} \end{cases}$$

where  $r = \|\mathbf{z}\|_2$ .

This method have been introduced to avoid the cell-crossing issue when a particle moves from one cell to another through the background grid.

In SPH associate, as this name suggests, a smooth kernel to approximate the solution. 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.

For instance, the quadratic B-spline, we called  $M_3$  defined with

$$M_3(r) = \frac{\alpha}{\varepsilon^d} \begin{cases} \frac{3}{4} - |q|^2 & 0 \leq |q| < \frac{1}{2} \\ \frac{1}{2} \left( \frac{3}{2} - |q| \right)^2 & \frac{1}{2} \leq |q| < \frac{3}{2} \\ 0 & \frac{3}{2} \leq |q| \end{cases} \quad (3)$$

with  $r = \|\mathbf{z}\|_2$  and  $q = r/\varepsilon$  and  $\alpha$  the normalization condition and  $d$  the spatial dimension. This kernel ensures the  $C^1$  continuity. The cubic kernel is another B-Spline kernel which is

$$M_4(r) = \frac{\alpha}{\varepsilon^d} \begin{cases} \frac{1}{6}(-|q| + 2)^3 - \frac{4}{6}(-|q| + 1)^3 & 0 \leq |q| \leq 1 \\ \frac{1}{6}(-|q| + 2)^3 & 1 \leq |q| \leq 2 \\ 0 & 2 \leq |q| \end{cases} \quad (4)$$

In this last case, the normalization factor  $\alpha$  is

$$\alpha = \begin{cases} 1; & 1\text{d} \\ 30/14\pi; & 2\text{d} \\ 3/2\pi; & 3\text{d} \end{cases}$$

Note that, those kernel have been define with the radial coordinate  $r$ . Another possibility would be to define the multidimensional kernel as the tensor product of the 1d kernel

on the normalized coordinate. This is what is used for the following application of the regridding operation 2.3.3 or the transfer define between particle and grid in the MPM scheme [35, 23].

For the periodic boundary problem described in section 4.1, we define an equivalent kernel function  $\phi_g^P = \sum_{n=-\infty}^{+\infty} \phi_g(r - nL)$ , where  $L$  is the periodic length. All the kernel properties are still verified on a single period.

## 2.3 Particle-based function manipulations

One of the main drawbacks of Lagrangian methods is that they can produce a highly distorted distribution of particles. To conserve the overlapping of particle shape function, and enable communication between particles, two main techniques have been introduced to reconstruct a new particle discretization, or to adapt the particle strengths to better fit the current field. The different operators will be introduced in the assimilation process in order to update particle solution of each members.

### 2.3.1 Approximation operator

The first category of solutions aims to improve the approximation of the field by modifying particle strength. For any particle location,  $\mathbf{z}_p$ , the continuous field could be evaluated  $\mathbf{u}(\mathbf{z}_p)$ .

A first approximation could be to use the particle approximation to reevaluate the particle intensities like in Equation (3) such as

$$\mathbf{U}_p = \int_{\Omega_p} \mathbf{u}(z) d\mathbf{z} = \mathbf{u}(\mathbf{z}_p) V_p.$$

This approximation is easily computable but do not ensure the conservation of the all the moment of the field.

### 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 obtain by solving a linear system based on the minimization of 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 compute on each particle positions  $\mathbf{z}_p$  given

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

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

Find 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

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

In this case, the solution is  $\mathbf{U} = (\Phi^T \Phi)^{-1} \Phi \mathbf{u}$ . 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 Tikhonov penalization term. The Ridge regression we used introduced a penalization on of the form  $\lambda \|\mathbf{U}\|_2^2$ , where  $\lambda$  is a penalization coefficient, such as the new problem is

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

given the following solution  $\mathbf{U}_{p,\text{ridge}} = (\Phi^T \Phi + \lambda^2 \mathbf{I})^{-1} \Phi \mathbf{u}$ .

### 2.3.3 Remeshing operator

A second type of method is based this time on a complete projection of the solution on a new regular grid of particles. The technique pioneered by Cottet and Koumoutsakos [11, 12] in the Vortex Method also plays a crucial role in the Particle-in-Cell (PIC) method for transferring particle intensities to a grid, as observed in methods such as the Material Point Method (MPM) [35, 16, 23].

Those method are based on a redistribution of the old intensities to the new regular distributed particle grid thanks to a kernel of redistribution  $W$ .

For the one dimensional case, we denote by  $z_I$  and  $z_p$  respectively the grid and the old particle location. The new particles are defined on a grid with regular spacing  $\ell_I$ . We define the particle intensities with  $\mathbf{U}_I$  and  $\mathbf{U}_p$ . By using the redistribution kernel  $W$  through a classical interpolation rule, the new particle intensities are

$$\mathbf{U}_p = \sum_I \mathbf{U}_I W \left( \frac{z_I - z_p}{\ell_I} \right).$$

The kernel  $W$  determines the type and quality of interpolation. First, the partition of unity is required

$$\sum_p W \left( \frac{z_I - z_p}{\ell_I} \right) \equiv 1$$

which leads to the conservation of the first order moment

$$\begin{aligned}
\sum_p U_p &= \sum_p \sum_I U_I W \left( \frac{z_I - z_p}{\ell_I} \right) \\
&= \sum_I U_I \sum_p W \left( \frac{z_I - z_p}{\ell_I} \right) \\
&= \sum_I U_I
\end{aligned}$$

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

$$\sum_I (z - z_I)^\alpha W \left( \frac{z - z_I}{\ell_I} \right) = 0, \quad (5)$$

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

$$\sum_I z_I^\alpha W \left( \frac{z_p - z_I}{\ell_I} \right) = z^\alpha,$$

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. For instance, if  $m=2$ , we obtain that

$$\begin{aligned}
\sum_p U_p (z - z_p) &= \sum_p \sum_I U_I W \left( \frac{z - z_I}{\ell_I} \right) (z_p - z_I) \\
&= \sum_p \sum_I U_I [(z - z_I) + (z_I - z_p)] W \left( \frac{z - z_I}{\ell_I} \right) \\
&= \sum_I U_I (z_I - z_p) + \sum_p (z - z_p) W \left( \frac{z - z_I}{\ell_I} \right) \\
&= \sum_I U_I (z - z_I).
\end{aligned}$$

As a redistribution kernel, one may use the piecewise linear interpolation function

$$W(z) = \begin{cases} 1 - |z| & 0 \leq |z| < 1 \\ 0 & 1 \leq |z|, \end{cases}$$

which ensure the conservation of moment 0.

For higher degrees of freedom, the B-spline function provides a smoothing function for higher order. For instance, the quadratic B-splines defined in 3 ensure moment conservation by respecting the property of equation 5.

However, if higher order B-spline improves the smoothness of the solution, their accuracy is limited to second order and can only interpolate exactly linear functions.

Monaghan [28] proposes a systematic way to increase the accuracy and maintain the smoothness based on the extrapolation. The idea is to construct a new kernel 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 to not be positive.

Finally, for multidimensional space, the redistribution kernel  $W$  could be obtain as the product of the one dimensional kernel applied to each coordinate. such as

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

### 3 Methods

This section presents the development of ensemble data assimilation techniques for particle-based simulations.

If the forward step is still the same in our approach as in the classical Ensemble Kalman Filter, the main challenge resides in the update in the analysis step.

Based on the analysis formula 1 and the two families of transformation 2.3 on the particle discretization, we propose two different EnKF-adapted filters.

The first common step is to compute the correction matrix  $\mathbf{F}$  defined in 2, which defines the linear combination of the members. This step is possible because the analysis is independent of the state discretization thanks to the observation matrix-free implementation. Thus the analysis depends only on the observation  $\mathbf{y}$ , the predictive observation  $\{\mathbf{h}_i\}_{i=1}^{N_{obs}}$  and the noise sample  $\{\boldsymbol{\epsilon}_i\}_{i=1}^{N_{obs}}$ .

The differences come with the state update process and the new state definition. On the one hand, the Remesh-EnKF filter 3.1 uses new particle discretization, which will be the same for all the members. On the other hand, the particle positions are still the same for the Particles-EnKF filter 3.2, but the intensities will be updated.

The Remesh-EnKF uses a regridding process to update the particle quantities directly. The second one is intended to keep as much as possible the forward member discretization based on the approximation operator.

The choice of the filter will depend to the

### 3.1 Remesh-EnKF

The first method consists of redefining the discrete discretization on a new common one to perform the update with the transformation described in 2.3.3. After the remeshing, the previous discretization is lost, and the simulation continues with the new set of particles. After the forecast step, the particle discretization for a member  $i$  is defined by the set  $\left\{(\mathbf{z}_{ip}, \hat{\mathbf{U}}_{ip}^f)\right\}_{p=1}^{N_i}$  with  $N_i$  the number of particles. After the remesh transformation 3.1, a new particles discretization on the regular grid is obtained and gives  $\left\{(\mathbf{z}_q, \mathbf{U}_{iq}^f)\right\}_{q=1}^{N_g}$  where  $(\mathbf{z}_q, \mathbf{U}_{iq}^f)$  are the new coordinates and intensities. The state of the forecast member could be directly expressed as the vector of intensities  $\mathbf{U}_i^f = [\mathbf{U}_1^f, \dots, \mathbf{U}_{N_g}^f]^T$  with the ordered particle intensities. The linear combination is directly performed on particle intensities for each member  $i = 0, \dots, N$  such as

$$\mathbf{U}_i^a = \mathbf{U}_i^f + \sum_j \mathbf{F}_{ij} \mathbf{U}_j^f$$

.

### 3.2 Particles-EnKF

The Particles-EnKF formulation will not change the forward particle discretization. Each member will keep the same particle positions during the analysis step. The particle volumes do not change also. The only change will 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.

In this method, the analysis functions  $u_i^a$  are approximate. The analysis fields are obtained at any spacial coordinates thanks to the particle approximation of each member field  $u_i^f$  such as  $\forall \mathbf{z} \in \Omega$

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

**mettre une petite remarque sur l'équivalence des formulations.** Note that the ensemble  $u_i^a$  is the same as obtain by updating the intensity in Remesh-EnKF.

Thus, solutions are also described on a particle discretization  $Z_i^a = \bigcup_k Z_k^f$ . Such as

$$\mathbf{u}_i^a(\mathbf{z}) = \sum_p \mathbf{U}_{ip}^f \varphi_{ip}(\mathbf{z}) + \sum_j \mathbf{F}_{ij} \sum_p \mathbf{U}_{jp}^f \varphi_{jp}(\mathbf{z}) \quad i = 1, \dots, N \quad (6)$$

One solution is to approximate this solution on the previous discretization such that  $Z_i^a = Z_i^f$  to avoid exponential growth of the number of particles.



This would be done thanks to agglomeration to reduce the number of particles like in [36], or with regression operation defined in section 2.3.1.

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.

### 3.3 Complexity

Before evaluating the accuracy of the two filters, we could first evaluate their computational complexity. In the Remesh-EnKF, the complexity is led by the remeshing step for the ensemble of size  $N$ . The analysis is just a matrix matrix multiplication.

The remeshing process, described in section 2.3.3, necessitates redistributing the particle strengths to a new grid of particles. A loop over the  $N_p$  particles is first performed. If the size of the kernel is finite of size  $N_k$ , the redistribution of one particle needs  $N_k^d$  kernel evaluations, where  $d$  is the space dimension. Thus, the complexity of the Remesh-filter is  $\mathcal{O}(NN_p N_k^d)$ .

Finally, the updated step needs to compute a matrix multiplication between two matrices of shape  $N^2$ , leading to a global complexity of  $\mathcal{O}(N^2 + NN_p N_k^d)$ .

On the other hand, the Particles-EnKF complexity is led by an evaluation of the analysis fields and the regression or approximation step. First, the  $N$  analyses fields have to be evaluated in the particle positions of each member. Due to the linear combination of the forecast field as described in equation 6 such as

$$\mathbf{u}_i^a(\mathbf{z}_k) = \sum_{p \in V_k} \mathbf{U}_{ip}^f \varphi_{ip}(\mathbf{z}_k) + \sum_j \mathbf{F}_{ij} \sum_{p \in V_k} \mathbf{U}_{jp}^f \varphi_{jp}(\mathbf{z}_k) \quad i = 1, \dots, N, \quad k = 1, \dots, N_{ip}.$$

A grid search strategy could be used instead. For each particle, a grid cell is determined from which it belongs. All the particles are classified in a grid of size  $N_g$ . For each position evaluation, the distance is evaluated only on the particles in the next cells. Because the kernel is usually symmetric, the evaluation could be divided by two.

A loop is performed over the  $N_g^d$  grid cells, and the distance of the particles of one cell is computed with the arround cells. We approximate the mean number of particles in each cell as  $N_{PIC} = \frac{NN_p}{N_g^d}$ . Thus, the complexity is estimated as  $\mathcal{O}(NN_p)$ .

Then, the particle strengths have to be computed. An approximation, as described in section 2.3.1, does not add complexity. However, a regression method, like in section 2.3.2, needs to solve  $N$  linear system of size  $N_p$ . If the system is already computed thanks to the computation of the distances, the resolution needs the inversion of a  $N_p$  matrix. This

step uses a conjugate gradient algorithm, the best solver for a sparse system. In this case, the complexity is about  $\mathcal{O}((N_0 + N_p)k)$  where  $k$  is the number of iterations and  $N_0$  is the non-zero value of the matrix to inverse, which is about  $\mathcal{O}(N_p)$ . Bringing everything together, the complexity is about  $\mathcal{O}(kNN_p)$ . Together the global complexity is about  $\mathcal{O}((k + 1)NN_p)$ .

## 4 Applications

### 4.1 1D density advection-diffusion problem

#### 4.1.1 Description of the problem

An initial exploration is conducted in a one-dimensional application to enhance the assessment of filter performance. Various filters are assessed in relation to a conventional Ensemble Kalman Filter (EnKF) utilized in a grid-based simulation, serving as a benchmark for both qualitative and quantitative evaluation. We define the following one-dimensional  $2\pi$ -periodic convection-diffusion problem such as

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

where  $z$  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  $2\pi$ -periodic heat kernel in one dimension, such as

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

Considering an initial condition characterized by a Gaussian shape expressed as  $u(z, 0) = \phi(z - z_0, Dt_0)$ , 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

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

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

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

For solving the convection-diffusion scheme, we employ the two steps of the viscous splitting algorithm outlined in Section 4.2. 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

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

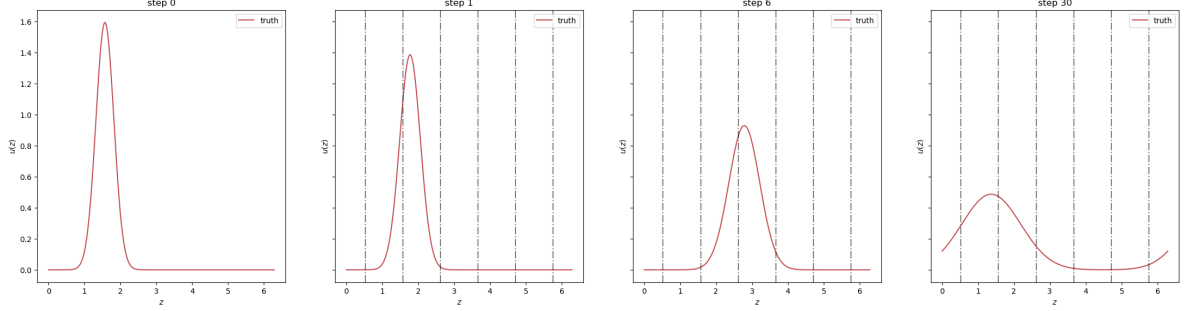


Figure 1: The analytical solution of the convection-diffusion problem evolves over time, with the final snapshot revealing a complete spatial period.

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) [17, 13] to approximate the laplacian term  $\frac{d^2 U_p}{dz^2}$ . Such as, the intensities  $U_p$  is 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  $\varepsilon$  the smoothing length of the Gaussian periodic kernel.

Our 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  $\varepsilon = 1.3h$ . On the contrary, the evolution model of the reference filter utilizes a central finite difference scheme on a grid with  $N_{grid} = 100$  nodes.

#### 4.1.2 Assimilation parameters and ensemble generation

We conduct  $N_{assim} = 30$  assimilation steps at evenly spaced intervals until the final time  $t_s = 2\frac{L}{v}$ . During each assimilation step, the field  $u$  is observed at six regularly spaced positions  $x_{obs}$ . The observational data is subject to additive noise, denoted as  $\eta \sim \mathcal{N}(0, \sigma_y \mathbf{I}_5)$ , where  $\sigma_y = 0.05$  and  $\mathbf{I}_5$  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  $\varepsilon_{mass}$  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.

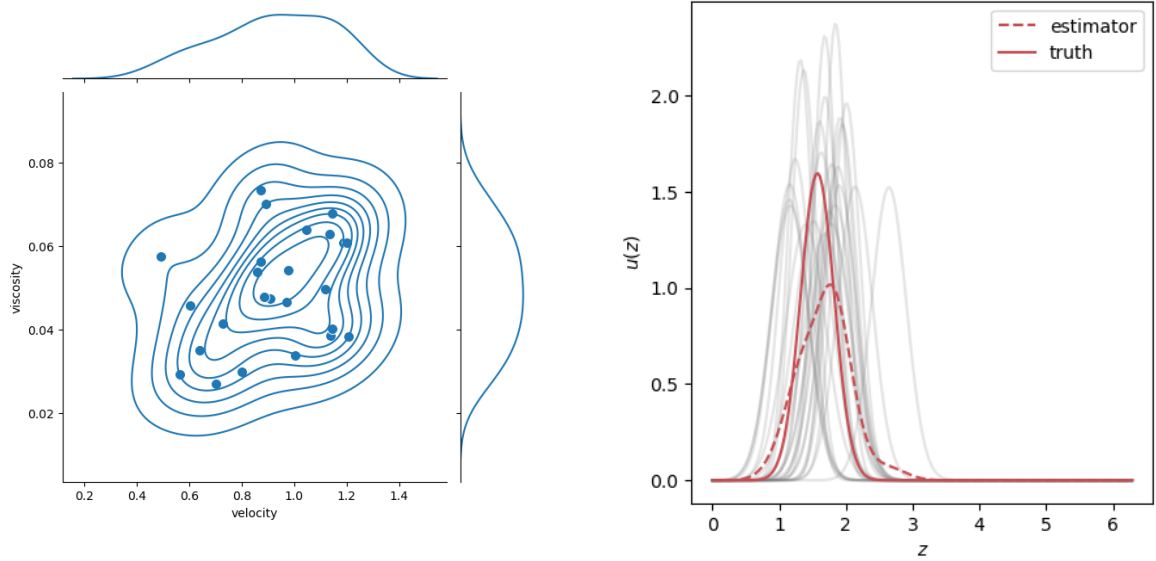


Figure 2: On the left the initial parameters sample,  $v$  in abscissa and  $D$  in ordinate. On the right is the initial ensemble state.

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

$$e_{L_2} = \frac{\mathbb{E} \left( \int_{\Omega} (u(z) - u^{EnKF}(z))^2 dz \right)^{1/2}}{\|u\|_{L_2}} \quad (7)$$

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

#### 4.1.3 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 three filters outlined in the Method section 3 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.

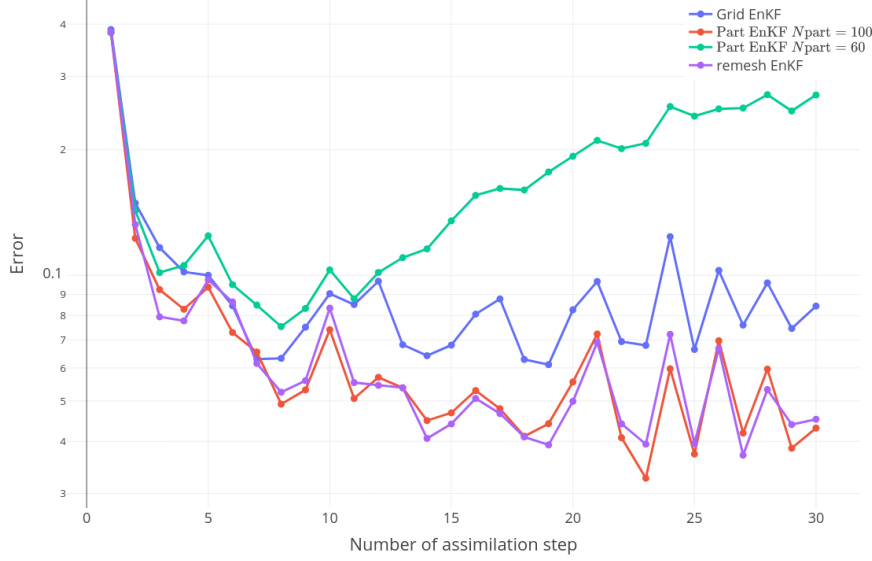


Figure 3: State error with respect to assimilation time step.

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 [20] 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's 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.

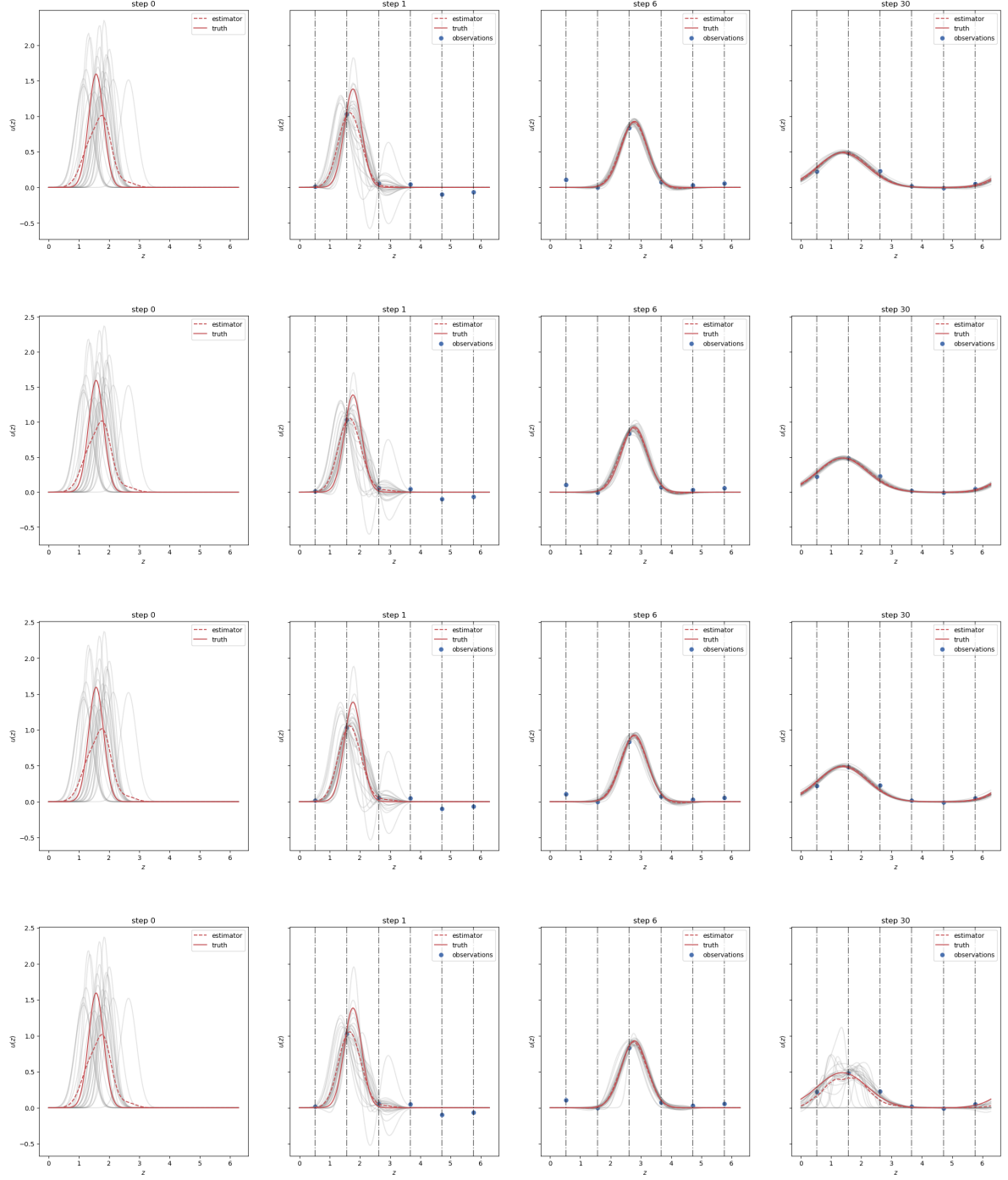


Figure 4: Data assimilation comparison over different filters. From top to bottom, with Grid Filter, RemeshEnKF, PartEnKF with  $N_{part} = 100$ , and PartEnKF with  $N_{part} = 60$ .

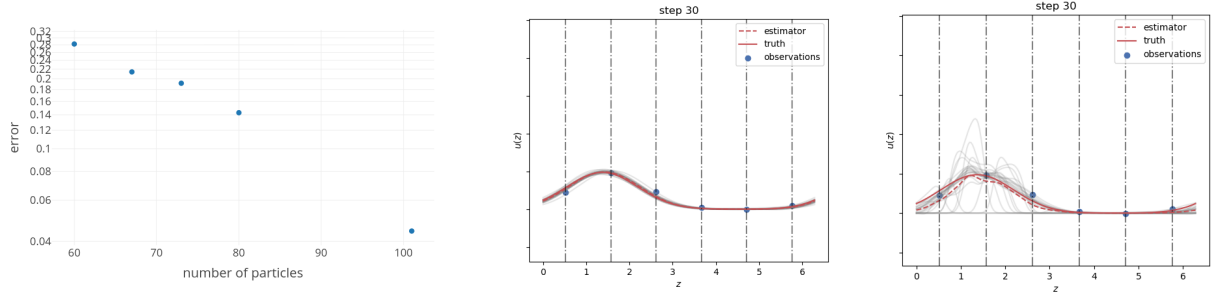


Figure 5: Left: Error with respect to particle support size, Middle: Final step for a support of 100 particles, Right: Final step for a support of 60 particles.

#### 4.1.4 Expanded state

In the case of an expanded state, we extend the update to parameters by introducing the couple  $(v, D)$  in the state definition, as explained in Section 2.1.4. This approach allows the calibration of these quantities. Figure 6d 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.

We also Quantitatively 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.



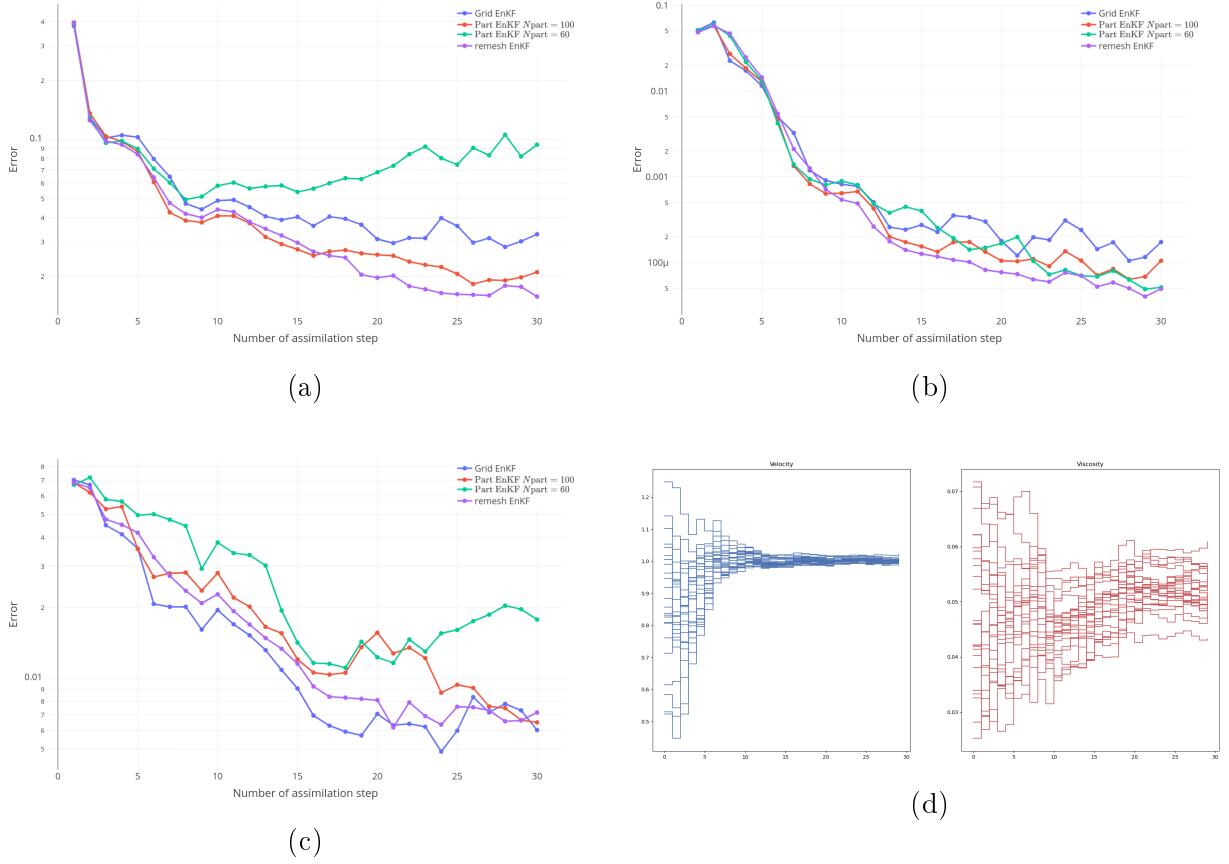


Figure 6: In a, b, c respectively state, velocity, and viscosity errors with respect to the assimilation time step. In d, the marginal distribution of velocity (left) and viscosity (right) over time (Remesh-EnKF).

## 4.2 2D vortex-in-cell problem

### 4.2.1 Description of the method

In this section, we apply the Vortex Method in a two-dimensional scenario, as outlined by Cottet et al. [11]. 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  $\omega$  denotes vorticity,  $\mathbf{v}$  represents velocity, and  $\nu$  stands for viscosity. In the context of 2D flow, vorticity is perpendicular to the flow plane, forming a scalar field denoted as  $\omega$ . 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  $\varphi_p$ , and a circulation  $\Gamma_p$ . For all points  $\mathbf{z}$  within the domain  $\Omega$ , the vorticity is expressed as

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

To address the Navier-Stokes equation, we employ a viscous splitting scheme, following the methodology outlined in [13], acknowledging the predominance of the convection term over viscosity. The initial phase involves solving the advection component for each particle in a Lagrangian context

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

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 [10, 4], 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.

$$\frac{d\mathbf{z}_p}{dt} = 0, \quad \frac{\omega_p}{dt} = \nu \Delta \omega(\mathbf{z}_p),$$

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

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

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

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

For further details on the computation, please refer to [13]. Consequently, our vortex model equation is solely dependent on the fluid's viscosity as a model parameter.

#### 4.2.2 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 [29]. 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  $\theta$ . 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  $\omega$  could be expressed as

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

where  $(r, \theta)$  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 7.

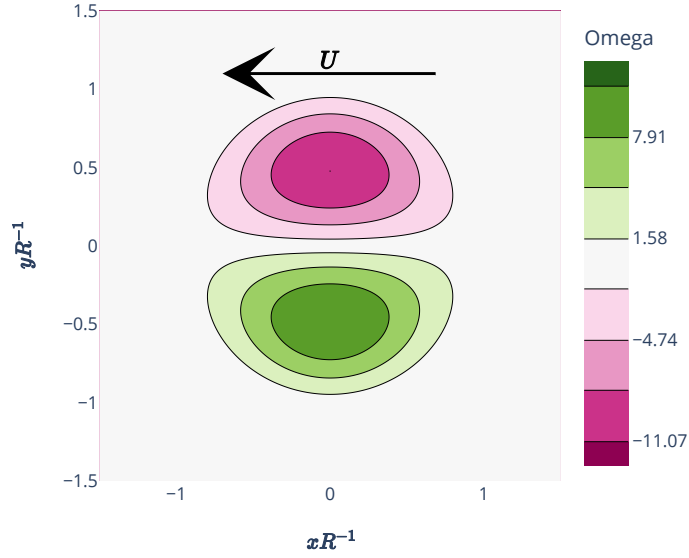


Figure 7: The Lamb-Chaplygin dipole vorticity field on a normalize space.

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  $0.25 \text{ m.s}^{-1}$ . 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 simulation the ground truth with a the vortex method for a fined discretization and fixed set of parameters described also in Table 1. This is typically referred to as a twin experiment. The trajectory of the ground truth is illustrate in the Figure 8 on a regularly spaced grid.



Figure 8: 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  $d_p$ , determining the particle volume as  $d_p^2$ . Another significant parameter is  $\varepsilon_\omega$ , associated with the remeshing process occurring either during the forecast (to prevent high distortion of the particle distribution) or during the Remesh-EnKF filter.  $\varepsilon_\omega$  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 9.

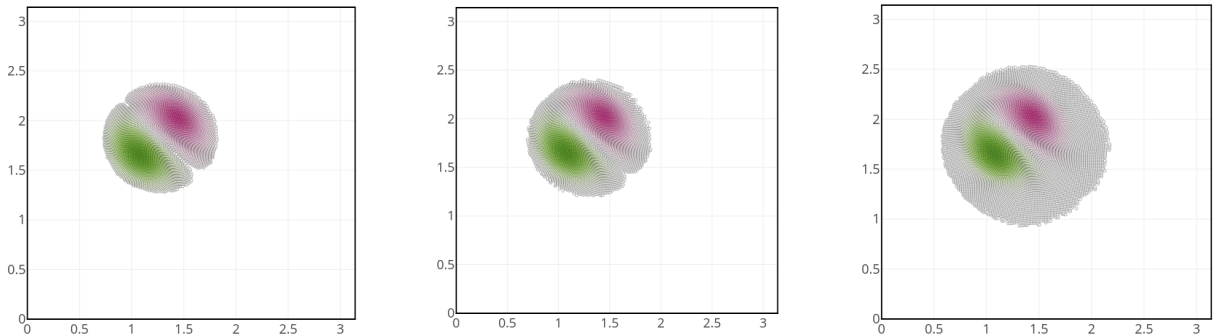


Figure 9: Effect of the parameter  $\varepsilon_\omega$  on the particle discretization of the solution for one member. From left to right, results for  $\varepsilon_\omega = 0.1, 0.01$ , and  $1.e - 6$ .

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

#### 4.2.3 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  $\theta$ , and the barycenter  $\mathbf{z}_{\text{mean}}$ . Additionally, the model viscosity  $\nu$  is also sampled. All the distributions are summarized in Table 3. The first six members are plotted in Figure 10.

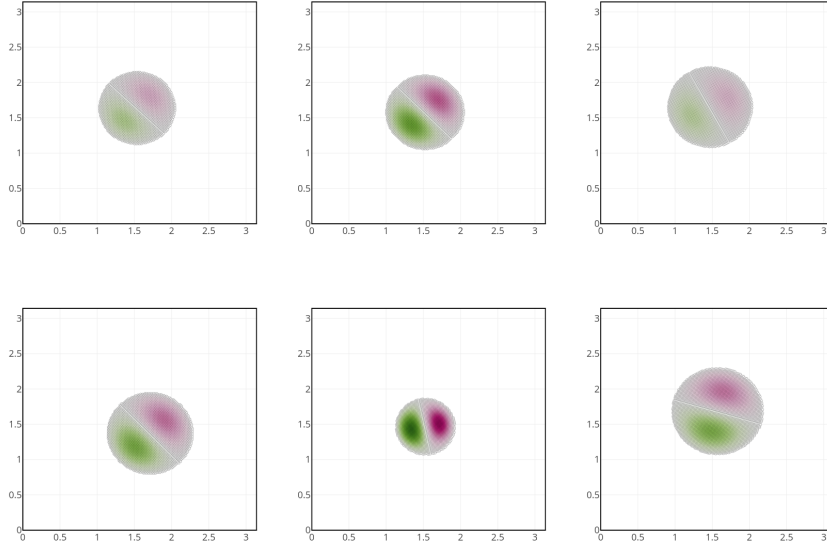


Figure 10: Six samples from the initial ensemble.

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. 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  $N_{\text{obs}}$ , measuring both components of the velocity. The observations follow a normal distribution  $\mathcal{N}(0, \sigma_{\text{obs}}^2 \mathbf{I} N_{\text{obs}})$ , indicating an ensemble of independent measurements, each characterized by a standard distribution of  $\sigma_{\text{obs}}$ . An example of observed velocity with and without noise is illustrated in Figure 11. We will employ the same metric, the relative  $L_2$ -error, similar to the one-dimensional application in 7 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  $f \in L_2$

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

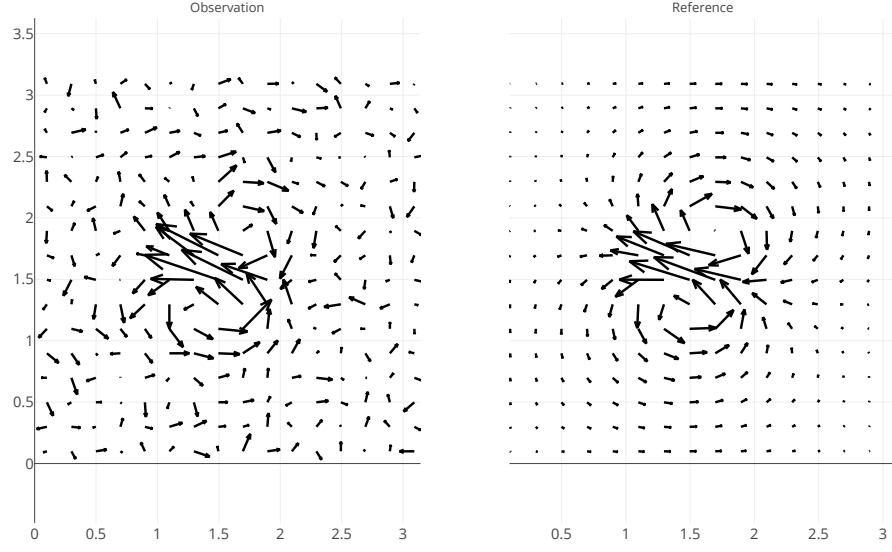


Figure 11: Observed and reference velocity fields. The error on each component is sample from a centred normal distribution with the nominal value  $\sigma_{\text{obs}} = 0.05$ .

where  $\mathbf{z}_c$  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 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.

#### 4.2.4 Results

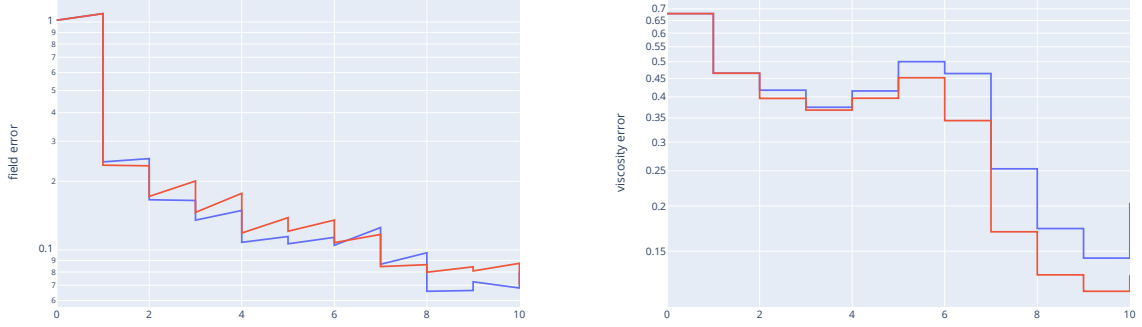


Figure 12: Error curves through assimilation steps. Left:  $L_2$ -error of the field, Right: Error for the viscosity parameter. With Part-EnKF in blue and Remesh-EnKF in red.

We start by analyzing the assimilation error over time. Figure 12 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 13 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  $1/\sigma_{\text{obs}}^2$ , the number of observations  $n_{\text{obs}}$ , the number of assimilation step  $N_{\text{assim}}$ . The results are regrouped in Figure 14. In this context, 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

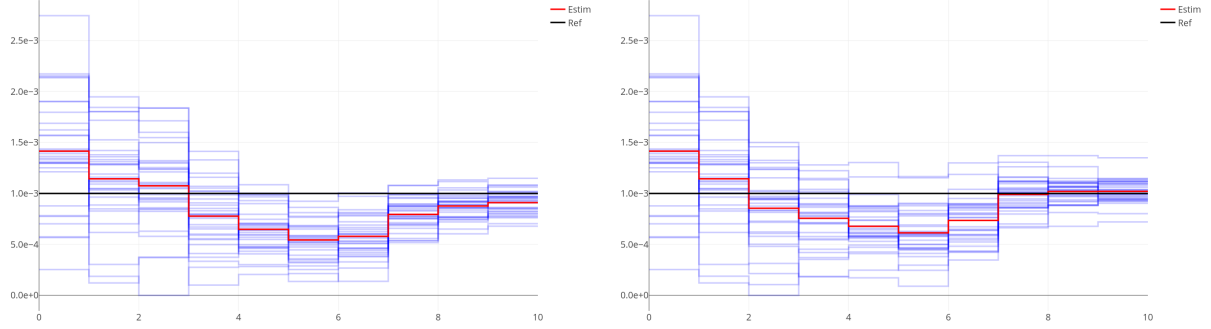


Figure 13: Evolution of the viscosity ensemble through assimilation. Left: for the Part-EnKF, Right: for the Remesh-EnKF. In blue the ensemble values, in red the estimate and black the reference.

assimilation parameters  $n_{\text{obs}}$ ,  $\sigma_{\text{obs}}$ , the conclusion is quite similar. The field error decreases with  $n_{\text{obs}}$ ,  $\sigma_{\text{obs}}$  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  $n_{\text{obs}}$ ,  $\sigma_{\text{obs}}$  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 15 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

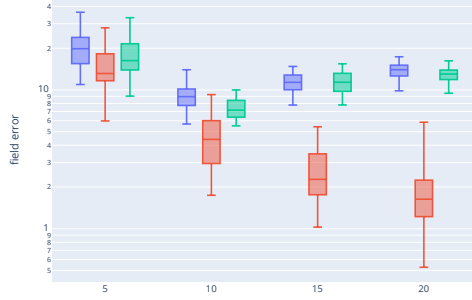




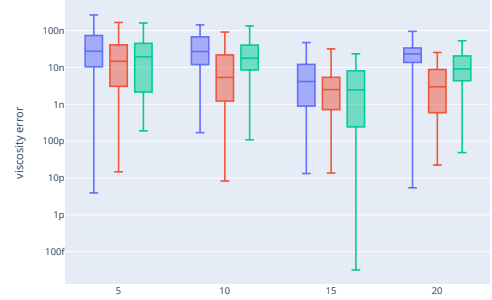
state error w.r.t.  $1/\sigma_{\text{obs}}^2$



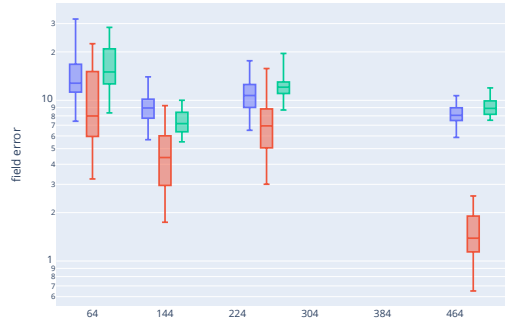
viscosity error w.r.t  $1/\sigma_{\text{obs}}^2$



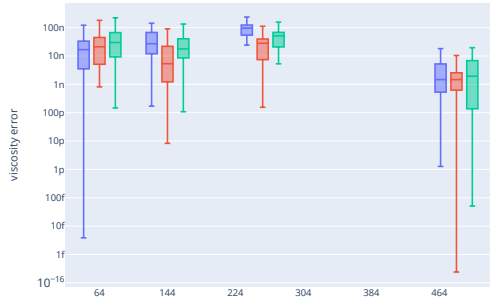
state error w.r.t.  $N_{\text{assim}}$



viscosity error w.r.t.  $N_{\text{assim}}$



state error w.r.t.  $N_{\text{obs}}$



viscosity error w.r.t  $N_{\text{obs}}$

Figure 14: 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.

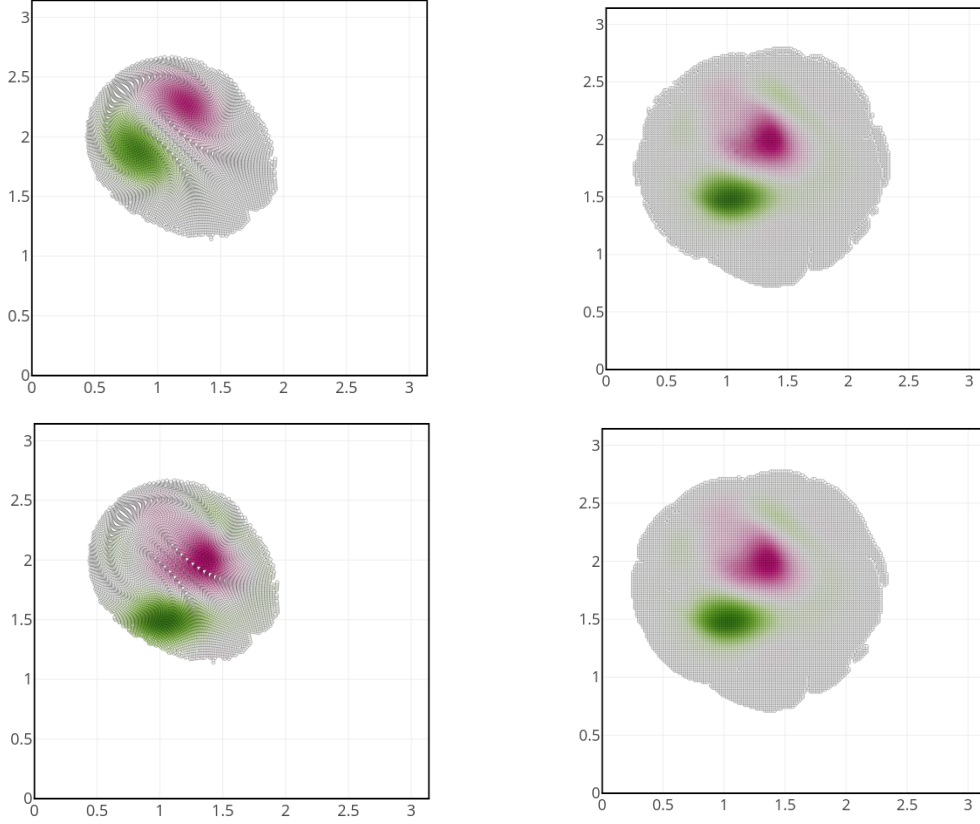
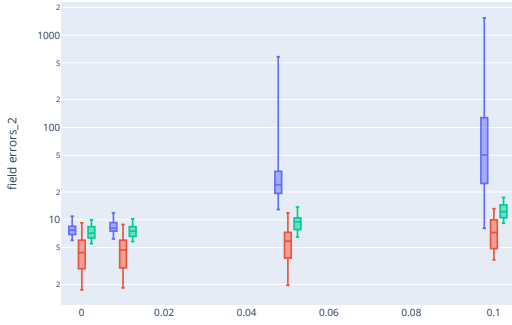


Figure 15: Assimilation of one member with a forecast discretization unadapted to the analyses solution. From left to right and up to down: the forecast member, the analyses member with the Remesh-EnKF filter, the analyses member with the Part-EnKF filter and finally the analyses member with the Part-Grid-EnKF. The forecast discretization use by the Part-EnKF is not always a well support of approximation for the analyses and introduce discretization errors.

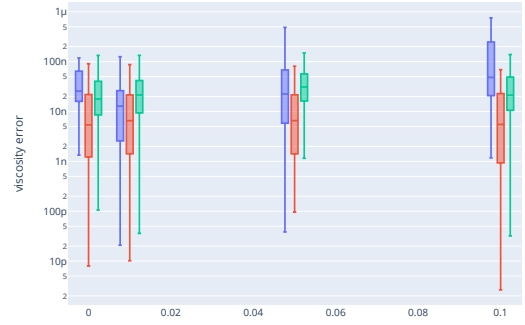
To evaluate the effect of the size of the support, we varied the value of  $\epsilon_\omega$ . We have seen in Figure 9 that this parameter affects the number of particles and, thus, the size of the support. We observe relatively bad results for very high  $\epsilon_\omega$  because the solution can 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 16. We

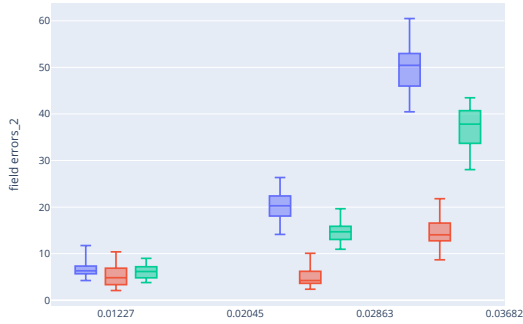
observe that the error for Part-EnKF and Part-Grid-EnKF increases proportionally with  $d_p$  as for the approximation error. The same order of magnitude for the different filters is observed for relatively small particle sizes. This confirmed the high effect of the particle approximation in this case. 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 4.1 and the choice of an adequate penalty coefficient to succeed in approximating the solution.



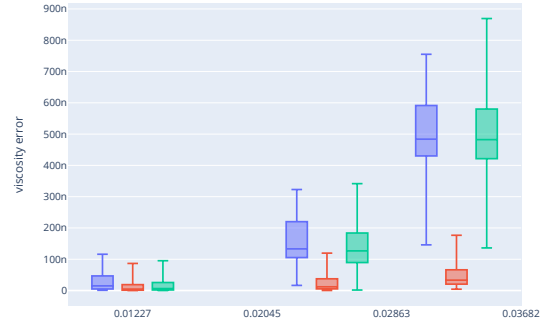
state error w.r.t.  $\varepsilon_\omega$



viscosity error w.r.t.  $\varepsilon_\omega$



state error w.r.t.  $d_p$



viscosity error w.r.t.  $d_p$

Figure 16: Error curves for simulation parameters. The effect of  $\varepsilon_\omega$  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 4.1, 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.

## 5 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 types of adapted Ensemble Kalman Filter schemes, relying on particle discretization to decompose a continuous solution into a sum of functions centered around them. These categories are dependent either on an update of the particle quantities or on remeshing the particle discretization on a common ensemble grid. The first strategy, named Part-EnKF evaluates the analysis field at particle locations to update the particle quantities. The second strategy, named Remesh-EnKF is based on the projection of every member onto a new common discretization of particles. In this way, the analysis is directly performed on the particle quantities. In any case, they share the common feature of evaluating the correction matrix independently of the state discretization. The different classes have been initially tested on a one-dimensional example and compared with a classical EnKF with a grid-based model. The results show comparability among the different filters, except for some 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 in 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 [6] or correction of the background ensemble alignment with the observation [30, 31] .

## A 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	$\mathbf{z}_{\text{ref}} = \left[\frac{\pi}{2}, \frac{\pi}{2}\right]^T$
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	$\sigma_{\text{obs}} = 5.0e - 2$
vorticity threshold	$\varepsilon_{\omega} = 1.0e - 4$
particle characteristic length	$dp = 0.01227$
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	$\theta \sim \mathcal{U}\left(\pi, \frac{\pi}{2}\right) (\text{rad.})$
barycenter	$z_{\text{mean},x} \sim \mathcal{N}\left(\frac{\pi}{2}, 0.1^2\right), \quad z_{\text{mean},x} \sim \mathcal{N}\left(\frac{\pi}{2}, 0.1^2\right)$
velocity	$U \sim \mathcal{U}(0, 0.5^2)$
viscosity	$v \sim \mathcal{N}(0.0015, 0.0005^2)$

## References

- [1] Lorena 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] S. Bardenhagen and Edward Kober. The Generalized Interpolation Material Point Method. *CMES - Computer Modeling in Engineering and Sciences*, 5, June 2004.
- [3] J. Thomas Beale. On the accuracy of vortex methods at large times. 1988.
- [4] Charles K Birdsall and Dieter Fuss. Clouds-in-clouds, clouds-in-cells physics for many-body plasma simulation. *Journal of Computational Physics*, 3(4):494–511, 1969.
- [5] Marc Bocquet. Introduction to the principles and methods of data assimilation in the geosciences. page 89, 2014.
- [6] Marc Bocquet, Pierre Vanderbeeken, Alban Farchi, Joffrey Dumont Le Brazidec, and Yelva Roustan. *Bridging classical data assimilation and optimal transport*. December 2023.
- [7] Bertrand Bonan, Nancy Nichols, Michael Baines, and Dale Partridge. Data assimilation for moving mesh methods with an application to ice sheet modelling. *Nonlinear Processes in Geophysics*, 24:515–534, September 2017.
- [8] 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.
- [9] Nan Chen, Quanling Deng, and Samuel N. Stechmann. Superfloe Parameterization with Physics Constraints for Uncertainty Quantification of Sea Ice Floes, March 2022. arXiv:2105.13569 [physics].
- [10] I.P Christiansen. Numerical simulation of hydrodynamics by the method of point vortices. *Journal of Computational Physics*, 13(3):363–379, 1973.
- [11] G.-H Cottet and Petros Koumoutsakos. *Vortex methods - theory and practice*. March 2000.
- [12] G.-H. Cottet, M.-L. Ould Salihi, and M. El Hamroui. Multi-purpose regridding in vortex methods. *ESAIM: Proceedings*, 7:94–103, 1999.
- [13] Georges-Henri Cottet and Sylvie Mas-Gallic. A particle method to solve the Navier-Stokes system. *Numerische Mathematik*, 57(1):805–827, December 1990.
- [14] P. A. Cundall and O. D. L. Strack. A discrete numerical model for granular assemblies. *Géotechnique*, 29(1):47–65, March 1979. Publisher: ICE Publishing.

- [15] Darwin Darakananda, Andre Fernando De Castro da Silva, Tim Colonius, and Jeff Eldredge. Data-assimilated low-order vortex modeling of separated flows. *Physical Review Fluids*, November 2018.
- [16] Alban de Vaucorbeil, Vinh Phu Nguyen, Sina Sinaie, and Jian Ying Wu. Material point method after 25 years: Theory, implementation, and applications. In *Advances in Applied Mechanics*, volume 53, pages 185–398. Elsevier, 2020.
- [17] 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.
- [18] Geir 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.
- [19] Geir Evensen, Femke C. Vossepoel, and Peter Jan van Leeuwen. *Data Assimilation Fundamentals: A Unified Formulation of the State and Parameter Estimation Problem*. Springer Textbooks in Earth Sciences, Geography and Environment. Springer International Publishing, Cham, 2022.
- [20] Bengt Fornberg and Natasha Flyer. Solving pdes with radial basis functions. *Acta Numerica*, 24:215–258, 2015.
- [21] 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.
- [22] Marco A. Iglesias, Kody J. H. Law, and Andrew M. Stuart. Ensemble Kalman methods for inverse problems. *Inverse Problems*, 29(4):045001, March 2013. Publisher: IOP Publishing.
- [23] Chenfanfu Jiang, Craig Schroeder, Andrew Selle, Joseph Teran, and Alexey Stomakhin. The affine particle-in-cell method. *ACM Transactions on Graphics*, 34(4):1–10, July 2015.
- [24] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, March 1960.
- [25] Mathieu Le Provost, Ricardo Baptista, Youssef Marzouk, and Jeff Eldredge. *A low-rank nonlinear ensemble filter for vortex models of aerodynamic flows*. January 2021.
- [26] Leon B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82:1013–1024, 1977.
- [27] Chloé Mimeau and Iraj Mortazavi. A Review of Vortex Methods and Their Applications: From Creation to Recent Advances. *Fluids*, 6(2):68, February 2021.



- [28] J. J Monaghan. Extrapolating B splines for interpolation. *Journal of Computational Physics*, 60(2):253–262, September 1985.
- [29] Paolo Orlandi. Vortex dipole rebound from a wall. *Physics of Fluids A: Fluid Dynamics*, 2(8):1429–1436, August 1990.
- [30] Sai Ravela, Kerry Emanuel, and Dennis McLaughlin. Data assimilation by field alignment. *Physica D: Nonlinear Phenomena*, 230(1-2):127–145, June 2007.
- [31] W. Steven Rosenthal, Shankar Venkataramani, Arthur J. Mariano, and Juan M. Restrepo. Displacement data assimilation. *Journal of Computational Physics*, 330:594–614, February 2017.
- [32] Giovanni Russo and John A. Strain. Fast triangulated vortex methods for the 2d euler equations. *Journal of Computational Physics*, 111:291–323, 1994.
- [33] 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.
- [34] Pietro Sperotto, Sandra Pieraccini, and Miguel Alfonso Mendez. A meshless method to compute pressure fields from image velocimetry. *Measurement Science and Technology*, 33, 2021.
- [35] D Sulsky, Z Chenb, and H L Schreyer. A particle method for history-dependent materials. page 18, 1994.
- [36] Yonghao Yue, Breannan Smith, Christopher Batty, Changxi Zheng, and Eitan Grinspun. Continuum Foam: A Material Point Method for Shear-Dependent Flows. *ACM Transactions on Graphics*, 34(5):1–20, November 2015.