A multigrid/ensemble Kalman filter strategy for assimilation of unsteady flows

G. Moldovan, G. Lehnasch, Laurent Cordier, M. Meldi

▶ **To cite this version:**

G. Moldovan, G. Lehnasch, Laurent Cordier, M. Meldi. A multigrid/ensemble Kalman filter strategy for assimilation of unsteady flows. Journal of Computational Physics, 2021, 443, pp.110481. 10.1016/j.jcp.2021.110481 . hal-03454796

HAL Id: hal-03454796
https://hal.science/hal-03454796

Submitted on 20 Nov 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A multigrid/ensemble Kalman filter strategy for assimilation of unsteady flows

G. Moldovan[a,*], G. Lehnasch[a], L. Cordier[a], M. Meldi[a]

[a] *Institut Pprime, CNRS - ISAE-ENSMA - Université de Poitiers, 11 Bd. Marie et Pierre Curie, Site du Futuroscope, TSA 41123, 86073 Poitiers Cedex 9, France*

Abstract

A sequential estimator based on the Ensemble Kalman Filter for Data Assimilation of fluid flows is presented in this research work. The main feature of this estimator is that the Kalman filter update, which relies on the determination of the Kalman gain, is performed exploiting the algorithmic features of the numerical solver employed as a model. More precisely, the multilevel resolution associated with the multigrid iterative approach for time advancement is used to generate several low-resolution numerical simulations. These results are used as ensemble members to determine the correction via Kalman filter, which is then projected on the high-resolution grid to correct a single simulation which corresponds to the numerical model. The assessment of the method is performed via the analysis of one-dimensional and two-dimensional test cases, using different dynamical equations. The results show an efficient trade-off in terms of accuracy and computational costs required. In addition, a physical regularization of the flow, which is not granted by classical KF approaches, is naturally obtained owing to the multigrid iterative calculations. The algorithm is also well suited for the analysis of unsteady phenomena and, in particular, for potential application to in-streaming Data Assimilation techniques.

Keywords: Kalman Filter, Data Assimilation, compressible flows

*Corresponding author, *gabriel-ionut.moldovan@ensma.fr*

1. Introduction

In Computational Fluid Dynamics (CFD), newly developed numerical methods are generally assessed in terms of accuracy via comparison with experimental data [1]. In practice, this validation step is far from being trivial since many sources of error are inevitably introduced in the simulations. First, the partial differential equations used to derive the numerical scheme may be restricted to oversimplified physical models, such as the Boussinesq approximation applied in thermal convection or the incompressibility condition. Second, the discretization process and the use of iterative numerical methods introduce computational errors in the representation of the flow features [2]. Third, boundary and initial conditions are usually very sophisticated in complex applications but detailed *a priori* knowledge is insufficiently available. Last, for very high Reynolds number configurations, turbulence/subgrid-scale modelling must be included in order to reduce the required computational costs [3]. All of these sources of error exhibit complex interactions owing to the non-linear nature of the dynamical models used in numerical application, such as the Navier-Stokes equations.

The experimental results are also affected by uncertainties and biases. In many cases, the set-up of the problem can be controlled up to a finite precision (alignment between the flow and the wind tunnel or immersed bodies, mass flow rate, ...). This kind of uncertainty, which is clearly affecting every physical system but cannot be exactly quantified, is usually referred to as *epistemic uncertainty*. In addition, experimental results are also affected by the precision (or bias) of the acquisition and measurement system. Thus, the main difficulty in the comparison between numerical and experimental results is understanding how much of the differences observed is due to an actual lack of precision of the numerical model, and how much is instead associated to differences in the set-up of investigation.

One possible strategy to obtain an optimal combination of the knowledge coming from simulations and experiments is to derive a state estimation which complies with both sources of information. The degree of precision of such esti-

mation is connected with the confidence in the sources of information. This has the advantage of naturally incorporating uncertainty and bias present in the sources of information in the analysis. Tools providing such state estimation are usually included in different disciplines, control theory for state observers [4], Data Assimilation (DA) [5, 6, 7] for weather prediction, ocean modelling and more recently mechanical engineering problems. Essentially, DA methods combine information issued from two sources: i) a *model*, which provides a dynamical description of the phenomenon in the physical domain, ii) a set of *observations*, which are usually sparse and/or local in time and space. These methods are classified in different families according to the way the state estimation is performed. One of the classical criterion of classification deals with the operative strategy used to derive the state estimation. *Variational* approaches resolve a constrained optimization problem over the parametric space characterizing the model (usually coefficients defining boundary conditions or physical models). The solution of the variational problem minimizes prescribed error norms so that the assimilated model complies with the observation provided over a specified time window. Methods from this family, which include 3D-Var and 4D-Var, usually exhibit very high accuracy [8, 9, 10, 11]. However, they are also affected by several drawbacks. First, the formulation of the adjoint problem that is introduced to perform the parametric optimization can be difficult, if not impossible, when automatic differentiation is not employed. Second, the adjoint problem is defined backward in time which may lead to numerical difficulties of resolution related to the unaffordable data storage that is needed, and the amplification of the adjoint solution that frequently happens when multi-scale interactions are dominant [5, 8, 12].

Another family of DA methods is represented by the *sequential* approaches. These methods, mostly based on Bayes' theorem, provide a probabilistic description of the state estimation. A well-known approach is the Kalman Filter (KF) [13]. Extensions and generalizations of this method have also been developed, such that the Extended Kalman Filter (EKF) [14] which is tailored for nonlinear systems, and the Ensemble Kalman Filter (EnKF) [6]. This class

of methods solves the state estimation problem by transporting error covariance matrices of the model and observations. These methods are usually more flexible than variational approaches (no required computation of first order sensitivities), but the time advancement and update of the covariance matrices are prohibitively expensive for large scale problems encountered in practical applications [15]. One possible strategy is reducing the order of the Kalman filter [16] or filtering the error covariance matrix. Inspired by a domain localization procedure, Meldi & Poux [17, 18] proposed a strategy based on an explicit filter of the error covariance matrix. The application of this estimator to different turbulent flows exhibited encouraging results considering the relatively small increase in computational resources. Another strategy for data assimilation for engineering applications is the Ensemble Kalman Filter [5, 6, 19], which relies on a Monte Carlo implementation of the Bayesian update problem. The EnKF (and follow-up models) was introduced as an extension of the original Kalman filter made for high-dimensional systems for which transporting the covariance matrix is not computationally feasible. EnKF replaces the covariance matrix by the sample covariance matrix computed from an ensemble of state vectors. The main advantage of EnKF is that advancing a high-dimensional covariance matrix is achieved by simply advancing each member of the ensemble. Several research works have been reported in the literature in recent years for application in fluid mechanics [20, 21, 22]. Statistical convergence is usually obtained for a typical ensemble size going from 60 to 100 ensemble members [5], which may still require prohibitive computational resources for realistic applications.

In addition, the state estimation obtained via sequential tools does not necessarily comply with a model solution i.e., the *conservativity* of the dynamical equations of the model is violated. This aspect is a potentially critical issue in fluid mechanics studies. Violation of conservativity may result in loss of conservation of some physical properties of the flow (such as mass conservation or momentum conservation) as well as in the emergence of non-physical discontinuities in the flow quantities. The aforementioned issues significantly affect the precision of the prediction of the flow and may eventually produce irreversible

instabilities in the time advancement of the dynamical model. A number of works in the literature have provided advancement in the form of additional constraints to be included in the state estimation process. Meldi & Poux [17] used a recursive procedure and a Lagrangian multiplier (the pressure field) to impose the zero-divergence condition of the velocity field for incompressible flows. Other proposals deal with imposing hard constraints in the framework of an optimization problem [23], ad-hoc augmented observation [24] and generalized regularization [25]. These approaches are responsible for a significant increase in the computational resources required, which is due to augmentation in size of the state estimation problem or to the optimization process, which usually needs the calculation of gradients of a cost function.

The investigation of physically constrained sequential state estimation is here performed using an advanced estimator strategy, which combines an EnKF approach and a multigrid method. For this reason, the algorithm will be referred to as Multigrid Ensemble Kalman Filter (MEnKF). Multigrid methods [26, 2] are a family of tools which employ multi-level techniques to obtain the time-advancement of the flow. In particular, the geometric multigrid [27] uses different levels of the resolution in the computational grid to obtain the final state. The method here proposed exploits algorithmic features of iterative solvers used in practical CFD applications. The EnKF error covariance matrix reconstruction is performed using information from a number of ensemble members which are generated over a coarse level mesh of a multigrid approach. While the main inspiration of this work deals with the integration of EnKF within the multigrid numerical algorithm, this procedure is reminiscent of reduced order / multi-level applications of EnKF strategy reported in the literature [28, 29, 30, 31]. The state estimation obtained at the coarse level and the associated ensemble statistics are used to obtain a single solution calculated on a high resolution mesh grid, similarly to the work by Debreu et al. [32] for variational DA. It will be shown that this procedure allows to i) reduce the computational costs of the EnKF and ii) ensure the conservativity and the smoothness of the final solution. In addition, owing to the algorithmic structure of the problem, all the

simulations on the fine and coarse level can be run simultaneously in parallel calculations, providing a tool able to perform in-streaming DA for unsteady flow problems.

The strategy is tested on different configurations, using several physical models represented by the Burger's equation and the compressible Euler and Navier-Stokes equations for one-dimensional and two-dimensional test cases.

The article is structured as follows. In Sec. 2, the sequential DA procedure is detailed including descriptions of the classical KF and EnKF methods. The numerical discretization and the multigrid strategy are also presented. In Sec. 3, the MEnKF algorithm is discussed. In Sec. 4, the MEnKF tool is used to investigate a one-dimensional case using the Burgers' equation. In Sec. 5, a second one-dimensional case is investigated, but in this case the dynamical model is represented by a Euler equation. In Sec. 6, we investigate the case of the two-dimensional compressible Navier-Stokes equations, with application to the spatially evolving mixing layer. Finally, in Sec. 7 concluding remarks are drawn.

2. Sequential data assimilation in fluid dynamics

In Sec. 2.1, we introduce sequential data assimilation methods starting with the Kalman filter. In Sec. 2.2, the procedures to transform a general transport equation into a discretized model usable in sequential DA are described. A brief description of the multigrid approach employed is also provided.

2.1. Sequential data assimilation

2.1.1. Kalman filter

The Kalman Filter (KF) provides an estimate of the state of a physical system at time k ($\mathbf{x}_k$), given the initial estimate $\mathbf{x}_0$, a set of observations, and the information of a dynamical model (e.g., first principle equations):

$$\mathbf{x}_k = \mathcal{M}_{k:k-1}\left(\mathbf{x}_{k-1}, \theta_k\right) + \eta_k \tag{1}$$

where $\mathcal{M}_{k:k-1}$ is a non-linear function[1] acting as state-transition model and θ_k contains the parameters that affect the state-transition. The term η_k is associated with uncertainties in the model prediction which, as discussed before, could emerge for example from incomplete knowledge of initial / boundary conditions. In the framework of KF applications, these uncertainties are usually modelled as a zero-mean Gaussian distribution characterized by a variance $\mathbf{Q}_k$, *i.e.* $\eta_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$. Indirect observations of $\mathbf{x}_k$ are available in the components of the observation vector $\mathbf{y}_k^{\mathrm{o}}$. These two variables are related by:

$$\mathbf{y}_k^{\mathrm{o}} = \mathcal{H}_k(\mathbf{x}_k) + \epsilon_k^{\mathrm{o}} \tag{2}$$

where $\mathcal{H}_k$ is the non-linear observation operator which maps the model state space to the observed space. The available measurements are also affected by

[1] Throughout this manuscript, we will use the standard notations employed in data assimilation studies. We will therefore make the difference between $\mathcal{M}_{k:k-1}$, which is the non-linear dynamical model, and $\mathbf{M}_{k:k-1}$, which is its linearized version.

uncertainties which are assumed to follow a zero-mean Gaussian distribution characterized by a variance $\mathbf{R}_k$, *i.e.* $\epsilon_k^{\mathrm{o}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$.

The model and observation errors can be described in first approximation by Gaussian stochastic processes; therefore the solution can be completely described by the first two moments of the state. Following the notation generally used in DA literature, the *forecast/analysis* states and error covariances are indicated as $\mathbf{x}_k^{\mathrm{f/a}}$ and $\mathbf{P}_k^{\mathrm{f/a}}$, respectively. The error covariance matrix is defined as $\mathbf{P}_k^{\mathrm{f/a}} = \mathbb{E}\left[\left(\mathbf{x}_k^{\mathrm{f/a}} - \mathbb{E}(\mathbf{x}_k^{\mathrm{f/a}})\right)\left(\mathbf{x}_k^{\mathrm{f/a}} - \mathbb{E}(\mathbf{x}_k^{\mathrm{f/a}})\right)^{\top}\right]$. In the framework of KF applications, a linear dynamical model ($\mathcal{M}_{k:k-1} \equiv \mathbf{M}_{k:k-1}$) and a linear observation model ($\mathcal{H}_k \equiv \mathbf{H}_k$) are considered. Also, a common simplifying hypothesis is that uncertainties in the model and in the set of observations are uncorrelated. The estimated state is obtained via a recursive procedure:

1. A predictor (forecast) phase, where the analysed state of the system at a previous time-step is used to obtain an *a priori* estimation of the state at the current instant. This prediction, which is obtained relying on the model only, is not conditioned by observation at time k:

$$\mathbf{x}_k^{\mathrm{f}} = \mathbf{M}_{k:k-1}\mathbf{x}_{k-1}^{\mathrm{a}} \tag{3}$$

$$\mathbf{P}_k^{\mathrm{f}} = \mathbf{M}_{k:k-1}\mathbf{P}_{k-1}^{\mathrm{a}}\mathbf{M}_{k:k-1}^{\top} + \mathbf{Q}_k \tag{4}$$

2. An update (analysis) step, where the state estimation is updated accounting for observation at the time k:

$$\mathbf{K}_k = \mathbf{P}_k^{\mathrm{f}}\mathbf{H}_k^{\top}\left(\mathbf{H}_k\mathbf{P}_k^{\mathrm{f}}\mathbf{H}_k^{\top} + \mathbf{R}_k\right)^{-1} \tag{5}$$

$$\mathbf{x}_k^{\mathrm{a}} = \mathbf{x}_k^{\mathrm{f}} + \mathbf{K}_k\left(\mathbf{y}_k^{\mathrm{o}} - \mathbf{H}_k\mathbf{x}_k^{\mathrm{f}}\right) \tag{6}$$

$$\mathbf{P}_k^{\mathrm{a}} = \left(I - \mathbf{K}_k\mathbf{H}_k\right)\mathbf{P}_k^{\mathrm{f}} \tag{7}$$

The optimal prediction of the state ($\mathbf{x}_k^{\mathrm{a}}$) is obtained via the addition to the predictor estimation ($\mathbf{x}_k^{\mathrm{f}}$) of a correction term determined via the so called *Kalman gain* $\mathbf{K}_k$. The classical KF algorithm is not suited for direct application to the analysis of complex flows since the classical KF formulation is developed

for linear systems. Applications to non-linear systems can be performed using more advanced techniques such as the extended Kalman filter [14] or exploiting features of the numerical algorithms used for numerical discretization [17]. The canonical Kalman filter is difficult to implement with realistic engineering models. The matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$ are usually unknown and their behaviour must be modelled. One simple, classical simplification is to consider that errors for each component are completely uncorrelated in space and from other components *i.e.* $\mathbf{Q}_k$ and $\mathbf{R}_k$ are considered to be diagonal [33, 34]. Also, KF relies on the transport of a very large error covariance matrix $\mathbf{P}_k$. It is therefore necessary to store it but also to invert very large matrices (see (5)).

2.1.2. Ensemble Kalman filter

The Ensemble Kalman Filter (EnKF) [35, 6] relies on the estimation of $\mathbf{P}_k$ by means of an ensemble. More precisely, the error covariance matrix is approximated using a finite ensemble of model states of size N_e. If the ensemble members are generated using stochastic Monte-Carlo sampling, the error in the approximation decreases with a rate of $\frac{1}{\sqrt{N_\mathrm{e}}}$.

Given an ensemble of forecast/analysed states at a certain instant k, the ensemble matrix is defined as:

$$\mathscr{E}_k^{\mathrm{f/a}} = \left[\mathbf{x}_k^{\mathrm{f/a},(1)}, \cdots, \mathbf{x}_k^{\mathrm{f/a},(N_\mathrm{e})}\right] \in \mathbb{R}^{N_x \times N_\mathrm{e}} \tag{8}$$

To reduce the numerical cost of implementation, the normalized ensemble anomaly matrix is then specified as:

$$\mathbf{X}_k^{\mathrm{f/a}} = \frac{\left[\mathbf{x}_k^{\mathrm{f/a},(1)} - \overline{\mathbf{x}_k^{\mathrm{f/a}}}, \cdots, \mathbf{x}_k^{\mathrm{f/a},(N_\mathrm{e})} - \overline{\mathbf{x}_k^{\mathrm{f/a}}}\right]}{\sqrt{N_\mathrm{e}-1}} \in \mathbb{R}^{N_x \times N_\mathrm{e}}, \tag{9}$$

where the ensemble mean $\overline{\mathbf{x}_k^{\mathrm{f/a}}}$ is obtained as:

$$\overline{\mathbf{x}_k^{\mathrm{f/a}}} = \frac{1}{N_\mathrm{e}} \sum_{i=1}^{N_\mathrm{e}} \mathbf{x}_k^{\mathrm{f/a},(i)} \tag{10}$$

The error covariance matrix $\mathbf{P}_k^{\mathrm{f/a}}$ can thus be estimated via the information derived from the ensemble. This estimation, hereafter denoted with the superscript e, can be factorized into:

$$\mathbf{P}_k^{\mathrm{f/a,e}} = \mathbf{X}_k^{\mathrm{f/a}} \left(\mathbf{X}_k^{\mathrm{f/a}}\right)^\top \in \mathbb{R}^{N_x \times N_x} \tag{11}$$

The goal of the EnKF is to mimic the BLUE (Best Linear Unbiased Estimator) analysis of the Kalman filter. For this, Burgers et al. [36] showed that the observation must be considered as a random variable with an average corresponding to the observed value and a covariance $\mathbf{R}_k$ (the so-called *data randomization* trick). Therefore, given the discrete observation vector $\mathbf{y}_k^{\mathrm{o}} \in \mathbb{R}^{N_y}$ at an instant k, the ensemble of perturbed observations is defined as:

$$\mathbf{y}_k^{\mathrm{o},(i)} = \mathbf{y}_k^{\mathrm{o}} + \epsilon_k^{\mathrm{o},(i)}, \quad \text{with} \quad i = 1, \cdots, N_{\mathrm{e}} \quad \text{and} \quad \epsilon_k^{\mathrm{o},(i)} \sim \mathcal{N}(0, \mathbf{R}_k). \tag{12}$$

A normalized anomaly matrix of the observations errors is defined as

$$\mathbf{E}_k^{\mathrm{o}} = \frac{1}{\sqrt{N_{\mathrm{e}} - 1}} \left[\epsilon_k^{\mathrm{o},(1)} - \overline{\epsilon_k^{\mathrm{o}}}, \epsilon_k^{\mathrm{o},(2)} - \overline{\epsilon_k^{\mathrm{o}}}, \cdots, \epsilon_k^{\mathrm{o},(N_{\mathrm{e}})} - \overline{\epsilon_k^{\mathrm{o}}}, \right] \in \mathbb{R}^{N_y \times N_{\mathrm{e}}} \tag{13}$$

where $\overline{\epsilon_k^{\mathrm{o}}} = \dfrac{1}{N_{\mathrm{e}}} \displaystyle\sum_{i=1}^{N_{\mathrm{e}}} \epsilon_k^{\mathrm{o},(i)}$.

The covariance matrix of the measurement error can then be estimated as

$$\mathbf{R}_k^{\mathrm{e}} = \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}}\right)^\top \in \mathbb{R}^{N_y \times N_y}. \tag{14}$$

By combining the previous results, we obtain (see [5]) the standard stochastic EnKF algorithm. The corresponding analysis step consists of updates performed on each of the ensemble members, as given by

$$\mathbf{x}_k^{\mathrm{a},(i)} = \mathbf{x}_k^{\mathrm{f},(i)} + \mathbf{K}_k^{\mathrm{e}} \left(y_k^{\mathrm{o},(i)} - \mathcal{H}_k \left(\mathbf{x}_k^{\mathrm{f},(i)}\right)\right) \tag{15}$$

The expression of the Kalman gain is

$$\mathbf{K}_k^{\mathrm{e}} = \mathbf{X}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^\top \left(\mathbf{Y}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^\top + \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}}\right)^\top\right)^{-1} \tag{16}$$

where $\mathbf{Y}_k^{\mathrm{f}} = \mathbf{H}_k \mathbf{X}_k^{\mathrm{f}}$.

A version of the Ensemble Kalman filter algorithm using the previously defined anomaly matrices is given in Appendix A.2. This is the version we use in our applications.

State-of-the-art approaches based on the EnKF are arguably the most advanced forms of state estimation available in the field of DA methods. These techniques have been extensively applied in the last decade in meteorology and geoscience [5]. Applications in mechanics and engineering are much more recent, despite a rapid increase in the number of applications in the literature. Among those, studies dealing with wildfire propagation [22], combustion [37], turbulence modeling [21] and hybrid variational-EnKF methods [10] have been reported. These applications reinforce the idea that approaches based on EnKF have a high investigative potential despite the highly non-linear, multiscale features of the flows studied by the fluid mechanics community.

2.1.3. Dual Ensemble Kalman filter

In this section, we extend the classical EnKF framework presented in Sec. 2.1.2 by considering the case of a parameterized model such as (1). The objective is to enable the model to generate accurate forecasts. For this, we need to determine good estimates of both model state variables $\mathbf{x}_k$ and parameters θ_k given erroneous observations $\mathbf{y}_k^{\mathrm{o}}$. The procedure developed by [38], called *dual estimation*, is here employed for this purpose. Two interactive filters are used, the former for the estimation of the parameters from a guessed state solution, the latter for the update of the state variables from the estimated previous parameters.

In the first step of the algorithm, the ensemble of the analysed parameters

is updated following the classical KF equation:

$$\theta_k^{\mathrm{a},(i)} = \theta_k^{\mathrm{f},(i)} + \mathbf{K}_k^{\theta,\mathrm{e}} \left(\mathbf{y}_k^{\mathrm{o},(i)} - \mathbf{y}_k^{\mathrm{f},(i)} \right) \quad \text{with} \quad i = 1, \cdots, N_e \tag{17}$$

where $\mathbf{y}_k^{\mathrm{f},(i)} = \mathcal{H}_k \left(\mathbf{x}_k^{\mathrm{f},(i)} \right)$.

The Kalman gain responsible for correcting the parameter trajectories in the ensemble is obtained as follows:

$$\mathbf{K}_k^{\theta,\mathrm{e}} = \mathbf{\Theta}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}} \right)^\top \left(\mathbf{Y}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}} \right)^\top + \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}} \right)^\top \right)^{-1}, \tag{18}$$

where the variable $\mathbf{\Theta}_k^{\mathrm{f}}$ plays the same role for the parameters as the variable $\mathbf{X}_k^{\mathrm{f}}$ defined in (9) for the states. We then have:

$$\mathbf{\Theta}_k^{\mathrm{f/a}} = \frac{\left[\theta_k^{\mathrm{f/a},(1)} - \overline{\theta_k^{\mathrm{f/a}}}, \cdots, \theta_k^{\mathrm{f/a},(N_\mathrm{e})} - \overline{\theta_k^{\mathrm{f/a}}} \right]}{\sqrt{N_\mathrm{e} - 1}} \in \mathbb{R}^{N_\theta \times N_\mathrm{e}} \tag{19}$$

with

$$\overline{\theta_k^{\mathrm{f/a}}} = \frac{1}{N_\mathrm{e}} \sum_{i=1}^{N_\mathrm{e}} \theta_k^{\mathrm{f/a},(i)} \tag{20}$$

Once new values of the model's parameters are inferred, we can update the state by EnKF (see Sec. 2.1.2). The Dual Ensemble Kalman filter allows to perform a recursive parametric inference / state estimation using the information from the ensemble members. The algorithm that we use is given in Appendix A.3.

2.2. From transport equation to multigrid resolution

The general expression for a conservation equation in local formulation over a continuous physical domain reads as:

$$\frac{\mathrm{D}\mathbf{x}}{\mathrm{D}t} = \frac{1}{\rho} \nabla \cdot \overline{\overline{\sigma}} + \mathbf{f} \tag{21}$$

where $\mathrm{D}/\mathrm{D}t$ is the total (or material) derivative of the physical quantity of investigation $\mathbf{x}$ and ρ is the flow density. The divergence operator is indicated as

$\nabla\cdot$ while $\overline{\overline{\sigma}}$ is the stress tensor. Finally, $\mathbf{f}$ represents the effects of volume forces. The evolution of the flow is obtained via time advancement of the discretized solution, which is performed in a physical domain where initial and boundary conditions are provided. A general expression of the discretized form of (21) for the time advancement from the step $k-1$ to k is given by:

$$\mathbf{x}_k = \mathbf{\Phi}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{b}_k \tag{22}$$

where $\mathbf{\Phi}_k$ is the *state transition model* which includes the discretized information of (21). In case of non-linear dynamics described by (21), the state-of-the-art algorithms used for the discretization process are able to preserve the non-linear information in the product $\mathbf{\Phi}_k \mathbf{x}_{k-1}$, up to a discretization error which is usually proportional to the size of the time step. The term $\mathbf{b}_k$ represents the *control vector* reflecting, for instance, the effect of the boundary conditions. $\mathbf{B}_k$ is the *control input model* which is applied to the control vector $\mathbf{b}_k$. Equation (22) is consistent with a time explicit discretization of (21). It is well known that this class of methods, despite the very high accuracy, may exhibit some unfavorable characteristics for the simulation of complex flows, such as limitations to the time step according to the Courant-Friedrichs-Lewy (CFL) condition [2]. To bypass this limitation, one possible alternative consists in using implicit schemes for time discretization. In this case, the general structure of the discretized problem is usually cast in the form:

$$\mathbf{\Psi}_k \mathbf{x}_k = \widetilde{\mathbf{\Psi}}_k \mathbf{x}_{k-1} + \widetilde{\mathbf{B}}_k \mathbf{b}_k := \mathbf{c}_k \tag{23}$$

where $\mathbf{\Psi}_k$, $\widetilde{\mathbf{\Psi}}_k$ and $\widetilde{\mathbf{B}}_k$ are matrices obtained via the discretization process. Obviously, considering $\mathbf{\Phi}_k = \mathbf{\Psi}_k^{-1} \widetilde{\mathbf{\Psi}}_k$, we retrieve (22). However, this manipulation is in practice not performed due to the prohibitive costs associated to large scale matrices inversions at each time step. Instead, an iterative procedure can be used until the residual δ^n, determined at the n-th iteration, falls below a pre-selected threshold value ε. In other words, the procedure is stopped when

$\|\delta^n\| = \|\mathbf{\Psi}_k \mathbf{x}_k^n - \mathbf{c}_k\| < \varepsilon$. Among the various iterative methods proposed in the literature, *multigrid* approaches are extensively used in CFD applications [39, 2]. The solution is found on the computational grid by updating an initial guess via multiple estimations obtained on a hierarchy of discretizations. Two well-known families of multigrid approaches exist, namely the *algebraic* multigrid method and the *geometric* multigrid method. With algebraic multigrid methods, a hierarchy of operators is directly constructed from the state transition model $\mathbf{\Psi}$. On the other hand, the geometric multigrid obtains the solution via a set of operations performed in two (or more) meshes. In this paper, focus on the latter technique and we consider the simplified case of two grids. Thereafter, the variables defined on the fine grid will be denoted with the superscript F ($\mathbf{x}^{\mathrm{F}}$ for instance), those defined on the coarse grid will be denoted with the superscript C ($\mathbf{x}^{\mathrm{C}}$ for instance).

The coarse-level representation $\mathbf{x}^{\mathrm{C}}$ is usually obtained suppressing multiple mesh elements from the initial fine-level one $\mathbf{x}^{\mathrm{F}}$. This operation may be defined by a coarsening ratio parameter r_{C}, which indicates the total number of elements on the fine grid over the number of elements conserved in the coarse grid. Among the numerous algorithms proposed for geometric multigrid, we use the Full Approximation Scheme (FAS), which is a well-documented strategy [26, 27]. A general formulation for a two-grid algorithm is now provided. The time subscript k is excluded for clarity. The superscript n represents the iteration step of the procedure.

1. Starting from an initial solution on the fine grid $\left(\mathbf{x}^0\right)^{\mathrm{F}}$ (which is usually equal to $\mathbf{x}$ at the previous time step $k-1$), an iterative procedure is applied to obtain a first solution $\left(\mathbf{x}^1\right)^{\mathrm{F}}$. A residual $\left(\delta^1\right)^{\mathrm{F}} = \mathbf{c}^{\mathrm{F}} - \mathbf{\Psi}^{\mathrm{F}}\left(\mathbf{x}^1\right)^{\mathrm{F}}$ is calculated.
2. $\left(\mathbf{x}^1\right)^{\mathrm{F}}$ and $\left(\delta^1\right)^{\mathrm{F}}$ are projected from the fine grid to the coarse grid space via a projection operator Π_{C}, so that $\left(\mathbf{x}^1\right)^{\mathrm{C}}$ and $\left(\delta^1\right)^{\mathrm{C}}$ are obtained. Similarly, the state transition model $\mathbf{\Psi}^{\mathrm{F}}$ is projected on the coarse grid (that is re-estimated based on the projection of the solution of the fine grid onto the

coarse grid) to obtain $\mathbf{\Psi}^{\mathrm{C}}$. Finally, we evaluate $\mathbf{c}^{\mathrm{C}} = \mathbf{\Psi}^{\mathrm{C}} \left(\mathbf{x}^1\right)^{\mathrm{C}} + \left(\delta^1\right)^{\mathrm{C}}$.

3. An iterative procedure is employed to obtain $\left(\mathbf{x}^2\right)^{\mathrm{C}}$ on the coarse grid using as initial solution $\left(\mathbf{x}^1\right)^{\mathrm{C}}$.

4. The updated variable on the fine grid is obtained as $\left(\mathbf{x}^2\right)^{\mathrm{F}} = \left(\mathbf{x}^1\right)^{\mathrm{F}} + \Pi_{\mathrm{F}} \left(\left(\mathbf{x}^2\right)^{\mathrm{C}} - \left(\mathbf{x}^1\right)^{\mathrm{C}}\right)$ where Π_{F} is a projection operator from the coarse grid to the fine grid.

5. At last, the final solution $\left(\mathbf{x}^3\right)^{\mathrm{F}}$ is obtained via a second iterative procedure on the fine grid starting from the intermediate solution $\left(\mathbf{x}^2\right)^{\mathrm{F}}$.

This procedure can be repeated multiple times imposing $\left(\mathbf{x}^0\right)^{\mathrm{F}} = \left(\mathbf{x}^3\right)^{\mathrm{F}}$ at the beginning of each cycle. When the convergence is reached, the fine grid solution at time instant k is equal to $\left(\mathbf{x}^3\right)^{\mathrm{F}}$. In this work, the two projection operators (Π_{F} and Π_{C}) are chosen to be 4-th order Lagrange interpolators.

3. Multigrid Ensemble KF method (MEnKF)

Despite the game-changing advantage that EnKF offers for the analysis of large-scale dynamical systems, the use of a sufficiently large ensemble (usually 60 to 100 members are required for convergence [5]) may still be prohibitive for advanced applications. In the following, we present an EnKF strategy which relies on the generation of the ensemble members on a sub-space (*i.e.* coarser mesh) of the original model. To do so, we exploit the multiple levels of resolution naturally used by the multigrid procedure for the time advancement of the flow. For the classical case of the FAS two-grid multigrid algorithm, which employs two levels of resolution (coarse and fine), the ensemble members calculated on the coarse mesh level are run with a single high-refinement simulation, which is updated using the coarse mesh assimilation results. For this reason, the computational costs and the memory storage of the physical variables are dramatically reduced. For sake of simplicity, the procedure is here detailed just for the FAS two-grid multigrid algorithm. However, it can be integrated in every implementation of a multigrid method. In particular, using more than two resolution levels may allow to generate a larger number of ensemble members on the coarsest grid level, which would bring a further reduction of the computational resources required.

The present multigrid-ensemble algorithm here proposed works through the steps described below. An overview of the assimilation cycle is presented in Fig. 1. In the following description, the notation $\mathbf{\Psi}$ might hold for both $\mathbf{\Psi}$ and $\widetilde{\mathbf{\Psi}}$ introduced in (23), depending on the choice of the time integration strategy.

1. **First iteration on the fine grid**. Starting from an initial solution on the fine grid $\left(\mathbf{x}^{\mathrm{F}}_{k-1}\right)^{\mathrm{a}}$, a forecasted state $\left(\mathbf{x}^{\mathrm{F}}_{k}\right)^{\mathrm{f}}$ is obtained by using $\overline{\theta^{\mathrm{a}}_{k}}$ as parameter for the model $\mathbf{\Psi}^{\mathrm{F}}$, *i.e.*

$$\left(\mathbf{x}^{\mathrm{F}}_{k}\right)^{\mathrm{f}} = \mathcal{M}^{\mathrm{F}}_{k:k-1}\left(\left(\mathbf{x}^{\mathrm{F}}_{k-1}\right)^{\mathrm{a}}, \overline{\theta^{\mathrm{a}}_{k}}\right)$$

2. **Projection on the coarse grid**. $\left(\mathbf{x}^{\mathrm{F}}_{k}\right)^{\mathrm{f}}$ is projected on the coarse grid

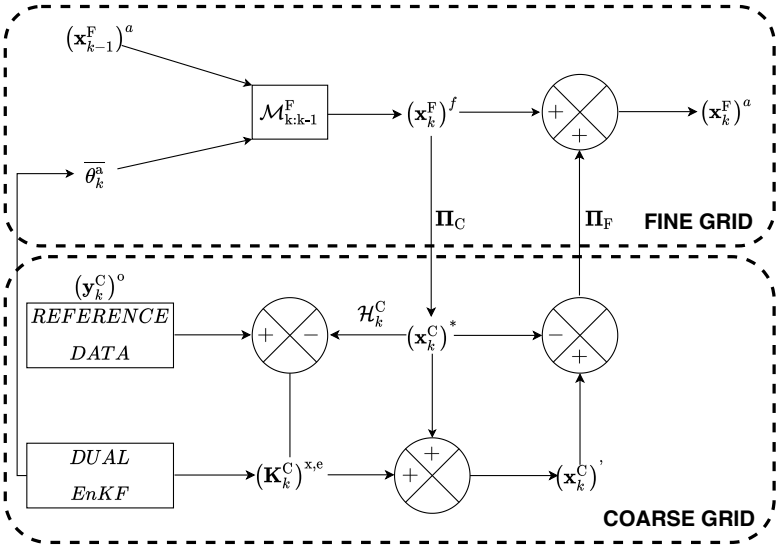


Figure 1: Schematic representation of the Multigrid Ensemble Kalman Filter (MEnKF). Two different levels of representation (fine and coarse grids) are used to obtain a data-driven fine grid estimation. The Dual Ensemble Kalman filter procedure is solved in the coarse grid. In the current implementation, the coarse level dual ensemble is not influenced by the fine grid simulations. The influence is then only one-directional, from coarse grid to fine grid. The full algorithm is given in Appendix A.

space via a projection operator Π_{C}, so that $(\mathbf{x}^{\mathrm{C}}_{k})^{*}$ is obtained, *i.e.*

$$(\mathbf{x}^{\mathrm{C}}_{k})^{*} = \Pi_{\mathrm{C}}\left((\mathbf{x}^{\mathrm{F}}_{k})^{\mathrm{f}}\right)$$

In this step, the flow field obtained on the fine mesh level can be used to optimize the behaviour of an ad-hoc model included in the time-marching process of the ensemble members to compensate the lack of resolution on the coarse grid and improve their accuracy. This aspect is discussed in the following.

3. **Time advancement of the ensemble members used in the Dual EnKF**. For each member i of the ensemble, the state matrix $(\mathbf{\Psi}^{\mathrm{C}})^{(i)}$ used for the advancement in time on the coarse grid is determined[2]. The ensem-

[2]While the advancement model is unique (the Navier-Stokes equations, for example), the discretization process contained in $\mathbf{\Psi}$ is also unique for each member of the ensemble. To distinguish them, it is therefore necessary to introduce an exponent i in the notations.

ble forecast $(\mathbf{x}_k^{\mathrm{C}})^{\mathrm{f},(i)}$ is corrected with the standard Dual EnKF procedure to obtain $(\mathbf{x}_k^{\mathrm{C}})^{\mathrm{a},(i)}$ as well as the parameters $\theta_k^{\mathrm{a},(i)}$. See details in Appendix A.4, Algo. 4.

4. **Determination of the state variables on the coarse grid**. This step provides the update of the physical state of the main simulation on the coarse grid. This state, which will be referred to as $(\mathbf{x}_k^{\mathrm{C}})'$, is obtained by classical iterative procedures on the coarse grid using the initial solution $(\mathbf{x}_k^{\mathrm{C}})^*$ if observations are not available (see step 3 of the FAS algorithm in Section 2.2). On the other hand, if observations are available, the Kalman gain matrix $(\mathbf{K}_k^{\mathrm{C}})^{x,\mathrm{e}}$ is used to determine the coarse grid solution $(\mathbf{x}_k^{\mathrm{C}})'$ through a Kalman filter estimation, *i.e.*

$$(\mathbf{x}_k^{\mathrm{C}})' = (\mathbf{x}_k^{\mathrm{C}})^* + (\mathbf{K}_k^{\mathrm{C}})^{x,\mathrm{e}} \left[(\mathbf{y}_k^{\mathrm{C}})^{\mathrm{o}} - \mathcal{H}_k^{\mathrm{C}} \left((\mathbf{x}_k^{\mathrm{C}})^*\right)\right]$$

5. **Final iteration on the fine grid**. The initial estimation $(\mathbf{x}_k^{\mathrm{F}})'$ of the final iteration of the fine grid state is determined using the results obtained on the coarse space: $(\mathbf{x}_k^{\mathrm{F}})' = (\mathbf{x}_k^{\mathrm{F}})^{\mathrm{f}} + \Pi_{\mathrm{F}} \left((\mathbf{x}_k^{\mathrm{C}})' - (\mathbf{x}_k^{\mathrm{C}})^*\right)$. The state $(\mathbf{x}_k^{\mathrm{F}})^{\mathrm{a}}$ is obtained from a final iterative procedure starting from $(\mathbf{x}_k^{\mathrm{F}})'$.

This algorithm has been specifically conceived to reduce the computational costs associated with the classical EnKF approach for large scale problems. For this, we combine a multigrid framework, frequently encountered in flow solvers, and a Dual Ensemble Kalman filter. Broadly speaking, our method falls into the class of *multilevel* techniques that aim at improving the estimation of statistics of expensive numerical simulations by considering different levels of resolution – in time or in space – of the same set of equations. In multilevel Monte Carlo applications, a small number of high-resolved solutions is combined with a larger number of low-resolution data [see 28, 29, 30, 31, for some applications]. Instead of considering additional simulation models for the same set of equations and several resolutions, it is also possible to reduce the variance of Monte Carlo methods by considering different sets of equations (surrogate models at differ-

ent levels of accuracy). This approach, called *multifidelity*, has recently been used with POD Galerkin reduced-order models [40]. At this point, it should be noted that, at the opposite of the classical filtering methods, our algorithm does not attempt to directly approximate the term $\mathbb{E}[\mathcal{M}_{k:k-1}((\mathbf{x}_k^F)^f, \theta_k)|\mathbf{y}_{1:k}^o]$. The specificity of MEnKF is that it is tailored for the simulation of compressible flows, for which spurious oscillations produced by the KF procedure may be responsible for irreversible numerical instabilities. For this reason, two important features must be discussed:

- The recursive structure of the algorithm allows for integration of iterative corrections for non-linear systems [42] as well as hard constraints (see the discussion in the introduction of [23, 24, 25]) to respect the conservativity of the model equations. However, these corrections may result in an increase of the computational resources required. Here, the multigrid algorithm itself is used for regularization (*i.e.* for smoothing the discontinuities in the physical variables produced by the update via Kalman Filter) of the flow. If an intentionally reduced tolerance is imposed in the iterative steps 4 and 5, the final solution will keep memory of the features of the state estimation produced in step 3. However, the iterative resolution will smooth the estimation via the state transition model $\mathbf{\Psi}$, which will perform a natural regularization of the flow. Clearly, if a reduced tolerance is imposed, the final solution will not necessarily respect the conservativity constraints of the model equations. However, one can argue that complete conservativity is not an optimal objective in this case if the model state at the beginning of the time step is not accurate.

- The state obtained on the fine grid level could be used to improve the accuracy of the calculation of the ensemble members via a second nested EnKF application. This nested cycle should use as observation the sampled data from the fine grid level prediction (variable $(\mathbf{x}_k^C)^*$ in the step 2 of the MEnKF algorithm) to infer the parametric behaviour of an ad-hoc modelling term $\gamma^{(i)}$ included in the time advancing model for the ensemble

members:

$$(\mathbf{x}_k^{\mathrm{C}})^{\mathrm{f,(i)}} = \mathcal{M}_{k:k-1}^{\mathrm{C}} \left(\left(\mathbf{x}_{k-1}^{\mathrm{C}} \right)^{\mathrm{a,(i)}}, \theta_k^{\mathrm{f,(i)}} \right) + \gamma^{(i)} \tag{24}$$

This second, internal EnKF procedure is used only to infer the parametric description of the terms $\gamma^{(i)}$ and no modification to the state is performed, in order to avoid the collapse of the solutions of the ensemble members. A similar procedure, although not in the framework of multigrid applications, has been very recently proposed by Brajard et al. [43]. In their work the model $\gamma^{(i)}$ measures the difference between the state obtained using two models with different accuracy. For application to turbulent flows, one could envision for example to run a Large Eddy Simulation (LES) ensemble on the coarse grid level and to use DNS results on the fine grid level to infer the behaviour of subgrid scale modelling for the LES. In the present work, we have chosen to exclude this inner EnKF loop from the solver i.e. $\gamma^{(i)} = 0$ for every ensemble member. This implies that the results from the coarse grid affect the simulation on the fine grid level, but not vice versa. This choice has been performed to quantify at first the accuracy of the MEnKF method in its simplest form. A dedicated study, beyond the scope of the present paper, will address this point and detail more precisely the true potential advantages of allowing the results obtained on the fine grid level to influence retro-actively the results of the ensemble members on the coarse grid. Current development about this aspect and future envisioned applications are further discussed in the perspectives included in Sec. 7.

Let us now consider how this choice affects the derivation of the state matrices for time advancement. In non-linear problems of interest in fluid mechanics, the state transition matrix $\mathbf{\Psi}$ includes information of the multi-scale interactions that are specific for every case investigated. The simplest possible choice, which is the one adopted in this work, is to calculate the coefficients of the matrices $\mathbf{\Psi}^{\mathrm{C}}$ and $(\mathbf{\Psi}^{\mathrm{C}})^{(i)}$ separately for each simulated

state. Thus, the similarities between the employed state matrices are limited to the use of the same discretization schemes / structure of $\mathbf{\Psi}$.

The advantages of our strategy with respect to classical approaches based on EnKF may be summarized in the following points:

- The RAM requirement necessary to store the N_e ensemble members during the assimilation is usually moderate. The reduction in computational costs is driven by N_e and by the size of the coarse variables. To illustrate this, let us consider the case of a simple two-level geometric multigrid approach for a 3D test case with a constant coarsening ratio $r_\mathrm{C} = 4$ and a size of ensembles $N_\mathrm{e} = 100$. Each ensemble member is then described by $4^3 = 64$ times less mesh elements than the single simulation on the fine grid. If one considers that one main simulation and 100 ensemble members are run simultaneously, and if the RAM requirement is normalized over the main simulation, this implies that R_RAM, the non-dimensional RAM requirement, is equal to $1 + 100/64 = 2.56$. In other words, the total cost in RAM is increased to just 2.56 times the cost of the simulation without EnKF. For $r_\mathrm{C} = 8$, the normalized RAM requirement is $R_\mathrm{RAM} = 1 + 100/8^3 = 1.195$, thus just a 20% increase in RAM requirements. This is clearly orders of magnitude more advantageous than a fine-grid classical EnKF application with $N_\mathrm{e} = 100$, since in this case $R_\mathrm{RAM} = N_\mathrm{e} = 100$.

- The computational cost relative to the ensemble forecast on the coarse grid can typically become less important than the cost of the single simulation retained on the fine grid depending on the r_c value. Considering that the ensemble members in the coarse grid and the simulation over the fine grid are running simultaneously, communication times are optimized.

- Owing to the iterative procedures of steps 4 and 5, regularization of the final solution is naturally obtained.

- The algorithm is here described and tested in the framework of geometric multigrid, but it can actually be integrated within other algorithmic struc-

tures. For iterative methods, the only essential operation to be performed is the determination of the state transition model $\mathbf{\Psi}^{\mathrm{C}}$ and of the projections Π_{C} and Π_{F}. This implies that the method can be easily extended to other popular procedures, such as the algebraic multigrid. If the multigrid operations are removed, then no regularization is obtained unless specific corrections are included.

This general algorithm may be easily tailored accounting for the complexity of the test case investigated, in particular for the requirements of iterative loops on both the coarse grid level and the fine grid level. The algorithm that we used to validate our approach is described in Appendix A.4.

4. Application: one-dimensional Burgers' equation

The MEnKF method introduced in Sec. 3 is now applied to the analysis of different test cases. Several dynamical systems of increasing complexity were chosen in order to highlight different properties of the algorithm. Also, a set of different tests is performed in order to obtain a comprehensive validation of the method. At first, let us consider a 1D Burgers' equation:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{\mathrm{Re}}\frac{\partial^2 u}{\partial x^2} \tag{25}$$

where x is the spatial coordinate, u the velocity and Re is the Reynolds number. Equation (25) is non-dimensionalized with a reference velocity u_0 and a reference length L_0. This equation is solved with a second-order centred finite difference scheme for the space derivatives and a first-order scheme for the time integration to obtain the general form of discretized representation as given by (23). A Dirichlet time-varying condition is imposed at the inlet:

$$u(x = 0, t) = 1 + \theta_1 \sin(2\pi t + \theta_2) \tag{26}$$

where θ_1 and θ_2 represent the amplitude and phase of the sinusoidal signal, respectively. The outlet boundary condition is extrapolated from the nearest points to the outlet using 4-th order Lagrange polynomials. The initial condition is $u(x, t = 0) = 1$ everywhere in the physical domain. The value of the Reynolds number is $\mathrm{Re} = 200$. The time advancement step is chosen as $\Delta t = 0.0002$. It is kept constant throughout the simulation. The analysis is performed over a physical domain of size $[0, 10]$. The distance between the computational nodes in the fine mesh is constant and set to $\Delta x = 0.0125$. This choice has been performed to discretize the characteristic length $L_0 = 1$ using 80 mesh elements. This also implies that the total number of nodes employed to perform the calculation is $N_x = 800$. A reference simulation is run on the fine grid with values $\theta_1 = 0.2$ for the amplitude and $\theta_2 = 0$ for the phase. Thereafter, the solution obtained by this reference simulation is called the *true* state or *truth*.

A flow visualization at $t = 10$ is shown in Fig. 2. For the investigated value of Reynolds number, the non-linear effects and viscous mechanisms can be clearly identified. One can see that the initial condition imposed at $t = 0$ has been completely advected outside the physical domain. Therefore, the simulation is no longer affected by the initial condition, and a so-called *fully developed* flow configuration is observed. The state obtained at $t = 10$ is then used to re-initialize the reference simulation at $t = 0$. The solution of this second run is finally sampled to obtain the set of observations.

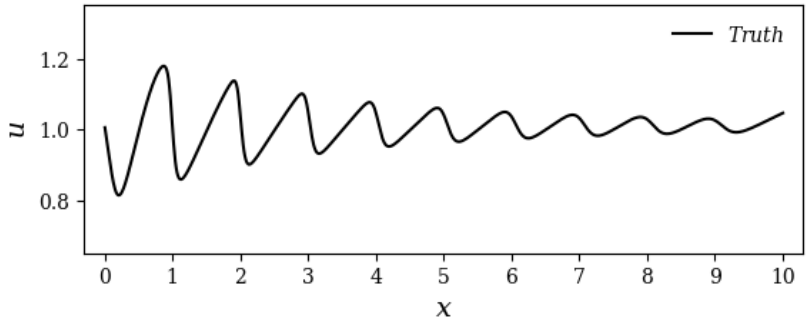


Figure 2: Solution of the 1D Burgers' equation at $t = 10$ for $\theta_1 = 0.2$ and $\theta_2 = 0$ (reference simulation, true state).

Data assimilation is performed in the following conditions:

- The observations are sampled each 30 time steps of the second run on the space domain $[0, 1]$ (80 sensors) and on the time window $[0, 19]$. This time horizon is long enough for the flow to be fully developed. For simplicity, we assume that the observations and the coarse-grid ensemble are represented on the same space. Therefore, $\mathcal{H}_k \equiv \mathcal{H}$ is a subsampling operator independent of time retaining only the points comprised in the coarse space domain $[0, 1]$. The observations are artificially perturbed using a constant in time Gaussian noise of covariance $\mathbf{R}_k \equiv \mathbf{R} = 0.0025\mathbf{I}$. This choice has been performed for every test case following the recommendations of [44], which extensively investigated the sensitivity of the EnKF to the noise / uncertainty in the model and in the observation.

- The *model* is chosen to be the discretized version of (25). The numerical test consists of one main simulation, which is run on the fine grid previously introduced, and an ensemble of $N_{\mathrm{e}} = 100$ coarse simulations used for assimilation purposes. The coefficients θ_1 and θ_2 are initially assumed to be described by Gaussian distributions, so that $\theta_1 \sim \mathcal{N}(0, \mathbf{Q}_{\theta_1})$ and $\theta_2 \sim \mathcal{N}(0.3, \mathbf{Q}_{\theta_2})$. The initial value of the covariance of the parameters is chosen equal to $\mathbf{Q}_{\theta_1}(t=0) = \mathbf{Q}_{\theta_2}(t=0) = 0.0025$. The values prescribed on the fine grid simulation are the mean values of the Gaussian distribution *i.e.* $\theta_1 = 0$ and $\theta_2 = 0.3$. Random values for the parameters are imposed at the inlet for each ensemble member on the coarse grid level. The initial mean values for the parameters are significantly different when compared with the values prescribed in the reference simulation, which are $\theta_1 = 0.2$ and $\theta_2 = 0$. This choice allows to analyse the rate of convergence of the optimization procedure. The initial condition $u(x, t=0) = 1$ is imposed to the fine-grid and coarse ensemble simulations. Thus, at $t = 0$, the reference simulation exhibits a very different behaviour when compared to the state imposed on the fine grid and for the ensemble members. This choice allows to explore the evolution of the first state estimation stages when the solution of the model could be very different from the observations. This transient is usually critical for complex flow investigation because it can be responsible of numerical instabilities. One can also see that the main fine grid simulation and the truth are identical in terms of mesh and numerical schemes, but for the former the parametric description via θ_1 and θ_2 needs to be optimized using the MEnKF.

The data estimation is performed at regular intervals of 30 time steps for a time window of $T_{\mathrm{DA}} = 19$ characteristic times, which encompass roughly 3000 DA analysis phases. The sensitivity of the parametric inference procedure to the resolution of the coarse simulations is investigated considering several coarsening ratios $r_{\mathrm{C}} = 1, 2, 4, 8, 16$. The fine grid is unchanged, so that the MEnKF is performed using information of progressively coarser grids as r_{C} increases.

The interest of this test is to analyse the loss of accuracy of the estimator as r_{C} increases and to ascertain the potential for efficient trade-off between accuracy and computational resources required for the estimation process. The performance of the estimator is analysed in terms of deviation to the *true* state, either by direct comparison or by estimating the Root Mean Square error as defined in (27). The developed algorithm is aimed at applications where, usually, an inexact state of the system is available *a priori*. Should scatter observations be integrated in the system through DA methods, the estimation shall progressively converge towards the *true* state. This implies that the estimator should undergo a transitional phase as the integrated observations are propagated in the entire domain.

The time-evolution of the estimation of θ_1 is shown in Fig. 3. Very rapid convergence ($t < 2$) is observed for $r_{\mathrm{C}} \leq 4$. In addition, the parameter estimation is extremely precise (discrepancy lower than 0.01% for $r_{\mathrm{C}} = 1$, lower than 2% for $r_{\mathrm{C}} = 4$). For higher values of the parameter r_{C}, the estimation of θ_1 becomes progressively more degraded. For the case $r_{\mathrm{C}} = 8$, θ_1 is initially overestimated and it finally converges to a value of $\theta_1 = 0.194$, 3% smaller that the true value. Larger errors in the optimization of θ_1 are observed for $r_{\mathrm{C}} = 16$. In this case, the optimized amplitude parameter is $\theta_1 = 0.27$, which is 35% larger than the true value.

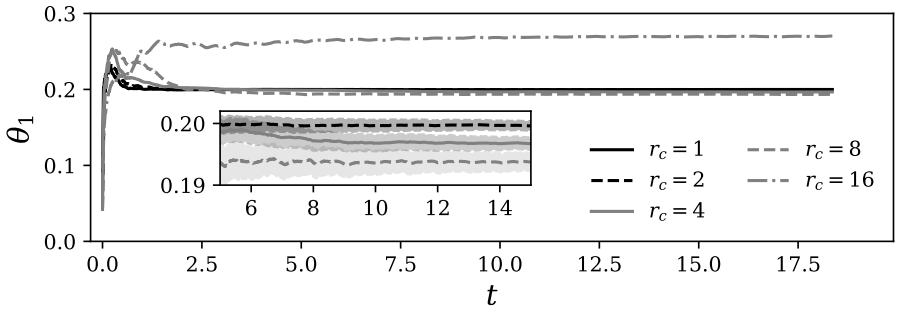


Figure 3: Values of the parameter θ_1 for different coarsening ratios $r_{\mathrm{C}} = 1, 2, 4, 8, 16$. In the zoomed region, the shaded area represents the 95% credible interval for the shown cases.

Similar considerations can be drawn by the analysis of the optimization of the

parameter θ_2, which is shown in Fig. 4. For $r_{\mathrm{C}} = 1$, 2, 4, we obtain an accurate prediction of the parameter, while a loss in accuracy is observed for the cases $r_{\mathrm{C}} = 8, 16$. This observation can be justified considering the number of mesh elements representing one characteristic length L_0 for these cases, which are 10 and 5 for $r_{\mathrm{C}} = 8$ and 16, respectively. Since one complete oscillatory cycle is performed on average over a characteristic length L_0, this means that the average phase angle between mesh elements is equal to 0.63 radians for $r_{\mathrm{C}} = 8$ and 1.26 radians for $r_{\mathrm{C}} = 16$. Thus, the values observed for the optimization of θ_2 for these two cases, which are around $\theta_2 \approx 0.15$, are significantly lower than the uncertainty due to the coarse-level resolution.

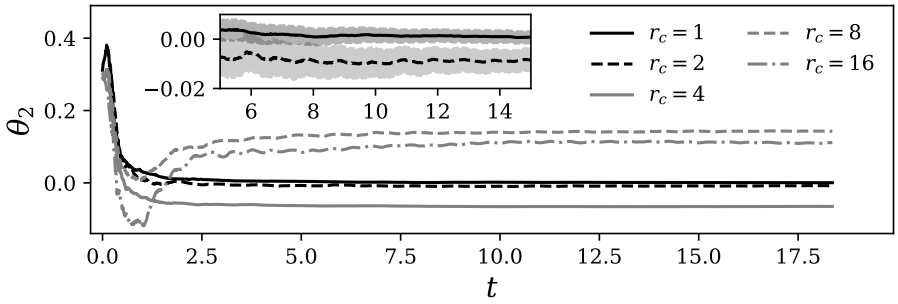


Figure 4: Values of the parameter θ_2 for different coarsening ratios $r_{\mathrm{C}} = 1, 2, 4, 8, 16$. In the zoomed region, the shaded area represents the 95% credible interval for the shown cases.

State estimation results for the case $r_{\mathrm{C}} = 1$ are shown in Fig. 5. This case, which is performed using the same grid for the coarse and fine mesh level, is equivalent to a standard Dual EnKF. However, owing to the final multigrid iterative loop, the final solution is naturally regularized. The results, which are shown for $t = 1$, 3.88, 10.60, show that the estimator successfully represents the behaviour of the dynamical system. A full domain advective time (*i.e.* 10 characteristic time units) must be simulated in order to observe the effect of the MEnKF in the whole domain. In fact, the parametric information imposed at the inlet for the ensemble members must affect the whole physical domain before a reliable correlation between the state variables can be established. However, once this initial transient is faded, the state estimation almost perfectly captures

the behaviour of the true state.

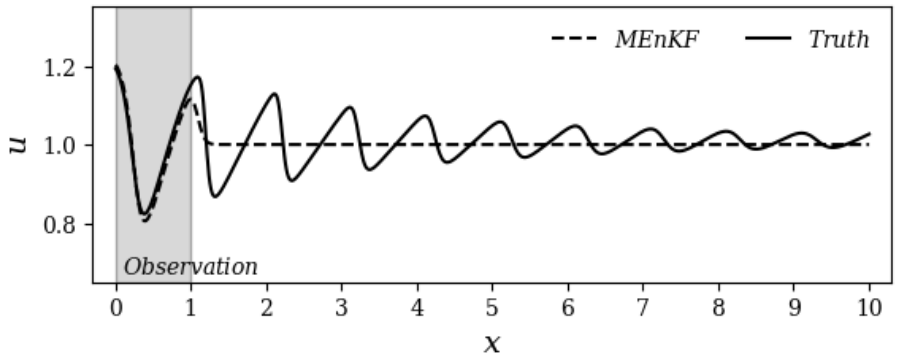
Results are now investigated for increasing values of r_{C}. Results for $r_{\mathrm{C}} = 8$ are shown in Fig. 6. Minor differences between the state estimation and the true state can be observed in this case. This discrepancy is due to the lack of resolution of the ensemble members. In fact, the resolution in this case is of 10 mesh elements per characteristic length. This number of points is arguably not enough to provide an accurate representation of the sinusoidal waves which are imposed at the inlet. However, one can see that no spurious numerical effects are observed as the estimator provides a smooth, continuous prediction of the velocity. The discrepancy between the true state and the state estimation is mainly associated with an erroneous calculation of the Kalman gain due to the under-resolution of the ensemble, which also affects the parameter estimation. A combined analysis of Fig. 3 and 6 shows that, due to the lack of accuracy in the estimation of θ_1, the variable u is over-predicted for $t < 2$ while it is slightly under-predicted for $t > 4$.

At last, the case for $r_{\mathrm{C}} = 16$ is shown in Fig. 7. In this case, the mesh elements are only 5 times smaller than the characteristic length L_0. Despite the important under-resolution of the ensemble members, which severely affects the estimation of θ_1 and θ_2, the state estimation still adequately represents the main features of the dynamical system.

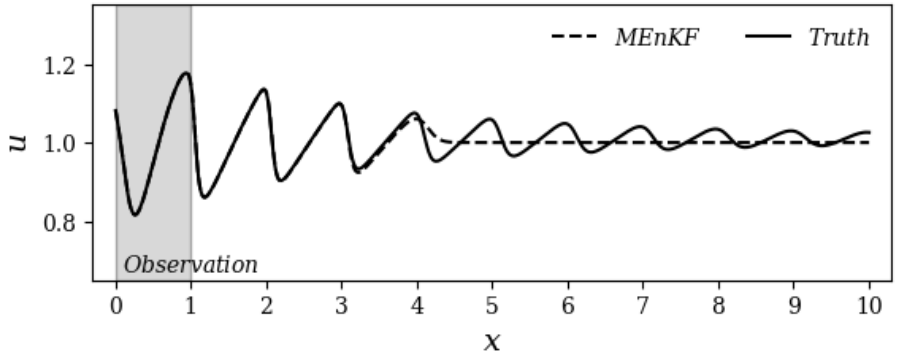
The discrepancy between the truth and the state estimation is measured via the time-dependent relative Root Mean Square Error (RMSE), *i.e.*

$$\mathrm{RMSE}(k) = \sqrt{\frac{\displaystyle\int_x \left[\left(u_k^{\mathrm{F}}\right)^{\mathrm{a}}(x) - \left(u_k^{\mathrm{F}}\right)^{\mathrm{True}}(x) \right]^2 \mathrm{d}x}{\displaystyle\int_x \left[\left(u_k^{\mathrm{F}}\right)^{\mathrm{True}}(x) \right]^2 \mathrm{d}x}} \tag{27}$$

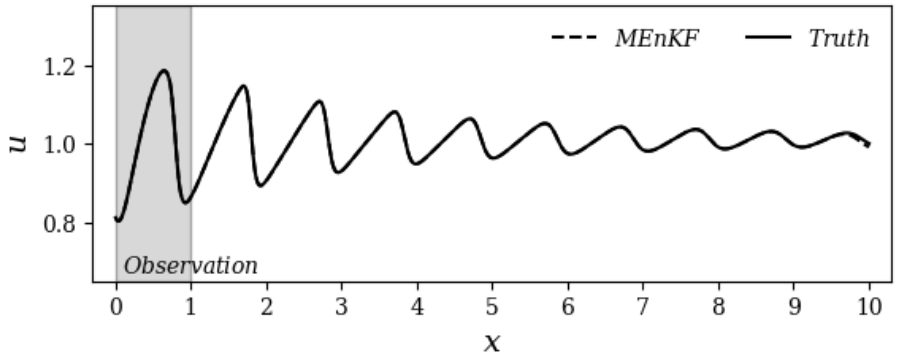
The results are shown in Fig. 8 for different values of the coarsening ratio r_{C}. One can see that the error decreases slowly for $t < 10$. This threshold time corresponds to a complete advective cycle in the physical domain. After this transient, the error may rapidly decrease before reaching a quasi-asymptotic



(a) $t = 1$

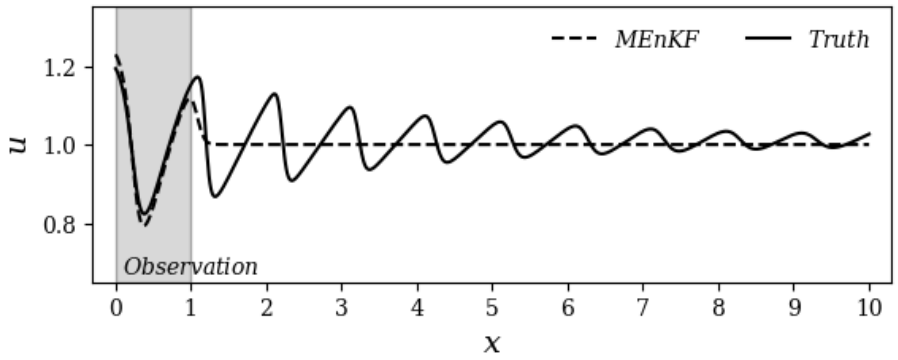


(b) $t = 3.88$

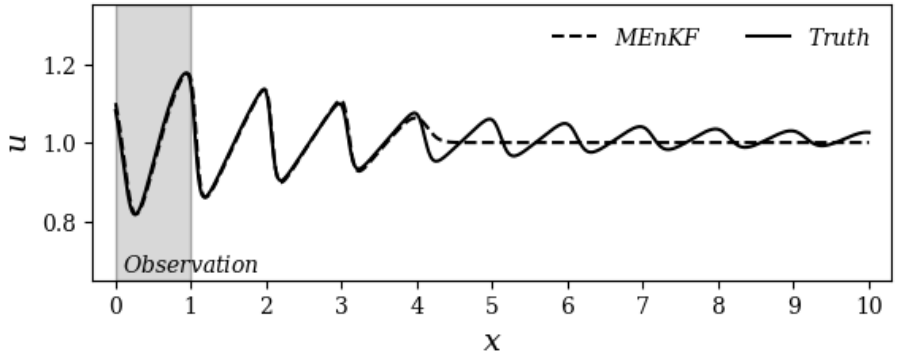


(c) $t = 10.60$

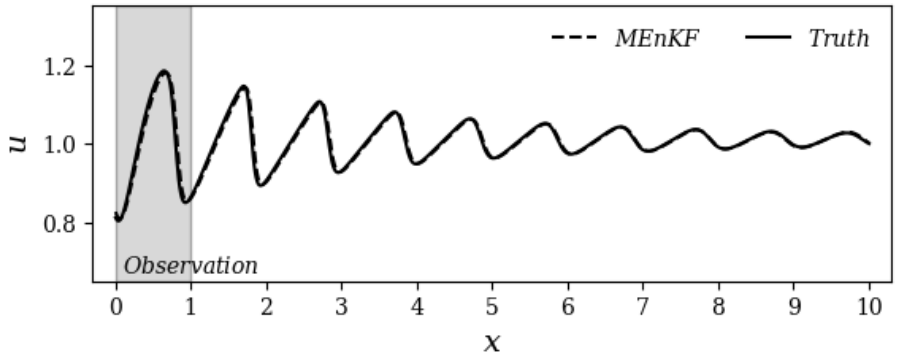
Figure 5: Estimations obtained by MEnKF for $r_{\mathrm{C}} = 1$ at $t = 1$ (a), $t = 3.88$ (b) and $t = 10.60$ (c). The times are dimensionless. The grey shaded area corresponds to the observation window.



(a) $t = 1$

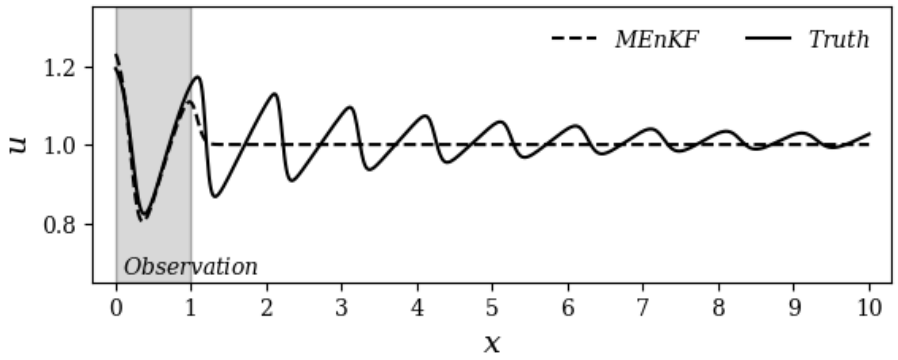


(b) $t = 3.88$

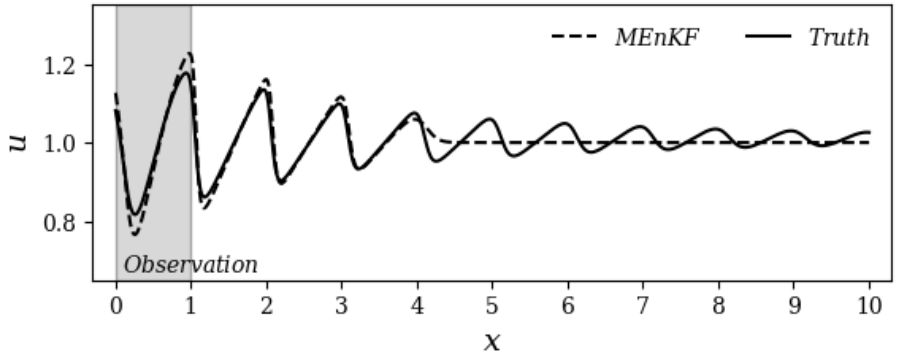


(c) $t = 10.60$

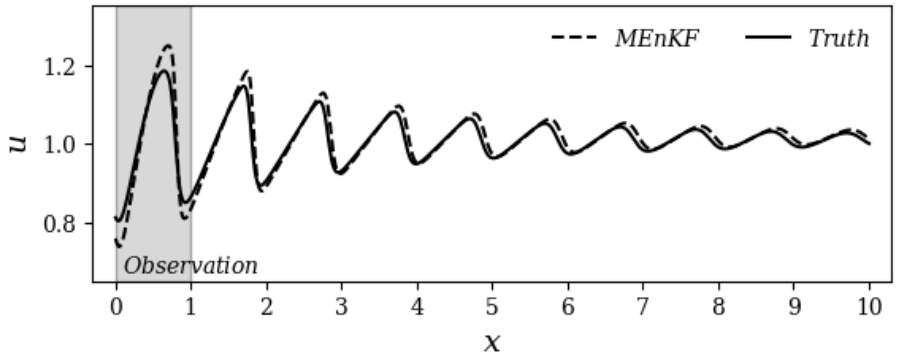
Figure 6: Estimations obtained by MEnKF for $r_C = 8$ at $t = 1$ (a), $t = 3.88$ (b) and $t = 10.60$ (c). The times are dimensionless. The grey shaded area corresponds to the observation window.



(a) $t = 1$



(b) $t = 3.88$



(c) $t = 10.60$

Figure 7: Estimations obtained by MEnKF for $r_C = 16$ at $t = 1$ (a), $t = 3.88$ (b) and $t = 10.60$ (c). The times are dimensionless. The grey shaded area corresponds to the observation window.

behaviour.

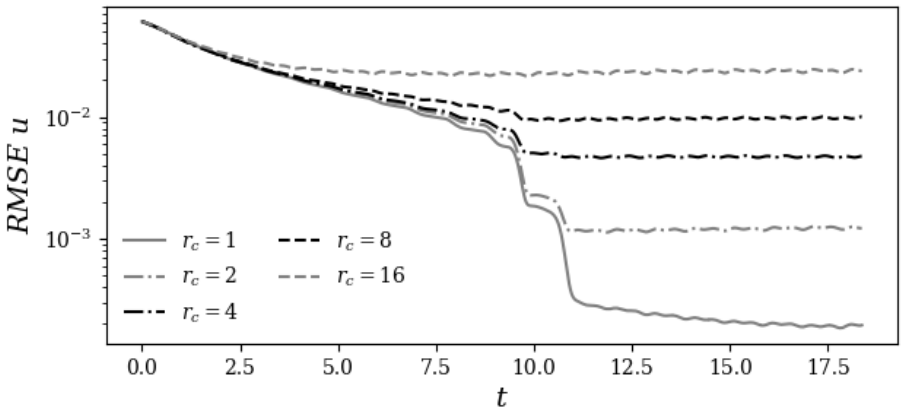


Figure 8: Time evolution of the RMS error of u for $r_{\mathrm{C}} = 1, 2, 4, 8, 16$.

One can also see that, once convergence is reached, the asymptotic RMSE value decreases with lower r_{C} values, as expected. However, lower r_{C} values are as well associated with larger computational costs, so that a trade-off between accuracy and required resources must be found. This aspect is further investigated considering the computational resources required to perform a full assimilation window for a given value of r_{C}. In Fig. 9, results are shown and normalized over the case $r_{\mathrm{C}} = 1$. One can see that the computational resources required rapidly decrease with increasing r_{C}, even for this simple one-dimensional test case. For large values of $r_{\mathrm{C}} = 8$, 16, one can see that the computational resources reach a plateau. Here the computational time to perform the DA procedures, which is the same for every r_{C}, is of similar order of magnitude of the calculation for the time advancement of the ensemble members.

In summary, the present analysis assesses the performance of the MEnKF tool for varying mesh resolution of the ensemble members. As expected, the accuracy of the state and parameter estimations diminishes for increasing r_{C}, but so does the computational cost. In addition, it was observed that the accuracy significantly drops when the mesh resolution is not able to provide a suitable description of the main scales characterizing the flow. Such a significant discrep-

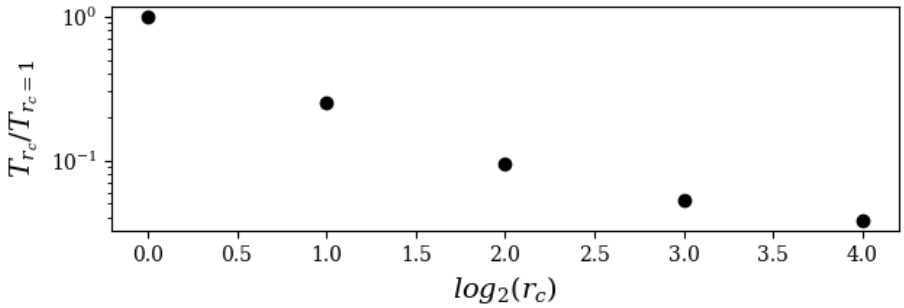


Figure 9: Computational time required to perform a full assimilation cycle for $r_C = 1, 2, 4, 8, 16$. Results are normalized over the computational time required for $r_C = 1$.

ancy for a relatively simple test case stresses how a minimal resolution threshold must be achieved in order to capture the essential physical features and to obtain a successful state estimation. In conclusion, the results of the parametric optimization may become unreliable with extreme under-resolution of the ensemble members. This finding is not surprising, as a minimum resolution level must be granted to capture the main features of the flow. Generally speaking, the level of resolution on the coarse grid must at least respect the Shannon Sampling Theorem [45], which states that at least two points per wavelength are necessary to capture the frequency of a given signal. However, this is not sufficient to approximate the function. Schmiechen [46] recommends 6-10 points per element for an accurate reconstruction. This threshold should never be crossed on the coarse grid. The dispersive and diffusive effects of the coarse grid on the state prediction deserve an in-depth analysis, which is out of the scope of this work. Here, the discretization schemes used are the same for the fine and coarse grids. However, one could argue that choosing adapted schemes for progressively coarser grids could improve the performance of the estimator.

5. Acoustic propagation of sinusoidal wave

The MEnKF strategy is now applied to a more complex physical system, namely the inviscid one-dimensional Euler equations:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} = 0 \tag{28}$$

$$\frac{\partial (\rho u)}{\partial t} + \frac{\partial ((\rho u) u)}{\partial x} + \frac{\partial p}{\partial x} = 0 \tag{29}$$

$$\frac{\partial (\rho E)}{\partial t} + \frac{\partial ((\rho E)\, u)}{\partial x} + \frac{\partial (p u)}{\partial x} = 0 \tag{30}$$

where ρ is the density, u is the velocity, p is the pressure and E is the total energy per unit mass. In this case, viscous effects are absent, but acoustic propagation affects the evolution of the flow. The equations are discretized in space using a second-order finite difference centred scheme. A first-order explicit Euler scheme is used for the time integration method. After discretization, a representation similar to (23) is obtained. A centred sixth-order numerical filter is included to damp numerical spurious oscillations [47]. We specifically analyse the acoustic propagation of a sinusoidal wave with a time-varying amplitude. To do so, a Dirichlet time-varying velocity condition is imposed at the inlet:

$$u(x = 0, t) = u_0 \left(1 + \theta(t) \sin(2\pi f_c t)\right) \tag{31}$$

The value of u_0 is set in order to impose an inlet Mach number $M = \frac{u_0}{a} = 0.4$, where a is the speed of the sound. The amplitude of variation in θ is sufficiently low to allow a flow evolution mainly driven by acoustic phenomena. The inlet velocity perturbation creates an acoustic wave that is transported along the domain with a speed equal to $u_0 + a$. The characteristic velocity and length scales are $u_c = u_0 + a$ and L_c, which is the wavelength of the signal imposed at the inlet. The characteristic time of the system is defined as $t_c = L_c/u_c$.

The sinusoidal behaviour of the velocity at the inlet is characterized by a constant frequency $f_c = 1/t_c$. However, the amplitude of the sinusoidal wave is driven by the time-varying parameter $\theta(t) = \theta_0 \left(1 + \sin\left(2\pi \frac{f_c}{10} t\right)\right)$, where

θ_0 is a constant. At the inlet, we set $\rho(x = 0, t) = \rho_0$ and $E(x = 0, t) = E_0 = e + 0.5u_0^2$, where e is the internal energy per unit of mass. By definition, $e = C_v T_0$ where C_v is the heat capacity at constant volume and T_0 the initial temperature of the flow. The outlet boundary condition is extrapolated from the nearest points to the outlet using 4-th order Lagrange polynomials. The initial condition imposed at $t = 0$ is $u(x, t = 0) = u_0$, $\rho(x, t = 0) = \rho_0$ and $E(x, t = 0) = E_0$ everywhere in the physical domain and for all the simulations (fine-grid and coarse-grid ensemble members). The fluid is considered an ideal gas with $C_v = 0.7171$, $\gamma = 1.4$, $\rho_0 = 1.17$ and $T_0 = 300$ in S.I. units.

The computational domain has been set to a size of $L_x = 10$. A uniform mesh distribution is used for every calculation. Similarly to the analysis in Sec. 4, 80 mesh elements are used to discretize the characteristic length L_c for a total of $N_x = 800$ elements in the domain. Finally, the normalized value of Δt is set to $\Delta t = 0.0006$.

A preliminary simulation is performed for $\theta_0 = 0.015$ (true state). A flow visualization of the wave patterns is shown in Fig. 10 at $t = 17.3$. The fully developed state obtained at $t = 10$ is used to initialize a new simulation from $t = 0$. This simulation is run for a total time of $T_{\mathrm{DA}} = 110$. As in Sec. 4, the fine grid data are projected on the coarse grid and sampled to obtain observations of ρu in the space region $x \in [0, 1]$. These observations are artificially perturbed using a Gaussian noise of variance $\mathbf{R} = 0.09\mathbf{I}$. The observation operator $\mathcal{H}$ selects the points in the coarse grid where ρu is available.

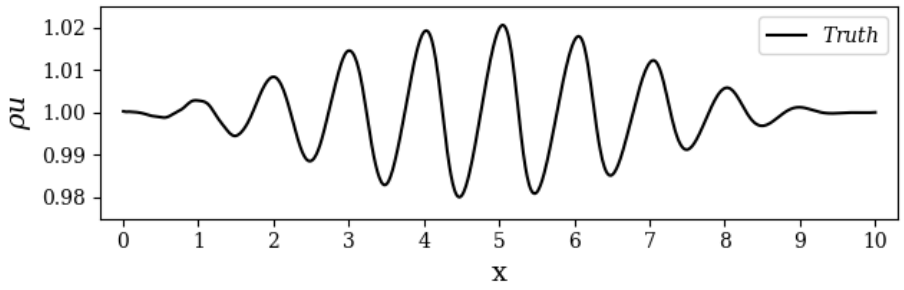


Figure 10: Solution ρu of the inviscid one-dimensional Euler equations at $t = 17.3$ for $\theta_0 = 0.015$ (*true* state).

The DA procedure is identical to the analysis presented in Sec. 4. The assimilation is composed of the base simulation, which is run on the fine grid, and $N_\mathrm{e} = 100$ simulations on the coarse-grid level. In this section, only one coarsening ratio is considered ($r_\mathrm{C} = 4$). The estimator is used to dynamically track the value of the parameter θ, which evolves in time. No a priori knowledge about the behaviour of the parameter is used. This time optimization is significantly more challenging when compared with the inference of a constant parameter as done in Sec. 4. A similar analysis using a classical Kalman smoother was recently proposed by Mons et al. [20].

For each coarse grid simulation of the estimator, θ is initially assumed to be a random Gaussian phenomenon $\theta \sim \mathcal{N}(0, \mathbf{Q}_\theta)$. The initial value of the covariance is $\mathbf{Q}_\theta(t = 0) = 6.4 \times 10^{-5}$. The value initially imposed at the inlet of the fine-grid simulation is $\theta = 0$, while random values are selected for each ensemble member on the coarse grid level following the normal distribution introduced above. The variance of the parameter θ for the ensemble members is artificially increased, as in the classical Dual EnKF algorithm. As described in Appendix A.3, we add to the estimated parameter of each member of the ensemble a Gaussian noise of covariance $\mathbf{\Sigma}_k^\theta = 10^{-10}\mathbf{I}$, which is reminiscent of the strategy used by Moradkhani et al. [38]. Extensive numerical tests have been performed and a sensitivity of the results to $\mathbf{\Sigma}_k^\theta$ has been observed. The value chosen for $\mathbf{\Sigma}_k^\theta$ has been set to avoid the collapse of the ensemble members over the state estimation while keeping the noise level for θ moderately low.

Let f_a be the number of analysis phases per characteristic time of simulation t_c, *i.e.* $f_a = t_c/t_a$. In the following, three different values of f_a are investigated: $f_a = 2$, 10, 55.

The estimator is run for a total simulation time $T_\mathrm{DA} = 110$, which encompasses 220 to 6000 DA analysis phases, depending on the value of f_a. At the end of each analysis, the mean value and the variance of the amplitude θ are updated following the Dual EnKF technique [38], similarly to what was done in Sec. 4.

The results for the estimation of the time-varying parameter θ are reported

in Fig. 11. For clarity, only the results for the assimilation window $t \in [40, 70]$ are shown. The precision of the parametric inference is measured via the relative error η defined as $\eta_k = \dfrac{\overline{\theta_k^a} - \theta_k^{\text{True}}}{\max_k \left|\theta_k^{\text{True}}\right|}$. The time evolution of θ is correctly estimated for the three values of f_a. This is an important result, considering that no a priori information was provided for the evolution of this parameter. A more detailed analysis reveals a lag in the parameter estimation. The application of a simple Kalman filter seems to be responsible for this result, while a Kalman Smoother (KS) should have been used to obtain a better synchronization. However, considering that the implementation of a KS is straightforward in this case and that observation is always provided close to the inlet, we considered that the increase in computational resources required by the KS was not needed. We find that the lag increases when a relatively small number of DA analyses is done. One can see that the prediction is significantly degraded for $f_a = 2$, while similar results are obtained for $f_a = 10, 55$. This observation is quantified by the time evolution of the relative error η, which is significantly large for $f_a = 2$. In addition, θ tends to be generally underestimated (around $10-20\%$) when it reaches its maximum value. This result is arguably associated with the under-resolution of the coarse level of the grid, where the gradients of physical variables are calculated with lower accuracy.

Now, results dealing with the state estimation are discussed. The predicted physical variable ρu, normalized over the initial value $\rho_0 u_0$, is shown in Fig. 12, 13 and 14 for $f_a = 2, 10$ and 55, respectively. For $f_a = 2$, the state estimation is significantly distant from the truth. It appears that the field correction applied via the Kalman gain is not able to compensate the poor estimation of θ. However, accurate results are observed for $f_a = 10$ and 55. Even though the value of the parameter θ is not exact, the state estimation including the correction via Kalman gain is very precise. For the case $f_a = 55$, almost no discernible difference is observed between the state estimation and the truth.

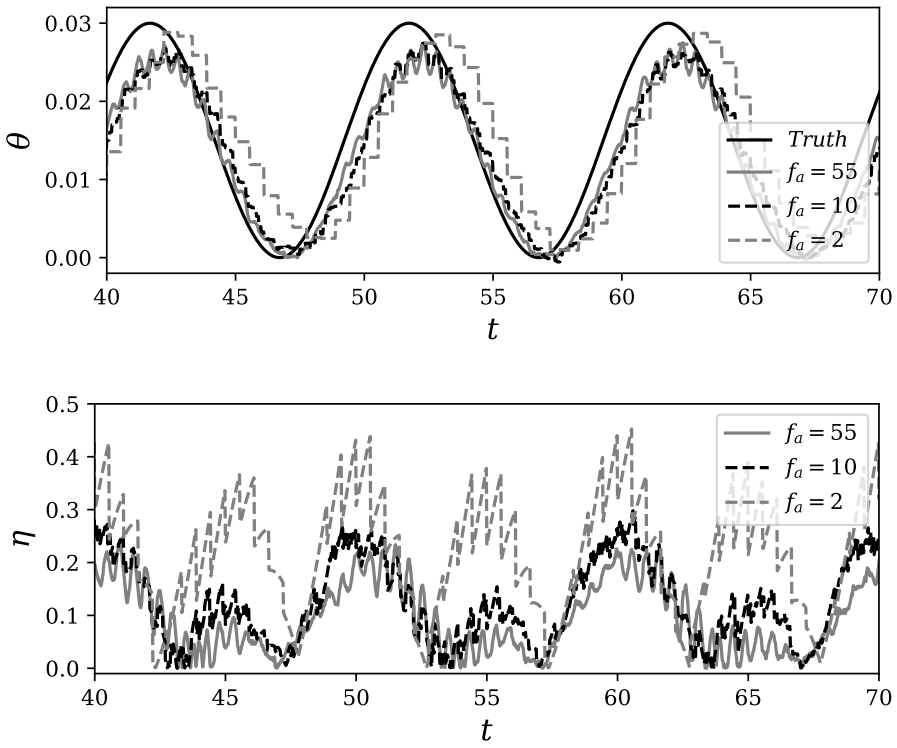
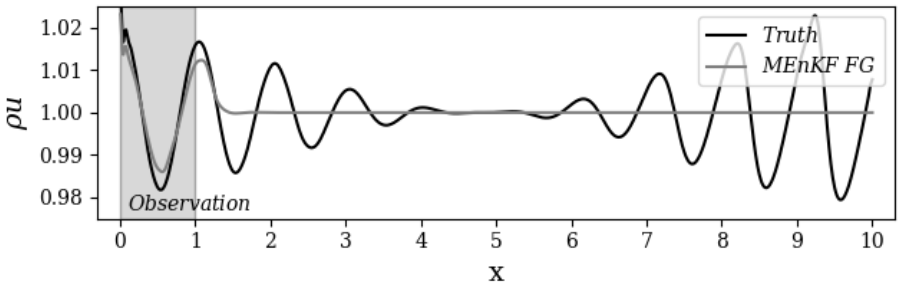
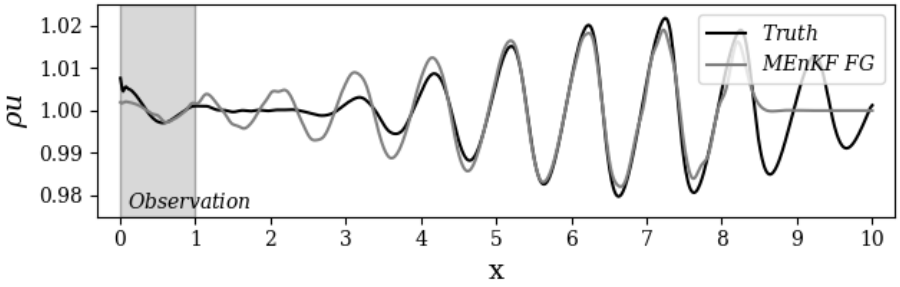


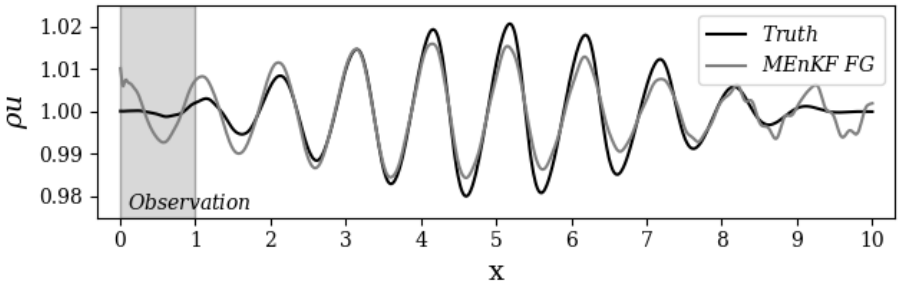
Figure 11: Time estimation of the parameter θ driving the amplitude of the sinusoidal acoustic wave for the assimilation window $t \in [40, 70]$. In the top image, results are shown for $f_a = 2$, 10, 55 and compared to the true value of θ. In the bottom image, the relative error η quantifying the parametric inference is shown.



(a) $t = 1.23$

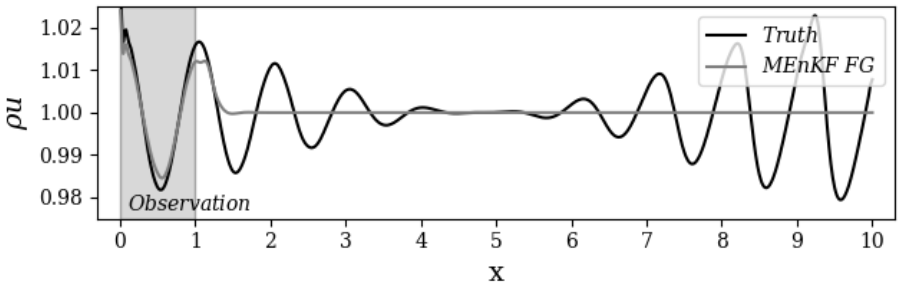


(b) $t = 8.32$

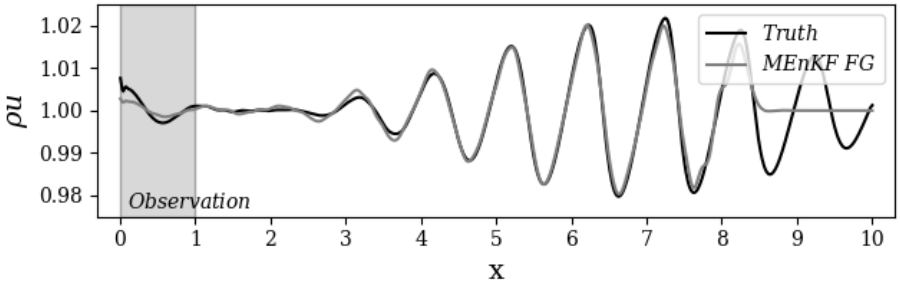


(c) $t = 16.30$

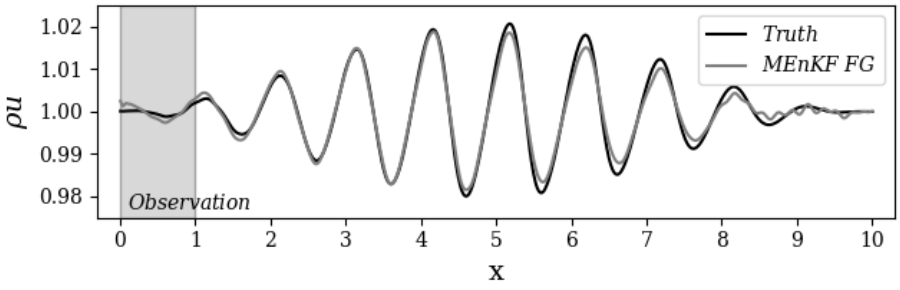
Figure 12: Estimations by MEnKF of the momentum ρu normalized by $\rho_0 u_0$ for $f_a = 2$ at $t = 1.23$ (a), $t = 8.32$ (b) and $t = 16.30$ (c). Times are given in t_c units. The grey shaded area corresponds to the observation window.



(a) $t = 1.23$

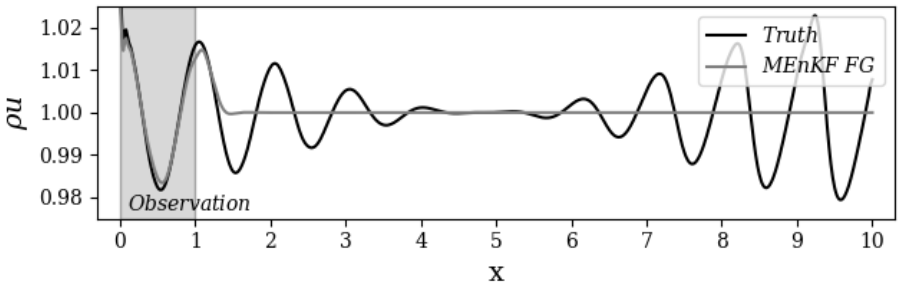


(b) $t = 8.32$

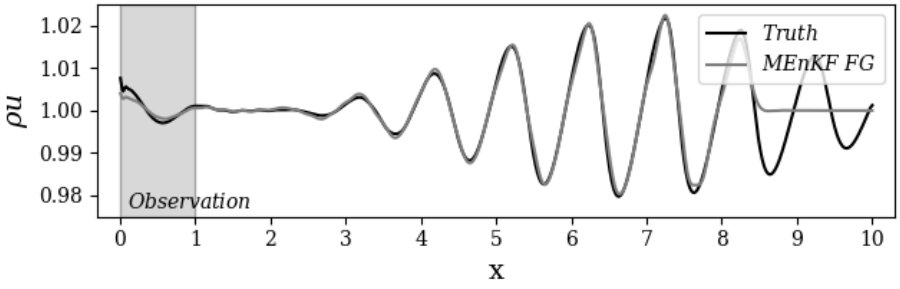


(c) $t = 16.30$

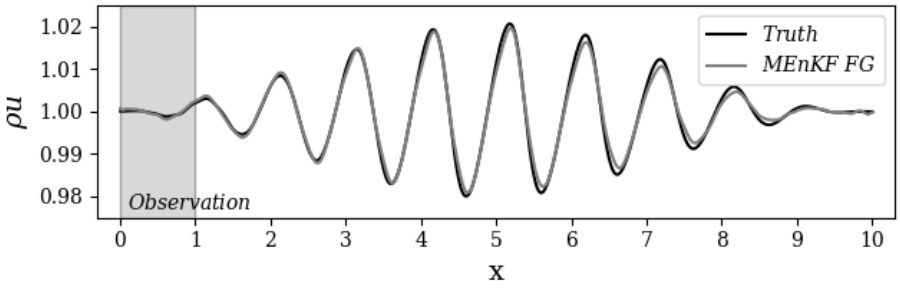
Figure 13: Estimations by MEnKF of the momentum ρu normalized by $\rho_0 u_0$ for $f_a = 10$ at $t = 1.23$ (a), $t = 8.32$ (b) and $t = 16.30$ (c). Times are given in t_c units. The grey shaded area corresponds to the observation window.



(a) $t = 1.23$



(b) $t = 8.32$



(c) $t = 16.30$

Figure 14: Estimations by MEnKF of the momentum ρu normalized by $\rho_0 u_0$ for $f_a = 55$ at $t = 1.23$ (a), $t = 8.32$ (b) and $t = 16.30$ (c). Times are given in t_c units. The grey shaded area corresponds to the observation window.

At last, the relative Root Mean Square Error (RMSE) defined as

$$\mathrm{RMSE}(k) = \sqrt{\frac{\int_x \left[\left((\rho u)_k^{\mathrm{F}}\right)^{\mathrm{a}}(x) - \left((\rho u)_k^{\mathrm{F}}\right)^{\mathrm{True}}(x)\right]^2 \mathrm{d}x}{\int_x \left[\left((\rho u)_k^{\mathrm{F}}\right)^{\mathrm{True}}(x)\right]^2 \mathrm{d}x}} \tag{32}$$

is shown in Fig. 15. The error achieves a quasi-constant asymptotic behaviour after a complete propagation of the signal in the physical domain ($t \approx 10 t_c$). As expected, a low global error is obtained for the cases $f_a = 10$ and $f_a = 55$. On the other hand, the error for $f_a = 2$ case is around $2 - 3$ times larger. The very small difference in performance between the cases $f_a = 10$ and $f_a = 55$ can be interpreted as a sign of convergence of the procedure. Another interpretation may be that the leading contribution of the error corresponds to the statistical error from using only 100 ensemble members in the estimation. We can however note that similar results were previously observed in three-dimensional simulations [18].

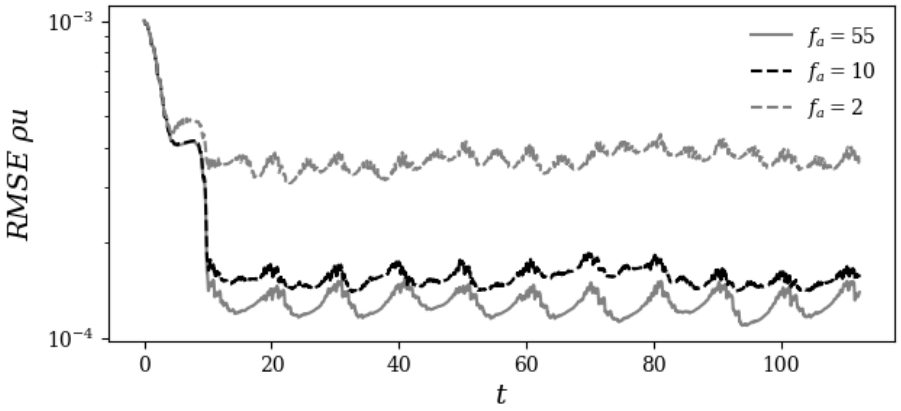


Figure 15: Time evolution of the RMS error of ρu for $f_a = 2$, 10, 55.

6. Spatially evolving compressible mixing layer

In this section, we consider the compressible Navier-Stokes equations in a two-dimensional physical domain:

$$\frac{\partial \rho}{\partial t} + div(\rho \mathbf{u}) = 0 \tag{33}$$

$$\frac{\partial (\rho \mathbf{u})}{\partial t} + div(\rho \mathbf{u} \otimes \mathbf{u}) = -\mathbf{grad}\, p + \mathbf{div}\, \overline{\overline{\tau}} \tag{34}$$

$$\frac{\partial (\rho E)}{\partial t} + div(\rho E \mathbf{u}) = -div(p\mathbf{u}) + div(\overline{\overline{\tau}}\mathbf{u}) + div\,(\lambda(T)\mathbf{grad}\, T) \tag{35}$$

where ρ is the density, $\mathbf{u}$ is the velocity (components u in the streamwise direction and v in the normal direction), p is the pressure, E is the total energy per unit of mass, $\overline{\overline{\tau}}$ is the tensor of the viscous constraints and T is the temperature. To obtain the representation given by (23), the equations are discretized using the finite difference method. Second-order centred schemes are used for the derivatives in space and a first-order scheme for the time integration. A centred sixth-order numerical filter is included to damp numerical spurious oscillations [47].

The two-dimensional spatially evolving mixing layer at Re = 100 is here investigated. For this value of Reynolds number, the flow exhibits unsteady features. It can be shown [48, 49] that the characteristics of the mixing layer are strongly affected by the inlet and, in particular, by imposed *ad hoc* time perturbations. The computational domain has been set to a size of $14L_c \times 6L_c$ in the streamwise direction x and normal direction y, respectively. The characteristic length L_c, which is taken as reference length from now on, is given by $L_c = A\delta_0$, where δ_0 is the initial vorticity thickness imposed at the inlet. The value of the parameter A is set in order to represent the most unstable wavelength determined by Linear Stability Theory (LST). At Re = 100, we have $A = 14.132$ [50]. The mesh resolution in the horizontal direction is constant for $x \in [0; 10]$. The size of the elements is $\Delta x = \frac{\delta_0}{8}$. For $x \geq 10$, a sponge zone is established with a coarsening ratio between successive elements which increases from 1.025 to 1.04. The resolution in the normal direction is constant and equal

to $\Delta y = \frac{\delta_0}{20}$ for $-0.18 \leq y \leq 0.18$. Outside this zone, the mesh elements increase in size moving away from the centreline with a constant coarsening ratio of 1.01.

The Reynolds number of the flow is calculated as $\mathrm{Re} = (U_1 - U_2)\delta_0/\nu = 100$ with asymptotic velocities set to $U_1 = 173.61$ and $U_2 = 104.17$. These values correspond to a Mach number $\mathrm{Ma} = 0.5$ and $\mathrm{Ma} = 0.3$, for each stream, respectively. The kinematic viscosity and thermal diffusivity of the flow are considered to be constant and their value is fixed to $\nu = 1.568 \times 10^{-5}$ and $\alpha = 22.07 \times 10^{-7}$, respectively. Finally, the flow is considered to be a perfect gas with $\gamma = 1.4$ and $C_v = 0.7171$. All these quantities are expressed in S.I. units. The inlet boundary condition is taken from [48]. For the velocity field, one has:

$$U_{\mathrm{in}}(y,t) = \frac{U_1 + U_2}{2} + \frac{U_1 - U_2}{2} \tanh\left(\frac{2y}{\delta_0}\right) + U_{\mathrm{Pert}}(y,t) \qquad -3 < y < 3 \quad (36)$$

$$V_{\mathrm{in}}(y,t) = 0 \quad (37)$$

where U_{in} is the streamwise velocity at the inlet and V_{in} is the normal velocity. U_{in} is estimated as a classical hyperbolic tangent profile plus a time-varying perturbation component:

$$U_{\mathrm{Pert}}(y,t) = \sum_{i=1}^{N_{\mathrm{in}}} \epsilon_i(t) \frac{U_1 + U_2}{2} [f_i(y) \sin(\omega_i t)], \quad (38)$$

where N_{in} is the total number of perturbation modes and ϵ_i quantifies the magnitude of each mode. The function $f_i(y) = \cos\left(4n_i \frac{y}{\delta_0}\right) h(y)$ controls the shape of the perturbation of the inlet velocity profile in the normal direction. The role of $h(y) = 1 - \tanh^2\left(\frac{2y}{\delta_0}\right)$ is to damp the perturbation component moving away from the centreline. The wavelength parameters n_i are tuned according to the LST results. In the following, we consider $N_{\mathrm{in}} = 1$ *i.e.* the inlet perturbation consists of a single mode. In the numerical tests, we follow [48] and set $n_1 = 0.4\pi$ and $\omega_1 = 1/t_c$, where $t_c = 2L_c/(U_1 + U_2)$ is the average

advection time. The inlet density is set to be constant so that $\rho_{\text{in}} = 1.177$, as well as the temperature $T_{\text{in}} = 300$ in S.I. units. The inlet total energy per unit mass E_{in} is calculated as $E_{\text{in}} = e + 0.5\left(U_{\text{in}}^2 + V_{\text{in}}^2\right)$, where the internal energy e is defined as $e = C_v T_{\text{in}}$. The outlet boundary conditions for all the variables present in the state vector are extrapolated from the nearest points to the outlet using 4-th order Lagrange polynomials. The zero gradient boundary condition is imposed on the transverse sides of the domain (at $y = -3$ and $y = 3$).

In this section, the parameter θ of the model corresponds to the single parameter ϵ_1, which is the time variable governing the amplitude of the perturbation. A reference simulation is run where ϵ_1 varies in time following a sinusoidal form: $\epsilon_1(t) = \epsilon(1 + \sin(\omega_\epsilon t))$. The values of the numerical parameters characterizing the perturbation are $\epsilon = 0.15$ and $\omega_\epsilon = 0.62\omega_1$. At $t = 0$, the variables of the fine grid and the coarse grid ensemble are initialized in the physical domain using the values imposed at the inlet. This implies that the physical quantities can exhibit initial variations in the y direction, while their value in the streamwise x direction is constant. For the case of the streamwise velocity, the velocity perturbation $U_{\text{Pert}} = 0$ at $t = 0$, see (38). A flow visualization of ρv at $t = 10$ is shown in Fig. 16 for this reference simulation. One can clearly observe the emergence of coherent structures with complex pairing patterns.

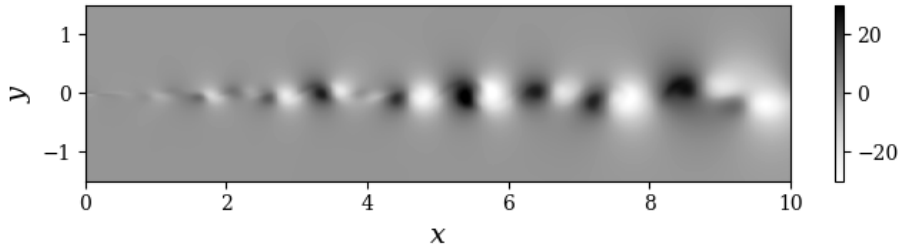


Figure 16: Visualization of the normal momentum ρv (S.I. units) for the 2D compressible Navier-Stokes equation. Reference simulation at $t = 10$ for a time-varying value of ϵ_1.

The DA procedure is performed using the following elements:

- The observation is sampled from the reference simulation shown in Fig. 16, which is run for a total simulation time of $T_{\text{DA}} = 40$ in t_c units. This value

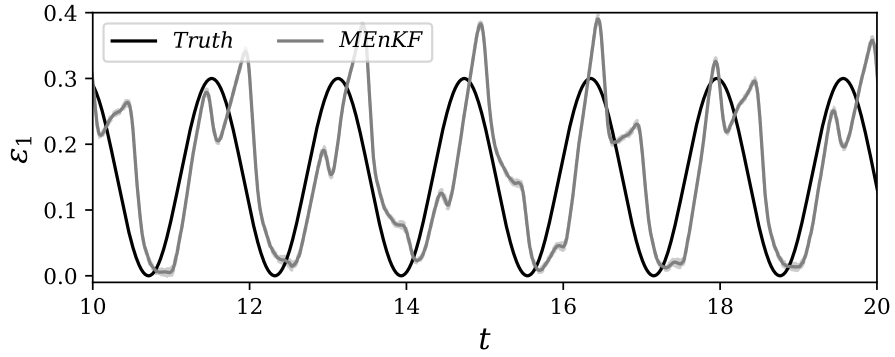
corresponds to four complete advections in the whole physical domain. A fully developed state obtained from a prior simulation at $t = 10$ is used to initialize the simulations at $t = 0$. Data are projected on the coarse grid and sampled every 30 time steps in the region $x \in [0, 0.55]$ and $y \in [-0.16, 0.16]$. Considering the results obtained in Sec. 5, the number of analysis phases per characteristic time of simulation ($f_a = 75$) is chosen sufficiently high to assure a good estimation. The observations are made from the instantaneous fields ρu and ρv. The data used as observation are artificially perturbed using a Gaussian noise of covariance $\mathbf{R} = \mathbf{I}$. The observation operator $\mathcal{H}$ acts as described in Sec. 5.

- The model is the discretized version of the system given by (33) - (35). The features of the fine mesh level were previously introduced. For the coarse grid level, a homogeneous coarsening ratio $r_{\mathrm{C}} = 4$ is employed. The initialization strategy used for the coarse ensemble simulations is identical to the one described above for the fine-grid reference case. We consider that no prior information is available on the time evolution of ϵ_1. At $t = 0$, this coefficient is fixed to be a random Gaussian value $\epsilon_1 \sim \mathcal{N}(0, \mathbf{Q}_a)$ where the initial value of $\mathbf{Q}_a(t = 0) = 0.0625$. Similarly to the cases analysed in Sec. 4, the value imposed on the main fine-grid simulation at $t = 0$ is $\epsilon_1 = 0$, while random values are imposed for each ensemble member on the coarse grid level. The size of the ensemble is $N_{\mathrm{e}} = 100$.

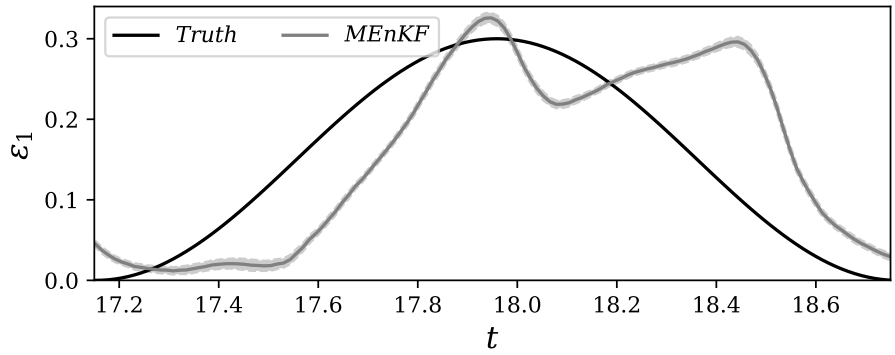
The estimation algorithm is run over a time window equal to $T_{\mathrm{DA}} = 40$ which encompass roughly 3200 DA analysis phases. At the end of each analysis, the mean value and the variance of the coefficient ϵ_1 are updated following the Dual EnKF technique [38], similarly to what was done in Sec. 4 and 5.

The time evolution of the estimated value of ϵ_1 is reported in Fig. 17. The overall sinusoidal trend is generally respected, although a relatively small phase lag is visible. This lag does not appear to be larger than the one previously observed for the one-dimensional case based on the Euler equation. This lag is due to the usage of a Kalman Filter approach instead of a Kalman Smoother, as

discussed in Sec. 5. However, in this case, some over prediction of the parameter is locally observed in time, which was not obtained for the wave propagation test case.



(a) Estimation history



(b) Zoom

Figure 17: Time evolution of the inferred values of ϵ_1 for the time-varying reference case. (a) Large time window. (b) Zoomed region. The shaded area represents the 95% credible interval for the estimated parameter.

765

The results obtained for the prediction of the normal momentum ρv are shown in Fig. 18. One can see that the combination of parameter and state estimations produces an accurate prediction of the flow. Minor discrepancies are observed when comparing the state estimation with the true state. In particular,
770 the momentum ρv does not exhibit spurious oscillations which could stem from the field correction determined via the Kalman gain. In order to evaluate the

respective influence of the parameter estimation step and state estimation phase, a test case is run in which only the parameter estimation is performed. That is, the state estimation obtained on the coarse-grid level is not included in the steps 4 and 5 of the algorithm presented in Sec. 3. While the results of the parameter estimation are the same for the two cases, one can see in Fig. 19 that the prediction is sensibly deteriorated.

This observation is quantified by the evaluation of the relative Root Mean Square Error (RMSE), defined as:

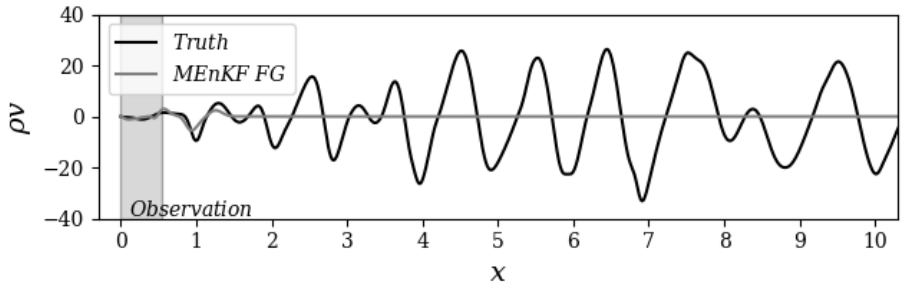
$$\mathrm{RMSE}(k) = \sqrt{\frac{\int_x \left[\left((\rho v)_k^{\mathrm{F}}\right)^{\mathrm{a}}(x) - \left((\rho v)_k^{\mathrm{F}}\right)^{\mathrm{True}}(x)\right]^2 \mathrm{d}x}{\int_x \left[\left((\rho v)_k^{\mathrm{F}}\right)^{\mathrm{True}}(x)\right]^2 \mathrm{d}x}} \tag{39}$$

The results, which are shown in Fig. 20, indicate that the accuracy of the complete algorithm is higher when compared to the case in which only the parameter estimation is performed. Therefore, the two operations concurrently provide an improvement in the prediction of the flow.

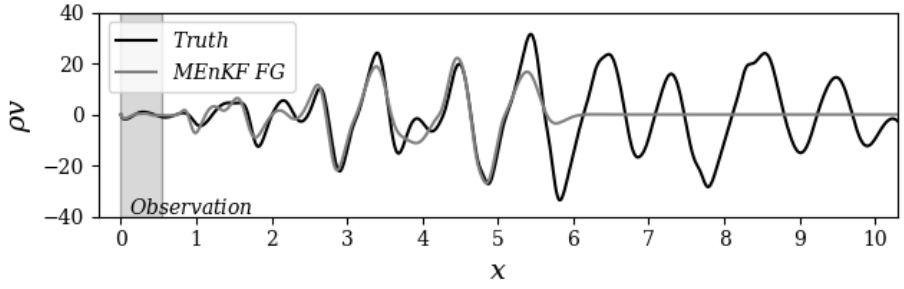
At last, an analysis of the conservativity of the algorithm is performed. As previously discussed, the state estimation obtained via EnKF does not necessarily comply with the dynamical equations of the model. This drawback can be responsible for discontinuities in the physical field, which can significantly affect the accuracy and stability of the global algorithm. The analysis is performed considering an indicator Γ_k^{F} which measures the conservation of the transversal momentum equation (34) in discretized form:

$$\frac{\left((\rho v)_k^{\mathrm{F}}\right)^{\mathrm{a}} - \left((\rho v)_{k-1}^{\mathrm{F}}\right)^{\mathrm{a}}}{\Delta t} - \mathcal{F}_{\rho v}\left(\rho_k, \mathbf{u}_k, p_k, \overline{\overline{\tau}}_k\right) := \Gamma_k^{\mathrm{F}}(x, y), \tag{40}$$

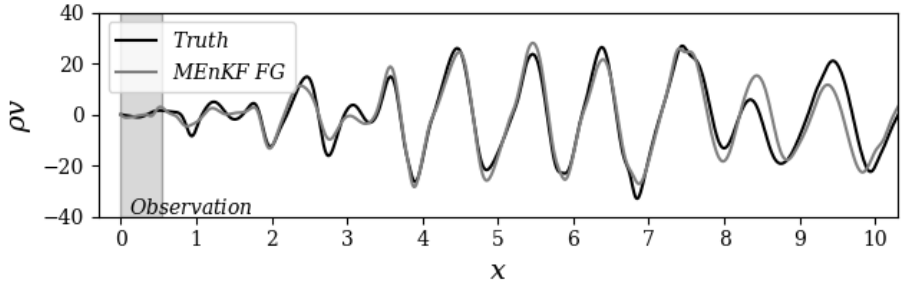
where $\mathcal{F}_{\rho v}$ represents the spatial discretization terms in the transversal momentum equation. In the forecast step performed via the model, $\Gamma_k^{\mathrm{F}} = 0$ down to a convergence rate δ which is prescribed. However, the value of Γ_k^{F}, at the end of a time step where a forecast-analysis is performed, is strictly connected with the computational strategy employed. Here, three scenarios are considered for



(a) $t = 1$

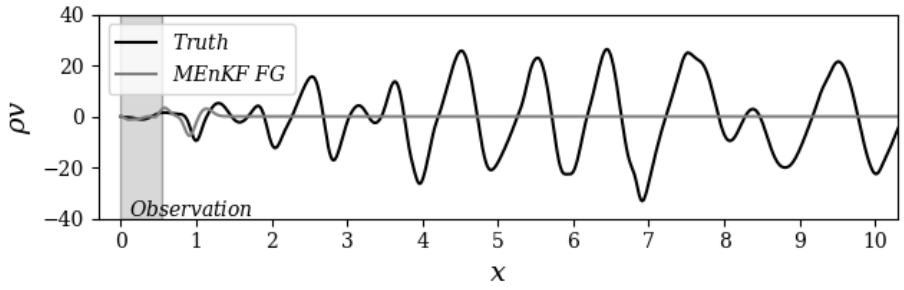


(b) $t = 5$

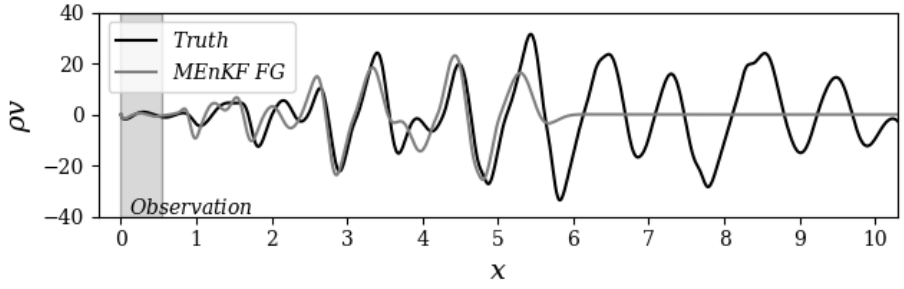


(c) $t = 30$

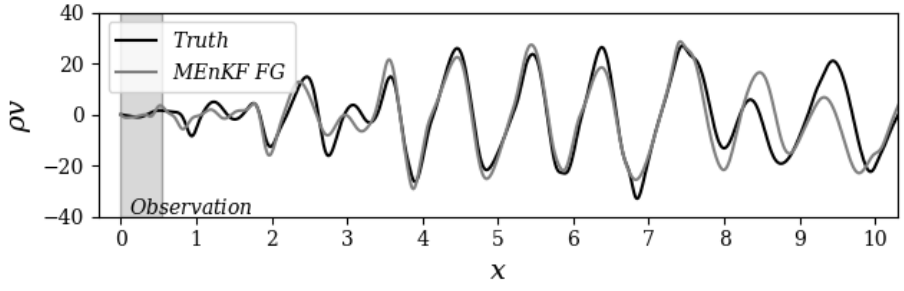
Figure 18: Estimations obtained by MEnKF of the momentum ρv (S.I. units) at the centreline $y = 0$ of the mixing layer. Results at $t = 1$ (a), $t = 5$ (b) and $t = 30$ (c) for the time-varying ϵ_1.



(a) $t = 1$



(b) $t = 5$



(c) $t = 30$

Figure 19: Estimations obtained by MEnKF of the momentum ρv (S.I. units) at the centreline $y = 0$ of the mixing layer. Here, MEnKF is only used to provide the estimation of ϵ_1. Results at $t = 1$ (a), $t = 5$ (b) and $t = 30$ (c) for the time-varying ϵ_1.

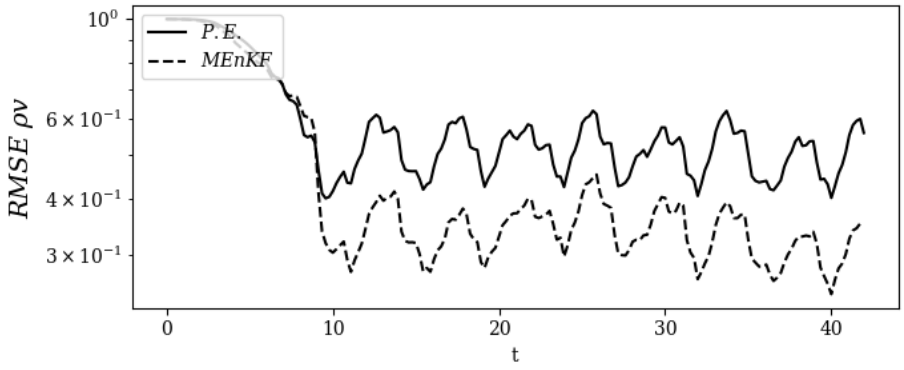


Figure 20: Time evolution of the RMS error of ρv for the case of a time-varying inlet parameter ϵ_1. The symbol P.E. corresponds to the case where MEnKF is only used for the estimation of ϵ_1. The notation MEnKF corresponds to the standard version of the algorithm, including parameter estimation and physical state correction via Kalman gain.

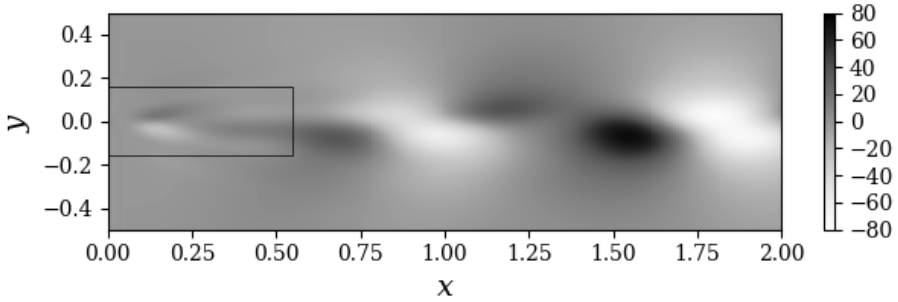
a simplified case where the amplitude of the inlet-perturbation is assumed to be constant ($\epsilon_1 = \text{ct} = 0.15$):

- S_1: A classical Dual EnKF is performed on the coarse grid and a fine-grid correction is obtained through the ensemble statistics. In this scenario, the state estimation obtained in the step 4 of the MEnKF algorithm presented in Sec. 3 is directly projected in the fine mesh space and used as final solution. The step 5 of the algorithm is not performed.

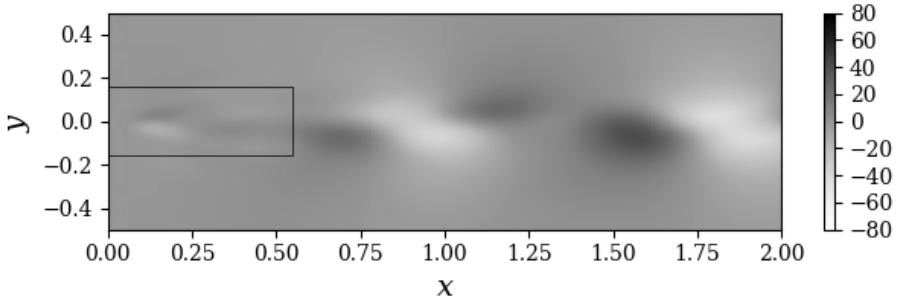
- S_2: A standard MEnKF algorithm, as described in Sec. 3.

- S_3: A MEnKF algorithm where the ensemble prediction is just used to estimate the unknown parameter of the system. No update of the physical solution is performed using the correction via Kalman gain.

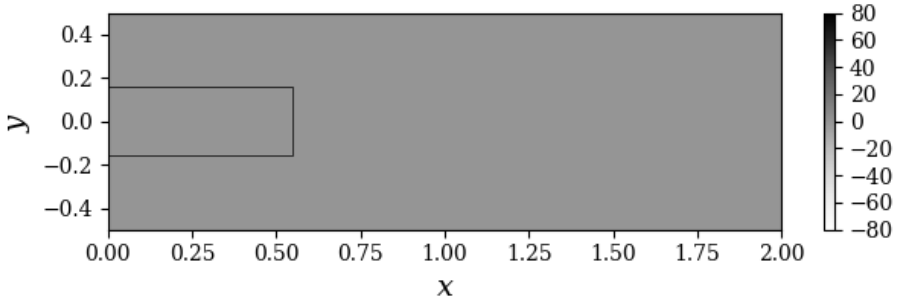
The results are shown in Fig. 21 after the first forecast / analysis step. For clarity, we introduce a normalized criterion $(\Gamma_k^{\text{F}})^* = \frac{\Gamma_k^{\text{F}}}{\Gamma_C}$ where Γ_C is defined as $\max_k \left| \frac{\left((\rho v)_k^{\text{F}}\right)^{\text{a}} - \left((\rho v)_{k-1}^{\text{F}}\right)^{\text{a}}}{\Delta t} \right|$. As expected, $\Gamma_k^{\text{F}} = 0$ everywhere when MEnKF is only used for the parameter estimation (scenario S_3). In this case, the time advancement of the solution is performed using the model only, which exactly complies with the discretized equation and respects conservativity (up to a con-



(a) Standard Dual EnKF on the coarse mesh.



(b) Standard MEnKF.



(c) MEnKF used only for parameter estimation.

Figure 21: Analysis of the conservativity of the dynamical model via the normalized quantity $\left(\Gamma_k^{\mathrm{F}}\right)^*$. Results are shown for three scenarios: (a) the classical Dual EnKF, (b) the classical MEnKF algorithm and (c) the MEnKF only used for parameter estimation.

vergence error which is negligible). On the other hand, results in Fig. 21 (a) show some lack of conservativity in the physical domain for the first scenario. This is also expected, since no constraint is imposed to force the Kalman gain correction to comply with the dynamical equations. Finally, results for the MEnKF are shown in Fig. 21 (b). The evolution of $(\Gamma_k^{\mathrm{F}})^*$ is very similar to the results observed for the first scenario. However, one can clearly see that this field appears to be sensibly smoothed out by the multigrid iterative procedures in step 4 and 5 of the MEnKF algorithm. As previously discussed, complete conservativity starting from an erroneous state at $k-1$ is possibly not an optimal objective, while one wants a regularized solution to avoid affecting the precision of the global calculation. On this last objective, the MEnKF appears to provide a better result when compared with the classical Dual EnKF, described in the first scenario. Considering also that the MEnKF showed better accuracy than the algorithm relying on parameter estimation only, one can conclude that the MEnKF provides an efficient compromise between global accuracy and regularization of the solution. In order to draw more information about this important aspect, the MEnKF algorithm needs to be tested for the simulation of three-dimensional compressible flows, where the Kalman gain correction may be responsible for important acoustic phenomena which are not observable in 2D and 1D dynamical systems.

7. Conclusions

Solving a data assimilation problem with a sequential approach is strongly constrained in real configurations by the excessive computational costs of the methods based on Kalman filters. In this paper, we proposed a strategy, called Multigrid Ensemble Kalman Filter (MEnKF), that relies on Dual Ensemble Kalman filter and targets data assimilation of unsteady fluid flows. This estimator exploits multigrid iterative features which are employed in many CFD codes for the resolution of complex applications in fluid mechanics. From this point of view, our method falls into the class of *multilevel* techniques that aim at improving the estimation of statistics of expensive numerical simulations by considering different levels of resolution of the same set of equations. These low-resolution solutions are then considered as the members of an EnKF approach to determine: *i*) a correction of the numerical model by projection of the estimation on the high-resolution grid and *ii*) an optimization of the free parameters driving the simulation. One of the main advantages of the proposed approach is that, owing to the iterative procedure for the calculation of the flow variables, the final solution is regularized. For turbulent flows, assimilating data from a coarse grid level to a fine grid level can be conceived as filtering the state estimation correction so that only the large scale contribution is kept. Classical turbulence theories such as K41 [51, 52] indicate that the characteristic features of the flow are intimately associated with the large scales, while the small scales exhibit a more universal behaviour. Within this framework, one could envision that the MEnKF will assimilate the flow dependent information associated with the large structures and let the model calculate the small scales, which would arguably exhibit more universal features.

Our method is assessed via the analysis of one-dimensional and two-dimensional test cases for different dynamical equations. First, the one-dimensional Burgers equation at $Re = 200$ is analysed. Here, the performance of MEnKF is assessed considering several coarsening ratios r_{C}, which determines the difference in resolution between the main simulation and the ensemble members. The results for

$r_C = 1$ (*i.e.* method equivalent to a classical EnKF) indicate that the State Estimation and the parametric optimization of the inlet parameters provide very high accuracy in the results. With increasing coarsening ratios the quality of the results is progressively degraded, but the main features of the flow are obtained even for very under-resolved ensemble members. In addition, higher r_C values are associated with significantly decreased computational costs, so that this method exhibit a potential to be explored for efficient trade-off between accuracy and resources required.

Then, MEnKF is used to track the time evolution of a free parameter for the case of a wave propagation, using a one-dimensional Euler model. Three cases are here investigated, varying the time window between successive assimilations. The estimator can efficiently represent the evolution in time of the parameter, as well as to provide an accurate state estimation. However, the global prediction is significantly degraded if the assimilation window is larger than a threshold value, which is arguably connected to the physical features of the flow.

At last, the analysis of the two-dimensional spatially evolving mixing layer at Re = 100 is performed. The algorithm appears to be well suited for the analysis of unsteady phenomena, in particular for the analysis of time varying free parameters of the simulation. These features are promising for potential application to in-streaming Data Assimilation techniques.

Future research will target improvement of the coarse grain resolution by using information from the physical state calculated on the fine grid. Preliminary tests have shown that, in the case of non-linear models, the parameter $\gamma^{(i)}$ in (24) can be optimized using data from the fine level of the grid, significantly improving the accuracy of the prediction of the ensemble members. This implies a more accurate estimation of the state and of the parameters on the coarse level, from which the main refined simulation will benefit. This process has the potential to improve even more the performance of the MEnKF model. Strategies for efficient application are currently under investigation. Moreover, we plan to test the MEnKF algorithm on more complex configurations involving complex geometries and more challenging parametric optimization problems. The

compressible effects are relatively low for the tests performed so far. Further research is planned on test-cases where the compressible effects are more accentuated.

Acknowledgements: Our research activities are supported by the Direction Générale de l'Armement (DGA) and the Région Nouvelle Aquitaine. LC would like to acknowledge the supplementary funding and excellent working conditions offered by the French Agence Nationale de la Recherche (ANR) in the framework of the ASTRID project "Turbulent flow closed-loop control with machine learning – FLOwCON" (ANR-17-ASTR0022) and project "Closed- loop control of the wake of a road vehicle – COWAVE" (ANR-17-CE22-0008). Prof. Heng Xiao is also warmly acknowledged for valuable discussion on the subject.

Conflict of interest: The Authors have no conflict of interests.

Appendix A. Data Assimilation algorithms

Appendix A.1. Kalman filter algorithm

The Kalman filter algorithm given in Sec. 2.1.1 corresponds to Fig. A.22.

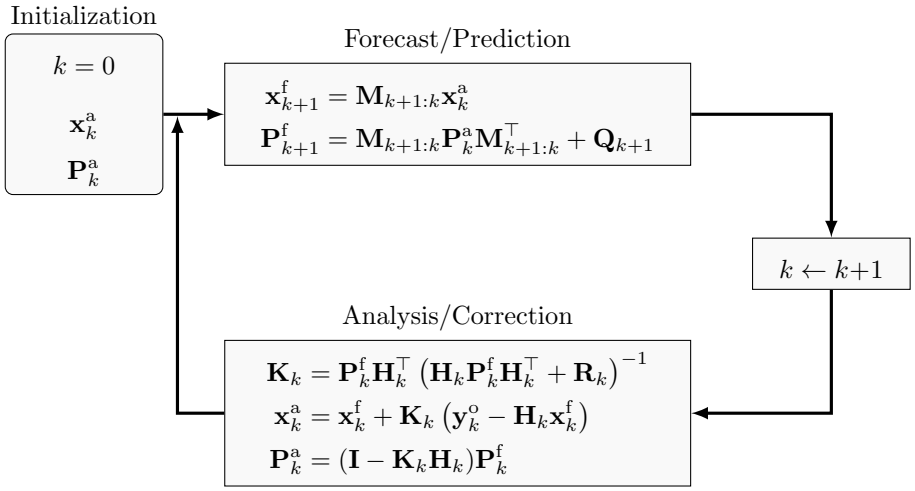


Figure A.22: Kalman Filter algorithm. The initialization is made with the analysed state.

Appendix A.2. Ensemble Kalman filter algorithm

An efficient implementation of the EnKF relying on anomaly matrices is given in Algo. 1. We have used the secant method described in [5] to change the definition of the variable $\mathbf{Y}_k^{\mathrm{f}}$.

Algorithm 1: Stochastic Ensemble Kalman Filter (slightly adapted from [5]). Use of anomaly matrices with $\mathbf{Y}_k^{\mathrm{f}} = \mathbf{H}_k \mathbf{X}_k^{\mathrm{f}}$.

Input: For $k = 0, \ldots, K$: the forward models $\mathcal{M}_{k:k-1}$, the observation models $\mathcal{H}_k$, the observation error covariance matrices $\mathbf{R}_k$

Output: $\{\mathbf{x}_k^{\mathrm{a},(i)}\}\,;\ k = 0, \cdots, K\,;\ i = 1, \cdots, N_{\mathrm{e}}$

begin

Initialize the ensemble of forecasts $\{\mathbf{x}_0^{\mathrm{f},(i)}\}\,;\ i = 1, \cdots, N_{\mathrm{e}}$

for $k = 0, \ldots, K$ **do**

Draw a statistically consistent observation set$\,;\ i = 1, \cdots, N_{\mathrm{e}}$

$$\mathbf{y}_k^{\mathrm{o},(i)} = \mathbf{y}_k^{\mathrm{o}} + \epsilon_k^{\mathrm{o},(i)} \quad \text{with} \quad \epsilon_k^{\mathrm{o},(i)} \sim \mathcal{N}(0, \mathbf{R}_k)$$

Compute the model counterparts of the observation set$\,;$ $i = 1, \cdots, N_{\mathrm{e}}$

$$\mathbf{y}_k^{\mathrm{f},(i)} = \mathcal{H}_k\left(\mathbf{x}_k^{\mathrm{f},(i)}\right)$$

Compute the ensemble means

$$\overline{\mathbf{x}_k^{\mathrm{f}}} = \frac{1}{N_{\mathrm{e}}} \sum_{i=1}^{N_{\mathrm{e}}} \mathbf{x}_k^{\mathrm{f},(i)}\ ;\ \overline{\mathbf{y}_k^{\mathrm{f}}} = \frac{1}{N_{\mathrm{e}}} \sum_{i=1}^{N_{\mathrm{e}}} \mathbf{y}_k^{\mathrm{f},(i)}\ ;\ \overline{\epsilon_k^{\mathrm{o}}} = \frac{1}{N_{\mathrm{e}}} \sum_{i=1}^{N_{\mathrm{e}}} \epsilon_k^{\mathrm{o},(i)}$$

Compute the normalized anomalies$\,;\ i = 1, \cdots, N_{\mathrm{e}}$

$$\left[\mathbf{X}_k^{\mathrm{f}}\right]_{:,i} = \frac{\mathbf{x}_k^{\mathrm{f},(i)} - \overline{\mathbf{x}_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}}\ ;\ \left[\mathbf{Y}_k^{\mathrm{f}}\right]_{:,i} = \frac{\mathbf{y}_k^{\mathrm{f},(i)} - \overline{\mathbf{y}_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}}\ ;\ \left[\mathbf{E}_k^{\mathrm{o}}\right]_{:,i} = \frac{\epsilon_k^{\mathrm{o},(i)} - \overline{\epsilon_k^{\mathrm{o}}}}{\sqrt{N_{\mathrm{e}} - 1}}$$

Compute the Kalman gain

$$\mathbf{K}_k^{\mathrm{e}} = \mathbf{X}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^\top \left(\mathbf{Y}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^\top + \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}}\right)^\top\right)^{-1}$$

Update the ensemble$\,;\ i = 1, \cdots, N_{\mathrm{e}}$

$$\mathbf{x}_k^{\mathrm{a},(i)} = \mathbf{x}_k^{\mathrm{f},(i)} + \mathbf{K}_k^{\mathrm{e}}\left(\mathbf{y}_k^{\mathrm{o},(i)} - \mathbf{y}_k^{\mathrm{f},(i)}\right)$$

Compute the ensemble forecast$\,;\ i = 1, \cdots, N_{\mathrm{e}}$

$$\mathbf{x}_{k+1}^{\mathrm{f},(i)} = \mathcal{M}_{k+1:k}(\mathbf{x}_k^{\mathrm{a},(i)})$$

Appendix A.3. Dual Ensemble Kalman filter algorithm

An efficient implementation of the Dual EnKF relying on anomaly matrices is given in Algo. 2. We have slightly adapted this algorithm from [38].

Appendix A.4. Multigrid Ensemble Kalman filter algorithm

The algorithm 4 represents a simplified, ready-to-use application of the conceptual methodology presented in Sec. 3. This algorithm was tailored for the

Algorithm 2: Dual Ensemble Kalman Filter (slightly adapted from [38]). Use of anomaly matrices with $\mathbf{Y}_k^{\mathrm{f}} = \mathbf{H}_k \mathbf{X}_k^{\mathrm{f}}$. We have $i = 1, \cdots, N_{\mathrm{e}}$.

Input: For $k = 1, \ldots, K$: the forward models $\mathcal{M}_{k:k-1}$, the observation models $\mathcal{H}_k$, the observation error covariance matrices $\mathbf{R}_k$

Output: $\{\theta_k^{\mathrm{a},(i)}\}$ and $\{\mathbf{x}_k^{\mathrm{a},(i)}\}$; $k = 0, \cdots, K$

begin

Initialize $\{\theta_0^{\mathrm{a},(i)}\}$ and $\{\mathbf{x}_0^{\mathrm{a},(i)}\}$

for $k = 1, \ldots, K$ **do**

Observation ensemble:

$$\mathbf{y}_k^{\mathrm{o},(i)} = \mathbf{y}_k^{\mathrm{o}} + \epsilon_k^{\mathrm{o},(i)} \quad \text{with} \quad \epsilon_k^{\mathrm{o},(i)} \sim \mathcal{N}(0, \mathbf{R}_k)$$

$$\mathbf{R}_k^{\mathrm{e}} = \frac{1}{N_{\mathrm{e}} - 1} \sum_{i=1}^{N_{\mathrm{e}}} \epsilon_k^{\mathrm{o},(i)} \left(\epsilon_k^{\mathrm{o},(i)}\right)^{\top}$$

Parameter forecast:

$$\theta_k^{\mathrm{f},(i)} = \theta_{k-1}^{\mathrm{a},(i)} + \tau_k^{(i)} \quad \text{with} \quad \tau_k^{(i)} \sim \mathcal{N}(0, \mathbf{\Sigma}_k^{\theta})$$

$$\mathbf{x}_k^{\mathrm{f},(i)} = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}^{\mathrm{a},(i)}, \theta_k^{\mathrm{f},(i)})$$

$$\mathbf{y}_k^{\mathrm{f},(i)} = \mathcal{H}_k\left(\mathbf{x}_k^{\mathrm{f},(i)}\right)$$

Compute the normalized anomalies

$$\left[\mathbf{\Theta}_k^{\mathrm{f}}\right]_{:,i} = \frac{\theta_k^{\mathrm{f},(i)} - \overline{\theta_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}} \; ; \; \left[\mathbf{Y}_k^{\mathrm{f}}\right]_{:,i} = \frac{\mathbf{y}_k^{\mathrm{f},(i)} - \overline{\mathbf{y}_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}} \; ; \; \left[\mathbf{E}_k^{\mathrm{o}}\right]_{:,i} = \frac{\epsilon_k^{\mathrm{o},(i)} - \overline{\epsilon_k^{\mathrm{o}}}}{\sqrt{N_{\mathrm{e}} - 1}}$$

Parameter update:

$$\mathbf{K}_k^{\theta,\mathrm{e}} = \mathbf{\Theta}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^{\top} \left(\mathbf{Y}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^{\top} + \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}}\right)^{\top}\right)^{-1}$$

$$\theta_k^{\mathrm{a},(i)} = \theta_k^{\mathrm{f},(i)} + \mathbf{K}_k^{\theta,\mathrm{e}} \left(\mathbf{y}_k^{\mathrm{o},(i)} - \mathbf{y}_k^{\mathrm{f},(i)}\right)$$

State forecast:

$$\mathbf{x}_k^{\mathrm{f},(i)} = \mathcal{M}_{k:k-1}(\mathbf{x}_{k-1}^{\mathrm{a},(i)}, \theta_k^{\mathrm{a},(i)})$$

$$\mathbf{y}_k^{\mathrm{f},(i)} = \mathcal{H}_k\left(\mathbf{x}_k^{\mathrm{f},(i)}\right)$$

Compute the normalized anomalies

$$\left[\mathbf{X}_k^{\mathrm{f}}\right]_{:,i} = \frac{\mathbf{x}_k^{\mathrm{f},(i)} - \overline{\mathbf{x}_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}} \; ; \; \left[\mathbf{Y}_k^{\mathrm{f}}\right]_{:,i} = \frac{\mathbf{y}_k^{\mathrm{f},(i)} - \overline{\mathbf{y}_k^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}} \; ; \; \left[\mathbf{E}_k^{\mathrm{o}}\right]_{:,i} = \frac{\epsilon_k^{\mathrm{o},(i)} - \overline{\epsilon_k^{\mathrm{o}}}}{\sqrt{N_{\mathrm{e}} - 1}}$$

State update:

$$\mathbf{K}_k^{x,\mathrm{e}} = \mathbf{X}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^{\top} \left(\mathbf{Y}_k^{\mathrm{f}} \left(\mathbf{Y}_k^{\mathrm{f}}\right)^{\top} + \mathbf{E}_k^{\mathrm{o}} \left(\mathbf{E}_k^{\mathrm{o}}\right)^{\top}\right)^{-1}$$

$$\mathbf{x}_k^{\mathrm{a},(i)} = \mathbf{x}_k^{\mathrm{f},(i)} + \mathbf{K}_k^{x,\mathrm{e}} \left(\mathbf{y}_k^{\mathrm{o},(i)} - \mathbf{y}_k^{\mathrm{f},(i)}\right)$$

Algorithm 3: Dual Ensemble Kalman filter of Algo. 2 applied on the coarse mesh. We have $i = 1, \cdots, N_{\mathrm{e}}$.

Input: For $k = 1, \ldots, K$: the forward models $\mathcal{M}^{\mathrm{C}}_{k:k-1}$, the observation models $\mathcal{H}^{\mathrm{C}}_k$, the observation error covariance matrices $\mathbf{R}^{\mathrm{C}}_k$

Output: $\{\theta^{\mathrm{a},(i)}_k\}$ and $\{(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{a},(i)}\}$; $k = 0, \cdots, K$

begin

Initialize $\{\theta^{\mathrm{a},(i)}_0\}$ and $\{(\mathbf{x}^{\mathrm{C}}_0)^{\mathrm{a},(i)}\}$;

for $k = 1, \ldots, K$ **do**

Parameter forecast:

$$\theta^{\mathrm{f},(i)}_k = \theta^{\mathrm{a},(i)}_{k-1} + \tau^{(i)}_k \quad \text{with} \quad \tau^{(i)}_k \sim \mathcal{N}(0, \mathbf{\Sigma}^{\theta}_k)$$

$$(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)} = \mathcal{M}^{\mathrm{C}}_{k:k-1}\left((\mathbf{x}^{\mathrm{C}}_{k-1})^{\mathrm{a},(i)}, \theta^{\mathrm{f},(i)}_k\right)$$

if *Observation available* **then**

$$(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)} = \mathcal{H}^{\mathrm{C}}_k\left((\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)}\right)$$

Observation ensemble:

$$(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{o},(i)} = (\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{o}} + (\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)} \quad \text{with} \quad (\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)} \sim \mathcal{N}(0, \mathbf{R}^{\mathrm{C}}_k)$$

$$(\mathbf{R}^{\mathrm{C}}_k)^{\mathrm{e}} = \frac{1}{N_{\mathrm{e}} - 1} \sum_{i=1}^{N_{\mathrm{e}}} (\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)} \left((\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)}\right)^{\top}$$

Compute the normalized anomalies

$$\left[\mathbf{\Theta}^{\mathrm{f}}_k\right]_{:,i} = \frac{\theta^{\mathrm{f},(i)}_k - \overline{\theta^{\mathrm{f}}_k}}{\sqrt{N_{\mathrm{e}} - 1}}; \left[\mathbf{Y}^{\mathrm{f}}_k\right]_{:,i} = \frac{(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)} - \overline{(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}}; [\mathbf{E}^{\mathrm{o}}_k]_{:,i} = \frac{(\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)} - \overline{(\epsilon^{\mathrm{C}}_k)^{\mathrm{o}}}}{\sqrt{N_{\mathrm{e}} - 1}}$$

Parameter update:

$$(\mathbf{K}^{\mathrm{C}}_k)^{\theta,\mathrm{e}} = \mathbf{\Theta}^{\mathrm{f}}_k \left(\mathbf{Y}^{\mathrm{f}}_k\right)^{\top} \left(\mathbf{Y}^{\mathrm{f}}_k \left(\mathbf{Y}^{\mathrm{f}}_k\right)^{\top} + \mathbf{E}^{\mathrm{o}}_k \left(\mathbf{E}^{\mathrm{o}}_k\right)^{\top}\right)^{-1}$$

$$\theta^{\mathrm{a},(i)}_k = \theta^{\mathrm{f},(i)}_k + (\mathbf{K}^{\mathrm{C}}_k)^{\theta,\mathrm{e}} \left((\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{o},(i)} - (\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)}\right)$$

State forecast:

$$(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)} = \mathcal{M}^{\mathrm{C}}_{k:k-1}\left((\mathbf{x}^{\mathrm{C}}_{k-1})^{\mathrm{a},(i)}, \theta^{\mathrm{a},(i)}_k\right)$$

$$(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)} = \mathcal{H}^{\mathrm{C}}_k\left((\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)}\right)$$

Compute the normalized anomalies

$$\left[\mathbf{X}^{\mathrm{f}}_k\right]_{:,i} = \frac{(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)} - \overline{(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}}; \left[\mathbf{Y}^{\mathrm{f}}_k\right]_{:,i} = \frac{(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)} - \overline{(\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f}}}}{\sqrt{N_{\mathrm{e}} - 1}}; [\mathbf{E}^{\mathrm{o}}_k]_{:,i} = \frac{(\epsilon^{\mathrm{C}}_k)^{\mathrm{o},(i)} - \overline{(\epsilon^{\mathrm{C}}_k)^{\mathrm{o}}}}{\sqrt{N_{\mathrm{e}} - 1}}$$

State update:

$$(\mathbf{K}^{\mathrm{C}}_k)^{x,\mathrm{e}} = \mathbf{X}^{\mathrm{f}}_k \left(\mathbf{Y}^{\mathrm{f}}_k\right)^{\top} \left(\mathbf{Y}^{\mathrm{f}}_k \left(\mathbf{Y}^{\mathrm{f}}_k\right)^{\top} + \mathbf{E}^{\mathrm{o}}_k \left(\mathbf{E}^{\mathrm{o}}_k\right)^{\top}\right)^{-1}$$

$$(\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{a},(i)} = (\mathbf{x}^{\mathrm{C}}_k)^{\mathrm{f},(i)} + (\mathbf{K}^{\mathrm{C}}_k)^{x,\mathrm{e}} \left((\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{o},(i)} - (\mathbf{y}^{\mathrm{C}}_k)^{\mathrm{f},(i)}\right)$$

Algorithm 4: Multigrid EnKF algorithm. We have $i = 1, \cdots, N_{\mathrm{e}}$.

begin

Initialize $\left\{ \left(\mathbf{x}_0^{\mathrm{F}}\right)^{\mathrm{a}}, \overline{\theta_0^{\mathrm{a}}}, \theta_0^{\mathrm{a},(i)}, \left(\mathbf{x}_0^{\mathrm{C}}\right)^{\mathrm{a},(i)} \right\}$

for $k = 1, \ldots, K$ **do**

Fine grid forecast:

$$\left(\mathbf{x}_k^{\mathrm{F}}\right)^{\mathrm{f}} = \mathcal{M}_{k:k-1}^{\mathrm{F}} \left(\left(\mathbf{x}_{k-1}^{\mathrm{F}}\right)^{\mathrm{a}}, \overline{\theta_k^{\mathrm{a}}} \right)$$

Dual EnKF on coarse mesh: Apply Algo. 3

if *Observation available* **then**

Projection on the coarse grid

$$\left(\mathbf{x}_k^{\mathrm{C}}\right)^{*} = \Pi_{\mathrm{C}} \left(\left(\mathbf{x}_k^{\mathrm{F}}\right)^{\mathrm{f}} \right)$$

Fine grid state correction using the ensemble statistics:

$$\left(\mathbf{x}_k^{\mathrm{C}}\right)^{'} = \left(\mathbf{x}_k^{\mathrm{C}}\right)^{*} + \left(\mathbf{K}_k^{\mathrm{C}}\right)^{x,\mathrm{e}} \left[\left(\mathbf{y}_k^{\mathrm{C}}\right)^{\mathrm{o}} - \mathcal{H}_k^{\mathrm{C}} \left(\left(\mathbf{x}_k^{\mathrm{C}}\right)^{*} \right) \right]$$

$$\left(\mathbf{x}_k^{\mathrm{F}}\right)^{'} = \left(\mathbf{x}_k^{\mathrm{F}}\right)^{\mathrm{f}} + \Pi_{\mathrm{F}} \left(\left(\mathbf{x}_k^{\mathrm{C}}\right)^{'} - \left(\mathbf{x}_k^{\mathrm{C}}\right)^{*} \right)$$

$\left(\mathbf{x}_k^{\mathrm{F}}\right)^{\mathrm{a}}$ is obtained through a matrix-splitting iterative procedure starting from $\left(\mathbf{x}_k^{\mathrm{F}}\right)^{'}$.

relatively simple physical models used in this work. While it may not be suited for complex three-dimensional applications, it proved an optimum trade-off in accuracy and computational resources for the present analysis.

First, when observation is not available, the two main forecast operations (fine grid forecast and ensemble coarse forecast) are performed using explicit time advancement schemes. This choice allows to reduce the computational costs. However, when observation is available, the following strategies are employed:

1. The two forecast operations (main simulation and ensemble members) are performed using an implicit matrix-splitting iterative procedure, using a single iteration. As stated in Sec. 3, the state transition model for each ensemble member is determined independently, but using the same structure and discretization schemes of the main simulation.
2. The number of iterative solutions for the main simulation on the coarse-grid level is equal to zero. That is, the solution from the first forecast is projected on the coarse grid, and the difference between the KF state estimation and this forecast is re-projected over the fine grid. Note that in this version, the communication between coarse ensemble and fine grid simulation is one directional, *i.e.* the coarse Dual EnKF simulation influences the fine grid forecast through parametric optimization and state correction, but it is not affected by the physical state calculated in the fine grid.
3. In the final iteration on the fine grid, an implicit matrix-splitting iterative procedure is employed, using a single iteration and a relaxation coefficient $\alpha = 0.5$. This choice, which provides the best compromise between accuracy and regularization, has been identified after extensive tests for the configurations investigated.

References

[1] W. L. Oberkampf, T. G. Trucano, Verification and validation in computational fluid dynamics, Progress in Aerospace Sciences 38 (3) (2002) 209–272. doi:https://doi.org/10.1016/S0376-0421(02)00005-2. URL http://www.sciencedirect.com/science/article/pii/S0376042102000052

[2] J. H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, Springer, 2002.

[3] S. B. Pope, Turbulent flows, Cambridge University Press, 2000.

[4] K. Zhou, J. C. Doyle, K. Glover, Robust and optimal control, Prentice Hall, 1996.

[5] M. Asch, M. Bocquet, M. Nodet, Data Assimilation: methods, algorithms, and applications, SIAM, 2016.

[6] G. Evensen, Data Assimilation: The Ensemble Kalman Filter, Springer-Verlag/Berlin/Heildelberg, 2009.

[7] S. B. Daley, Atmospheric Data Analysis, Cambridge University Press, 1991.

[8] A. Onder, J. Meyers, Optimal control of a transitional jet using a continuous adjoint method, Computers & Fluids 126 (2016) 12–24.

[9] D. P. G. Foures, N. Dovetta, D. Sipp, P. J. Schmid, A data-assimilation method for Reynolds-averaged Navier-Stokes-driven mean flow reconstruction, Journal of Fluid Mechanics 759 (2014) 404–431.

[10] V. Mons, Q. Wang, T. A. Zaki, Kriging-enhanced ensemble variational data assimilation for scalar-source identification in turbulent environments, Journal of Computational Physics 398 (2019) 108856.

[11] P. Chandramouli, E. Memin, D. Heitz, 4d large scale variational data assimilation of a turbulent flow with a dynamics error model, J. Comp. Phys.

412 (2020) 109446. doi:https://doi.org/10.1016/j.jcp.2020.109446. URL http://www.sciencedirect.com/science/article/pii/S0021999120302205

[12] Z. Sirkes, E. Tziperman, Finite Difference of Adjoint or Adjoint of Finite Difference?, Monthly Weather Review 125 (1997) 3373–3378.

[13] R. E. Kalman, A new approach to linear filtering and prediction problems, Journal of Basic Engineering 82 (1960) 35–45.

[14] W. G., B. G., An Introduction to the Kalman Filter, Technical report - tr 95-041, University of North Carolina at Chapel Hill (2006).

[15] D. Rozier, F. Birol, E. Cosme, P. Brasseur, J.-M. Brankart, J. Verron, A Reduced-Order Kalman Filter for Data Assimilation in Physical Oceanography, SIAM Review 49 (3) (2007) 449–465. doi:10.1137/050635717.

[16] T. Suzuki, Reduced-order Kalman-filtered hybrid simulation combining particle tracking velocimetry and direct numerical simulation, Journal of Fluid Mechanics 709 (2012) 249 – 288.

[17] M. Meldi, A. Poux, A reduced order model based on Kalman Filtering for sequential Data Assimilation of turbulent flows, Journal of Computational Physics 347 (2017) 207–234.

[18] M. Meldi, Augmented Prediction of Turbulent Flows via Sequential Estimators: Sensitivity of State Estimation to Density of Time Sampling for Available Observation, Flow, Turbulence and Combustion 101 (2018) 389–412.

[19] G. Evensen, The ensemble Kalman Filter for combined state and parameter estimation - Monte Carlo techniques for data assimilation in large systems, IEEE Control Systems 29 (2009) 83–104.

[20] V. Mons, J. C. Chassaing, T. Gomez, P. Sagaut, Reconstruction of unsteady viscous flows using data assimilation schemes, Journal of Computational Physics 316 (2016) 255–280.

[21] H. Xiao, J. L. Wu, J. X. Wang, R. Sun, C. Roy, Quantifying and reducing model-form uncertainties in reynolds-averaged navier–stokes simulations: A data-driven, physics informed bayesian approach, Journal of Computational Physics 324 (2016) 115–136.

[22] M. C. Rochoux, S. Ricci, D. Lucor, B. Cuenot, A. Trouve, Towards predictive data-driven simulations of wildfire spread - Part I: Reduced-cost Ensemble Kalman Filter based on a Polynomial Chaos surrogate model for parameter estimation, Natural Hazards and Earth System Sciences 14 (2015) 2951–2973.

[23] D. Simon, T. L. Chia, Kalman filtering with state equality constraints, IEEE Trans. Aerosp. Electron. Syst. 38 (2002) 128–136.

[24] G. Nachi, Kalman filtering in the presence of state space equality constraints, IEEE Chinese Control Conference (2007) 107–133.

[25] X. L. Zhang, C. M. Ströfer, H. Xiao, Regularized ensemble kalman methods for inverse problems, J. Comp. Phys. 416 (2020) 109517.

[26] A. Brandt, Multi-level adaptive solutions to boundary-value problems, Mathematics of Computation 31 (1977) 333–390.

[27] P. Wesseling, C. W. Oosterlee, Geometric multigrid with applications to computational fluid dynamics, Journal of Computational and Applied Mathematics 128 (2001) 311–334.

[28] H. Hoel, K. J. H. Law, R. Tempone, Multilevel ensemble kalman filtering, SIAM J. Numer. Anal. 54(3) (2016) 1813–1839.

[29] A. Siripatana, L. Giraldi, O. P. Le Maître, O. M. Knio, I. Hoteit, Combining ensemble kalman filter and multiresolution analysis for efficient assimilation into adaptive mesh models, Computational Geosciences 23 (2019) 1259–1276.

[30] K. Fossum, T. Mannseth, A. S. Stordal, Kalman filtering with state equality constraints, Assessment of multilevel ensemble-based data assimilation for reservoir history matching 24 (2020) 217–239.

[31] K. Law, H. Hoel, R. Tempone, F. Nobile, A. Chernov, Multilevel ensemble kalman filtering for spatio-temporal processes, NUMERISCHE MATHEMATIK (Nov. 2020). doi:10.1007/s00211-020-01159-3.

[32] L. Debreu, E. Neveu, E. Simon, F.-X. Le Dimet, A. Vidard, Multigrid solvers and multigrid preconditioners for the solution of variational data assimilation problems, Quarterly Journal of the Royal Meteorological Society 142 (2015) 515–528.

[33] G. Stonebridge, Diagonal Approximations to the ObservationError Covariance Matrix in Sea Ice ThicknessData Assimilation, Master of science, University of Waterloo (2017).

[34] S. L. Brunton, B. R. Noack, Closed-Loop Turbulence Control: Progress and Challenges, Applied Mechanics Review 67 (2015) 050801.

[35] G. Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using monte-carlo methods to forecast error statistics, Journal of Geophysical Research 99 (1994) 10143–10162. doi:10.1029/94JC00572.

[36] G. Burgers, P. J. Van Leeuwen, G. Evensen, On the analysis scheme in the ensemble kalman filter, Monthly Weather Review 126 (06 1998). doi:10.1175/1520-0493(1998)126<1719:ASITEK>2.0.CO;2.

[37] J. W. Labahn, H. Wu, B. Coriton, J. H. Frank, M. Ihme, Data assimilation using high-speed measurements and LES to examine local extinction events in turbulent flames, Proceedings of the Combustion Institute 37 (2019) 2259–2266.

[38] M. Hamid, S. Sorooshian, H. Gupta, P. Houser, Dual state-parameter estimation of hydrological models using ensemble kalman filter, Advances in

Water Resources 28 (2005) 135–147. doi:10.1016/j.advwatres.2004.09.002.

[39] W. Hackbusch, Multi-grid methods and applications, Springer Berlin, 1985.

[40] A. A. Popov, C. Mou, T. Iliescu, A. Sandu, A multifidelity ensemble kalman filter with reduced order control variates, arXiv (2020) 1–28.

[41] A. A. Gorodetsky, G. Geraci, M. S. Eldred, J. D. Jakeman, A generalized approximate control variate framework for multifidelity uncertainty quantification, Journal of Computational Physics 408 (2020) 109257.

[42] P. Sakov, D. S. Olivier, L. Bertino, An iterative enkf for strongly nonlinear systems, Monthly Weather Review 140 (2011) 1988–2004.

[43] J. Brajard, A. Carrassi, M. Bocquet, L. Bertino, Combining data assimilation and machine learning to infer unresolved scale parametrisation, arxiv 2009.04318v1 (2020) 1–16.

[44] P. Tandeo, P. Ailliot, M. Bocquet, A. Carrassi, T. Miyoshi, M. Pulido, Y. Zhen, Joint Estimation of Model and Observation Error Covariance Matrices in Data Assimilation: a Review, Monthly Weather Review 148 (2020) 3973–3994. doi:https://doi.org/10.1175/MWR-D-19-0240.1.

[45] A. J. Jerri, The shannon sampling theorem—its various extensions and applications: A tutorial review, Proceedings of the IEEE 65 (11) (1977) 1565–1596. doi:10.1109/PROC.1977.10771.

[46] P. Schmiechen, Travelling wave speed coincidence, Ph.D. thesis, Imperial College of Science, Technology and Medicine, University of London (1997).

[47] C. Bogey, C. Bailly, A family of low dispersive and low dissipative explicit schemes for flow and noise computations, Journal of Computational Physics 194 (2004) 194–214. doi:10.1016/j.jcp.2003.09.003.

[48] J. Ko, D. Lucor, P. Sagaut, Sensitivity of two-dimensional spatially developing mixing layers with respect to uncertain inflow conditions, Physics Fluids 20 (2008) 077102.

[49] M. Meldi, M. V. Salvetti, P. Sagaut, Quantification of errors in large-eddy simulations of a spatially evolving mixing layer using polynomial chaos, Physics of Fluids 24 (2012) 035101.

[50] W. A. McMullan, S. Gao, C. M. Coats, A comparative study of inflow conditions for two- and three-dimensional spatially developing mixing layers using large eddy simulation, International Journal for Numerical Methods in Fluids 55 (6) (2007) 589–610. doi:https://doi.org/10.1002/fld.1482.
URL https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.1482

[51] A. N. Kolmogorov, The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers, Comptes rendus (Doklady) de l'Academie des sciences de l'URSS XXXII (30) (1941) 301.

[52] A. N. Kolmogorov, Dissipation of energy in the locally isotropic turbulence, Comptes rendus (Doklady) de l'Academie des sciences de l'URSS XXXII (32) (1941) 16–18.