# Real-time algorithm to estimate and predict unsteady flows from few measurements using reduced-order models

Valentin Resseguier, Matheus Ladvig, Dominique Heitz

▶ **To cite this version:**

Valentin Resseguier, Matheus Ladvig, Dominique Heitz. Real-time algorithm to estimate and predict unsteady flows from few measurements using reduced-order models. 2021. hal-03445455v1

**HAL Id: hal-03445455**
**https://hal.science/hal-03445455v1**

Preprint submitted on 24 Nov 2021 (v1), last revised 2 Mar 2022 (v3)

**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.

# Real-time algorithm to estimate and predict unsteady flows from few measurements using reduced-order models

Valentin Resseguier*, Matheus Ladvig

*Lab, SCALIAN DS, Espace Nobel, 2 Allée de Becquerel, Rennes, France*

Dominique Heitz

*INRAE, OPAALE , Rennes, France*

**Abstract**

To successfully monitor and actively control hydrodynamic and aerodynamic systems (e.g., wind turbine blade, wind farm, hydrofoil, aircraft wings), it can be critical to estimate and predict the unsteady flow around them in real-time. We propose to introduce a new algorithm to couple onboard measurements with fluid dynamics simulations and data in real-time without the need to rely on an extensive computational infrastructure. This coupling is achieved by combining a Proper Orthogonal Decomposition Galerkin method, a model under location uncertainty stochastic closure, and a particle filtering scheme. We focus our numerical tests on a two- and a three-dimensional wake flows at low and moderate Reynolds numbers, respectively. Using a single measurement point, we obtain almost optimal flow estimations for up to 14 vortex shedding cycles after the learning time window.

*Keywords:* Fluid dynamics, reduced order model, uncertainty quantification, stochastic closure, particle filtering

## 1. INTRODUCTION

Accurate aeroelastic and aerodynamic active control can require state observers. It concerns, among others, gust and load alleviation, wind farm production maximization, active flutter suppression, flight and sail stability augmentation. However, estimating – or even predicting – an unsteady turbulent flow state from sparse measurements in real-time can be challenging. Through statistical estimation techniques, sensor observations can be assimilated to flow dynamical models' predictions, but some difficulties must first be overcome to be a viable strategy.

Firstly, the simulation must resolve sufficient spatiotemporal scales for the system to be stable, specifically in real-time. Though tempting, tackling this problem with purely data-driven fluid dynamics models (e.g., through machine learning techniques) may not be accurate and/or robust enough with turbulent flows, or else it would need a large amount of data to assimilate. Note that this last statement is also true for time-wise fluid flow estimation that

*Corresponding author
*Email address:* valentin.resseguier@scalian.com (Valentin Resseguier)

does not consider underlying time dynamics, such as supervised linear and non-linear interpolations [e.g. 1]. On the other hand, pure physics-based model simulations (e.g., Large Eddy Simulation, Reynolds-averaged Navier–Stokes) are often too slow for real-time applications. Using coarser meshes to speed up the simulations would most likely push them to miss critical spatiotemporal scales. Hence, Reduced Order Models (ROM) represent a excellent compromise (see, e.g., [2] for some aeroelastic applications). In particular, the Proper Orthogonal Decomposition (POD) -Galerkin method is a model that relies on physical equations while constraining the solution to live inside a small subspace learned from data (see, e.g., [3] for turbulent mixing layer dynamics analysis). Nevertheless, the unsteady CFD ROM state of the art limits itself to deterministic ROMs (often linear and/or with purely data-driven calibration) with limited prediction capabilities when the ROM dimension is low. It is probably mainly due to the chaotic and intermittent nature of turbulence and closure problems. Indeed, intermittency – ubiquitous in turbulence – is related to rare events and long-memory processes. These long-memory processes make learning from a finite time window more complicated because turbulence data are hardly exhaustive. Therefore, the learned or partially-learned turbulent flow ROMs remain inexact and uncontrolled in the long run, owing to their chaotic nature (intrinsic sensitivity to perturbations [4]) and the growth of accumulated error along time. Outside the learning time interval, predictions become less and less accurate. Additionally, we believe that ROM deterministic closures can hardly be accurate in the long run. Indeed, energy fluxes between temporal modes corresponding to orthogonal divergence-free spatial modes (e.g., curl of Fourier modes or POD modes) of a real incompressible flow are described by dyads and triads. They bring energy from one temporal mode toward another temporal mode, possibly thanks to a third temporal mode [5]. It is still true to treat the mean flow as one of the spatial modes. Unfortunately, in ROM, the mode truncation breaks many of these triads because the third mode is missing [6, 7]. The missing negative and positive energy fluxes create unstable modes and over-damped modes, respectively. Consequently, to stabilize ROMs, authors commonly introduce an additional deterministic term (typically an eddy viscosity term) [8, 9, 10] or an additional constraint [4], along with a possible calibration on available data [11, 12, 13]. However, few authors address the missing positive energy fluxes issue. One reason is that adding relevant terms, which increase energy, is much more difficult in a deterministic framework. Nevertheless, these positive energy fluxes are essential to maintain the linearly-stable temporal modes variability and thus to obtain coherent ROM dynamics in long-time integrations.

The simulation-measurement coupling, known in the literature as data assimilation, can be in itself a challenging task too. However, thanks to the weather forecast community, there have been many advances in the field, with promising research and many operational perspectives thriving lately [14]. Several variational data assimilation methods have been proposed to jointly learn the ROM (or part of the ROM) and the state in fluid dynamics reduced-order context [e.g. 15, 11, 16, 17, 12]. Nevertheless, those approaches generally require a large amount of dense data to be assimilated – typically two-dimensional flow observables like particle image velocimetry (PIV). In such a case, fluid flow *estimation* hardly necessitates ROM procedures since the problem is over-determined. A least-

square estimation can provide the first velocity modes (a kind of mode-based velocity interpolation). For *prediction* of unsteady turbulent flows, the use of ROM is more relevant, but those methods often have bad prediction skills (outside the learning interval).

A smaller amount of data to assimilate shall be compensated somehow by a better prior ROM and an adapted data assimilation algorithm. Algorithms fully addressing non-linearities of fluid mechanics are limited by the available computational power and the dynamical model's accuracy quantification. The use of our proposed ROM can overcome the former. The so-called dynamics under location uncertainty [18, 19] – a random fluid mechanics framework explicitly designed for this purpose – can handle the later. Inspired from the theoretical work of [20], [18] has introduced that stochastic closure and [19] generalized it. It has led to huge improvements in uncertainty and model error quantification both in high-dimensional CFD [21, 22, 23] and reduced state spaces [24, 25].

This paper will be organized as follows: section 2 will recall the main aspects of POD-Galerkin ROM, section 3 will present our main contribution: a randomized version of POD-Galerkin ROM, section 4 explains the data assimilation procedure, and finally, section 6 will showcase its potential through some of our numerical results.

## 2. POD-ROM

Reduced Order Models (ROM) aims to reduce the computational cost of simulations by drastically constricting the solution's degrees of freedom. This gain is typically enabled by a combination of simulation data and modeling based on physical equations. In CFD, solutions, i.e., the velocity fields, have as many degrees of freedom as grid points in the spatial domain (typically in the order of $10^6$). Thus, to achieve such a dimensionality reduction using ROMs, velocity fields are traditionally decomposed as follows:

$$\boldsymbol{v}(\boldsymbol{x},t) = \underbrace{\boldsymbol{w}(\boldsymbol{x},t)}_{\substack{\text{Resolved}\\ \text{by the ROM}}} + \underbrace{\boldsymbol{v}'(\boldsymbol{x})}_{\substack{\text{Unresolved}\\ \text{by the ROM}}}, \tag{1}$$

with

$$\boldsymbol{w}(\boldsymbol{x},t) = \underbrace{\overline{\boldsymbol{v}}(\boldsymbol{x})}_{\substack{\text{Time}\\ \text{averaged}}} + \underbrace{\sum_{i=1}^{n} b_i(t)\boldsymbol{\phi}_i(\boldsymbol{x})}_{\substack{\text{Unsteady}\\ \text{component}}}, \tag{2}$$

with $1 \leqslant n \leqslant 10^2$. Proper orthogonal decomposition (POD) learns the time-averaged $\overline{\boldsymbol{v}}(\boldsymbol{x})$ and the spatial modes $\boldsymbol{\phi}(\boldsymbol{x})$ through principal component analysis (PCA) applied to a set of high-resolution simulation solutions (learning set). Then, the physical equations (e.g., Navier-Stokes equations) can be projected onto these spatial modes, thus providing a system of $n$ coupled ordinary differential equations that describe the evolution of the temporal modes $b_i(t)$. Through the time integration of this low-dimensional system, and from equation (2), we can predict an a priori estimation of the velocity field at any given time. Consequently, this ROM construction scheme can be a midpoint between fully data-driven methods and pure physics-based models. It uses both the available simulation data and physical modeling to infer reliable predictions more efficiently.

## 3. MODEL UNCERTAINTY QUANTIFICATION

Models under location uncertainty (LU) are a type of random CFD model [18, 21, 23] that provides both an efficient closure (i.e., an efficient way to model the effect of the neglected dynamics degrees of freedom $v'$) and a quantification of the error induced by this closure. Though for it to be tractable, two assumptions must be made: the time decorrelation of the unresolved velocity component $\boldsymbol{v}'$ (see eq. (1)) and the stochastic transport – up to some forces $\boldsymbol{F}$ – of the resolved velocity component $\boldsymbol{w}$. With Ito stochastic calculus notations (see Appendix A), this reads:

$$\frac{Dw_k}{Dt} = \partial_t w_k + \left(\boldsymbol{w} - \tfrac{1}{2}\left(\boldsymbol{\nabla}\cdot\boldsymbol{a}\right)^T + \boldsymbol{v}'\right)\cdot\boldsymbol{\nabla} w_k - \tfrac{1}{2}\boldsymbol{\nabla}\cdot(\boldsymbol{a}\boldsymbol{\nabla} w_k) \quad = \quad F_k, \tag{3}$$

with

$$a_{pq} = \overline{v'_p v'_q}\ \tau_{v'}, \tag{4}$$

being the unresolved velocity (Eulerian) absolute diffusivity, $\overline{v'_p v'_q}$ the time average of $v'_p v'_q$ and $\tau_{v'}$ the unresolved velocity correlation time. Compared to classical fluid dynamics conservation equations, models under location uncertainty involve three new terms: a turbulent diffusion $\frac{1}{2}\boldsymbol{\nabla}\cdot(\boldsymbol{a}\boldsymbol{\nabla} w_k)$, a large-scale advecting velocity correction $-\frac{1}{2}\left(\boldsymbol{\nabla}\cdot\boldsymbol{a}\right)^T$, and a multiplicative noise $\boldsymbol{v}'\cdot\boldsymbol{\nabla} w_k$. To express the uncertainty induced by the dynamic truncation (inherent to any closure in CFD), we ran multiple simulations in parallel using the stochastic model to efficiently realize the most probable future states of the fluid system. Since that stochastic closure is based on physics, its robustness is proved, and calibrations can be performed from available physical quantities. Almost no tuning nor fitting is hence required.

[26] makes use of this formalism in a POD-Galerkin context for data analysis, but without considering the noise term $v'\cdot\boldsymbol{\nabla} w_k$. Here, we do consider this noise term. As in [25], we have implemented the POD-Galerkin of the Navier-Stokes model under location uncertainty (3). We obtain the following ROM:

$$\frac{\mathrm{d}b_i(t)}{\mathrm{d}t} = \mathcal{M}_i\left(\boldsymbol{b}(t), \dot{\boldsymbol{\beta}}(t)\right) \triangleq \underbrace{\sum_{p=0}^{n} l_{pi}\ b_p(t) + \sum_{p=0}^{n}\sum_{q=0}^{n} \boldsymbol{c}_{pqi}\ b_p(t) b_q(t)}_{\text{Usual POD-Galerkin terms}} + \underbrace{\sum_{p=0}^{n} f_{pi}\ b_p(t) + \sum_{p=0}^{n}\sum_{k=1}^{n} \tilde{\alpha}^R_{pik}\ b_p(t)\dot{\beta}_k(t)}_{\text{New POD-LU-Galerkin terms}}, \tag{5}$$

where $(\dot{\beta}_k)_k$ are $n$ independent one-dimensional white noises and by convention $b_0 = 1$, $\mathcal{M}_0 = 0$ and $\boldsymbol{\phi}_0 = \overline{v}$. Using the physical equations (3) and (4) and corresponding technical statistical estimators based on stochastic calculus, we computed the ROM coefficients $l$, $f$, $c$, and $\tilde{\alpha}^R$ from the resolved spatial modes $\boldsymbol{\phi}_i$, the resolved temporal modes $b_i$, and the POD residual velocity $v'$. Interested readers can refer to Appendix B or to [25] for more details. The low-dimensional system (5) incorporates noise terms, thus enabling model uncertainty quantification and strengthening its forecasting capabilities to allow for more reliable forecasts beyond the learning period.

## 4. PARTICLE FILTERING

The final block in our stochastic POD-ROM model's pipeline is a particle filter [27] in order to integrate the real-time measurements from all the different sensors. We first explain why we choose particle filtering rather than more usual methods like Ensemble Kalman Filter (EnKF). We then point out again the principle of this data assimilation algorithm.

In fluid mechanics, the equations describing the system behavior are non-linear, non-Gaussian, and high-dimensional. Therefore, the Kalman filter, the well-known Gaussian data assimilation technique, does not have good performance. Moreover, the curse of dimensionality prevents the construction of the huge state covariance matrix. Theoretically, the proper method to approach such non-linear and non-Gaussian dynamics is particle filtering [28]. However, this method often requires a vast ensemble of realizations. When the state dimension increases, the problem quickly becomes intractable as even more realizations are needed , and each new realization comes with a severe additional computational cost. That is why most fluid dynamics practitioners prefer the EnKF or variational data assimilation approaches, especially in weather forecasts Note that there are some exceptions and variants of particle filtering to address higher and higher dimensions [29, 30, 31, 32, 33]. Variational methods are indeed competitive, especially the widely-used 4D-Var algorithm [34] and its variants. Nevertheless, the 4D-Var algorithm necessitates adjoint codes and neglects the statistical non-stationarity and the non-Gaussianity of models' errors. Moreover, ensemble-based data assimilation methods are easier to parallelize because one can generally forecast each realization independently between two assimilation steps. The EnKF and its variants (e.g., square-root EnKF with localized and inflated covariance) are less sensitive to the curse of dimensionality than particle filters, but the state correction is linear, which can lead to non-physical solutions [e.g. 23]. Here, we can use a fully-nonlinear filter – the particle filter – for two reasons. Firstly, the POD-ROM makes the state dimension small enough. Secondly, models under location uncertainty efficiently spread small ensembles over the state space without introducing new errors [24, 23, 25]. Therefore, particle filtering accommodates a small ensemble in this context, which strongly reduces its computational cost.

Algorithm 1 points out again the skeleton of the most known particle filter, known as sequential importance resampling (SIR) [35, 36]. We used this algorithm in this paper. The first step initializes an ensemble of $N_p$ independent states and associates an equal weight to each state. The independent states are called particles or realizations . They are forecast in time with the evolution model (5). Each time a new measurement is assimilated, weights are updated based on their respective likelihoods. An additional step called re-sampling prevents the weights variances from increasing with time. Such a problem is known as degeneracy and would result in poor state estimations. Eventually, the ensemble forms the shape of the posterior distribution and gives the state estimation.

**Algorithm 1** Particle Filter SIR with ROM under location uncertainty

**Initialization**

- Compute the ROM coefficients $l$, $f$, $c$, and $\tilde{\alpha}^R$ from a simulation output dataset (see Appendix B for definitions). ▷ POD-Galerkin
- Compute matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ (see Appendix D for definitions). ▷ Log-likelihood matrices
- Sample $\begin{pmatrix} b_1^{(j)}(0) \\ \vdots \\ b_n^{(j)}(0) \end{pmatrix} \overset{iid}{\sim} \mathcal{N}\left(0, \begin{pmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{pmatrix}\right)$ where $\lambda_i = \overline{b_i^2}$. ▷ Initializing the first state

**Loop over time** $t$

*Importance sampling*

- $\boldsymbol{b}^{(j)}(t) = \boldsymbol{b}^{(j)}(t - \mathrm{d}t) + \mathcal{M}\left(\boldsymbol{b}^{(j)}(t - \mathrm{d}t), \dot{\boldsymbol{\beta}}^{(j)}(t - \mathrm{d}t)\right) \mathrm{d}t$. ▷ State transition
- If an observation $y(t)$ is available at the current time $t$:
  - $l_j(t) = l\left(\boldsymbol{y}(t)|\boldsymbol{b}^{(j)}(t)\right) = (\boldsymbol{b}^{(j)}(t))^T \boldsymbol{A}\, \boldsymbol{b}^{(j)}(t) + \boldsymbol{y}(t)^T \boldsymbol{B}\, \boldsymbol{b}^{(j)}(t))$; ▷ Log-likelihood up to a constant (see Appendix D for the proof of this formula)
  - $l_j(t) = l_j(t) - \max_j l_j(t) + 90$; ▷ Add a constant to prevent numerical errors when applying exp
  - $W_j(t) = \exp\left(l_j(t)\right)$; ▷ Computing weights
  - $\boldsymbol{W}_j(t) = \frac{W_j(t)}{\sum_{m=1}^{N_p} W_m(t)}$; ▷ Normalization

*Re-sampling*

  - Each new temporal mode $b^{(j)}(t)$ is replaced by one of the old temporal modes $b^{(1)}(t)$, ..., $b^{(N_p)}(t)$ with probability $\boldsymbol{W}_1(t)$, ..., $\boldsymbol{W}_{N_p}(t)$, respectively. ▷ Resampling

**Final posterior distribution** at a time $t$ larger than measurement times $t_1, \ldots, t_K$

$p\left(\boldsymbol{b}(t)|\boldsymbol{y}(t_1), \ldots, \boldsymbol{y}(t_K)\right) \approx \sum_{j=1}^{N_p} \frac{1}{N_p}\, \delta\left(\boldsymbol{b}(t) - \boldsymbol{b}^{(j)}(t)\right)$. ▷ Posterior Distribution

## 5. MEASUREMENTS TO ASSIMILATE

Theoretically, our method can assimilate any measurements. Here, we choose a widely-used fluid flow velocimetry technique: particle image velocimetry (PIV). The method principle uses solid particles submerged in a fluid combined with a high-power light emission source. A camera with a high frame rate takes photos of the fluid flow with the submerged particles. Image processing and optical flow algorithms are then applied to the image sequence to estimate the fluid flow velocity.

Specifically, our algorithm was tested with a cropped two-dimensional, two-component PIV (2D2C PIV) experimental arrangement with a linear observation model as in (7).

$$\boldsymbol{y} = \boldsymbol{\mathcal{H}}[\boldsymbol{v}] + \boldsymbol{\epsilon}_y, \tag{6}$$

$$= \sum_{i=0}^{n} \boldsymbol{\mathcal{H}}[\boldsymbol{\phi}_i]\, b_i + (\boldsymbol{\mathcal{H}}[\boldsymbol{v}'] + \boldsymbol{\epsilon}_y)\,, \tag{7}$$

where $\boldsymbol{y}$ is the measured signal, $\boldsymbol{\epsilon}_y$ represents the PIV measurement error. The linear operator $\boldsymbol{\mathcal{H}}$ incorporates a 3-dimensional spatial smoothing operation, the occlusion of the horizontal plane, and its corresponding component in the velocity field to mimic the PIV measurement process. The parameters inside $\boldsymbol{\mathcal{H}}$ and $\epsilon_y$ are estimated using experimental data, comparing the hot-wire and PIV measurements' spectrum (using a Taylor assumption). Appendix C details the definition of $\boldsymbol{\mathcal{H}}$ and the measurement noise covariance. Additionally, to make the data assimilation task more challenging, the information relating to a subset of points in the grid was obscured through the operator $\boldsymbol{\mathcal{H}}$. Indeed, estimating a vector $b$ of $n \sim 10$ components from a noisy linearly-dependent observation vector $\boldsymbol{y}$ of $M_{PIV} \sim 10^4 (\gg n)$ components is often an over-determined problem and could otherwise be solved using a straightforward least-squares procedure. Note that the strong influence of the unresolved velocity $\boldsymbol{v}'$ on the final observation model's uncertainty is taken into account through a noise term in (7).

For the first tests presented here, we only consider synthetically generated measurements that were produced using the aforementioned observation model (6). It allows us to know the exact values of the velocity field everywhere on the 3-dimensional space to perform the algorithm validation.

## 6. NUMERICAL RESULTS

We have performed numerical experiments with two test flows in order to evaluate the prediction skills of the proposed data assimilation method. For comparison, we have pursued the same numerical evaluations with two state-of-the-art POD-ROMs, presented below in this section. We first evaluate the prediction skills visually from the vorticity and the Q-criterion of the reconstructed velocity fields. Then, we quantitatively compare the reconstructed velocity errors of each method. After this, we numerically investigate the influence of the number and positions of measurements and the ensemble size. Finally, we provide some insights into our algorithm complexity.

Our data assimilation procedure has been applied to two different cylinder wake flows, two- and three-dimensional

at Reynolds numbers 100 and 300, respectively. In wake flows, vortices are shed from the cylinder in a pseudo-periodic way. Before our data assimilation tests, we performed direct numerical simulations (DNS) using Incompact3d, a high-order flow solver based on the discretization of the incompressible Navier-Stokes equations [37]. These fully-resolved simulations are illustrated in top of figure 1 and top right of figure 3. A big difference in terms of degrees of freedom exists between these two flows. At Reynolds number 100, the flow remains laminar, whereas, at Reynolds number 300, the flow is already three-dimensional and very complex. The simulation's spatial grids correspond to state-space dimensions of about $10^4$ and $10^7$, respectively. During the learning period, 140 and 80 vortex shedding cycles are used to construct the ROM. The remaining vortex shedding cycles are used to build synthetic measurements and to test the method. For both flows, a single spatial resolution point of the synthetic PIV data is assimilated ten times for each vortex shedding cycle. The observation point – visible in figure 1 – is outside but close to the recirculation zone. Specifically, its coordinates are $x = 1.31D$ (streamwise direction), $z = 0$ (spanwise direction), and $y = 1.27D$ (orthogonal direction), where the cylinder is centered on $(0, 0, -)$ and has a diameter $D$. We refer to this setting as observation case 1. Other observation cases will be considered at the end of this section.

## 6.1 State-of-the-art ROMs

To better appreciate the prediction skills of our methodology, we will compare it to state-of-the-art algorithms. As a baseline, we will consider two state-of-the-art POD-ROMs. To facilitate the comparison, we use the same observation model (7), assimilation method (particle filter SIR), and ensemble size (100 particles).

### 6.1.1 Deterministic state-of-the-art ROM

The first state-of-the-art ROM considered is a usual deterministic POD-Galerkin with optimally fitted eddy (via least square):

$$\frac{\mathrm{d}b_i(t)}{\mathrm{d}t} = \underbrace{\sum_{p=0}^{n} l_{pi}\, b_p(t) + \sum_{p=0}^{n}\sum_{q=0}^{n} \boldsymbol{c}_{pqi}\, b_p(t) b_q(t)}_{\text{Usual POD-Galerkin terms}} + \underbrace{\left(\frac{\nu^{\mathrm{ev}}}{\nu} - 1\right) \sum_{p=0}^{n} l_{pi}\, b_p(t)}_{\text{Fitted on the resolved temporal modes dynamics}} , \tag{8}$$

where $\nu^{\mathrm{ev}}$ is the fitted eddy viscosity, and $\nu$ is the molecular viscosity. This POD-ROM is deterministic but ensemble-based data assimilation can still be used by randomizing initial conditions.

### 6.1.2 Stochastic state-of-the-art ROM

Unfortunately, simple randomization of the initial conditions is known to lead to error underestimation in fluids dynamics usually [38, 39, 40, 41]. So, to keep things simple, we define a second baseline POD-ROM by adding a white noise term to the first baseline ROM. It constitutes a simple *adhoc* randomization technique for a given

dynamical system through a Gaussian additive forcing. Despite its potential lack of physical relevance, such a strategy is very often adopted in data-assimilation applications [24].

$$\frac{\mathrm{d}b_i(t)}{\mathrm{d}t} = \underbrace{\sum_{p=0}^{n} l_{pi}\; b_p(t) + \sum_{p=0}^{n}\sum_{q=0}^{n} \boldsymbol{c}_{pqi}\; b_p(t) b_q(t)}_{\text{Usual POD-Galerkin terms}} + \underbrace{\left(\frac{\nu^{\mathrm{ev}}}{\nu} - 1\right)\sum_{p=0}^{n} l_{pi}\; b_p(t) + \sum_{k=1}^{n} \sigma_{ik}^{\mathrm{ev}}\; \dot{\beta}_k(t)}_{\text{Fitted on the resolved temporal modes dynamics}}, \tag{9}$$

where $\nu^{\mathrm{ev}}$ is the fitted eddy viscosity, $\nu$ is the molecular viscosity, and $(\boldsymbol{\sigma}^{\mathrm{ev}})(\boldsymbol{\sigma}^{\mathrm{ev}})^T$ is the fitted additive noise covariance matrix. For more details about this state-of-the-art stochastic ROM, one can refer to [25] for example.

[42] also propose a similar POD-ROM to be used with a particle filter. The authors estimate a whole additional linear term and not just an eddy viscosity coefficient. Their noise term forces the POD-ROM at each assimilation step only (not at every time step) and has a tuned variance. Their observation model is somehow simpler as their observation noise has a tuned spatially-uniform variance.

## 6.2 Qualitative analysis of the prediction skills

To appreciate the skills of each method, we will compare the predicted velocity fields of the different methods to a reference velocity field. A natural reference field is the DNS field, which will be denoted $\boldsymbol{v}_{\mathrm{dns}}$. Nevertheless, the solution of any POD-ROM is confined to the affine space spanned by the POD modes (the set of fields which write as (2)). Therefore, the best[1] field reachable by a POD-ROM-based method is the orthogonal projection of the DNS field onto this reduced affine space, i.e., the projection onto the POD basis. Let us denote $\widetilde{\Pi}_{\phi}[\boldsymbol{v}_{\mathrm{dns}}]$ this new reference field. It reads:

$$\widetilde{\Pi}_{\phi}[\boldsymbol{v}_{\mathrm{dns}}] \triangleq \overline{\boldsymbol{v}} + \sum_{i=1}^{n} b_i^{\mathrm{ref}} \boldsymbol{\phi}_i \qquad \text{with} \qquad b_i^{\mathrm{ref}} \triangleq \left(\boldsymbol{\phi}_p, \boldsymbol{v}_{\mathrm{dns}} - \overline{\boldsymbol{v}}\right). \tag{10}$$

This projected field is the theoretical performance limit: it corresponds to the case of exactly known temporal modes $b_i(t) = b_i^{\mathrm{ref}}(t)$ $(i \leqslant n)$ in the equation (2). We have represented these projected fields in the second panel of figures 1 and 2 for the Reynolds number 100 and at the top left of figures 3 and 4 for the Reynolds number 300 respectively. Note that when the Reynolds number increases, even though the projected velocity field is still meaningful, it can be relatively far from the DNS field.

### 6.2.1 Reynolds number 100

Figures 1 and 2 show the vorticity field[2] of the data assimilation results at Reynolds number 100, with $n = 2$ and 8 modes, respectively. Despite the small amount of information in the measurements, our ROM shows a

[1] Best in terms of $L^2$ norm. Note that by definition, the POD is also the best modal decomposition in terms of $L^2$ norm for a given degree of freedom $n$.

[2] The vorticity field is the curl of the velocity field and is a classical visual tool of CFD to visualize 2D vortices.

excellent predictive power even outside the learning period. The proposed observers are very close to the projected references, positively demonstrating the vast potential of our approach. For $n = 8$, the projected reference is also very close to the DNS field. In order to better appreciate the potential of our method, figures 1 and 2 also display the predictions from the stochastic state-of-the-art POD-ROM (9) (the POD-Galerkin with optimally fitted eddy viscosity and additive noise). The fitted additive noise variance of this POD-ROM is huge in order to encompass the significant ROM error. Therefore, the prior probability distribution generated by the state-of-the-art ROM is hardly informative, while the single measurement gives little additional information. That is why this state-of-the-art method fails to estimate the proper flow. This behavior worsens when the ROM order $n$ increases since the dynamics become more complicated, the is more erroneous, and the fitted additive noise variance is more prominent. The results of the deterministic POD-ROM method (8) are even worse and are hence not discussed here. The skills of this model will be presented later in section 6.3. In contrast, the physical structure of our POD-ROM – and in particular of its skew-symmetric multiplicative noise – guarantees an efficient prior probability distribution [25] and then an accurate data assimilation.

#### 6.2.2 Reynolds number 300

Figures 3 and 4 show the Q-criterion[3] isosurfaces of the data assimilation results at Reynolds number 300, with $n = 2$ and 8 modes, respectively. Again, the proposed observers are almost identical to the projected references. We point out again that these references (top left plots in figures 3 and 4) are the theoretical performance limits: they correspond to the case of exactly known temporal modes $b_i(t)$ $(i \leqslant n)$ in the equation (2). The differences between these theoretical optimums and the DNS reference correspond to the unresolved velocity $v'$, mainly restricted to small-scales 3-dimensional effects at Reynolds number 300. In contrast, the stochastic state-of-the-art POD-ROM method strongly differs from the reference. The reasons are the same that for Reynolds number 100, i.e. the noise structure of the state-of-the-art POD-ROM is not adapted.

### 6.3 Quantitative analysis of the prediction skills

Vorticity fields and Q-criterion isosurfaces provide good qualitative analyses. For a quantitative one, figure 5 plots the global velocity prediction normalized error for Reynolds numbers 100 and 300, from the end of the learning period until 20 and 14 vortex shedding cycles later, respectively. Let us specify how we define and compute this prediction error. If we denote $\boldsymbol{w}^{\text{est}} = \sum_{i=0}^{n} b_i^{\text{est}} \boldsymbol{\phi}_i$ the estimated velocity field, the expression of the Mean Square Error (MSE) simplifies due the orthogonality properties of the orthogonal projection $\widetilde{\Pi}_{\phi}[\boldsymbol{v}_{\text{ref}}]$ and of the POD modes

[3] Q-criterion is a classical visual tool of CFD to visualize vortices.

Reference simulation $\boldsymbol{v}_{\text{dns}}$
(2D DNS at Reynolds 100:
state space dimension
of about $10^4$)

Projection of the DNS
onto the POD basis $\widetilde{\Pi}_{\phi}[\boldsymbol{v}_{\text{dns}}]$
(State space
of dimension 2)

Our reduced data assimilation prediction
(ROM state space
of dimension 2)

State-of-the-art method prediction
(ROM state space
of dimension 2)

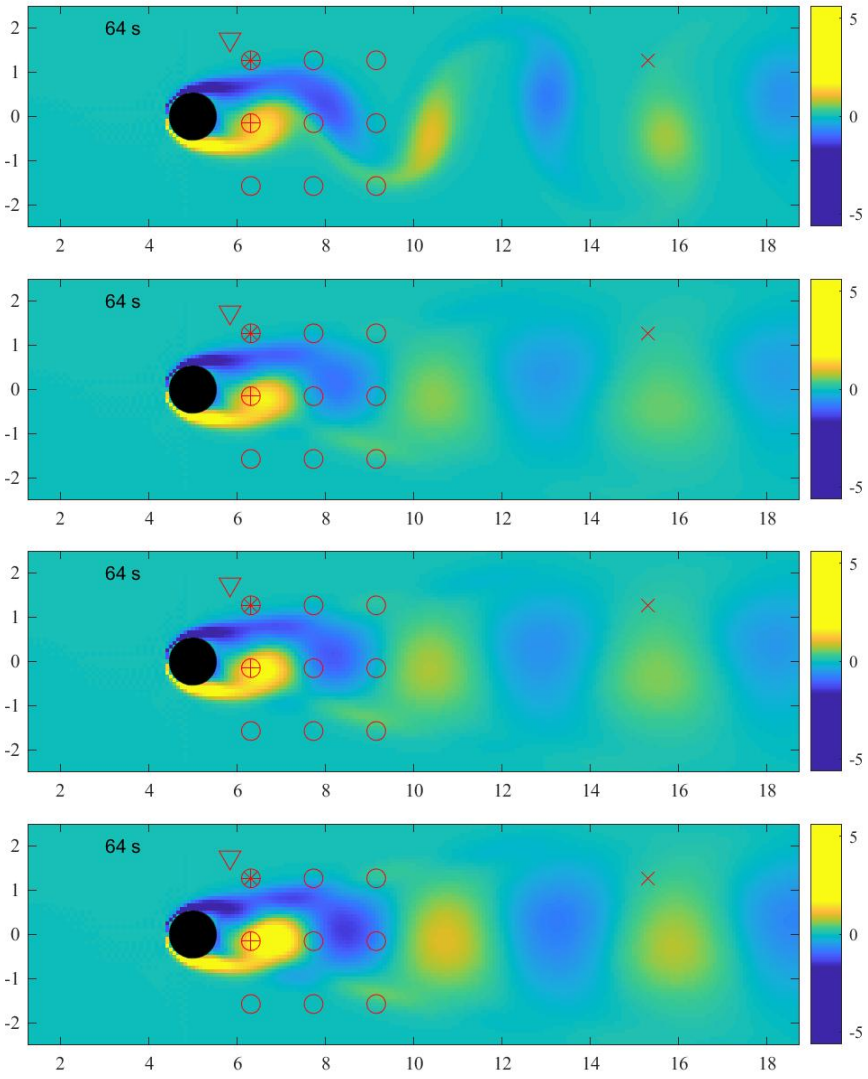


Figure 1: Vorticity field – of (from top to bottom) the 2D DNS at Reynolds number 100, the reference 2-dimensional representation (projection of the DNS onto the POD modes), our POD-ROM approach, and the state-of-the-art stochastic POD-ROM method.The red signs indicate the measurement locations for the different observation cases considered: case 1 (star symbol) (considered in sections 6.2 and 6.3 and figures 1-5 and 8), case 2 (circle symbols), case 3 (cross symbol close to the cylinder) (both considered in section 6.4.1 and figure 6), case 4 (triangle symbol), and case 5 (downstream cross symbol) (both considered in section 6.4.1 and figure 7).

Reference simulation $\boldsymbol{v}_{\text{dns}}$
(2D DNS at Reynolds 100:
state space dimension
of about $10^4$)

Projection of the DNS
onto the POD basis $\widetilde{\Pi}_{\phi}[\boldsymbol{v}_{\text{dns}}]$
(State space
of dimension 8)

Our reduced data assimilation prediction
(ROM state space
of dimension 8)

State-of-the-art method prediction
(ROM state space
of dimension 8)

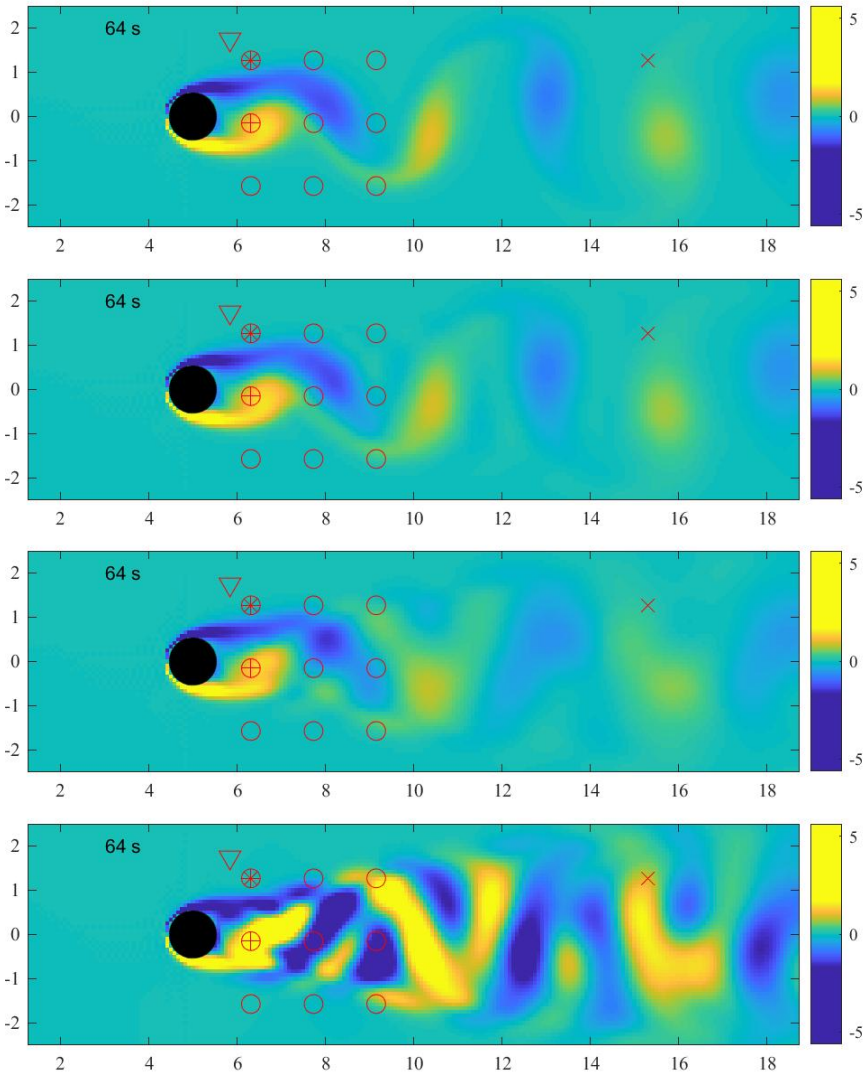


Figure 2: Vorticity field – 13 vortex shedding cycles after the learning period – of (from top to bottom) the 2D DNS at Reynolds number 100, the reference 8-dimensional representation (projection of the DNS onto the POD modes), our POD-ROM approach, and the state-of-the-art stochastic POD-ROM method.The red signs indicate the measurement locations for the different observation cases considered: case 1 (star symbol) (considered in sections 6.2 and 6.3 and figures 1-5 and 8), case 2 (circle symbols), case 3 (cross symbol close to the cylinder) (both considered in section 6.4.1 and figure 6), case 4 (triangle symbol), and case 5 (downstream cross symbol) (both considered in section 6.4.1 and figure 7).

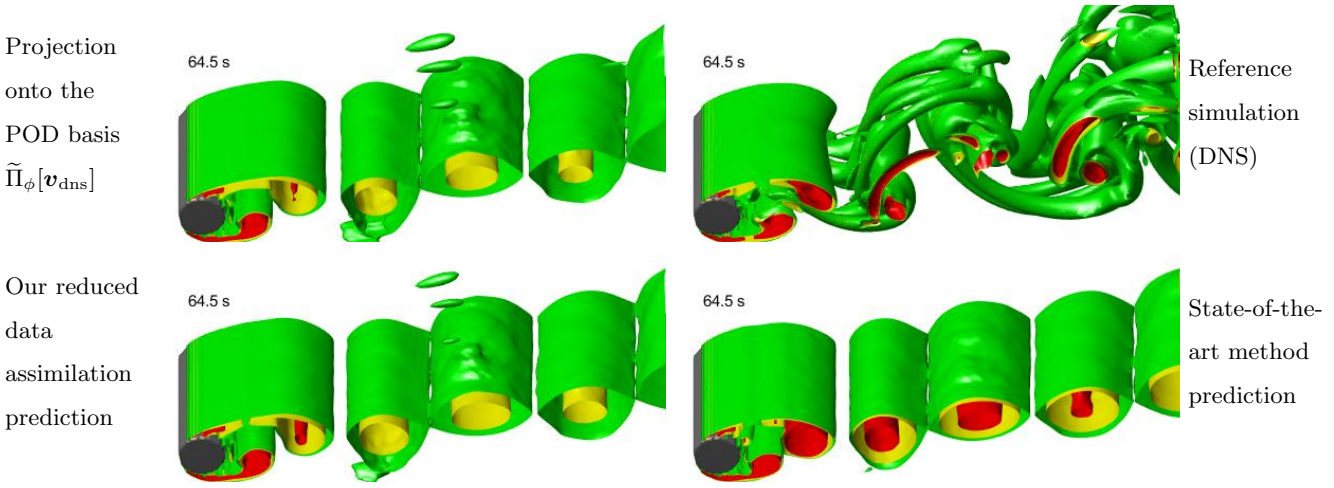


Figure 3: Q-criterion – 13 vortex shedding cycles after the learning period – from the reference 2-dimensional representation (projection of the 3D DNS at Reynolds 300 onto the POD modes) (top left), the DNS (top right), our POD-ROM approach (bottom left), and the state-of-the-art stochastic POD-ROM method (bottom right).

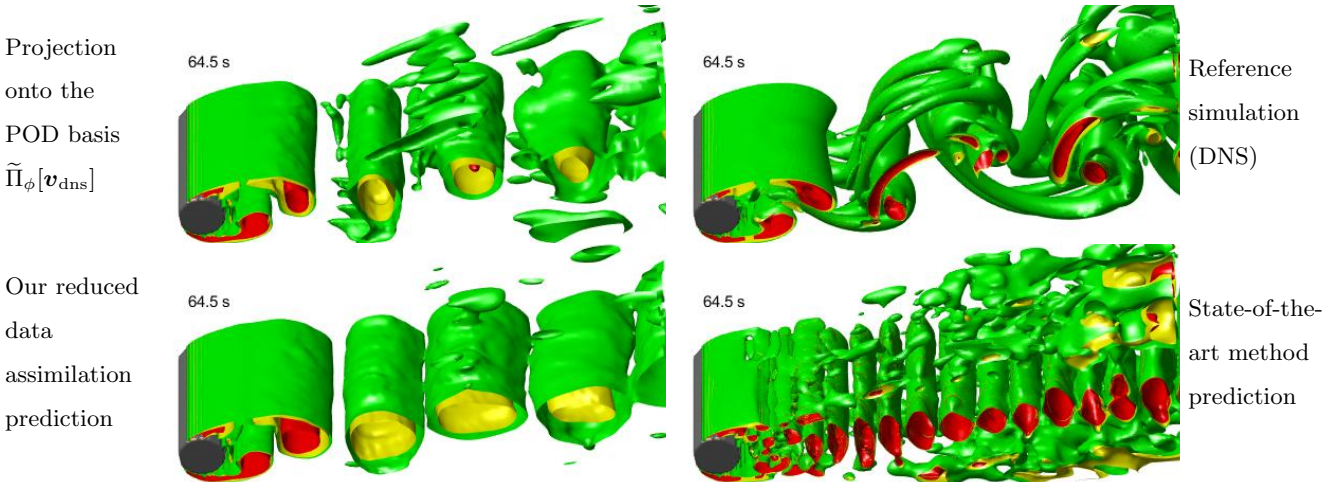


Figure 4: Q-criterion – 13 vortex shedding cycles after the learning period – from the reference 8-dimensional representation (projection of the 3D DNS at Reynolds 300 onto the POD modes) (top left), the DNS (top right), our POD-ROM approach (bottom left), and the state-of-the-art stochastic POD-ROM method (bottom right).

$\boldsymbol{\phi}_i$:

$$\text{MSE} \triangleq \int_\Omega \left\| \boldsymbol{w}^{\text{est}} - \boldsymbol{v}_{\text{dns}} \right\|^2, \tag{11}$$

$$= \int_\Omega \| \underbrace{\boldsymbol{w}^{\text{est}} - \widetilde{\Pi}_\phi[\boldsymbol{v}_{\text{dns}}]}_{=\sum_{i=1}^n \left(b_i - b_i^{\text{ref}}\right)\boldsymbol{\phi}_i} \|^2 + \int_\Omega \| \underbrace{\boldsymbol{v}_{\text{dns}} - \widetilde{\Pi}_\phi[\boldsymbol{v}_{\text{dns}}]}_{=\boldsymbol{v}'} \|^2, \tag{12}$$

$$= \underbrace{\sum_{i=1}^n \left(b_i - b_i^{\text{ref}}\right)^2}_{\text{ROM-dependent}} + \underbrace{\int_\Omega \left\| \boldsymbol{v}_{\text{dns}} - \widetilde{\Pi}_\phi[\boldsymbol{v}_{\text{dns}}] \right\|^2}_{\text{ROM-independent}}, \tag{13}$$

where the integrations are performed over the spatial space denoted $\Omega$ and $\| \bullet \|$ is the usual Euclidean norm of $\mathbb{R}^2$ or $\mathbb{R}^3$. The first error term depends on which ROM and data assimilation we choose, whereas the second depends solely on the POD modes $\boldsymbol{\phi}_i$. Therefore, the last term is the same for the proposed method and the state-of-the-art POD-ROMs. Note that this term is time-dependent in general. In figure 5, we normalize the MSE by the mean kinetic energy (MKE) in the fluid moving frame (averaged over the training set), MKE $= \overline{\int_\Omega \|\boldsymbol{v} - \boldsymbol{v}_\infty\|^2}$, with $\boldsymbol{v}_\infty = (1, 0, 0)^T$ the velocity far from the cylinder. Then, we take the square root of the normalized MSE. Subtracting the far fluid frame velocity, $\boldsymbol{v}_\infty$, makes the MKE independent of the spatial domain choice $\Omega$ for a spatial domain large enough. It makes our normalized RMSE also independent of the spatial domain choice (for a spatial domain large enough) and hence more objective. Note that even divided by the volume (or surface) of $\Omega$, $\overline{\int_\Omega \|\boldsymbol{v}\|^2}$ still depends on the spatial domain choice because the velocity variability (the wake) is localized in a region of the spatial domain.

In the plots, we readily see that our method outperforms the stochastic state-of-the-art POD-ROMs. Note that when the light blue line goes above the solid black line, it means that taking all temporal modes equal to zero is more accurate than using the stochastic state-of-the-art POD-ROM. The performances of this state-of-the-art POD-ROM are also getting worse for larger $n$, unlike our method performances. In figure 5, we also superimpose the results of the deterministic POD-ROM. As expected, the variance of the corresponding ensemble quickly decreases, which leads to variance underestimation, and thus filter degeneracy, wrong estimation, and – at Reynolds number 300 – divergence in time of the POD-ROM estimation. This variance decay is an illustration of the classical alignment of ensembles along with unstable directions [43, 44], which is due to the over-damping of stable temporal modes [6, 7]. This over-damping and, more generally, the variance decay is induced by the missing positive energy fluxes toward each temporal mode (the mode truncation removes many triad-based fluxes) and the stabilizing corrective terms (here the eddy viscosity term).

### 6.4 Influence of the available resources

In practice, the result of our algorithm can be influenced by several factors, such as the amount and the quality of the available measurement or the available computational power.

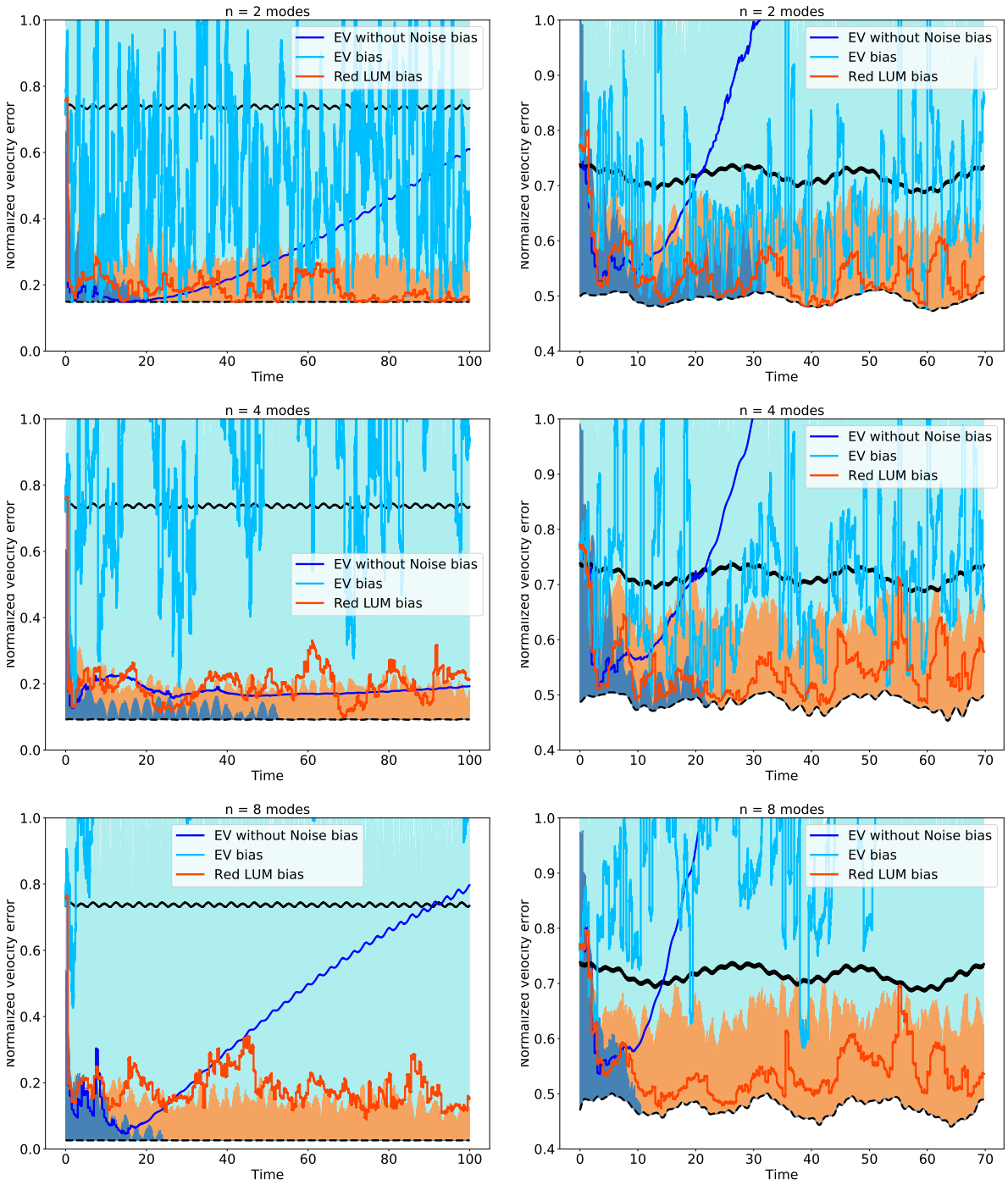


Figure 5: Global velocity prediction normalized error (after the learning period) for the wake flow at Reynolds 100 (left) and 300 (right) in the observation case 1 with, from top to bottom $n = 2, 4$ and 8 modes for our POD-ROM method (orange), the stochastic state-of-the-art POD-ROM method (light blue), and the deterministic state-of-the-art POD-ROM method (dark blue). The shaded colors (light orange, light blue, and grey) correspond to the respective estimated *a posteriori* standard deviations. The dashed black line at the bottom is the POD truncation error. The solid black line at the top is the error obtained by setting all temporal modes $b_i$ to 0, i.e. keeping only the time averaged velocity $\overline{v}$.

| Case name | Description | $x/D$ | $y/D$ | $z/D$ | Symbol on figures 1 and 2 |
|---|---|---|---|---|---|
| Case 1 | 1 observation point near the recirculation zone | 1.31 | 1.27 | 0 | $*$ |
| Case 2 | $3 \times 3$ observation points | 1.31 | 1.27 | 0 | $\circ$ |
| | | 2.73 | 1.27 | 0 | |
| | | 4.15 | 1.27 | 0 | |
| | | 1.31 | $-0.15$ | 0 | |
| | | 2.73 | $-0.15$ | 0 | |
| | | 4.15 | $-0.15$ | 0 | |
| | | 1.31 | $-1.57$ | 0 | |
| | | 2.73 | $-1.57$ | 0 | |
| | | 4.15 | $-1.57$ | 0 | |
| Case 3 | 1 observation point inside the recirculation zone | 1.31 | $-0.15$ | 0 | $+$ |
| Case 4 | 1 point outside the wake | 0.84 | 1.74 | 0 | $\nabla$ |
| Case 5 | 1 observation point far downstream | 10.31 | 1.27 | 0 | $\times$ |

Table 1: Description of the considered observation cases.

#### 6.4.1 Influence of the available measurements

We have simulated different observation cases of the wake flow at Reynolds number 300, summarized in table 1. We now focus on this Reynolds number because the estimation problem is more difficult at Reynolds number 300 than at Reynolds number 100. Figures 6 and 7 plot the corresponding estimation errors for the stochastic POD-ROM approach and for the method we propose.

As expected, all methods improve when more measurements are available (observation case 2, left panel of figure 6). The stochastic state-of-the-art ROM remains less efficient than our method. However, the two methods tend to become similar for a larger number of observation points – typically for a number of observations $M_y$ superior to the dimension $n$ of the reduced space. It is coherent with the good prediction results obtained by [42] with a particle filter and a POD-ROM similar to the stochastic state-of-the-art POD-ROM (9), with typically 9 observations points. Note however that, for many observation points, the observation model alone is often an over-determined least-square problem (number of equations superior to the number of unknown variables), and that a dynamic ROM is hardly needed for flow estimation. Note also that when the number of observation points increases too much, the likelihood becomes very peaky, and the particle filtering algorithm needs to be improved to prevent filter degeneracy. For our wake flow example, tempering and non-Gaussian jittering [29, 32] have been found to be an efficient solution even for about $M_y \sim 10^4$ observations points (not shown). Considering a single observation point may be considered as an over-complicated estimation problem compared to practical applications. Nevertheless,

industrial applications often rely on cheap sensors which provide very little information on the flow. Additionally, many parameters (e.g., boundary conditions, Reynolds number) are unknown or poorly known in more realistic situations and should be estimated on top of the state variables. Thus, more information is needed to constraint the estimation problem, and this information may not be available in the few available measurements. We believe that ROM relying more on physical a priori can bring that missing information, as our proposed POD-ROM does here by performing data assimilation from a single measurement.

We also challenge the data assimilation algorithms with a single observation point at different locations. For an observation point inside the recirculation zone (observation case 3, right panel of figure 6), the state-of-the-art ROM slightly improves but its RMSE remains close or above the zero solution RMSE, i.e. the error obtained by setting all temporal modes $b_i$ to 0. The Red LUM method prediction skills are equivalent to observation case 1. For an observation point outside the wake (observation case 4, left panel of figure 7), all method skills strongly deteriorate, as expected. For an observation point far downstream (observation case 5, right panel of figure 7), the state-of-the-art method keeps similar results. The skills of our method slightly deteriorate but remain much better than the state of the art. To conclude, the results depend on this location, but remain satisfactory as long as the observation point is in contact with the wake. Results seem more accurate for observations close to the recirculation zone.

We have also performed tests with observation data available less often, precisely one assimilation per vortex shedding cycle. As expected, fewer observations (in time or space) always seem to deteriorate the results. For our POD-ROM method, assimilating $3 \times 3$ observations points one time per vortex shedding cycle is more or less equivalent to assimilating a single observation point (observation case 1) five times per vortex shedding cycle (not shown).

### 6.4.2 Influence of the number of realizations

Figure 8 shows how RMSE varies with the ensemble size $N_p$ for our approach and for the stochastic state-of-the-art POD-ROM method. When the ROM dimension $n$ increases, a larger ensemble size seems to be needed. Our approach converges faster with the ensemble size than the state-of-the-art one. Only 100 realizations are needed with our method. It is also due to the specific structure of the noise. It is well adapted to represent the potential ROM errors, and thus few samples are sufficient to cover the likely system evolutions. Note that a smaller ensemble size means a smaller CPU cost and is of great advantage for real-time applications with embedding systems.

## 6.5 Complexity

Several hours of runs on a supercomputer are necessary to generate the simulation dataset with the off-line high-resolution highly-optimized CFD code.

In contrast, the off-line ROM construction takes a few hours on a laptop with our original non-parallelized

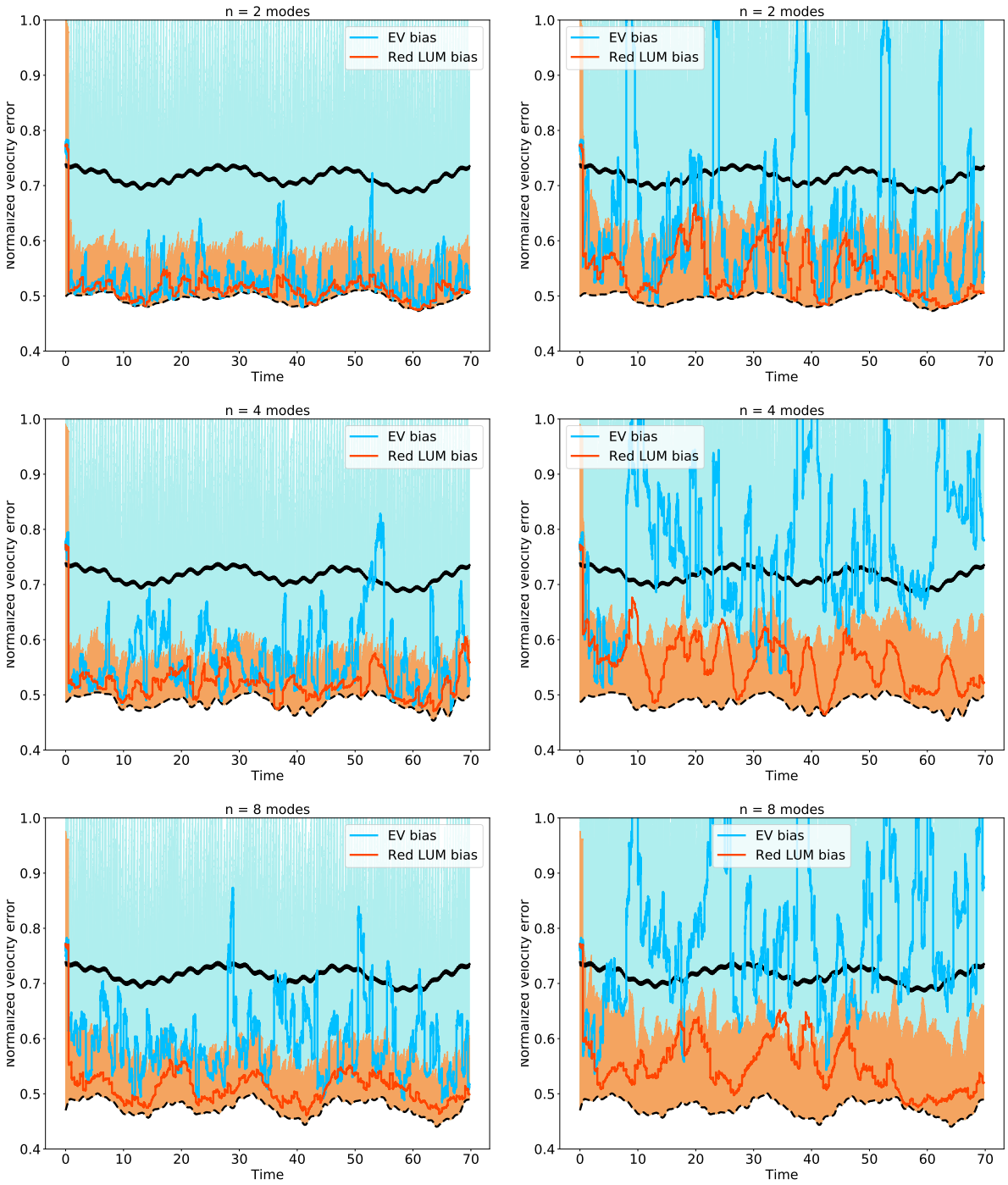


Figure 6: Global velocity prediction normalized error (after the learning period) for the wake flow at Reynolds number 300 in the observation cases 2 (9 observation points, left) and 3 (1 observation point inside the recirculation zone, right) with, from top to bottom $n = 2, 4$ and 8 modes for our POD-ROM method (orange) and the stochastic state-of-the-art POD-ROM method (light blue). The shaded colors (light orange and light blue) correspond to the respective estimated *a posteriori* standard deviations. The dashed black line at the bottom is the POD truncation error. The solid black line at the top is the error obtained by setting all temporal modes $b_i$ to 0, i.e. keeping only the time averaged velocity $\overline{v}$.

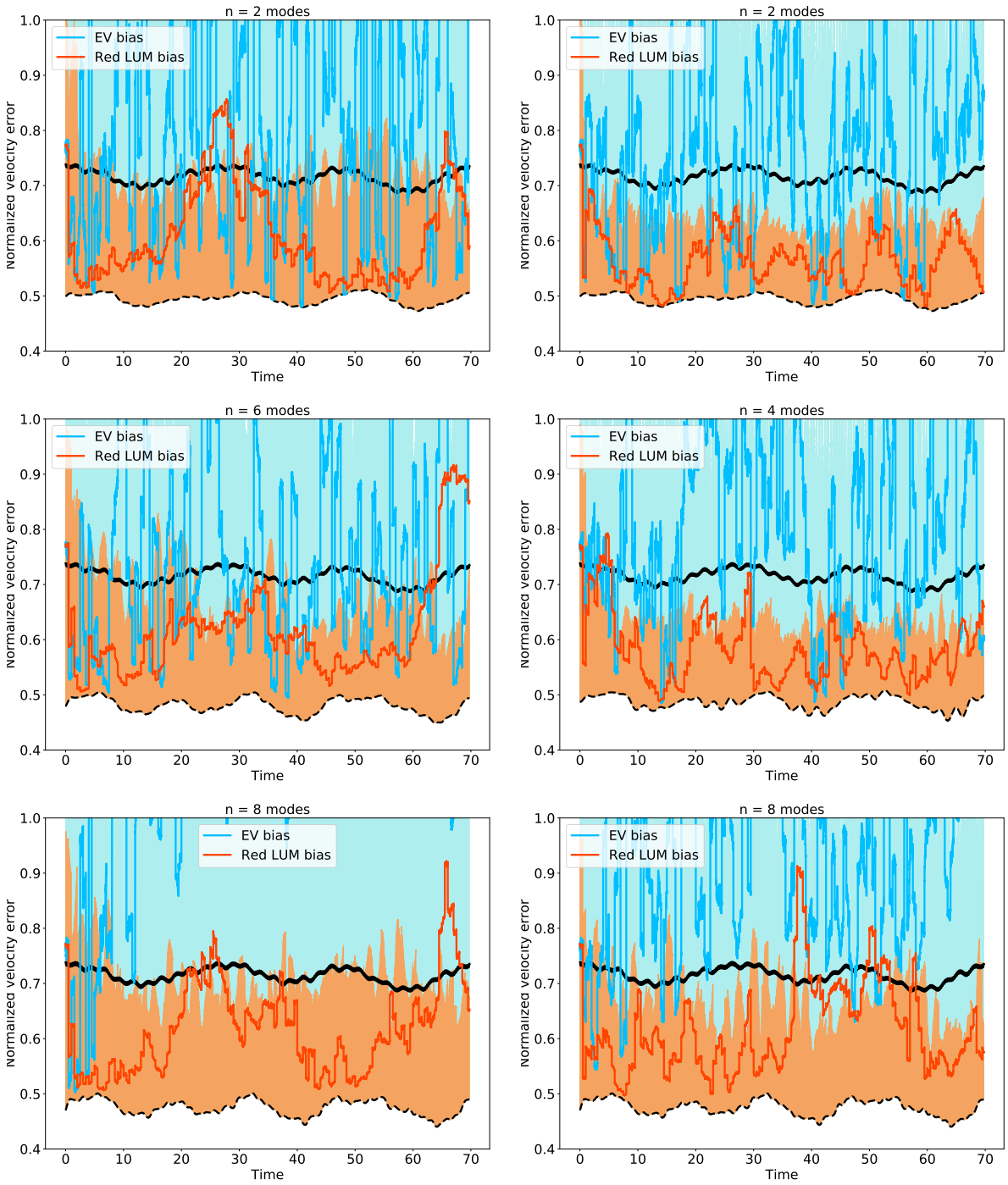


Figure 7: Global velocity prediction normalized error (after the learning period) for the wake flow at Reynolds 300 in the observation cases 4 (1 point outside the wake, left) and 5 (1 observation point far downstream, right) with, from top to bottom $n = 2, 4$ and 8 modes for our POD-ROM method (orange) and the stochastic state-of-the-art POD-ROM method (light blue). The shaded colors (light orange and light blue) correspond to the respective estimated *a posteriori* standard deviations. The dashed black line at the bottom is the POD truncation error. The solid black line at the top is the error obtained by setting all temporal modes $b_i$ to 0, i.e. keeping only the time averaged velocity $\overline{v}$.

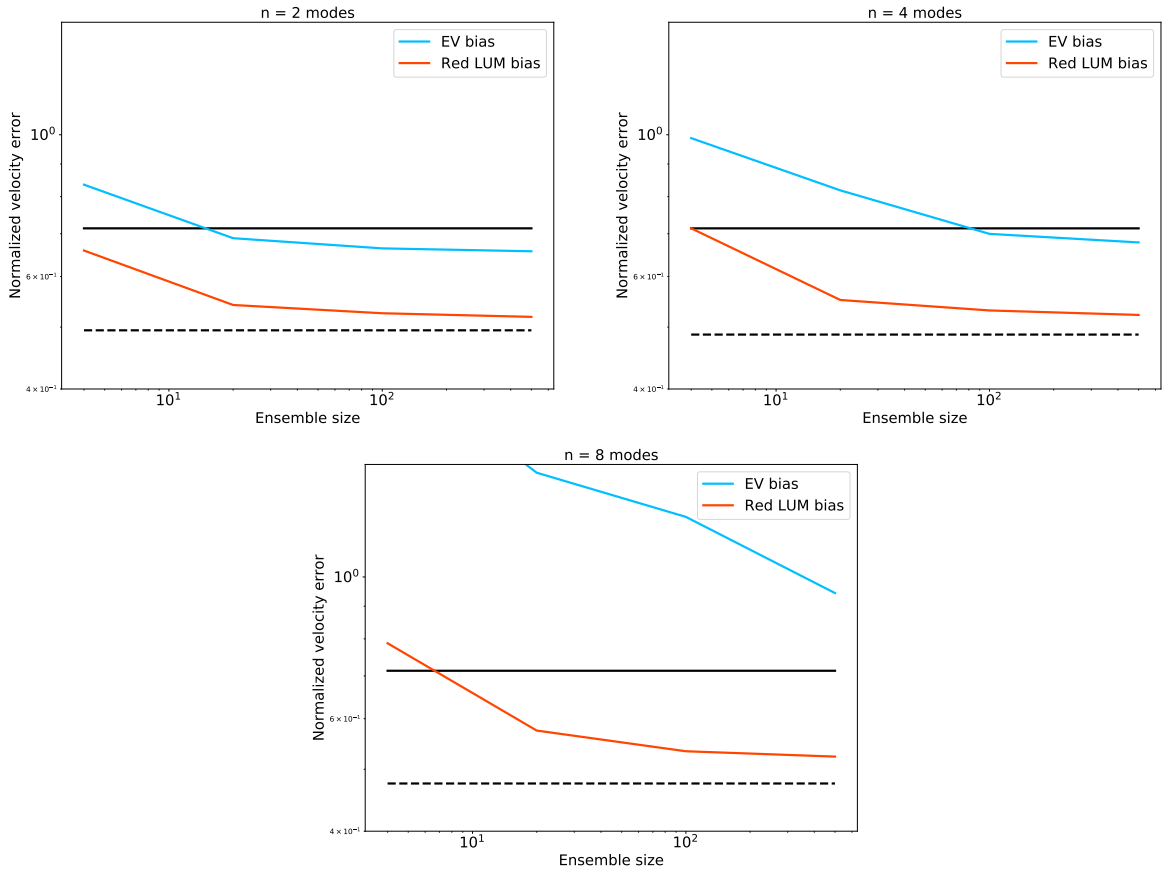


Figure 8: Loglog plots of the global velocity prediction normalized error (averaged on a time window of about 10 vortex shedding cycles, beginning about 2 vortex shedding cycles after the learning period) as a function of the ensemble size $N_p$ for the wake flow at Reynolds 300 with, from left to right and from top to bottom $n = 2, 4$ and 8 modes for our POD-ROM method (orange) and the stochastic state-of-the-art POD-ROM method (light blue). The dashed black line at the bottom is the POD truncation error. The solid black line at the top is the error obtained by setting all temporal modes $b_i$ to 0, i.e. keeping only the time averaged velocity $\overline{v}$.

MATLAB code used for this study. We have now implemented the POD-ROM building algorithm in C++ using the OpenFOAM-based library ITHACA-FV [45]. Our C++ code is not parallelized yet. However, a run now takes minutes only on a laptop, except for the POD decomposition – the velocity temporal covariance computation in particular. This step can be very long for a large dataset (here, about $10^4$ velocity snapshots of large dimensions).

The on-line data assimilation algorithm runs in a few minutes (about real-time) on a laptop computer, with a non-parallelized Python implementation. The time between two assimilation steps (0.5 s) is much larger than the time step used for the ROM time integration. Therefore, most of the CPU cost is from the ROM time integration. Between two assimilation steps, each realization is simulated independently. So, the CPU cost varies approximately linearly with the ensemble size, and future code parallelization could be done easily. The CPU cost also scales with the number of coefficients of our ROM, says $O(n^3)$. In practice, to choose the value of the ROM dimension $n$, we can target a specific solution accuracy. The eigenvalues of the POD decomposition express the amount of solution energy that can be encompassed for a specific value of $n$. A considerable speed-up should be expected when converting the online data assimilation code to C++, parallelizing the code, and using a better time integration scheme (adapted to stochastic differential equations). At term, the online algorithm should allow for measurements to be assimilated on the fly.

## 7. CONCLUSION

In this paper, we have presented a novel, accurate yet efficient method to estimate and predict a flow velocity field in the entire space from sparse measurements. Our algorithm is based on a stochastic, low-dimensional system built using physics equations and simulated DNS data, enabling the coupling with real-time measurements. Comparisons with state-of-the-art approaches have shown a remarkable improvement. A single measurement point was enough to accurately predict the unsteady velocity field in the whole 3-dimensional space with our method. The particle filtering with models under location uncertainty has shown an excellent performance in treating particle impoverishment. We believe that the data assimilation strategy used in this work can be extended to other fluid models and real-time fluid data assimilation.

Naturally, as a continuation of this work, we will perform some tests with actual particle velocimetry (PIV) measurements. Some enhancements to address more turbulent flows are also underway using the OpenFOAM-based Ithaca-FV library [45, 46]. Finally, we envisage the introduction of parametric ROMs and ROM construction from noisy and/or incomplete data.

## APPENDIX A. STOCHASTIC CALCULUS

Stochastic calculus is a field of Mathematics devoted to differential equations involving noises. Numerous tools exist in this framework, e.g., theoretical moments computation, statistical estimations, or simulation of both finite-

dimensional differential equations [47] and partial differential equations [48, 49, 50, 51]. Several notations co-exist in this field, in particular Stratonovich or Itō conventions. Here, we will mention the Ito notation – used in this paper – and some of its advantages. The discussion about relying whether on one notation or the other is recurrent in physics, but it is beyond the scope of this paper. For a more complete discussion on this subject, the reader can refer to sections 1.6 (pages 10-12) and 10.1 (pages 189-190) of [52]. For this short note, we highlight an important point: under appropriate assumptions, it is easy to switch from one notation to the other [47]. As a consequence, the most convenient form can be used to tackle a given issue.

In the Ito convention, time derivatives correspond to first-order forward-in-time differentiations. In particular, $\partial_t w_k$ in equation (3) and $\frac{\mathrm{d}b_i(t)}{\mathrm{d}t}$ in equation (5) stand for $\frac{w_k(\boldsymbol{x},t+dt)-w_k(\boldsymbol{x},t)}{dt}$ and $\frac{b_i(t+dt)-b_i(t)}{dt}$ respectively, both with an infinitesimal time step $dt$. A main advantage of this formalism is the straightforward simulation method associated to this forward-in-time differentiation. Other stochastic calculus conventions need much more complicate integration schemes. Moreover, the Itō notation more explicitly identifies and separates the zero-mean noise terms – $\boldsymbol{v}' \cdot \boldsymbol{\nabla} w_k$ in equation (3) and $\tilde{\alpha}^R_{pik}\ b_p(t)\dot{\beta}_k(t)$ in equation (5) – from the other effects induced by randomization of equations (e.g., diffusion, noise-induced drift). A consequence is a much easier derivation of the evolution law of moments.

# APPENDIX B. ESTIMATIONS OF THE REDUCED LOCATION UNCERTAINTY MODEL'S COEFFICIENTS

We here specify the definition of each term of the reduced location uncertainty model (5), initially proposed by [25]. In this appendix, $(\boldsymbol{\zeta},\boldsymbol{\xi}) \triangleq \int_\Omega \boldsymbol{\zeta}\cdot\boldsymbol{\xi}$ denotes the scalar product of the vectorial functions $\boldsymbol{\zeta}$ and $\boldsymbol{\xi}$. $\boldsymbol{\mathcal{P}}$ denotes the non-local Leray operator $\boldsymbol{\mathcal{P}} = \mathbb{I}_d - \boldsymbol{\nabla}\boldsymbol{\nabla}^T\Delta^{-1}$. This projection, which requires the resolution of a Poisson equation is used to simplify the fluid mechanics equations by removing the pressure term.

## Appendix B.1 Time down-sampling rate

Under the LU Navier-Stokes model hypothesis, the unresolved term of the velocity field $\boldsymbol{v}'$ corresponds to a noise uncorrelated in time. This assumption is consistent with the fact that the higher-order coefficients of the reduced-order solution often tend to have a shorter correlation time in fluid dynamics systems. However, in practice, this assumption is not found to be precisely accurate, and it is a recurrent issue for the data-driven modeling of systems combining fast and slowly evolving components [53, 54, 55, 56]. Consequently, a time down-sampling scheme is used to force the noise terms to be as uncorrelated as possible.

Assuming that the spatially averaged covariance function has a Gaussian form with a standard deviation equal to the correlation time $\tau$, a simple expression allows us to compute this correlation time. For a given velocity correlation matrix $C^v_{ij} = (\boldsymbol{v}_{\mathrm{obs}}(\bullet,t_i),\boldsymbol{v}_{\mathrm{obs}}(\bullet,t_j))$ (evaluated during the POD decomposition), the following unresolved velocity

| Physical meaning | Full-order term | ROM term |
|---|---|---|
| Molecular viscous dissipation | $\boldsymbol{L} = \frac{1}{Re}\Delta$ | $l_{pi} = \left(\boldsymbol{\phi}_i, \boldsymbol{L}(\boldsymbol{\phi}_p)\right)$ |
| Usual advection | $\boldsymbol{C}(\boldsymbol{w}, \bullet) = -\left(\boldsymbol{w}\cdot\boldsymbol{\nabla}\right)$ | $c_{pqi} = \left(\boldsymbol{\phi}_i, \boldsymbol{\mathcal{P}C}(\boldsymbol{\phi}_p, \boldsymbol{\phi}_q)\right)$ |
| Turbulent diffusion + Advecting velocity correction | $\boldsymbol{F}_{\text{dif}} = \boldsymbol{\nabla}\cdot\left(\frac{1}{2}\boldsymbol{a}\boldsymbol{\nabla}\bullet\right) + \frac{1}{2}\left(\boldsymbol{\nabla}\cdot\boldsymbol{a}\right)\boldsymbol{\nabla}$ | $f_{pi} = \left(\boldsymbol{\phi}_i, \boldsymbol{\mathcal{P}F}(\boldsymbol{\phi}_p)\right)$ |
| Absolute diffusivity | $\boldsymbol{a}(\boldsymbol{x}) = \Delta t\ \overline{\boldsymbol{v}'_{\text{obs}}\left(\boldsymbol{v}'_{\text{obs}}\right)^T}(\boldsymbol{x})$ $\boldsymbol{v}'_{\text{obs}} = \boldsymbol{v}_{\text{obs}} - \widetilde{\Pi}_\phi[\boldsymbol{v}_{\text{obs}}]$ | |

Table B.2: Deterministic terms and noise-indueced terms of the reduced location uncertainty model.

correlation matrix can be computed:

$$C^{v'}_{ij} = \left(\boldsymbol{v}'_{\text{obs}}(\bullet, t_i), \boldsymbol{v}'_{\text{obs}}(\bullet, t_j)\right) = C^{v}_{ij} - \sum_{k=1}^{n} b^{\text{obs}}_k(t_i) b^{\text{obs}}_k(t_j), \quad 0 \leqslant i, j \leqslant N-1, \tag{B.1}$$

and its associated stationary covariance function

$$\text{Cov}_s(t_p) = \frac{1}{N-p}\sum_{q=0}^{N-1-p} C^{v'}_{q,q+p}, \quad 0 \leqslant p \leqslant N-1. \tag{B.2}$$

We then estimate the correlation time as follow:

$$\widehat{\tau} \quad = \quad \sqrt{2\frac{\overline{\text{Cov}_s^2}}{\left(\frac{\Delta \text{Cov}_s}{\Delta t}\right)^2}}, \tag{B.3}$$

using a forward Euler temporal discretization of the stationary covariance:

$$\frac{\Delta \text{Cov}_s}{\Delta t}(t_p) \quad \triangleq \quad \frac{Cov_s(t_p + \Delta t) - Cov_s(t_p)}{\Delta t}, \quad 0 \leqslant p \leqslant N-1. \tag{B.4}$$

Before computing the estimations presented in the following of this appendix, we use this estimated correlation time $\widehat{\tau}$ to down-sample both the entire DNS velocity dataset and the observed coefficients of the reduced order solution, leaving us with a time step $\Delta t \approx \widehat{\tau}$.

## Appendix B.2 Deterministic terms and noise-induced terms

The deterministic coefficients of the proposed POD-ROM are summarized in table B.2.

## Appendix B.3 Noise correlations estimation

Let us now focus on noise statistics estimation.

For any function $\boldsymbol{\xi}$, we introduce the linear functional:

$$K_{jq}[\boldsymbol{\xi}] \triangleq \left(\boldsymbol{\phi}_j, -\boldsymbol{\mathcal{P}}\left[(\boldsymbol{\xi}\cdot\boldsymbol{\nabla})\,\boldsymbol{\phi}_q\right] + \delta_{q0}\,\nu\Delta\boldsymbol{\xi}\right), \quad 1 \leqslant j \leqslant n, \quad 0 \leqslant q \leqslant n. \tag{B.5}$$

Using this notation, the noise's covariance can be estimated as follows:

$$\widehat{\Sigma^{\boldsymbol{\alpha}}_{pi,qj}} = \frac{\Delta t}{\lambda^{\text{obs}}_p}\, K_{jq}\left[\overline{b^{\text{obs}}_p \left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)'' \boldsymbol{v}'_{\text{obs}}}\right], \quad 1 \leqslant i,j \leqslant n, \quad 0 \leqslant p,q \leqslant n, \tag{B.6}$$

where $b^{\text{obs}}_0 = \lambda^{\text{obs}}_0 = 1$ and for $1 \leqslant i \leqslant n$,

$$b^{\text{obs}}_i = (\boldsymbol{\phi}_i, \boldsymbol{v}_{\text{obs}}), \tag{B.7}$$

$$\lambda^{\text{obs}}_i = \overline{(b^{\text{obs}}_i)^2}, \tag{B.8}$$

$$\left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)'' = \left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)' - \overline{\left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)'}, \tag{B.9}$$

$$\left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)' = \left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right) - \left(\left(\boldsymbol{b}^{\text{obs}}\right)^T (\boldsymbol{l}+\boldsymbol{f})_{\bullet i} + \left(\boldsymbol{b}^{\text{obs}}\right)^T \boldsymbol{c}_{\bullet\bullet i}\, \boldsymbol{b}^{\text{obs}}\right), \tag{B.10}$$

$$\left(\frac{\Delta b^{\text{obs}}_i}{\Delta t}\right)(t_k) = \frac{b^{\text{obs}}_i(t_k+\Delta t) - b^{\text{obs}}_i(t_k)}{\Delta t}, \quad 0 \leqslant k \leqslant N-1. \tag{B.11}$$

To ensure the noise's covariance matrix to have the desired symmetric non-negative structure, we only keep the symmetric part of the estimated tensor (B.6) and set its possible negative eigenvalues to zero. [25] prove the consistency of this estimator.

Then, we reduce the noise dimension through a tensorial PCA of $\widehat{\boldsymbol{\Sigma}^{\boldsymbol{\alpha}}}$, eventually only keeping the $n$ first eigenvectors. $\left(\tilde{\boldsymbol{\alpha}}^R_k\right)_k \in \mathbb{R}^{(n+1)\times n}$ are the matrix forms of the first $n$ eigenvectors (weighted by the corresponding eigenvalues' square roots). Since we consider a multiplicative noise and the temporal coefficients $b_i$ have various amplitudes $\sqrt{\lambda_i}$, the covariance matrix $\widehat{\boldsymbol{\Sigma}^{\boldsymbol{\alpha}}}$ is adequately re-normalized by the amplitudes $\sqrt{\lambda_i}$ before applying the PCA.

# APPENDIX C. DESIGN OF THE OBSERVATION MODEL

An observation model aims at representing the link between measured values and an observed state. Here the state is $\boldsymbol{b}$ and the observation model is represented in equation C.1

$$\boldsymbol{y} = \boldsymbol{\mathcal{H}}[\boldsymbol{v}] + \boldsymbol{\epsilon}_y = \underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{v}} + \boldsymbol{L}_F\dot{W}, \tag{C.1}$$

where $\boldsymbol{y}$ is the vector of the $M_y$ PIV measurements to assimilate, $\boldsymbol{\epsilon}_y = \boldsymbol{L}_F\dot{W}$ is the PIV measurement noise, $\dot{\boldsymbol{W}}$ is a vector of $M_{PIV}$ independent white noise (in discrete time), $\boldsymbol{L}_F{\boldsymbol{L}_F}^T$ is the covariance matrix of the PIV measurement noise, $\underline{\boldsymbol{v}}$ is the reshaped version of $\boldsymbol{v}$ as a single vector of $M \times d$ coefficients and $\underline{\boldsymbol{\mathcal{H}}}$ is a reshaped version of the linear operator $\boldsymbol{\mathcal{H}}$ as $M_y \times (M \times d)$ matrix. For instance, if $d = 3$, we have $\underline{\boldsymbol{v}}(t) =$

$(v_1(\boldsymbol{x}_1,t),\ldots,v_1(\boldsymbol{x}_M,t),v_2(\boldsymbol{x}_1,t),\ldots,v_2(\boldsymbol{x}_M,t),v_3(\boldsymbol{x}_1,t),\ldots,v_3(\boldsymbol{x}_M,t))^T$. The two-scale representation $\boldsymbol{v}(t)=\boldsymbol{w}(t)+\boldsymbol{\sigma}\dot{\boldsymbol{B}}(t)$ can be extended to the reshaped version of $\boldsymbol{v}$:

$$\underline{\boldsymbol{v}}=\underline{\boldsymbol{w}}+\underline{\boldsymbol{v}'}=\underline{\boldsymbol{\Phi}}\,\boldsymbol{b}+\underline{\underline{\boldsymbol{\sigma}}}\,\underline{\dot{\boldsymbol{B}}}, \tag{C.2}$$

with $\underline{\boldsymbol{w}}$ a reshaped vector of $(M\times d)$ coefficients, $\dot{\boldsymbol{B}}$ is a vector of $(M\times d)$ independent white noise (in discrete time), $\underline{\underline{\boldsymbol{\sigma}}}$ a matrix of dimension $(M\times d)\times(M\times d)$ and $\underline{\boldsymbol{\Phi}}$ the reshaped matrix $\underline{\boldsymbol{\Phi}}=\left(\underline{\boldsymbol{\phi}_0},\ldots,\underline{\boldsymbol{\phi}_n}\right)$ with dimension $(M\times d)\times n$. Finally, the observation model equation can be represented as the equation C.3

$$\boldsymbol{y} \;=\; \underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\Phi}}\boldsymbol{b}+\underline{\boldsymbol{\mathcal{H}}}\,\underline{\underline{\boldsymbol{\sigma}}}\,\underline{\dot{\boldsymbol{B}}}+\boldsymbol{L}_F\dot{W}. \tag{C.3}$$

The matrices $\underline{\boldsymbol{\mathcal{H}}}$ and $\boldsymbol{L}_F$ are unknown and must be estimated. The $\underline{\boldsymbol{\mathcal{H}}}$ matrix is a spatial transformation from one space to another and mathematically it executes a spatial filtering and a cropping on the velocity field. We decompose the two operations through two operators:

$$\underline{\boldsymbol{\mathcal{H}}}=\boldsymbol{H}_{\text{crop}}\boldsymbol{H}_{\text{PIV}}, \tag{C.4}$$

where $\boldsymbol{H}_{\text{crop}}$ is a rectangular matrix of 0 and 1 that defined the cropping of the PIV image and $\boldsymbol{H}_{\text{PIV}}$ is a matrix representing the PIV measurement and postprocessing. The $\boldsymbol{L}_F$ matrix is a quantification of uncertainty from the measures after post treatment. In order to estimate $\boldsymbol{H}_{\text{PIV}}$ and $\boldsymbol{L}_F$, we have to compare the PIV data with a reference accurate measurement method : the hot wire.

Let us first focus on the matrix $\boldsymbol{H}_{\text{PIV}}$. Since we focus on 2D2C PIV, that matrix should include the slicing of the 3D data and the selection of the horizontal velocity components through two matrices of 0 and 1, that we name $\boldsymbol{H}_{\text{2D}}$ and $\boldsymbol{H}_{\text{2C}}$ respectively:

$$\boldsymbol{H}_{\text{PIV}}=\boldsymbol{H}_{\text{2D}}\boldsymbol{H}_{\text{2C}}\boldsymbol{H}_{\text{Blur}}, \tag{C.5}$$

with $\boldsymbol{H}_{\text{Blur}}$ to encode the blurring induced by the PIV measurement. This blurring is approximated by an isotropic 3D spatial convolution on each velocity component:

$$v_k^{\text{PIV}}=h^S*v_k+\epsilon_k^{\text{PIV}}, \tag{C.6}$$

where $h^S$ is the spatial filter and $\epsilon^{\text{PIV}}$ the measurement noise. We consider an isotropic Gaussian filter.

$$h^S(x,y,z)=h^s(x)h^s(y)h^s(z), \tag{C.7}$$

with

$$h^s(x)=A_h\exp\left(-\tfrac{1}{2}\left(\frac{x}{\sigma_s^h}\right)^2\right). \tag{C.8}$$

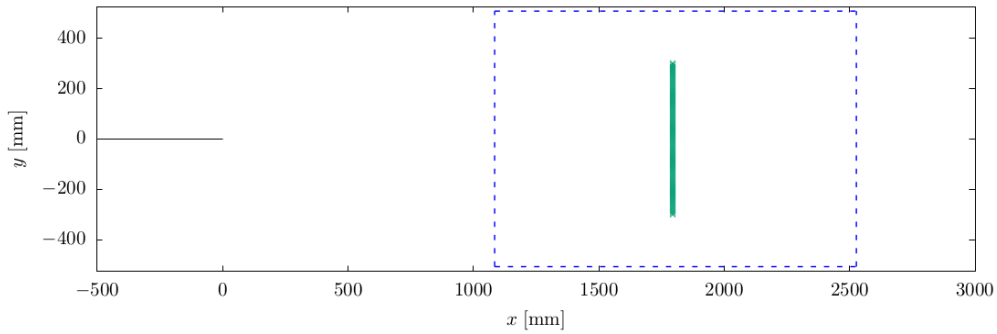


Figure C.9: Measure configuration plan of PIV and hot-wire.[57]

In Fourier space, the modulus of the filter is also a Gaussian function and can then be estimated as:

$$|\hat{h}^S| \approx \frac{|\hat{v}_k^{\mathrm{PIV}}|}{|\hat{v}_k|}. \tag{C.9}$$

Unfortunately, the hot wire data do not have a significant spatial extension. Accordingly, for our experimental data comparison we must focus on the temporal signatures. We will estimate a blurring temporal filter $h^t$ instead of the blurring spatial filter $h^s$. And subsequently, we will transform the temporal filter into a spatial filter using a Taylor assumption. Let us now estimate the blurring temporal filter $h^t$ from the experimental, keeping a Gaussian filter form.

Since the large-scale structures should be well reconstructed by the PIV, we can expect that the first Fourier mode is $\hat{h}^t(0) = 1$. This sets the amplitude of the filter. In Fourier space, the modulus of the filter is Gaussian and its logarithm writes:

$$\log\left(\frac{|\hat{v}_k^{\mathrm{PIV}}(f)|}{|\hat{v}_k(f)|}\right) \approx \log\left(|\hat{h}^t(f)|\right) = -\alpha f^2. \tag{C.10}$$

This gives a simple way to fit the last filter parameter: the standard deviation $\sigma_t^h = \sqrt{\frac{\alpha}{2\pi^2}}$. It is estimated with a simple linear regression in loglog plot.

The data used to estimate the matrices were presented in the work of [57]. The measure configuration is detailed by the Figure C.9, where the blue dotted line boundsthe PIV area estimation and the green line represents the hot wire converters position. The sampling frequency used in PIV is 500 Hertz and 6000 Hertz in hot wire. The hot wire data spectrum and the PIV data spectrum are compared in figure C.10. The Gaussian fitting of $\hat{h}^t$ is illustrated in figure C.11.

After that, the Taylor hypothesis is used again to transform the time filter $h^t$ into a space filter $h^s$ by the axis rescaling $x = Ut$ where $U$ is the mean flow velocity. Therefore, the space filter $h^s$ is a Gaussian function with a standard deviation $\sigma_x^h = U\sigma_t^h$.

Eventually, the filter $h^s$ is used to define the PIV blurring matrix $\boldsymbol{H}_{\mathrm{Blur}}$. According to equation (C.7), we can write the matrix $\boldsymbol{H}_{\mathrm{PIV}}$ as a reshaped version of a tensor product of 3 matrices representing the spatial smoothing

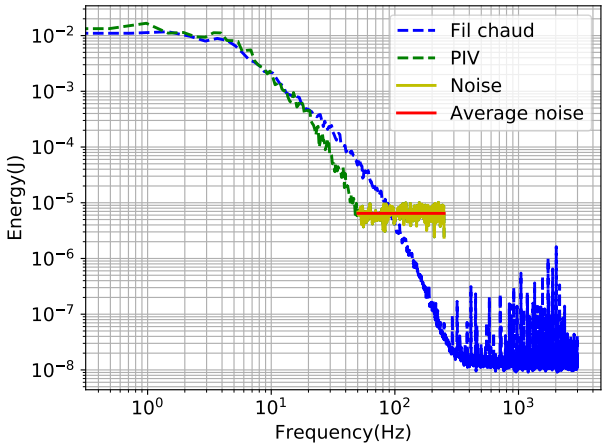


Figure C.10: Spectral analysis of PIV measurements versus hot-wire measurements.

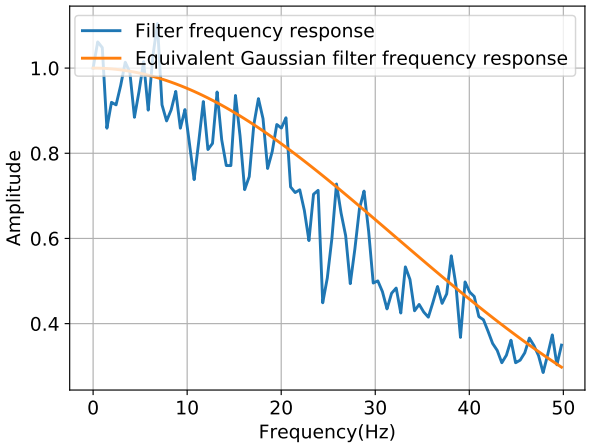


Figure C.11: Gaussian fitting of $|\hat{h}^t|$.

along $x$, $y$ and $z$:

$$\boldsymbol{H}_{\text{Blur}} = \underline{\boldsymbol{H}^x_{\text{PIV}} \otimes \boldsymbol{H}^y_{\text{PIV}} \otimes \boldsymbol{H}^z_{\text{PIV}}}. \tag{C.11}$$

Each of this matrix is defined in the same way though the filter $h^s$. In order to restrict the number of non-zero coefficient of those huge matrices, we crop the filter after one standard deviation $h(x) = h^s(x)\mathbb{1}_{\{|x|\leqslant\sigma_x\}}$. and

$$\boldsymbol{H}^x_{\text{PIV}} = \begin{bmatrix} h(-n_h\Delta x) & \dots & h(0) & \dots & h(n_h\Delta x) & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & h(-n_h\Delta x) & \dots & h(0) & \dots & h(n_h\Delta x) & \dots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & \dots & h(-n_h\Delta x) & \dots & h(0) & \dots & h(n_h\Delta x) \end{bmatrix}, \tag{C.12}$$

where $n_h$ is the floor function of $\sigma_x/\Delta x$.

This finally leads to the observation matrix

$$\underline{\boldsymbol{\mathcal{H}}} = \boldsymbol{H}_{\text{crop}}\boldsymbol{H}_{\text{PIV}} = \boldsymbol{H}_{\text{crop}}\boldsymbol{H}_{\text{2D}}\boldsymbol{H}_{\text{2C}}\underline{\boldsymbol{H}^x_{\text{PIV}} \otimes \boldsymbol{H}^y_{\text{PIV}} \otimes \boldsymbol{H}^z_{\text{PIV}}}. \tag{C.13}$$

On top of the observation matrix, we need to estimate the matrix $\boldsymbol{L}_F$. To do so, we use the noisy part of the PIV spectrum (see figure C.10). To simplify we assume that the noise $\boldsymbol{\epsilon}_y$ is white in time and in space. We estimate the noise variance $\sigma^2_\epsilon$ from the PIV spectrum. With the cropping of this PIV image, this leads to:

$$\boldsymbol{L}_{PIV} = \boldsymbol{H}_{\text{crop}}\left(\sigma_\epsilon \mathbb{I}_d\right) = \sigma_\epsilon \boldsymbol{H}_{\text{crop}}. \tag{C.14}$$

## APPENDIX D. LOG-LIKELIHOOD EXPRESSION

The log-likelihood is necessary in order to assimilate the PIV measurements. Since the observation model (C.3) is linear with an additive Gaussian noise $\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\sigma}}\,\underline{\dot{\boldsymbol{B}}} + \boldsymbol{L}_F\dot{W}$, the log-likelihood is a quadratic function of the state $\boldsymbol{b}$:

$$p\left(\boldsymbol{y}(t)|\boldsymbol{b}(t)\right) \propto \exp\left(-\tfrac{1}{2}\|\boldsymbol{y}(t) - \underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\Phi}}\,\boldsymbol{b}(t)\|^2_{\boldsymbol{\Sigma}^{-1}}\right), \tag{D.1}$$

$$\propto \exp\left(-\tfrac{1}{2}\left(\boldsymbol{y}^T(t)\boldsymbol{\Sigma}^{-1}\boldsymbol{y}(t) + \boldsymbol{y}^T(t)\boldsymbol{B}\boldsymbol{b}(t) + \boldsymbol{b}^T(t)\boldsymbol{A}\boldsymbol{b}(t)\right)\right), \tag{D.2}$$

with $\boldsymbol{\Sigma}$ the covariance matrix of the whole additive noise $\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\sigma}}\,\underline{\dot{\boldsymbol{B}}} + \boldsymbol{L}_F\dot{W}$

$$\boldsymbol{\Sigma} = \left(\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\sigma}}\right)\left(\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\sigma}}\right)^T + \boldsymbol{L}_F\boldsymbol{L}_F^T, \tag{D.3}$$

and

$$\boldsymbol{A} = -\tfrac{1}{2}\left(\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\Phi}}\right)^T\boldsymbol{B}, \tag{D.4}$$

$$\boldsymbol{B} = \boldsymbol{\Sigma}^{-1}\left(\underline{\boldsymbol{\mathcal{H}}}\,\underline{\boldsymbol{\Phi}}\right). \tag{D.5}$$

In practice, $\underline{\underline{\boldsymbol{\sigma}}}$ and $\underline{\underline{\boldsymbol{\sigma}}}\,\underline{\underline{\boldsymbol{\sigma}}}^T$ are huge and cannot be even memorize. Depending on the PIV cropping, $\boldsymbol{\Sigma}$ can be huge as well which prevents the computation of its inverse. Therefore, for the computation of $\boldsymbol{\Sigma}$ and its inverse only, we estimate $\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)^T$ with the following time average:

$$\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)^T = \overline{\left(\boldsymbol{\mathcal{H}}[\boldsymbol{v}']\right)\left(\boldsymbol{\mathcal{H}}[\boldsymbol{v}']\right)^T} = \boldsymbol{H}_{\text{crop}}\ \overline{\left(\boldsymbol{H}_{\text{PIV}}\ \underline{\boldsymbol{v}'}\right)\left(\boldsymbol{H}_{\text{PIV}}\ \underline{\boldsymbol{v}'}\right)^T}\ \boldsymbol{H}_{\text{crop}}^T. \tag{D.6}$$

Moreover, we neglect the spatial correlation in $\boldsymbol{\mathcal{H}}[\boldsymbol{v}']$. This makes $\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)\left(\underline{\boldsymbol{\mathcal{H}}}\ \underline{\underline{\boldsymbol{\sigma}}}\right)^T$ and then $\boldsymbol{\Sigma}$ quasi-diagonal and enables the computation of $\boldsymbol{\Sigma}^{-1}$.

Besides, the factor $\exp\left(-\frac{1}{2}\boldsymbol{y}(t)^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y}(t)\right)$ in equation (D.2) is the same for every realizations $\boldsymbol{b}^{(j)}$. So, it will disappear in the state distribution normalization and we do not need to compute it:

$$p\left(\boldsymbol{y}(t)|\boldsymbol{b}(t)\right) \quad \propto \quad \exp\left(\boldsymbol{y}^T(t)\boldsymbol{B}\boldsymbol{b}(t) + \boldsymbol{b}^T(t)\boldsymbol{A}\boldsymbol{b}(t)\right). \tag{D.7}$$

## SAMPLE CREDIT AUTHOR STATEMENT

**Valentin Resseguier:** Conceptualization, Methodology, Software, Visualization, Writing. **Matheus Ladvig:** Software, Investigation, Formal analysis. **Dominique Heitz:** Data Curation, Methodology.

## ACKNOWLEDGMENTS

We warmly thank Pranav Chandramouli for the generation of the three-dimensional wake flow direct numerical simulation (DNS) data and Romain Schuster for providing the experimental data. We also thank Darryl D. Holm, Dan Crisan, Igor Shevchenko and above all Wei Pan for the insightful discussions.

## FUNDING

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. Two of the authors have been employed by a private company (Scalian) when performing this work.

## REFERENCES

[1] N. B. Erichson, L. Mathelin, Z. Yao, S. L. Brunton, M. W. Mahoney, J. N. Kutz, Shallow neural networks for fluid flow reconstruction with limited sensors, Proceedings of the Royal Society A 476 (2238) (2020) 20200097.

[2] J. P. Thomas, E. H. Dowell, K. C. Hall, Three-dimensional transonic aeroelasticity using proper orthogonal decomposition-based reduced-order models, Journal of Aircraft 40 (3) (2003) 544–551.

[3] C. Braud, D. Heitz, G. Arroyo, L. Perret, J. Deleville, J. Bonnet, Low-dimensional analysis, using POD, for two mixing layer-wake interactions, International Journal of Heat and Fluid Flow 3 (3) (2004) 351–363.

[4] L. Fick, Y. Maday, A. T. Patera, T. Taddei, A stabilized pod model for turbulent flows over a range of reynolds numbers: Optimal parameter sampling and constrained projection, Journal of Computational Physics 371 (2018) 214–243.

[5] A. Majda, Statistical energy conservation principle for inhomogeneous turbulent dynamical systems, Proceedings of the National Academy of Sciences 112 (29) (2015) 8937–8941.

[6] T. Sapsis, Attractor local dimensionality, nonlinear energy transfers and finite-time instabilities in unstable dynamical systems with applications to two-dimensional fluid flows, Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences 469 (2153). doi:10.1098/rspa.2012.0550.

[7] T. Sapsis, A. Majda, Blending modified Gaussian closure and non-Gaussian reduced subspace methods for turbulent dynamical systems, Journal of Nonlinear Science 23 (6) (2013) 1039–1071.

[8] N. Aubry, P. Holmes, J. Lumley, E. Stone, The dynamics of coherent structures in the wall region of a turbulent boundary layer, J. Fluid Mech. 192 (1988) 115–173.

[9] W. Cazemier, R. Verstappen, A. Veldman, Proper orthogonal decomposition and low-dimensional models for driven cavity flows, Phys. Fluids 10 (7) (1998) 1685–1699.

[10] Z. Wang, I. Akhtar, J. Borggaard, T. Iliescu, Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison, Computer Methods in Applied Mechanics and Engineering 237 (2012) 10–26.

[11] M. Buffoni, S. Camarri, A. Iollo, M. V. Salvetti, Low-dimensional modelling of a confined three-dimensional wake flow, Journal of Fluid Mechanics 569 (2006) 141–150.

[12] L. Cordier, B. R. Noack, G. Tissot, G. Lehnasch, J. Delville, M. Balajewicz, G. Daviller, R. K. Niven, Identification strategies for model-based control, Experiments in fluids 54 (8) (2013) 1580.

[13] X. Xie, M. Mohebujjaman, L. G. Rebholz, T. Iliescu, Data-driven filtered reduced order modeling of fluid flows, SIAM Journal on Scientific Computing 40 (3) (2018) B834–B857.

[14] I. M. Navon, Data assimilation for numerical weather prediction: a review, in: Data assimilation for atmospheric, oceanic and hydrologic applications, Springer, 2009, pp. 21–65.

[15] M. Couplet, C. Basdevant, P. Sagaut, Calibrated reduced-order pod-galerkin system for fluid flow modelling, Journal of Computational Physics 207 (1) (2005) 192–220.

[16] J. D'adamo, N. Papadakis, E. Memin, G. Artana, Variational assimilation of pod low-order dynamical systems, Journal of Turbulence 8 (9) (2007) 1–22.

[17] G. Artana, A. Cammilleri, J. Carlier, E. Mémin, Strong and weak constraint variational assimilations for reduced order fluid flow modeling, J. Comp. Phys 231 (8) (2012) 3264–3288.

[18] E. Mémin, Fluid flow dynamics under location uncertainty, Geophysical & Astrophysical Fluid Dynamics 108 (2) (2014) 119–146.

[19] V. Resseguier, E. Mémin, B. Chapron, Geophysical flows under location uncertainty, part I random transport and general models, Geophysical & Astrophysical Fluid Dynamics 111 (3) (2017) 149–176.

[20] R. Mikulevicius, B. Rozovskii, Stochastic Navier–Stokes equations for turbulent flows, SIAM Journal on Mathematical Analysis 35 (5) (2004) 1250–1310.

[21] V. Resseguier, E. Mémin, B. Chapron, Geophysical flows under location uncertainty, part II quasi-geostrophy and efficient ensemble spreading, Geophysical & Astrophysical Fluid Dynamics 111 (3) (2017) 177–208.

[22] V. Resseguier, W. Pan, B. Fox-Kemper, Data-driven versus self-similar parameterizations for stochastic advection by lie transport and location uncertainty, Nonlinear Processes in Geophysics 27 (2) (2020) 209–234.

[23] V. Resseguier, L. Li, G. Jouan, P. Dérian, E. Mémin, C. Bertrand, New trends in ensemble forecast strategy: uncertainty quantification for coarse-grid computational fluid dynamics, Archives of Computational Methods in Engineering (2020) 1–82.

[24] B. Chapron, P. Dérian, E. Mémin, V. Resseguier, Large-scale flows under location uncertainty: a consistent stochastic framework, Quarterly Journal of the Royal Meteorological Society 144 (710) (2018) 251–260.

[25] V. Resseguier, A. M. Picard, E. Mémin, B. Chapron, Quantifying truncation-related uncertainties in unsteady fluid dynamics reduced order models, Submitted to SIAM/ASA Journal on Uncertainty Quantification. https://hal.archives-ouvertes.fr/hal-03169957.

[26] V. Resseguier, E. Mémin, D. Heitz, B. Chapron, Stochastic modelling and diffusion modes for proper orthogonal decomposition models and small-scale flow analysis, Journal of Fluid Mechanics 826 (2017) 888–917.

[27] A. Doucet, A. Johansen, A tutorial on particle filtering and smoothing: Fifteen years later, Handbook of Nonlinear Filtering 12 (2009) 656–704.

[28] A. Doucet, N. De Freitas, N. Gordon, Sequential Monte Carlo methods in practice, Springer, 2001.

[29] N. Kantas, A. Beskos, A. Jasra, Sequential monte carlo methods for high-dimensional inverse problems: A case study for the navier–stokes equations., SIAM/ASA Journal on Uncertainty Quantification 2.1 (2014) 464–489.

[30] P. Rebeschini, R. Van Handel, et al., Can local particle filters beat the curse of dimensionality?, The Annals of Applied Probability 25 (5) (2015) 2809–2866.

[31] A. Beskos, D. Crisan, A. Jasra, K. Kamatani, Y. Zhou, A stable particle filter for a class of high-dimensional state-space models, Advances in Applied Probability 49 (1) (2017) 24–48.

[32] C. Cotter, D. Crisan, D. D. Holm, W. Pan, I. Shevchenko, A particle filter for stochastic advection by lie transport: A case study for the damped and forced incompressible two-dimensional euler equation, SIAM/ASA Journal on Uncertainty Quantification 8 (4) (2020) 1446–1492.

[33] A. Farchi, M. Bocquet, Comparison of local particle filters and new implementations., Nonlinear Processes in Geophysics 25 (4).

[34] F.-X. Le Dimet, O. Talagrand, Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects, Tellus A 38 (2) (1986) 97–110.

[35] J. S. Liu, Monte Carlo strategies in scientific computing, Springer Science & Business Media, 2008.

[36] A. Doucet, X. Wang, Monte carlo methods for signal processing: a review in the statistical signal processing context, IEEE Signal Processing Magazine 22 (6) (2005) 152–170.

[37] S. Laizet, E. Lamballais, High-order compact schemes for incompressible flows: a simple and efficient method with the quasi-spectral accuracy, J. Comp. Phys. 228 (15) (2009) 5989–6015.

[38] J. Berner, S.-Y. Ha, J. Hacker, A. Fournier, C. Snyder, Model uncertainty in a mesoscale ensemble prediction system: Stochastic versus multiphysics representations, Monthly Weather Review 139 (6) (2011) 1972–1995.

[39] C. Franzke, T. O'Kane, J. Berner, P. Williams, V. Lucarini, Stochastic climate theory and modeling, Wiley Interdisciplinary Reviews: Climate Change 6 (1) (2015) 63–78.

[40] G. Gottwald, J. Harlim, The role of additive and multiplicative noise in filtering complex dynamical systems, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science 469 (2155) (2013) 20130096.

[41] L. Mitchell, G. Gottwald, Data assimilation in slow-fast systems using homogenized climate models, Journal of the atmospheric sciences 69 (4) (2012) 1359–1377.

[42] R. Kikuchi, T. Misaka, S. Obayashi, International journal of computational fluid dynamics real-time prediction of unsteady flow based on pod reduced-order model and particle filter, International Journal of Computational Fluid Dynamics 30 (4) (2016) 285–306.

[43] A. Trevisan, F. Uboldi, Assimilation of standard and targeted observations within the unstable subspace of the observation-analysis-forecast cycle system, Journal of the atmospheric sciences 61 (1) (2004) 103–113.

[44] G.-H. Ng, D. McLaughlin, D. Entekhabi, A. Ahanin, The role of model dynamics in ensemble Kalman filter performance for chaotic systems, Tellus A 63 (5) (2011) 958–977.

[45] G. Stabile, S. Hijazi, A. Mola, S. Lorenzi, G. Rozza, POD-Galerkin reduced order methods for CFD using finite volume discretisation: vortex shedding around a circular cylinder, Communications in Applied and Industrial Mathematics 8 (1) (2017) 210–236.

[46] G. Stabile, G. Rozza, Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible navier–stokes equations, Computers & Fluids 173 (2018) 273–284.

[47] B. Oksendal, Stochastic differential equations, Spinger-Verlag, 1998.

[48] H. Kunita, Stochastic flows and stochastic differential equations, Vol. 24, Cambridge university press, 1997.

[49] G. Da Prato, J. Zabczyk, Stochastic Equations in Infinite Dimensions, Encyclopedia of Mathematics and its Applications, Cambridge University Press, 1992.

[50] C. Prévôt, M. Röckner, A concise course on stochastic partial differential equations, Vol. 1905, Springer, 2007.

[51] M. Choi, T. Sapsis, G. Karniadakis, On the equivalence of dynamically orthogonal and bi-orthogonal methods: Theory and numerical simulations, Journal of Computational Physics 270 (2014) 1–20.

[52] V. Resseguier, Mixing and fluid dynamics under location uncertainty, Ph.D. thesis, Rennes 1 university (2017).

[53] R. Azencott, A. Beri, A. Jain, I. Timofeyev, Sub-sampling and parametric estimation for multiscale dynamics, Communications in Mathematical Sciences 11 (4) (2013) 939–970.

[54] R. Azencott, A. Beri, I. Timofeyev, Adaptive sub-sampling for parametric estimation of Gaussian diffusions, Journal of Statistical Physics 139 (6) (2010) 1066–1089.

[55] A. Papavasiliou, G. Pavliotis, A. Stuart, Maximum likelihood drift estimation for multiscale diffusions, Stochastic Processes and their Applications 119 (10) (2009) 3173–3210.

[56] G. Pavliotis, A. Stuart, Parameter estimation for multiscale diffusions, Journal of Statistical Physics 127 (4) (2007) 741–781.

[57] R. Schüster, Développement d'une méthode de mesure basée image pour caractériser en grande taille les flux d'air intérieurs, Ph.D. thesis, Université de Rennes I (2019).