



**HAL**  
open science

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

Gabriel Ionut Moldovan, Guillaume Lehnasch, Laurent Cordier, Marcello  
Meldi

► **To cite this version:**

Gabriel Ionut Moldovan, Guillaume Lehnasch, Laurent Cordier, Marcello Meldi. A multigrid/ensemble Kalman filter strategy for assimilation of unsteady flows. 2020. hal-03088525

**HAL Id: hal-03088525**

**<https://hal.science/hal-03088525>**

Preprint submitted on 26 Dec 2020

**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<sup>a,\*</sup>, G. Lehnasch<sup>a</sup>, L. Cordier<sup>a</sup>, M. Meldi<sup>a</sup>

<sup>a</sup>*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 dynamic 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](mailto: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 es-

timination 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]. Third, standard variational data assimilation methods are *intrusive* from a computational point of view, requiring the development of an adjoint calculation code, or the availability of the source code when the use of automatic differentiation is planned. In most cases, modifications are required in order to maintain the code or extend the applications (change of the definitions of errors, for instance). For commercial software without open-source licence, these modifications are expensive. *Non-intrusive* methods that

require no modification of original calculation code are therefore preferable in data assimilation.

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 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 consists in 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. A more popular strategy for data assimilation of 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]. Despite the interest of this non-intrusive technique, and the possibility to perform efficient parametric inference, the computational costs can still be prohibitive for realistic applications. Statistical convergence

is usually obtained for a typical ensemble size going from 60 to 100 ensemble members [5].

In addition, the state estimation obtained via sequential tools does not necessarily comply with a model solution i.e. the *conservativity* of the dynamic 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, we refer to our algorithm 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. This procedure is reminiscent of reduced order / multilevel applications of EnKF strategy reported in the literature [28, 29, 30, 31]. However, the state estimation obtained at the coarse level is 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. Because of the algorithmic structure of the problem, all of 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 algorithm called Multigrid Ensemble Kalman Filter (MEnKF) is discussed. In Sec. 4, the MEnKF 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, namely 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. The objective is to gradually arrive at the Dual ensemble Kalman Filter methodology (Dual EnKF), which is an essential ingredient of the estimator proposed in this work. In Sec. 2.2, the steps that are necessary 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 role of the Kalman filter (KF) is to provide 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 measurements or observations, and the information of a dynamical model (e.g., first principle equations):

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

where  $\mathcal{M}_{k:k-1}$  is a non linear function acting as state-transition model and  $\theta_k$  contains the parameters that affect the state-transition. The term  $\eta_k$  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  $\mathbf{y}_k^o$ . These two variables are related by:

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

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

The model and observation errors being Gaussian, we can show that the solution is characterized entirely 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  $\mathbf{x}_k^{f/a}$  and  $\mathbf{P}_k^{f/a}$ , respectively. The error covariance matrix is defined as  $\mathbf{P}_k^{f/a} = \mathbb{E} \left[ \left( \mathbf{x}_k^{f/a} - \mathbb{E}(\mathbf{x}_k^{f/a}) \right) \left( \mathbf{x}_k^{f/a} - \mathbb{E}(\mathbf{x}_k^{f/a}) \right)^\top \right]$ . In the case of a linear dynamical model ( $\mathcal{M}_{k:k-1} \equiv \mathbf{M}_{k:k-1}$ ) and a linear ob-



servation model ( $\mathcal{H}_k \equiv \mathbf{H}_k$ ), the estimated state is obtained via the following 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  $k$ :

$$\mathbf{x}_k^f = \mathbf{M}_{k:k-1} \mathbf{x}_{k-1}^a \quad (3)$$

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

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

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

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

$$\mathbf{P}_k^a = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^f \quad (7)$$

The optimal prediction of the state ( $\mathbf{x}_k^a$ ) is obtained via the addition to the predictor estimation ( $\mathbf{x}_k^f$ ) 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. First of all, KF classical 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 because of i) nonlinearities in the models and observation operators, ii) poorly known error statistics and iii) a prohibitive computational cost. Considering point ii), 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]. Point iii) is

easily explained by the fact that in engineering applications, the dimension of the physical state  $\mathbf{x}_k$  is huge ( $N \in [10^6, 10^9]$ ). By examining the equations of the canonical Kalman filter, it is immediate to see that the procedure 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)). To bypass this computational cost, one popular strategy is to rely on a statistical description obtained via an ensemble of trajectories of the model dynamics.

### 2.1.2. Ensemble Kalman filter

The Ensemble Kalman Filter (EnKF) [6], introduced by Evensen in 1994 [35], 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  $\frac{1}{\sqrt{N_e}}$ . In the following, we are describing the *stochastic EnKF* flavour of EnKF for which random sampling noise is introduced in the analysis [5].

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

$$\mathcal{E}_k^{\text{f/a}} = \left[ \mathbf{x}_k^{\text{f/a},(1)}, \dots, \mathbf{x}_k^{\text{f/a},(N_e)} \right] \in \mathbb{R}^{N_x \times N_e} \quad (8)$$

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

$$\mathbf{X}_k^{\text{f/a}} = \frac{\left[ \mathbf{x}_k^{\text{f/a},(1)} - \overline{\mathbf{x}_k^{\text{f/a}}}, \dots, \mathbf{x}_k^{\text{f/a},(N_e)} - \overline{\mathbf{x}_k^{\text{f/a}}} \right]}{\sqrt{N_e - 1}} \in \mathbb{R}^{N_x \times N_e}, \quad (9)$$

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

$$\overline{\mathbf{x}_k^{\text{f/a}}} = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{x}_k^{\text{f/a},(i)} \quad (10)$$

The error covariance matrix  $\mathbf{P}_k^{\text{f/a}}$  can thus be estimated via the informa-

tion derived from the ensemble. This estimation, hereafter denoted with the superscript  $e$ , can be factorized into:

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

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  $\mathbf{y}_k^o \in \mathbb{R}^{N_y}$  at an instant  $k$ , the ensemble of perturbed observations is defined as:

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

In a similar way to what we did for the ensemble matrices of the forecast/analysed states, we define the normalized anomaly matrix of the observations errors as

$$\mathbf{E}_k^o = \frac{1}{\sqrt{N_e - 1}} \left[ \epsilon_k^{\text{o,(1)}} - \bar{\epsilon}_k^o, \epsilon_k^{\text{o,(2)}} - \bar{\epsilon}_k^o, \dots, \epsilon_k^{\text{o,(N_e)}} - \bar{\epsilon}_k^o \right] \in \mathbb{R}^{N_y \times N_e} \quad (13)$$

where  $\bar{\epsilon}_k^o = \frac{1}{N_e} \sum_{i=1}^{N_e} \epsilon_k^{\text{o,(i)}}$ .

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

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

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

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

The expression of the Kalman gain is

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

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

Interestingly, it can be shown that the computation of the tangent linear operator  $\mathbf{H}_k$  can be avoided. The details can be found in [5, Sec. 6.3.3]. 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  $\theta_k$  given erroneous observations  $\mathbf{y}_k^o$ . One approach is provided by *joint estimation* where state and parameter vectors are concatenated into a single joint state vector (state augmentation). After [38], the drawback of such strategy is that, by increasing the number of unknown model states and parameters, the degree of freedom in the system increases and makes the estimation unstable and in-

tractable, especially in the non linear dynamical model. For this reason, we follow the procedure developed by [38], called *dual estimation*. The principle is to apply successively two interactive filters, one for the estimation of the parameters from a guessed state solution, the other for the updating 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:

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

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

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

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

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

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

with

$$\overline{\theta_k^{f/a}} = \frac{1}{N_e} \sum_{i=1}^{N_e} \theta_k^{f/a,(i)} \quad (20)$$

Knowing a better approximation of the model's parameters, 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:

$$\frac{D\mathbf{x}}{Dt} = \frac{1}{\rho} \nabla \cdot \bar{\sigma} + \mathbf{f} \quad (21)$$

where  $D/Dt$  is the total (or material) derivative of the physical quantity of investigation  $\mathbf{x}$  and  $\rho$  is the flow density. The divergence operator is indicated as  $\nabla \cdot$  while  $\bar{\sigma}$  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:

$$\mathbf{x}_k = \Phi_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{b}_k \quad (22)$$

where  $\Phi_k$  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  $\Phi_k \mathbf{x}_{k-1}$ , 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:

$$\mathbf{\Psi}_k \mathbf{x}_k = \tilde{\mathbf{\Psi}}_k \mathbf{x}_{k-1} + \tilde{\mathbf{B}}_k \mathbf{b}_k = \mathbf{c}_k \quad (23)$$

where  $\mathbf{\Psi}_k$ ,  $\tilde{\mathbf{\Psi}}_k$  and  $\tilde{\mathbf{B}}_k$  are matrices obtained via the discretization process. Obviously, considering  $\mathbf{\Phi}_k = \mathbf{\Psi}_k^{-1} \tilde{\mathbf{\Psi}}_k$ , 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 is used until the residual  $\delta^n$ , determined at the  $n$ -th iteration, falls below a pre-selected threshold value  $\varepsilon$ . 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, we consider the simplified case of two grids. Thereafter, the variables defined on the fine grid will be denoted with the superscript F ( $\mathbf{x}^F$  for instance), those defined on the coarse grid will be denoted with the superscript C ( $\mathbf{x}^C$  for instance).

The coarse-level representation  $\mathbf{x}^C$  is usually obtained suppressing multiple mesh elements from the initial fine-level one  $\mathbf{x}^F$ . 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  $(\mathbf{x}^0)^F$  (which is usually equal to  $\mathbf{x}$  at the previous time step  $k - 1$ ), an iterative procedure is applied to obtain a first solution  $(\mathbf{x}^1)^F$ . A residual  $(\delta^1)^F = \mathbf{c}^F - \Psi^F(\mathbf{x}^1)^F$  is calculated.
2.  $(\mathbf{x}^1)^F$  and  $(\delta^1)^F$  are projected from the fine grid to the coarse grid space via a projection operator  $\Pi_C$ , so that  $(\mathbf{x}^1)^C$  and  $(\delta^1)^C$  are obtained. Similarly, the state transition model  $\Psi^F$  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  $\Psi^C$ . Finally, we evaluate  $\mathbf{c}^C = \Psi^C(\mathbf{x}^1)^C + (\delta^1)^C$ .
3. An iterative procedure is employed to obtain  $(\mathbf{x}^2)^C$  on the coarse grid using as initial solution  $(\mathbf{x}^1)^C$ .
4. The updated variable on the fine grid is obtained as  $(\mathbf{x}^2)^F = (\mathbf{x}^1)^F + \Pi_F((\mathbf{x}^2)^C - (\mathbf{x}^1)^C)$  where  $\Pi_F$  is a projection operator from the coarse grid to the fine grid.
5. At last, the final solution  $(\mathbf{x}^3)^F$  is obtained via a second iterative procedure on the fine grid starting from the intermediate solution  $(\mathbf{x}^2)^F$ .

This procedure can be repeated multiple times imposing  $(\mathbf{x}^0)^F = (\mathbf{x}^3)^F$  at the beginning of each cycle. When the convergence is reached, the fine grid solution at time instant  $k$  is equal to  $(\mathbf{x}^3)^F$ . Performing part of the calculations on a coarse grid level provides two main advantages [2]. First, a significant reduction in the computational resources is obtained since the calculations performed over the coarse grid are usually much less expensive than a full set of iterations over the fine grid. Second, spurious high-frequency numerical oscillations, due to the discretization error in  $\mathbf{x}^F$ , are naturally filtered via the operators  $\Pi_C$  and  $\Pi_F$ , which globally improves the quality of the numerical solution. In this paper, the two projections ( $\Pi_F$  and  $\Pi_C$ ) are defined as 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 too prohibitive for advanced applications. In the following, we present a Kalman filter strategy which relies on the use of a coarse mesh for the ensemble Kalman filter step. These computations on the coarse mesh are jointly run with a single high-refinement simulation, which is updated using the coarse mesh assimilation results. For this reason, the clock time required for the time advancement of the ensemble members and the memory storage of the physical variables are dramatically reduced. The multigrid-ensemble algorithm works through the steps described below. In the following description, the notation  $\Psi$  might hold for both  $\Psi$  and  $\tilde{\Psi}$  introduced in (23), depending on the choice of the time integration strategy. For its part, the state  $(\mathbf{x}_k^c)^*$  represents the projection on the coarse-grid of  $(\mathbf{x}_k^f)^f$ , the fine-grid solution. Finally,  $(\mathbf{x}_k^c)'$  denotes the coarse-grid state obtained by an ensemble Kalman filtering process applied on the coarse grid. In the FAS multigrid method described in Sec. 2.2,  $(\mathbf{x}_k^c)^*$  and  $(\mathbf{x}_k^c)'$  correspond to  $(\mathbf{x}^1)^c$  and  $(\mathbf{x}^2)^c$ , respectively.

1. **First iteration on the fine grid.** Starting from an initial solution on the fine grid  $(\mathbf{x}_{k-1}^f)^a$ , a forecasted state  $(\mathbf{x}_k^f)^f$  is obtained by using  $\overline{\theta}_k^a$  as parameter for the model  $\Psi^F$ .
2. **Projection on the coarse grid.**  $(\mathbf{x}_k^f)^f$  is projected on the coarse grid space via a projection operator  $\Pi_C$ , so that  $(\mathbf{x}_k^c)^*$  is obtained. Similarly, the state matrix  $\Psi^F$  is projected on the coarse grid to obtain  $\Psi^C$ . This critical passage will be discussed in detail 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  $(\Psi^C)^{(i)}$  used for the advancement in time on the coarse grid is determined<sup>1</sup>. The ensemble forecast  $(\mathbf{x}_k^c)^{f,(i)}$  is corrected with the standard Dual EnKF procedure to obtain  $(\mathbf{x}_k^c)^{a,(i)}$  as well as the parameters  $\theta_k^{a,(i)}$ .

---

<sup>1</sup>While the advancement model is unique (the Navier-Stokes equations, for example), the discretization process contained in  $\Psi$  is unique for each member of the ensemble. To distinguish them, it is therefore necessary to introduce an exponent  $i$  in the notations.

4. **Determination of the state variables on the coarse grid.** If observations are not available, the state  $(\mathbf{x}_k^c)'$  is obtained using the classical multigrid procedure. On the other hand, if observations are available, the ensemble error covariance matrix  $(\mathbf{P}_k^c)^{f,e}$  is used to determine the coarse grid solution  $(\mathbf{x}_k^c)'$  through a Kalman filter estimation.
5. **Final iteration on the fine grid.** A first estimation  $(\mathbf{x}_k^f)'$  of the fine grid state is determined using the results obtained on the coarse space:  $(\mathbf{x}_k^f)' = (\mathbf{x}_k^f)^f + \Pi_F \left( (\mathbf{x}_k^c)' - (\mathbf{x}_k^c)^* \right)$ . The state  $(\mathbf{x}_k^f)^a$  is obtained from a final iterative procedure starting from  $(\mathbf{x}_k^f)'$ .

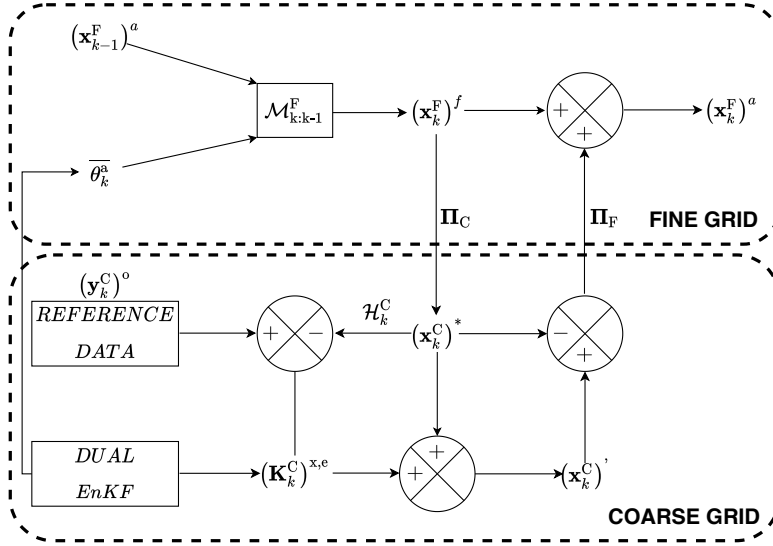


Figure 1: Schematic representation of the Multigrid Ensemble Kalman Filter (MEnKF). Two different levels of representation (fine and coarse grids) are used to obtain a data-driven fine grid estimation. The Dual Ensemble Kalman filter procedure is solved in the coarse grid. The full algorithm is given in Appendix A.

An overview of the assimilation cycle described in the previous algorithm is presented in Fig. 1. Two important aspects need to be discussed:

- As previously stated,  $\Psi^c$  and its role in the determination of the matrices  $(\Psi^c)^{(i)}$  is an essential step in the MEnKF strategy. In non-linear problems of interest in fluid mechanics, the state transition matrix  $\Psi$  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  $\Psi^c$  and  $(\Psi^c)^{(i)}$  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  $\Psi$ . However, one can envision to use the non-linear information conserved in  $\Psi^c$ , which is supposedly accurate, to improve the accuracy of the prediction of the ensemble members. This aspect is discussed in the perspectives included in Sec. 7.

- The recursive structure of the algorithm allows for integration of iterative corrections for non-linear systems [40] 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. In fact, 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  $\Psi$ , 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 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, let us consider the case of a simple two-level geometric multigrid approach

for a 3D test case with a constant coarsening ratio  $r_c = 4$  and a size of ensembles  $N_e = 100$ . 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  $R_{\text{RAM}}$ , 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  $r_c = 8$ , the normalized RAM requirement is  $R_{\text{RAM}} = 1 + 100/8^3 = 1.195$ , thus just a 20% increase in RAM requirements. This is clearly orders of magnitude more advantageous than a fine-grid classical EnKF application with  $N_e = 100$ , since in this case  $R_{\text{RAM}} = N_e = 100$ .

- 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 structures. For iterative methods, the only essential operation to be performed is the determination of the state transition model  $\Psi^c$  and of the projections  $\Pi_c$  and  $\Pi_f$ . This implies that the method can be easily extended to other popular procedures, such as the algebraic multigrid. If the multigrid operations are removed, the method becomes a classical multilevel EnKF method, which relies on just one level of resolution for the ensemble members. However, in this case, 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:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \frac{1}{\text{Re}} \frac{\partial^2 u}{\partial x^2} \quad (24)$$

where  $x$  is the spatial coordinate,  $u$  the velocity and  $\text{Re}$  is the Reynolds number. Equation (24) is non-dimensionalized with a reference velocity  $u_0$  and a reference length  $\lambda$ . This equation is solved with a second-order centered 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:

$$u(x = 0, t) = u_0 (1 + \theta_1 \sin(\omega t + \theta_2)) \quad (25)$$

where  $u_0 = 1$  and  $\omega = 2\pi f_c$ . The characteristic frequency  $f_c$  is set to 1 and the characteristic length  $\lambda$  is defined as  $\lambda = \frac{u_0}{f_c}$ . The characteristic time is defined as  $t_c = \lambda/u_0$ . This implies that, over a complete period of oscillation  $1/f_c$ , the solution is advected of a length  $\lambda$  by the velocity  $u_0$ .  $\theta_1$  and  $\theta_2$  represent the amplitude and phase of the sinusoidal signal, respectively. Note that from now on, the units are the characteristic magnitudes. The outlet boundary condition is extrapolated from the nearest points to the outlet. The initial condition imposed for  $t = 0$  is  $u(x, t = 0) = u_0$  everywhere in the physical domain ( $\theta_2 = 0$  for the reference simulation). The value of the Reynolds number is  $\text{Re} = 200$ . 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  $\lambda$  using 80 mesh elements. This also implies that the total number of nodes employed to perform the calculation is  $N_x = 800$ . A simulation of reference 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.

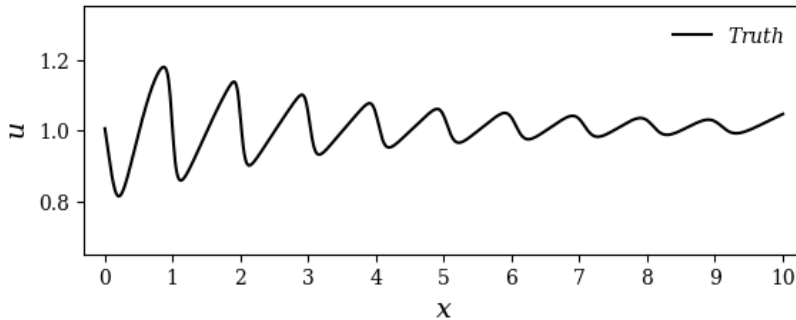


Figure 2: Solution of the 1D Burgers' equation at  $t = 10$  for  $\theta_1 = 0.2$  and  $\theta_2 = 0$  (true state).

Data assimilation is performed in the following conditions:

- The observations are sampled each 30 time steps of the reference simulation on the space domain  $[0, 1]$  (80 sensors) and on the time window  $[10, 29]$ . The observations are sampled as soon as the flow is fully developed. These observations are artificially perturbed using a Gaussian noise of variance  $R = 0.0025$ .
- The *model* is chosen to be the discretized version of (24). The numerical test consists of one main simulation, which is run on the fine grid previously introduced, and an ensemble of  $N_e = 100$  coarse simulations used for assimilation purposes. The coefficients  $\theta_1$  and  $\theta_2$  are initially assumed

to be described by Gaussian distributions, so that  $\theta_1 \sim \mathcal{N}(0, Q_{\theta_1})$  and  $\theta_2 \sim \mathcal{N}(0.3, Q_{\theta_2})$ . The initial value of the covariance of the parameters is chosen equal to  $Q_{\theta_1}(t=0) = Q_{\theta_2}(t=0) = 0.0025$ . 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 analyze the rate of convergence of the optimization procedure. Also, the initial condition  $u(x, 0) = u_0$  is imposed for every simulation.

The data estimation is run for a time window of  $T_{\text{DA}} = 19$  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  $r_c = 1, 2, 4, 8, 16$ . 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 time-evolution of the estimation of  $\theta_1$  is shown in Fig. 3. Very rapid convergence (less than  $2t_c$ ) is observed for  $r_c \leq 4$ . In addition, the parameter estimation is extremely precise (discrepancy lower than 0.01% for  $r_c = 1$ , lower than 2% for  $r_c = 4$ ). For higher values of the parameter  $r_c$ , the estimation of  $\theta_1$  becomes progressively more degraded. For the case  $r_c = 8$ ,  $\theta_1$  is initially overestimated and it finally converges to a value of  $\theta_1 = 0.194$ , 3% smaller than the true value. Larger errors in the optimization of  $\theta_1$  are observed for  $r_c = 16$ . In this case, the optimized amplitude parameter is  $\theta_1 = 0.27$ , which is 35% larger than the true value.

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

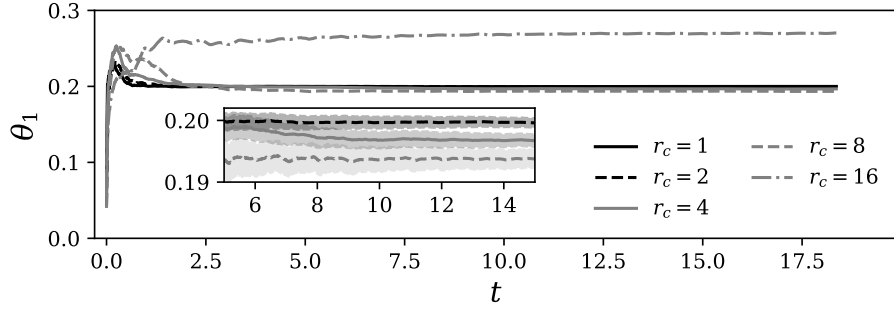


Figure 3: Values of the parameter  $\theta_1$  for different coarsening ratios  $r_C = 1, 2, 4, 8, 16$ . In the zoomed region, the shaded area represents the 95% credible interval for the shown cases.

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

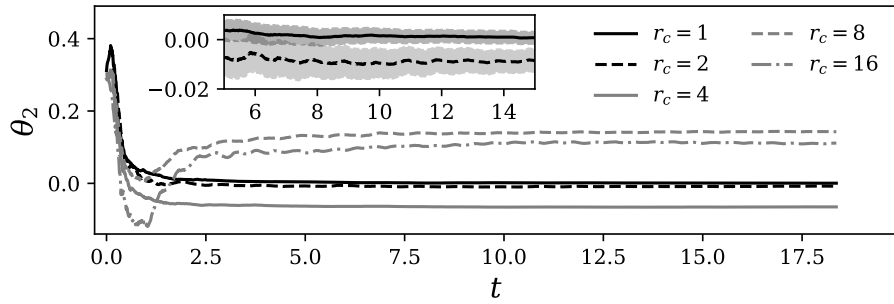


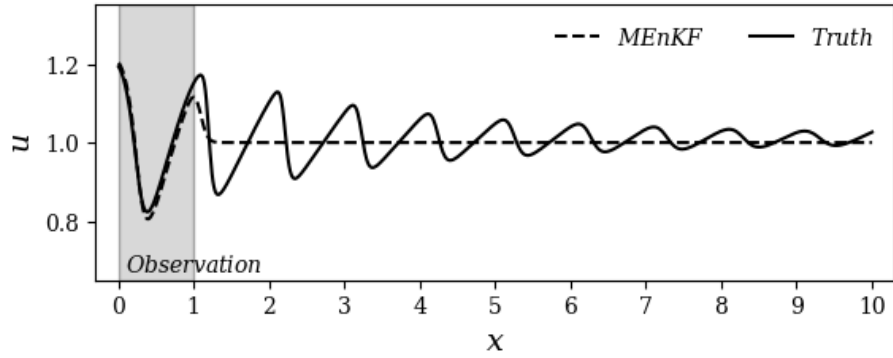
Figure 4: Values of the parameter  $\theta_2$  for different coarsening ratios  $r_C = 1, 2, 4, 8, 16$ . In the zoomed region, the shaded area represents the 95% credible interval for the shown cases.



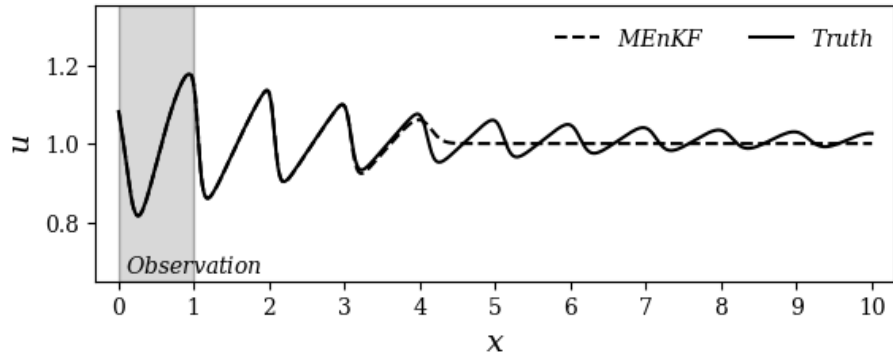
State estimation results for the case  $r_c = 1$  are shown in Fig. 5. This case, which is performed using the same grid for the coarse and fine mesh level, is equivalent to a standard Dual EnKF. However, owing to the final multigrid iterative loop, the final solution is naturally regularized. The results, 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  $r_c = 8$  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  $\theta_1$ , the variable  $u$  is over-predicted for  $t < 2t_c$  while it is slightly under-predicted for  $t > 4t_c$ .

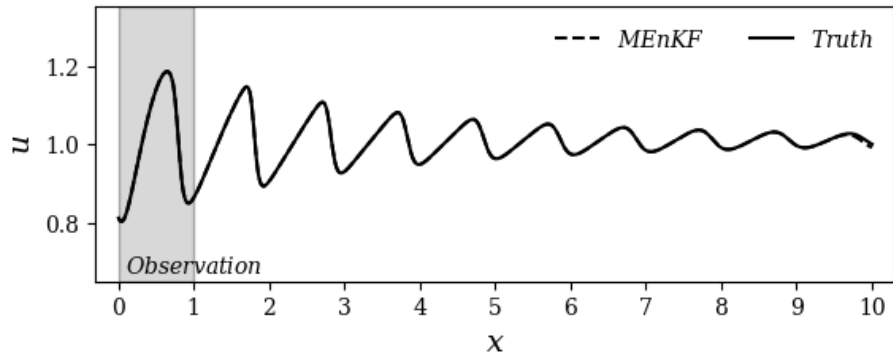
At last, the case for  $r_c = 16$  is shown in Fig. 7. In this case, the mesh elements are only 5 times smaller than the characteristic length  $\lambda$ . Despite the important under-resolution of the ensemble members, which severely affects the estimation of  $\theta_1$  and  $\theta_2$ , the state estimation still adequately represents the main features of the dynamical system.



(a)  $t = 1$

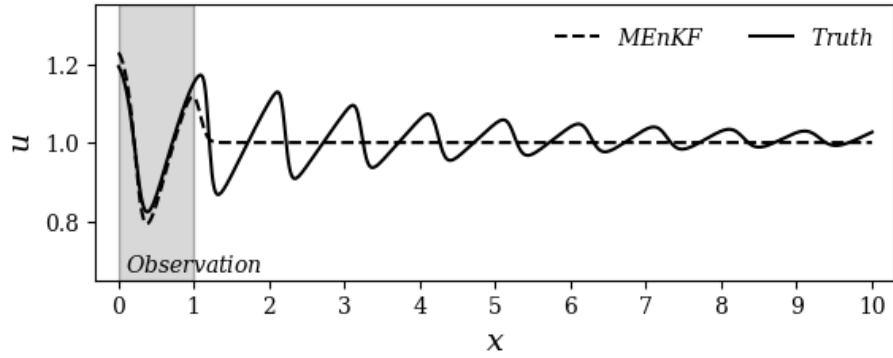


(b)  $t = 3.88$

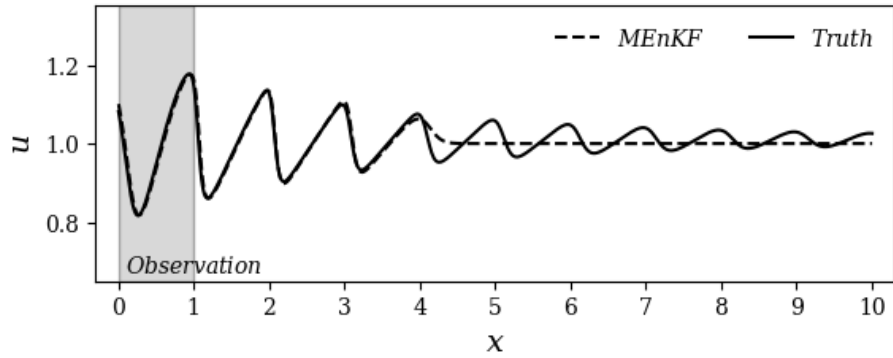


(c)  $t = 10.60$

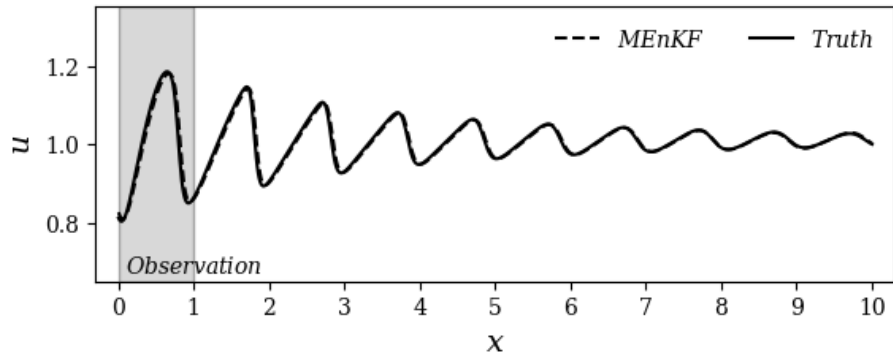
Figure 5: Estimations obtained by MEnKF for  $r_C = 1$  at  $t = 1$  (a),  $t = 3.88$  (b) and  $t = 10.60$  (c). Times are given in  $t_c$  units. The grey shaded area corresponds to the observation window.



(a)  $t = 1$

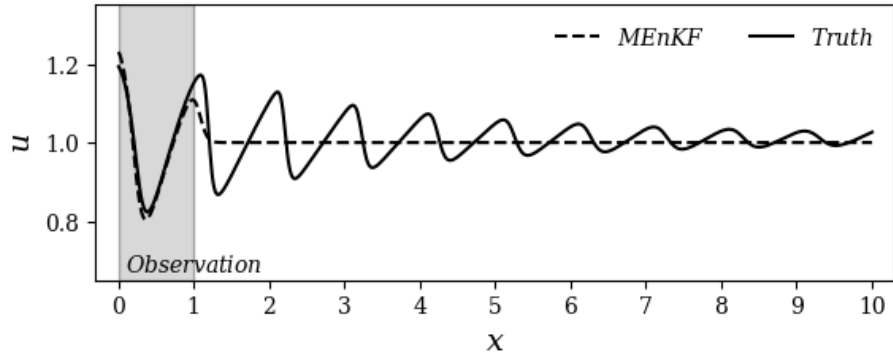


(b)  $t = 3.88$

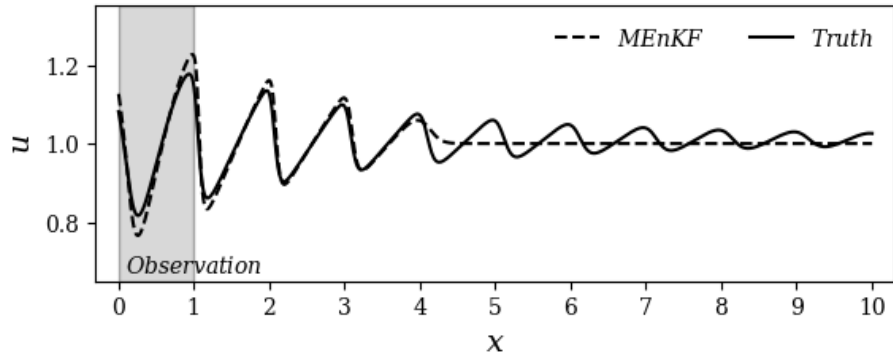


(c)  $t = 10.60$

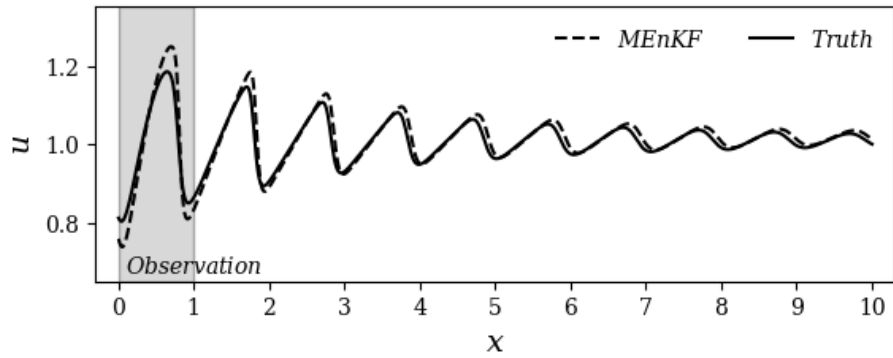
Figure 6: Estimations obtained by MEnKF for  $r_C = 8$  at  $t = 1$  (a),  $t = 3.88$  (b) and  $t = 10.60$  (c). Times are given in  $t_c$  units. The grey shaded area corresponds to the observation window.



(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). Times are given in  $t_c$  units. The grey shaded area corresponds to the observation window.

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

$$\text{RMSE}(k) = \sqrt{\frac{\int_x \left[ (u_k^{\text{F}})^{\text{a}}(x) - (u_k^{\text{F}})^{\text{True}}(x) \right]^2 dx}{\int_x \left[ (u_k^{\text{F}})^{\text{True}}(x) \right]^2 dx}} \quad (26)$$

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  $t < 10t_c$ . 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 behaviour.

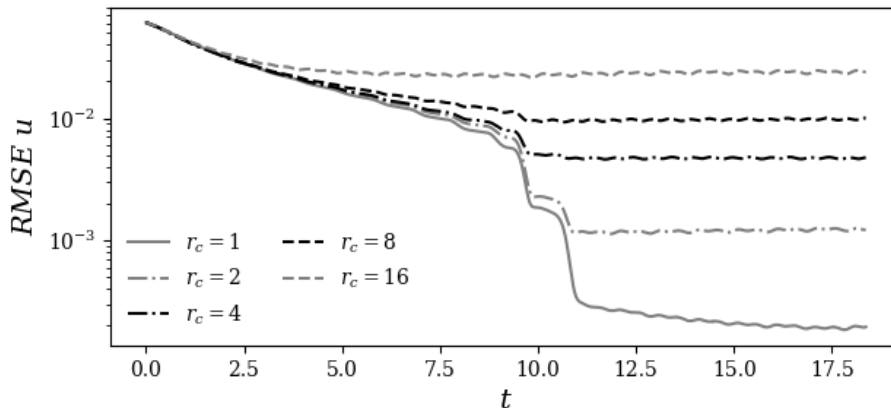


Figure 8: Time evolution of the RMS error of  $u$  for  $r_C = 1, 2, 4, 8, 16$ .

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 each  $r_C$  value. In Fig. 9, results are shown and normalized over the case  $r_C = 1$ . 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  $r_C = 8, 16$ , 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.

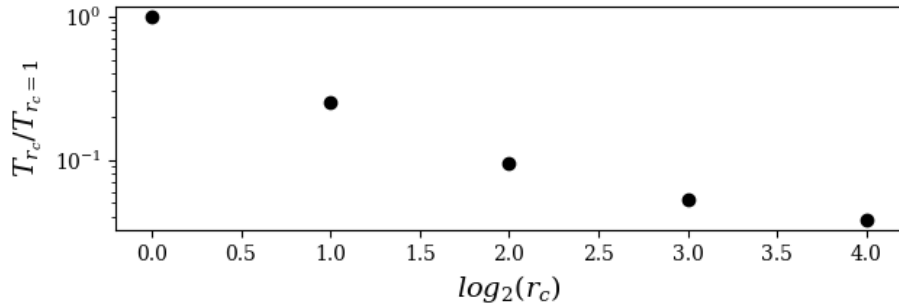


Figure 9: Computational time required to perform a full assimilation cycle for  $r_C = 1, 2, 4, 8, 16$ . Results are normalized over the computational time required for  $r_C = 1$ .

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 discrepancy 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. 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:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} = 0 \quad (27)$$

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

$$\frac{\partial(\rho E)}{\partial t} + \frac{\partial((\rho E)u)}{\partial x} + \frac{\partial(pu)}{\partial x} = 0 \quad (29)$$

where  $\rho$  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 using the finite difference method. Second-order centered schemes are used for the derivatives in space and a first-order scheme for the time integration to obtain the general form of discretized representation as given by (23). A centered sixth-order numerical filter is included to damp numerical spurious oscillations [41]. The numerical algorithm is used to analyse the acoustic propagation of a sinusoidal wave with a time-varying amplitude. To do so, a Dirichlet time-varying condition is imposed at the inlet for the velocity:

$$u(x = 0, t) = u_0 (1 + \theta(t) \sin(\omega t)) \quad (30)$$

$u_0$  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  $\theta$  is set at sufficiently low value 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 of  $u_0 + a$ . The characteristic velocity and length of the test case are  $u_c = u_0 + a$  and  $\lambda$ . Similarly to the analysis in Sec. 4,  $\lambda$  represents the wavelength of the signal imposed at the inlet. The characteristic time of the system is defined as  $t_c = \lambda/u_c$ .

The sinusoidal behaviour of the velocity at the inlet is characterized by

a constant frequency  $f_c = 1/t_c$  with  $\omega = 2\pi f_c$ . However, the time-varying amplitude of the sinusoidal wave is driven by the parameter  $\theta(t) = \theta_0(1 + \sin(\omega_\theta t))$ , where  $\theta_0$ ,  $\omega_\theta = 2\pi f_c/b$  and  $b$  are constants. The density at the inlet is fixed  $\rho(x = 0, t) = \rho_0$  as well as the total energy per unit mass  $E(x = 0, t) = E_0$ , which is calculated as  $E_0 = e + 0.5u_0^2$ . The outlet boundary condition is extrapolated from the nearest points to the outlet. 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. The fluid is an ideal gas with  $\gamma = 1.4$ ,  $\rho_0 = 1.17$  and  $T_0 = 300$  in S.I. units. Note that from now on, the units of measure are the characteristic magnitudes.

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 a characteristic length  $\lambda$  for a total of  $N_x = 800$  elements in the domain. Finally, the normalized value of  $\Delta t$  is set to  $\Delta t = 0.0006$ .

A preliminary simulation is performed using an explicit Euler scheme for time derivative. The parameters which describe the amplitude of the velocity oscillation imposed at the inlet are set to  $\theta_0 = 0.015$  (true value) and  $b = 10$ . A flow visualization of the wave patterns is shown in Fig. 10 at  $t = 17.3$ . The fully developed state obtained for  $t = 10$  is used to initialize a new simulation from  $t = 0$ . This simulation is run for a total time of  $T_{\text{ref}} = 110$ . As in Sec. 4, this simulation is used as true state and it is sampled every 30 time steps to obtain observations in the space region  $x \in [0, 1]$ . The sampled data, which is used as observation, is artificially perturbed using a Gaussian noise of variance  $R = 0.09$ .

The DA procedure is identical to the analysis presented in Sec. 4. The assimilation is composed by the base simulation, which is run on the fine grid, and  $N_e = 100$  simulations on a coarse-grid level. In this section, only one coarsening ratio is considered ( $r_c = 4$ ). The estimator is used to dynamically track the value of the parameter  $\theta$ , which evolves in time. No a priori knowledge about the behaviour of the parameter is used. This time optimization is much



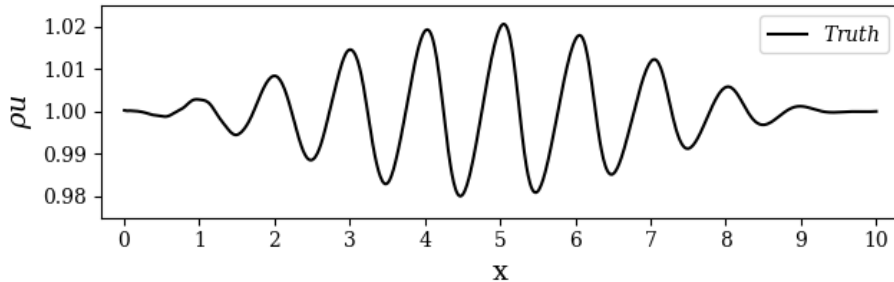


Figure 10: Field  $\rho u$  of the one-dimensional inviscid Euler equations at  $t = 17.3$  for  $\theta_0 = 0.015$  and  $\omega_\theta = \omega/10$  (true state).

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,  $\theta$  is initially assumed to be a random Gaussian phenomenon  $\theta \sim \mathcal{N}(0, Q_\theta)$ . The initial value of the covariance is  $Q_\theta(t = 0) = 6.4 \times 10^{-5}$ . As shown in the previous case, 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  $\theta$  for the ensemble members is artificially increased, as in the classical Dual EnKF algorithm. As described in Appendix A.4, we add to the estimated parameter of each member of the ensemble a Gaussian noise of diagonal covariance  $\Sigma_k^\theta = 10^{-10}$ . Extensive numerical tests have been performed and the results show the importance of  $\Sigma_k^\theta$ . The value chosen for  $\Sigma_k^\theta$  is arbitrary and works only for this test-case. In general, the estimation of  $\Sigma_k^\theta$  is challenging, since a priori information about the dynamic behaviour of  $\theta$  is required. Nonetheless,  $\theta$  can be determined heuristically: the artificial variance added to the parameters,  $\Sigma_k^\theta$ , should be of the same order of magnitude as the true rate of change of  $\theta$  in a single assimilation cycle. When this parameter is underpredicted, the variance in  $\theta$  is too small for the Dual EnKF to perform, while showing an oscillatory behaviour in the parameter estimation if  $\Sigma_k^\theta = 10^{-10}$  is overpredicted.

This study is performed for three different values of  $f_a$ , the normalized period between successive assimilations defined as  $f_a = t_c/t_a$ . This normalized analysis frequency corresponds to the number of analysis phases per characteristic time of simulation  $t_c$ . Three different values of  $f_a$  are investigated:  $f_a = 2, 10, 55$ .

The estimator is run for a total simulation time  $T_{\text{ref}} = 110$ , 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  $\theta$  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  $\theta$  are reported in Fig. 11. The time evolution of  $\theta$  are 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 were not deserved. 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  $f_a = 10, 55$ . In addition,  $\theta$  tends to be underestimated (around 10%) 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  $\rho u$ , 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  $\theta$ . However, accurate results are observed for  $f_a = 10$  and  $55$ . Even though the

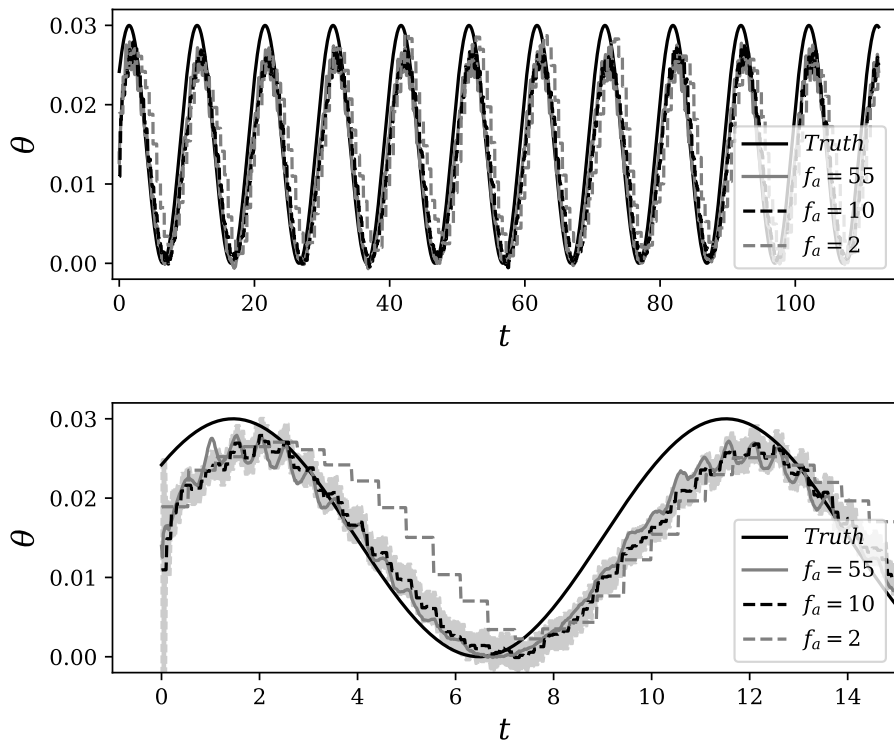


Figure 11: Time estimation of the parameter  $\theta$  driving the amplitude of the sinusoidal acoustic wave. Results are shown for  $f_a = 2, 10, 55$  and compared to the true value of  $\theta$ . In the top image,  $\theta$  is shown in the whole assimilation window  $t \in [0, 110]$ . In the bottom image, a zoom for  $t \in [0, 15]$  is shown. The shaded area represents the 95% credible interval for the case  $f_a = 10$ .

value of the parameter  $\theta$  is not exact, the total 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.

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

$$\text{RMSE}(k) = \sqrt{\frac{\int_x [((\rho u)_k^{\text{F}})^{\text{a}}(x) - ((\rho u)_k^{\text{F}})^{\text{True}}(x)]^2 dx}{\int_x [((\rho u)_k^{\text{F}})^{\text{True}}(x)]^2 dx}} \quad (31)$$

is shown in Fig. 15. The error achieves a quasi-constant asymptotic behaviour after a complete propagation of the signal in the physical domain ( $t \approx 10t_c$ ). 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  $f_a = 55$  indicates that, once a minimal threshold in the assimilation period  $t_a$  is reached, the prediction of the estimator exhibits a robust convergence. Similar results were previously observed in three-dimensional simulations [18].

## 6. Spatially evolving compressible mixing layer

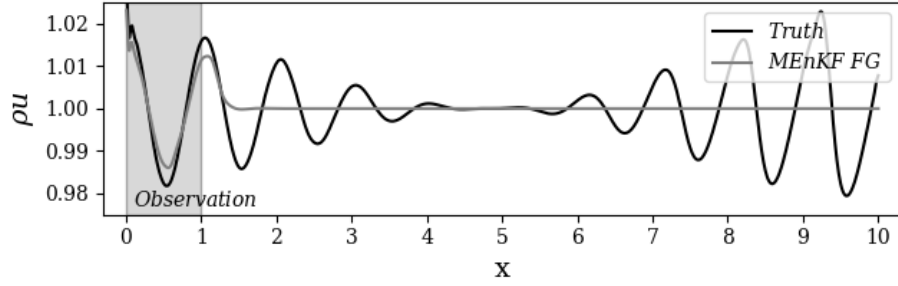
In this section, we consider the compressible Navier-Stokes equations in a two-dimensional physical domain:

$$\frac{\partial \rho}{\partial t} + \text{div}(\rho \mathbf{u}) = 0 \quad (32)$$

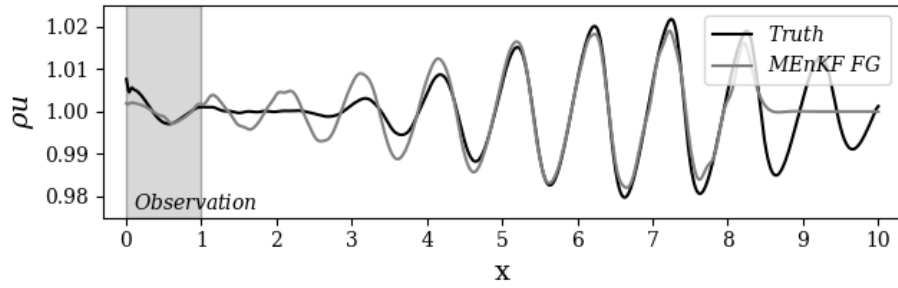
$$\frac{\partial (\rho \mathbf{u})}{\partial t} + \text{div}(\rho \mathbf{u} \otimes \mathbf{u}) = -\mathbf{grad} p + \mathbf{div} \bar{\tau} \quad (33)$$

$$\frac{\partial (\rho E)}{\partial t} + \text{div}(\rho E \mathbf{u}) = -\text{div}(p \mathbf{u}) + \text{div}(\bar{\tau} \mathbf{u}) + \text{div}(\lambda(T) \mathbf{grad} T) \quad (34)$$

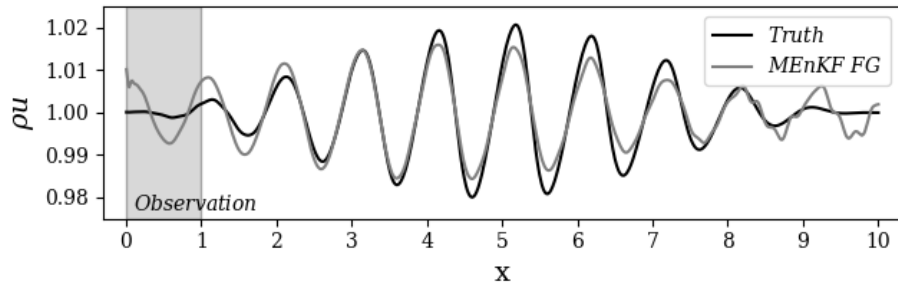
where  $\rho$  is the density,  $\mathbf{u}$  is the velocity (components  $u$  in the streamwise direction and  $v$  in the normal direction),  $p$  is the pressure,  $E$  is the total energy per unit of mass,  $\bar{\tau}$  is the tensor of the viscous constraints and  $T$  is the temperature. To obtain the representation given by (23), the equations are discretized using the finite difference method. Second-order centered schemes are used for



(a)  $t = 1.23$

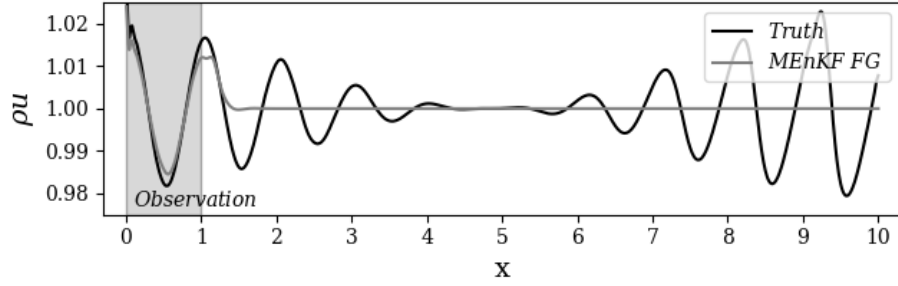


(b)  $t = 8.32$

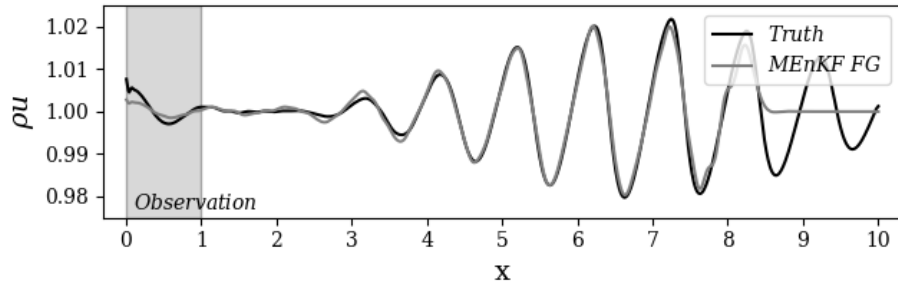


(c)  $t = 16.30$

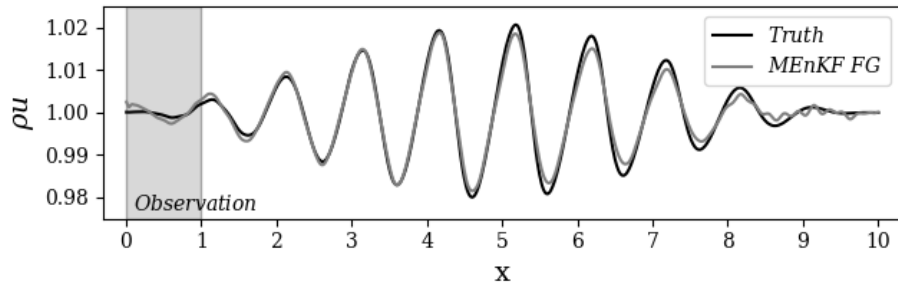
Figure 12: Estimations by MEnKF of the momentum  $\rho u$  normalized by  $\rho_0 u_0$  for  $f_a = 2$  at  $t = 1.23$  (a),  $t = 8.32$  (b) and  $t = 16.30$  (c). Times are given in  $t_c$  units. The grey shaded area corresponds to the observation window.



(a)  $t = 1.23$

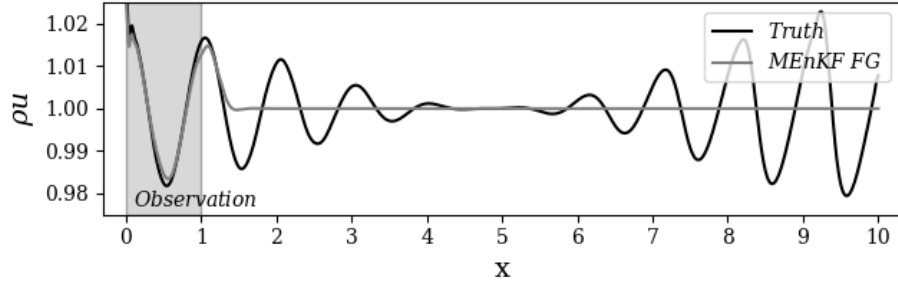


(b)  $t = 8.32$

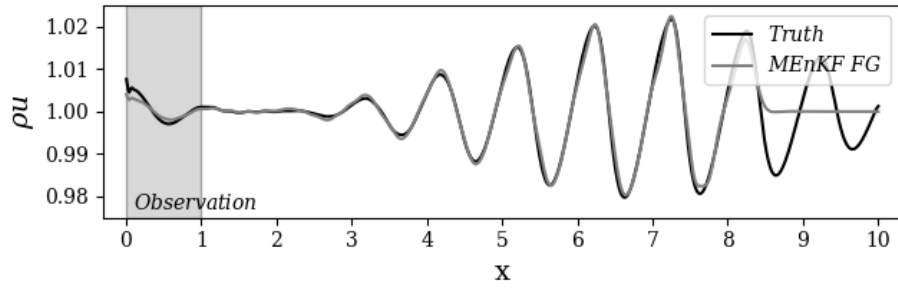


(c)  $t = 16.30$

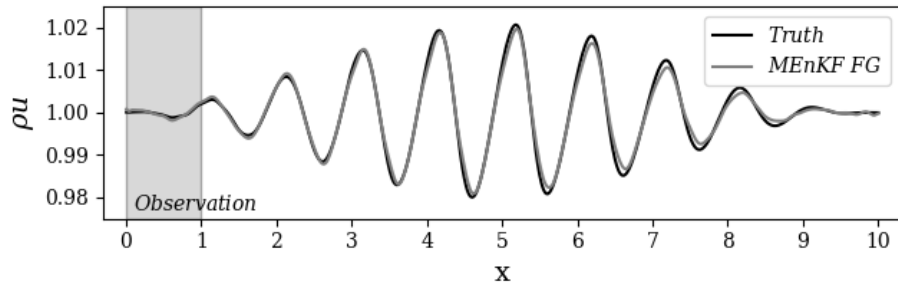
Figure 13: Estimations by MEnKF of the momentum  $\rho u$  normalized by  $\rho_0 u_0$  for  $f_a = 10$  at  $t = 1.23$  (a),  $t = 8.32$  (b) and  $t = 16.30$  (c). Times are given in  $t_c$  units. The grey shaded area corresponds to the observation window.



(a)  $t = 1.23$



(b)  $t = 8.32$



(c)  $t = 16.30$

Figure 14: Estimations by MEnKF of the momentum  $\rho u$  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.

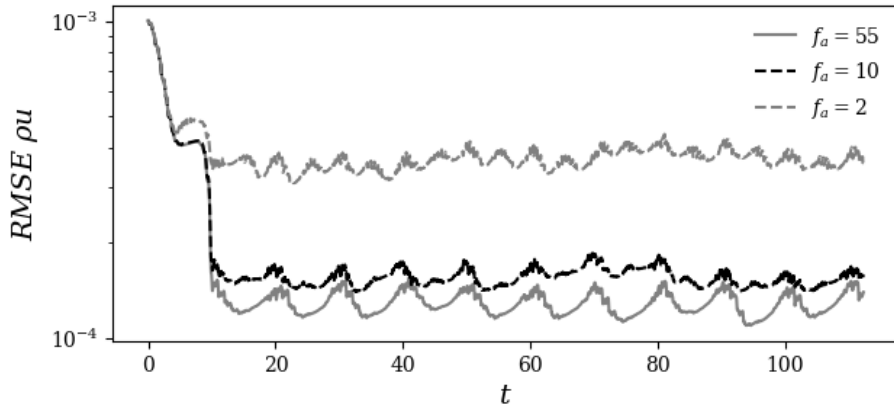


Figure 15: Time evolution of the RMS error of  $\rho u$  for  $f_a = 2, 10, 55$ .

the derivatives in space and a first-order scheme for the time integration. A centered sixth-order numerical filter is included to damp numerical spurious oscillations [41].

As flow configuration, we consider the two-dimensional spatially evolving mixing layer at  $Re = 100$ . For this value of Reynolds number, the flow exhibits unsteady features. It can be shown [42, 43] 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  $14\Lambda \times 6\Lambda$  in the streamwise direction  $x$  and normal direction  $y$ , respectively. The characteristic length  $\Lambda$ , which is taken as reference length from now on, is given by  $\Lambda = A\delta_0$ , where  $\delta_0$  is the initial vorticity thickness imposed at the inlet. The value of the parameter  $A$  is determined from the most unstable wavelength determined by Linear Stability Theory (LST). At  $Re = 100$ , we have  $A = 14.132$ . The mesh resolution in the horizontal direction is constant for  $x \leq 10$ . 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 centerline with a constant coarsening ratio of 1.01.



The Reynolds number of the flow is calculated as  $\text{Re} = (U_1 - U_2)\delta_0/\nu = 100$  with asymptotic velocities set to  $U_1 = 173.61$  and  $U_2 = 104.17$ . These values correspond to a Mach number  $\text{Ma} = 0.5$  and  $\text{Ma} = 0.3$ , 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. All these quantities are expressed in S.I. units. The inlet boundary condition is taken from [42]. For the velocity field, one has:

$$U_{\text{in}} = \frac{U_1 + U_2}{2} + \frac{U_1 - U_2}{2} \tanh\left(\frac{2y}{\delta_0}\right) + U_{\text{pert}} \quad -3 < y < 3 \quad (35)$$

$$V_{\text{in}} = 0 \quad (36)$$

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

$$U_{\text{pert}} = \sum_{i=1}^{N_{\text{in}}} \epsilon_i \frac{U_1 + U_2}{2} [f_i(y) \sin(\omega_i t)], \quad (37)$$

where  $N_{\text{in}}$  is the total number of perturbation modes and  $\epsilon_i$  quantifies the magnitude of each mode. The function  $f_i(y) = \cos(4n_i \frac{y}{\delta_0})h(y)$  controls the shape of the perturbation of the inlet velocity profile in the normal direction. The role of  $h(y) = 1 - \tanh^2(\frac{2y}{\delta_0})$  is to damp the perturbation component moving away from the centerline. The wavelength parameters  $n_i$  are tuned according to the LST results. In the following, we consider  $N_{\text{in}} = 1$  *i.e.* the inlet perturbation consists of a single mode. The inlet density is set to be constant so that  $\rho_{\text{in}} = 1.177$ , as well as the temperature  $T = 300$  in S.I. units.

In this section, the parameters  $\theta$  of the model correspond to  $\epsilon_i$ , the variable governing the amplitude of the perturbation. Two different scenarios are studied. In the first one, the parameter  $\epsilon_i$  is constant. In the second one, it is a time-dependent coefficient. In the first case, the reference simulation is done

using a constant single mode for the inlet perturbation  $\epsilon_1 = \epsilon = \text{cte}$ . In the second case,  $\epsilon_1$  varies in time following a sinusoidal form:  $\epsilon_1 = \epsilon(1 + \sin(\omega_\epsilon t))$ . The values of the numerical parameters characterizing the perturbation are  $\epsilon = 0.15$ ,  $n_1 = 0.4\pi$ ,  $\omega_1 = 1/t_c$  and  $\omega_\epsilon = 0.62\omega_1$ , where  $t_c = 2\Lambda/(U_1 + U_2)$  is the average advection time. A flow visualization of  $\rho v$  at  $t = 10$  is shown in Fig. 16 for the two cases. In both cases, one can clearly observe the emergence of coherent structures. When a constant value of  $\epsilon_1$  is used, the coherent structures are virtually aligned. When  $\epsilon_1$  is time-varying, one can observe the emergence of more complex pairing patterns.

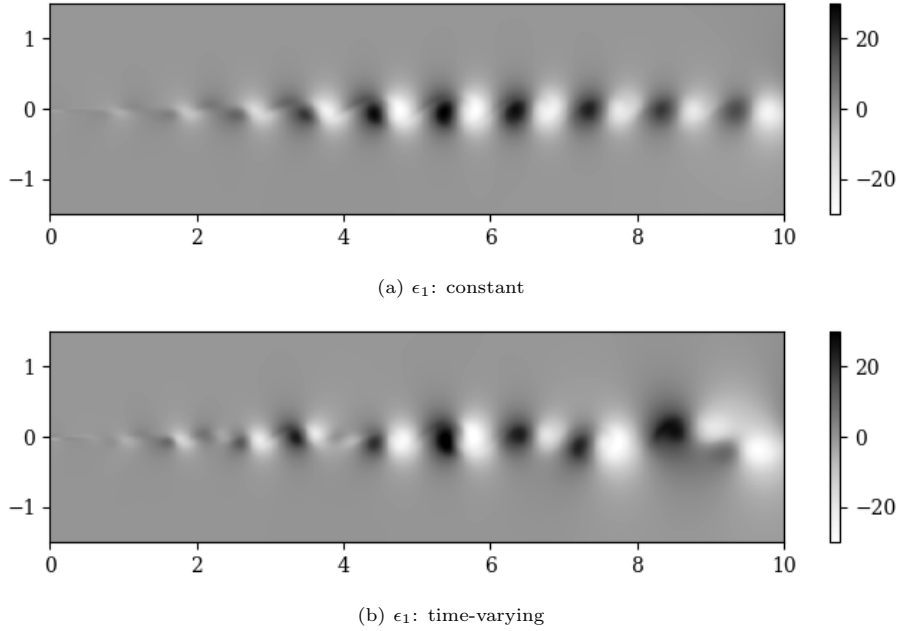


Figure 16: Visualization of the normal momentum  $\rho v$  (S.I. units) for the 2D compressible Navier-Stokes equation. Reference simulation at  $t = 10$  for a constant value of  $\epsilon_1$  (a), and a time-varying value (b).

The DA procedure is performed using the following elements:

- The model is the discretized version of the system given by (32) - (34). The features of the fine mesh level were previously introduced. For the coarse grid level, a homogeneous coarsening ratio  $r_C = 4$  is employed. The

simulation is initialized in the physical domain using a simple hyperbolic tangent profile with no perturbation (see (35)). We consider that no prior information is available on the time evolution of  $\epsilon_1$ . At  $t = 0$ , this coefficient is fixed to be a random Gaussian phenomenon  $\epsilon_1 \sim \mathcal{N}(0, Q_a)$  where the initial value of  $Q_a(t = 0) = 0.0625$ . Similarly to the cases analyzed 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  $N_e = 100$ .

- The observation is sampled from the reference simulations shown in Fig. 16, which are run for a total simulation time of  $T_{\text{ref}} = 40$  in  $t_c$  units. A fully developed state obtained from a prior simulation at  $t = 10$  is used to initialize the simulations at  $t = 0$ . Data are 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 update frequency is chosen sufficiently high to assure a good estimation. The observations are made from the instantaneous fields  $\rho u$  and  $\rho v$ . The data used as observation are artificially perturbed using a Gaussian noise of variance  $R = 1$ .

The estimation algorithm is run over a time window equal to  $T_{\text{DA}} = 40$  which encompass roughly 3200 DA analysis phases. This value corresponds to four complete advections in the whole physical domain. At the end of each analysis, the mean value and the variance of the coefficient  $\epsilon_1$  are updated following the Dual EnKF technique [38], similarly to what was done in Sec. 4.

First, results dealing with the constant value of  $\epsilon_1$  are discussed. The parameter estimation for this case is almost exact, as observed in Fig. 17. The convergence towards the exact value of  $\epsilon_1$  is obtained in roughly 0.5 units of time, after a first transient where a slight over-prediction is observed. When combined with the correction obtained via the Kalman gain, the accuracy of the state estimation is excellent, as shown in Fig. 18.

The value of  $\rho v$  that is predicted at the centerline of the mixing layer is shown in Fig. 18. At  $t = 1$ , the flow is strongly affected by the KF corrections.

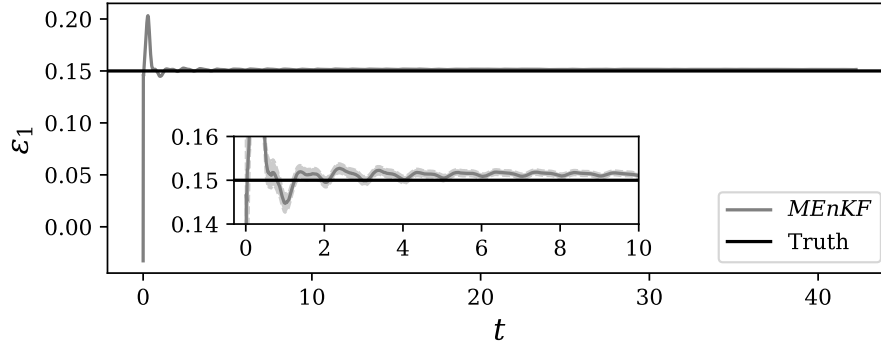
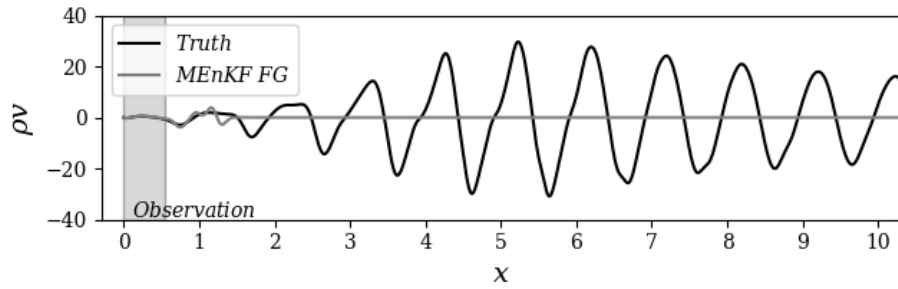


Figure 17: Time evolution of the inferred values of  $\epsilon_1$ . The truth corresponds to  $\epsilon_1 = \text{cte} = 0.15$ . In the zoomed region, the shaded area represents the 95% credible interval for the estimated parameter.

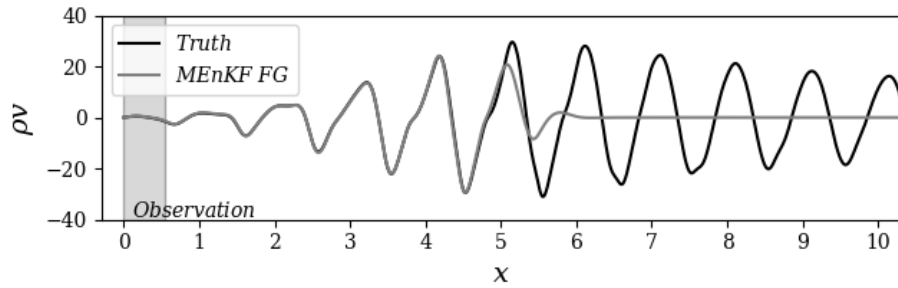
After five characteristic times, the flow is perfectly matching the true state upstream. Finally, after the initial transient is dissipated, we observe that the state estimation almost perfectly matches the true state in the whole physical domain.

The results dealing with the time-varying parameter  $\epsilon_1$  are now discussed. The time evolution of the estimated value of  $\epsilon_1$  is reported in Fig. 19. 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. The presence of this delay has probably the same reasons as 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.

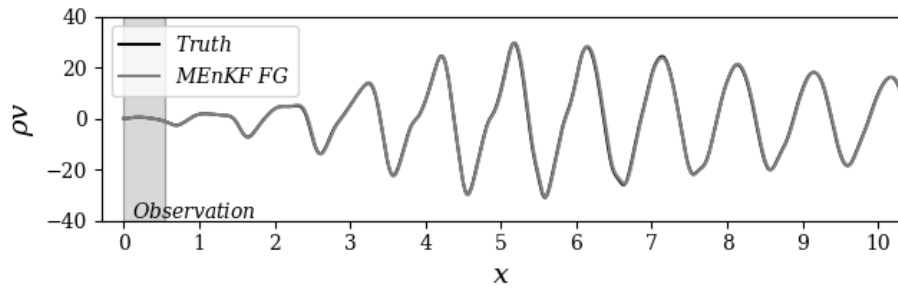
The results obtained for the prediction of the normal momentum  $\rho v$  are shown in Fig. 20. One can see that the combination of parameter and state estimations produces an accurate prediction of the flow. Minor differences are observed with the true state. In particular, the momentum  $\rho v$  does 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



(a)  $t = 1$

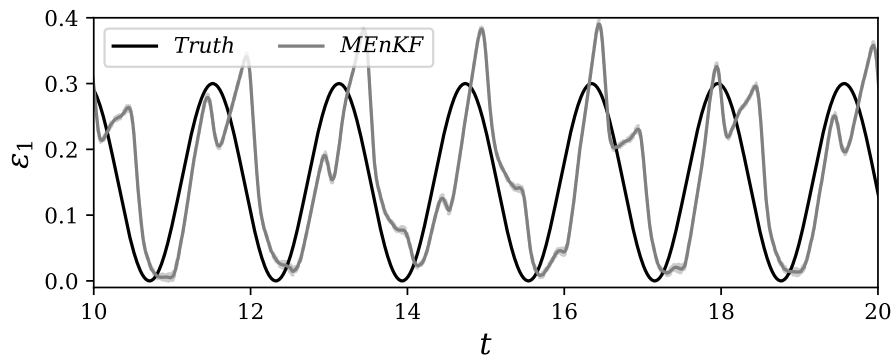


(b)  $t = 5$

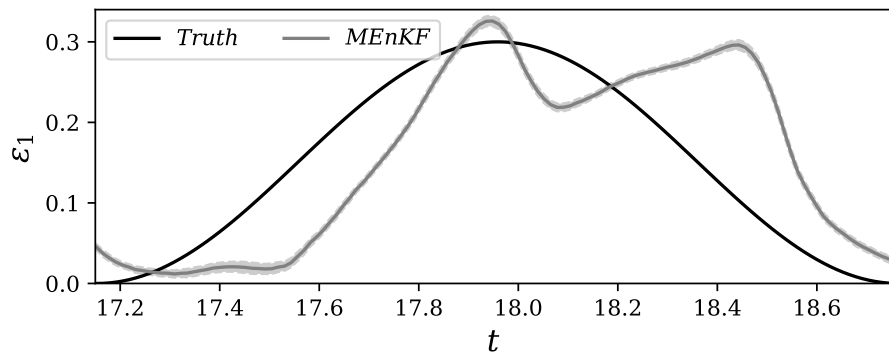


(c)  $t = 30$

Figure 18: Estimations obtained by MEnKF of the momentum  $\rho v$  (S.I. units) at the centerline  $y = 0$  of the mixing layer. Results at  $t = 1$  (a),  $t = 5$  (b) and  $t = 30$  (c) for the case  $\epsilon_1 = \text{cte} = 0.15$ .



(a) Estimation history



(b) Zoom

Figure 19: Time evolution of the inferred values of  $\epsilon_1$  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.

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. 21 that the prediction is sensibly deteriorated.

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

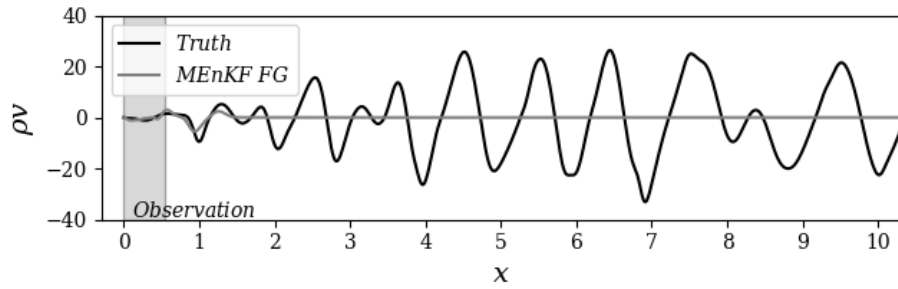
$$\text{RMSE}(k) = \sqrt{\frac{\int_x [((\rho v)_k^{\text{F}})^{\text{a}}(x) - ((\rho v)_k^{\text{F}})^{\text{True}}(x)]^2 dx}{\int_x [((\rho v)_k^{\text{F}})^{\text{True}}(x)]^2 dx}} \quad (38)$$

The results, which are shown in Fig. 22, 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.

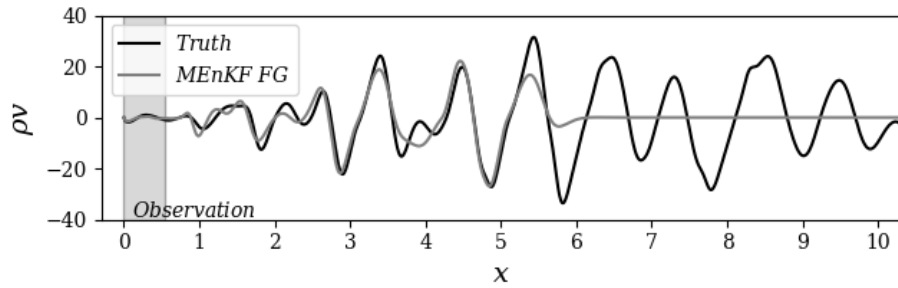
At last, an analysis of the conservativity of the algorithm is performed. As previously discussed, the state estimation obtained via EnKF does not necessarily comply with the dynamical equations of the model. This drawback can be responsible for discontinuities in the physical field, which can significantly affect the accuracy and stability of the global algorithm. The analysis is performed considering an indicator  $\Gamma_k^{\text{F}}$  which measures the conservation of the transversal momentum equation (33) in discretized form:

$$\frac{((\rho v)_k^{\text{F}})^{\text{a}} - ((\rho v)_{k-1}^{\text{F}})^{\text{a}}}{\Delta t} - \mathcal{F}_{\rho v}(\rho_k, \mathbf{u}_k, p_k, \bar{\tau}_k) = \Gamma_k^{\text{F}}, \quad (39)$$

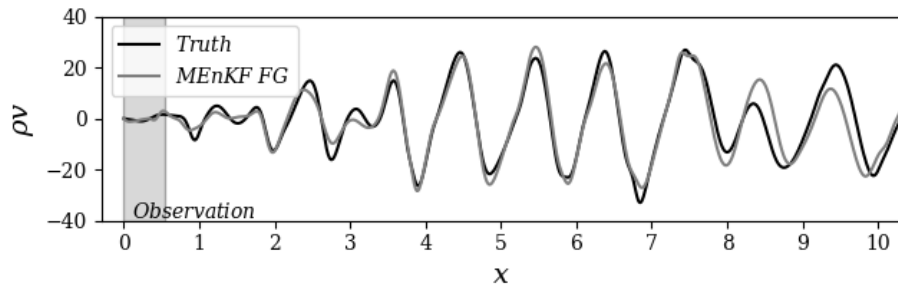
where  $\mathcal{F}_{\rho v}$  represents the spatial discretization terms in the transversal momentum equation. In the forecast step performed via the model,  $\Gamma_k^{\text{F}} = 0$  down to a convergence rate  $\delta$  which is prescribed. However, the value of  $\Gamma_k^{\text{F}}$ , at the end of a time step where a forecast-analysis is performed, is strictly connected with the computational strategy employed. Here, three cases are considered for  $\epsilon_1 = \text{cte} = 0.15$ :



(a)  $t = 1$



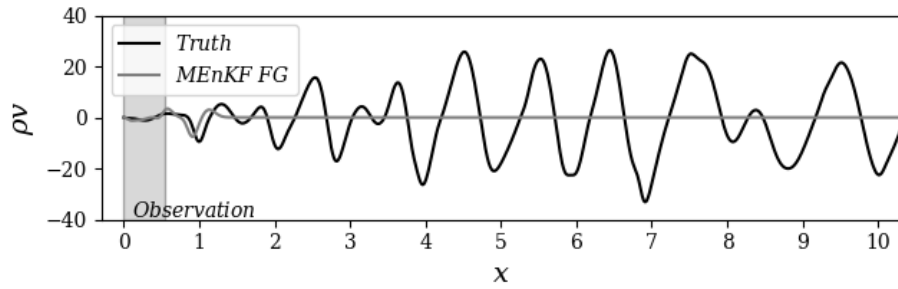
(b)  $t = 5$



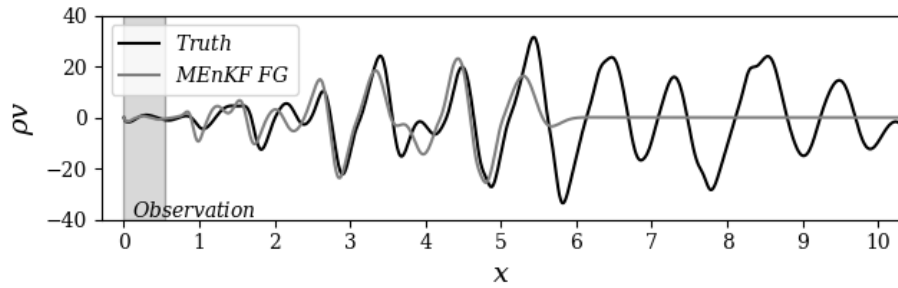
(c)  $t = 30$

Figure 20: Estimations obtained by MEnKF of the momentum  $\rho v$  (S.I. units) at the centerline  $y = 0$  of the mixing layer. Results at  $t = 1$  (a),  $t = 5$  (b) and  $t = 30$  (c) for the time-varying  $\epsilon_1$ .

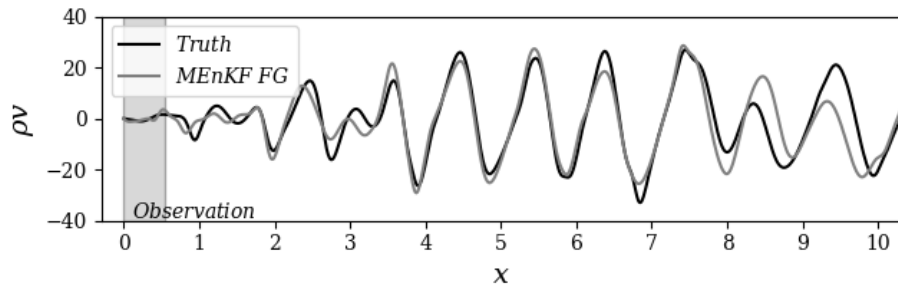




(a)  $t = 1$



(b)  $t = 5$



(c)  $t = 30$

Figure 21: Estimations obtained by MEnKF of the momentum  $\rho v$  (S.I. units) at the centerline  $y = 0$  of the mixing layer. Here, MEnKF is only used to provide the estimation of  $\epsilon_1$ . Results at  $t = 1$  (a),  $t = 5$  (b) and  $t = 30$  (c) for the time-varying  $\epsilon_1$ .

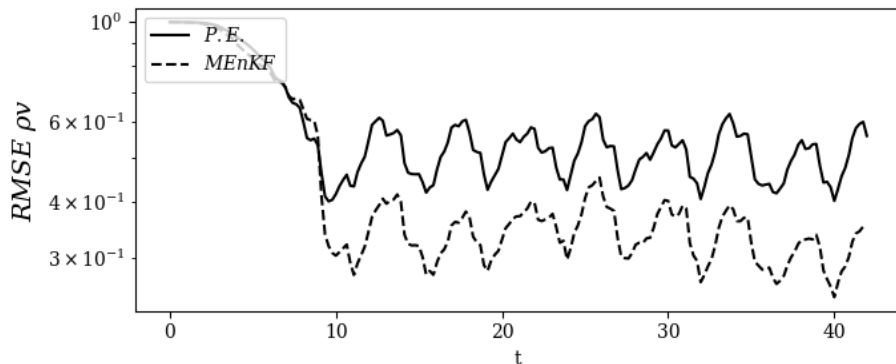
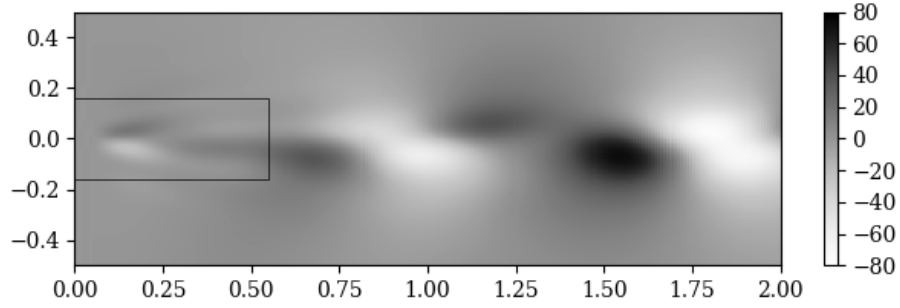


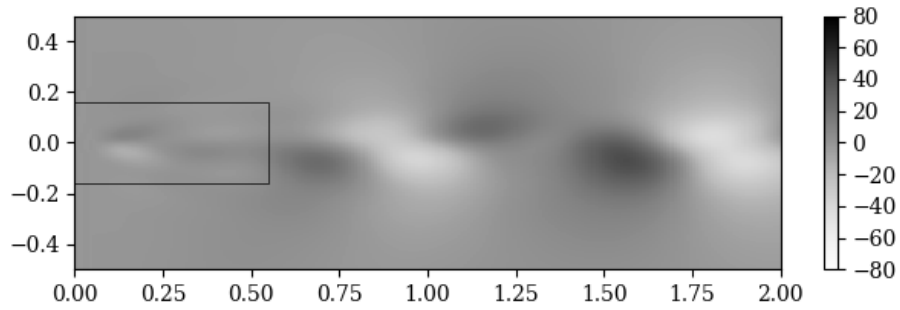
Figure 22: Time evolution of the RMS error of  $\rho v$  for the case of a time-varying inlet parameter  $\epsilon_1$ . The symbol P.E. corresponds to the case where MEnKF is only used for the estimation of  $\epsilon_1$ . The notation MEnKF corresponds to the standard version of the algorithm, including parameter estimation and physical state correction via Kalman gain.

- 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.
- A standard MEnKF algorithm, as described in Sec. 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.

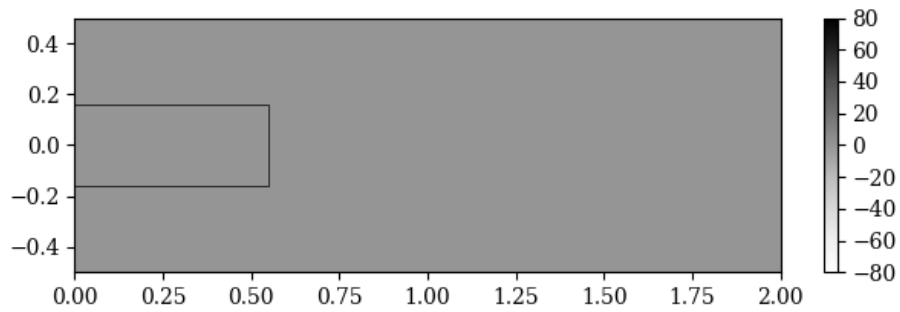
The results are shown in Fig. 23 after the first forecast / analysis step. For clarity, we introduce a normalized criterion  $(\Gamma_k^F)^* = \frac{\Gamma_k^F}{C}$  where  $C$  is defined as  $\max_k \left| \frac{((\rho v)_k^F)^a - ((\rho v)_{k-1}^F)^a}{\Delta t} \right|$ . As expected,  $\Gamma_k^F = 0$  everywhere when MEnKF is only used for the parameter estimation. Here, 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 convergence error which is negligible). On the other hand, results in Fig. 23 (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



(a) Standard Dual EnKF on the coarse mesh.



(b) Standard MEnKF.



(c) MEnKF used only for parameter estimation.

Figure 23: Analysis of the conservativity of the dynamical model via the normalized quantity  $(\Gamma_k^F)^*$ . 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.

with the dynamical equations. Finally, results for the MEnKF are shown in Fig. 23 (b). The evolution of  $(\Gamma_k^F)^*$  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

A sequential estimator based on a Kalman filter approach for Data Assimilation of fluid flows is presented in this research work. This estimator exploits iterative features which are employed in several CFD codes for the resolution of complex applications in fluid mechanics. 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 then employed as ensemble members to determine i) 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 and ii) an optimization of the free parameters driving the simulation. One of the main advantages of the model is that, owing to the iterative procedure for the calculation of the flow

variables, the final solution is regularized.

The method, which is referred to as Multigrid Ensemble Kalman Filter (MEnKF), is assessed via the analysis of one-dimensional and two-dimensional test cases and using different dynamic equations. First, the one-dimensional Burgers equation for  $Re = 200$  is analyzed. 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 EnKF) indicate that the State Estimation and the parametric optimization of the inlet 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 for  $Re = 100$  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 over this method will target improvement of the projector operators between coarse and fine grid level, in particular for the state matrix  $\Psi$ . Preliminary tests have shown that, in the case of non-linear models, information from the main, refined simulation can be projected on the coarse mesh

level to improve the predictive capabilities of the ensemble members. This implies a more accurate prediction of the state estimation and of the parameter optimization on the coarse level, from which the main refined simulation will benefit. This loop has the potential to improve even more the performance of the MEnKF model, and strategies for efficient application are currently under investigation. Moreover, we plan to test the MEnKF algorithm to 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. Prof. Heng Xiao is 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.24.

### *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  $\mathbf{Y}_k^f$ .

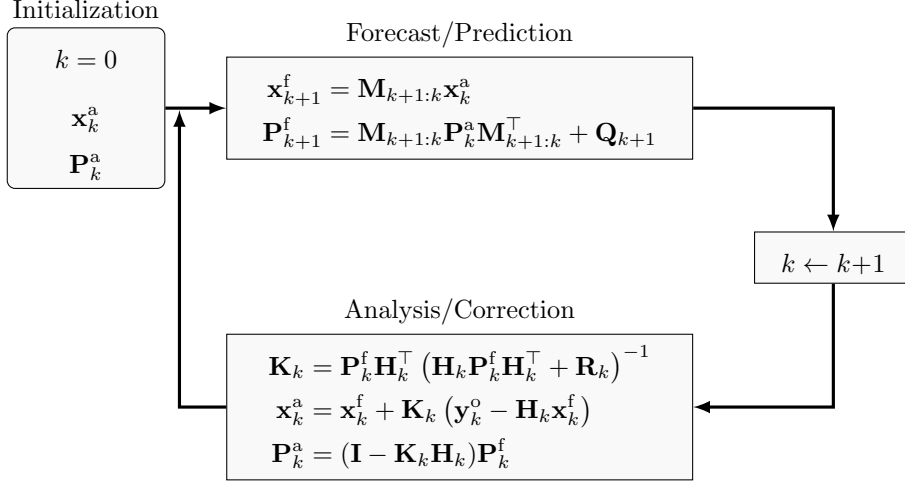


Figure A.24: Kalman Filter algorithm. The initialization is made with the analysed state.

#### 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 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 of all, 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

---

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

---

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

**Output:**  $\{\mathbf{x}_k^{a,(i)}\}$ ;  $k = 0, \dots, K$ ;  $i = 1, \dots, N_e$

**begin**

- 1: Initialize the ensemble of forecasts  $\{\mathbf{x}_0^{f,(i)}\}$ ;  $i = 1, \dots, N_e$
  - for**  $k = 0, \dots, K$  **do**
  - 2:   Draw a statistically consistent observation set;  $i = 1, \dots, N_e$   
 $\mathbf{y}_k^{o,(i)} = \mathbf{y}_k^o + \epsilon_k^{o,(i)}$  with  $\epsilon_k^{o,(i)} \sim \mathcal{N}(0, \mathbf{R}_k)$
  - 3:   Compute the model counterparts of the observation set;  
 $i = 1, \dots, N_e$   
 $\mathbf{y}_k^{f,(i)} = \mathcal{H}_k(\mathbf{x}_k^{f,(i)})$
  - 4:   Compute the ensemble means  
 $\overline{\mathbf{x}}_k^f = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{x}_k^{f,(i)}$ ;  $\overline{\mathbf{y}}_k^f = \frac{1}{N_e} \sum_{i=1}^{N_e} \mathbf{y}_k^{f,(i)}$ ;  $\overline{\epsilon}_k^o = \frac{1}{N_e} \sum_{i=1}^{N_e} \epsilon_k^{o,(i)}$
  - 5:   Compute the normalized anomalies;  $i = 1, \dots, N_e$   

$$[\mathbf{X}_k^f]_{:,i} = \frac{\mathbf{x}_k^{f,(i)} - \overline{\mathbf{x}}_k^f}{\sqrt{N_e - 1}}; [\mathbf{Y}_k^f]_{:,i} = \frac{\mathbf{y}_k^{f,(i)} - \overline{\mathbf{y}}_k^f}{\sqrt{N_e - 1}}; [\mathbf{E}_k^o]_{:,i} = \frac{\epsilon_k^{o,(i)} - \overline{\epsilon}_k^o}{\sqrt{N_e - 1}}$$
  - 6:   Compute the Kalman gain  

$$\mathbf{K}_k^e = \mathbf{X}_k^f (\mathbf{Y}_k^f)^\top \left( \mathbf{Y}_k^f (\mathbf{Y}_k^f)^\top + \mathbf{E}_k^o (\mathbf{E}_k^o)^\top \right)^{-1}$$
  - 7:   Update the ensemble;  $i = 1, \dots, N_e$   
 $\mathbf{x}_k^{a,(i)} = \mathbf{x}_k^{f,(i)} + \mathbf{K}_k^e \left( \mathbf{y}_k^{o,(i)} - \mathbf{y}_k^{f,(i)} \right)$
  - 8:   Compute the ensemble forecast;  $i = 1, \dots, N_e$   
 $\mathbf{x}_{k+1}^{f,(i)} = \mathcal{M}_{k+1:k}(\mathbf{x}_k^{a,(i)})$
-



---

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

---

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

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

**begin**

1: Initialize  $\{\theta_0^{a,(i)}\}$  and  $\{\mathbf{x}_0^{a,(i)}\}$   
**for**  $k = 1, \dots, K$  **do**

2: Observation ensemble:  

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

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

3: Parameter forecast:  

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

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

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

4: Compute the normalized anomalies  

$$[\boldsymbol{\Theta}_k^f]_{:,i} = \frac{\theta_k^{f,(i)} - \overline{\theta}_k^f}{\sqrt{N_e - 1}}; [\mathbf{Y}_k^f]_{:,i} = \frac{\mathbf{y}_k^{f,(i)} - \overline{\mathbf{y}}_k^f}{\sqrt{N_e - 1}}; [\mathbf{E}_k^o]_{:,i} = \frac{\epsilon_k^{o,(i)} - \overline{\epsilon}_k^o}{\sqrt{N_e - 1}}$$

5: Parameter update:  

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

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

6: State forecast:  

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

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

7: Compute the normalized anomalies  

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

8: State update:  

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

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

---

---

**Algorithm 3:** Dual Ensemble Kalman filter of Algo. 2 applied on the coarse mesh. We have  $i = 1, \dots, N_e$ .

---

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

**Output:**  $\{\theta_k^{a,(i)}\}$  and  $\{(\mathbf{x}_k^C)^{a,(i)}\}$ ;  $k = 0, \dots, K$   
**begin**

```

1:   Initialize  $\{\theta_0^{a,(i)}\}$  and  $\{(\mathbf{x}_0^C)^{a,(i)}\}$ ;
   for  $k = 1, \dots, K$  do
2:     Parameter forecast:
        $\theta_k^{f,(i)} = \theta_{k-1}^{a,(i)} + \tau_k^{(i)}$  with  $\tau_k^{(i)} \sim \mathcal{N}(0, \Sigma_k^\theta)$ 
        $(\mathbf{x}_k^C)^{f,(i)} = \mathcal{M}_{k:k-1}^C \left( (\mathbf{x}_{k-1}^C)^{a,(i)}, \theta_k^{f,(i)} \right)$ 

       if Observation available then
3:         Observation ensemble:
            $(\mathbf{y}_k^C)^{o,(i)} = (\mathbf{y}_k^C)^o + (\epsilon_k^C)^{o,(i)}$  with  $(\epsilon_k^C)^{o,(i)} \sim \mathcal{N}(0, \mathbf{R}_k^C)$ 
            $(\mathbf{R}_k^C)^e = \frac{1}{N_e - 1} \sum_{i=1}^{N_e} (\epsilon_k^C)^{o,(i)} \left( (\epsilon_k^C)^{o,(i)} \right)^\top$ 

4:         Compute the normalized anomalies
            $[\Theta_k^f]_{:,i} = \frac{\theta_k^{f,(i)} - \overline{\theta_k^f}}{\sqrt{N_e - 1}}$ ;  $[\mathbf{Y}_k^f]_{:,i} = \frac{(\mathbf{y}_k^C)^{f,(i)} - \overline{(\mathbf{y}_k^C)^f}}{\sqrt{N_e - 1}}$ ;  $[\mathbf{E}_k^o]_{:,i} = \frac{(\epsilon_k^C)^{o,(i)} - \overline{(\epsilon_k^C)^o}}{\sqrt{N_e - 1}}$ 

5:         Parameter update:
            $(\mathbf{K}_k^C)^{\theta,e} = \Theta_k^f (\mathbf{Y}_k^f)^\top \left( \mathbf{Y}_k^f (\mathbf{Y}_k^f)^\top + \mathbf{E}_k^o (\mathbf{E}_k^o)^\top \right)^{-1}$ 
            $\theta_k^{a,(i)} = \theta_k^{f,(i)} + (\mathbf{K}_k^C)^{\theta,e} \left( (\mathbf{y}_k^C)^{o,(i)} - (\mathbf{y}_k^C)^{f,(i)} \right)$ 

6:         State forecast:
            $(\mathbf{x}_k^C)^{f,(i)} = \mathcal{M}_{k:k-1}^C \left( (\mathbf{x}_{k-1}^C)^{a,(i)}, \theta_k^{a,(i)} \right)$ 
            $(\mathbf{y}_k^C)^{f,(i)} = \mathcal{H}_k^C \left( (\mathbf{x}_k^C)^{f,(i)} \right)$ 

7:         Compute the normalized anomalies
            $[\mathbf{X}_k^f]_{:,i} = \frac{(\mathbf{x}_k^C)^{f,(i)} - \overline{(\mathbf{x}_k^C)^f}}{\sqrt{N_e - 1}}$ ;  $[\mathbf{Y}_k^f]_{:,i} = \frac{(\mathbf{y}_k^C)^{f,(i)} - \overline{(\mathbf{y}_k^C)^f}}{\sqrt{N_e - 1}}$ ;  $[\mathbf{E}_k^o]_{:,i} = \frac{(\epsilon_k^C)^{o,(i)} - \overline{(\epsilon_k^C)^o}}{\sqrt{N_e - 1}}$ 

8:         State update:
            $(\mathbf{K}_k^C)^{x,e} = \mathbf{X}_k^f (\mathbf{Y}_k^f)^\top \left( \mathbf{Y}_k^f (\mathbf{Y}_k^f)^\top + \mathbf{E}_k^o (\mathbf{E}_k^o)^\top \right)^{-1}$ 
            $(\mathbf{x}_k^C)^{a,(i)} = (\mathbf{x}_k^C)^{f,(i)} + (\mathbf{K}_k^C)^{x,e} \left( (\mathbf{y}_k^C)^{o,(i)} - (\mathbf{y}_k^C)^{f,(i)} \right)$ 

```

---

---

**Algorithm 4:** Multigrid EnKF algorithm. We have  $i = 1, \dots, N_e$ .

---

```

begin
1: Initialize  $\{(\mathbf{x}_0^F)^a, \bar{\theta}_0^a, \theta_0^{a,(i)}, (\mathbf{x}_0^C)^{a,(i)}\}$ 
   for  $k = 1, \dots, K$  do
2:   Fine grid forecast:
       
$$(\mathbf{x}_k^F)^f = \mathcal{M}_{k:k-1}^F((\mathbf{x}_{k-1}^F)^a, \bar{\theta}_k^a)$$

3:   Dual EnKF on coarse mesh: Apply Algo. 3
   if Observation available then
4:     Projection on the coarse grid
       
$$(\mathbf{x}_k^C)^* = \Pi_C((\mathbf{x}_k^F)^f)$$

5:     Fine grid state correction using the ensemble statistics:
       
$$(\mathbf{x}_k^C)' = (\mathbf{x}_k^C)^* + (\mathbf{K}_k^C)^{x,e} [(\mathbf{y}_k^C)^o - \mathcal{H}_k^C((\mathbf{x}_k^C)^*)]$$

       
$$(\mathbf{x}_k^F)^a = (\mathbf{x}_k^F)^f + \Pi_F((\mathbf{x}_k^C)' - (\mathbf{x}_k^C)^*)$$

6:     Matrix-Splitting iterative procedure on the final solution
       starting from  $(\mathbf{x}_k^F)^a$ .

```

---

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.
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](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/Heidelberg, 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. Trounev, 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] 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.
- [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] P. Sakov, D. S. Olivier, L. Bertino, An iterative enkf for strongly nonlinear systems, *Monthly Weather Review* 140 (2011) 1988–2004.
- [41] 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.
- [42] 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.
- [43] 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.