Data driven estimation of fluid flows: long-term prediction of velocity fields using machine learning

Pierre Dubois, Thomas Gomez, Laurent Planckaert, Laurent Perret

Association Aéronautique et Astronautique de France

55th 3AF International Conference on Applied Aerodynamics
12 — 14 April 2021, Poitiers – France

FP06-AERO2020+1-dubois

Data driven estimation of fluid flows: long-term prediction of velocity fields using machine learning

Pierre Dubois[(1)], Thomas Gomez [(1)], Laurent Planckaert [(1)] and Laurent Perret [(2)]
[(1)] Univ. Lille, CNRS, ONERA, Arts et Metiers Institute of Technology, Centrale Lille
UMR 9014 - LMFL - Laboratoire de Mécanique des fluides de Lille
Kampé de Fériet, F-59000 Lille, France
[(2)]Centrale Nantes, LHEEA UMR CNRS 6598, Nantes, France

ABSTRACT

This paper gives a framework for the data-driven estimation of an unsteady fluid flow field. The strategy combines machine learning tools for the reduction, the reconstruction and the prediction of the considered system. The reduction is performed by linear autoencoding while support vector regression and dynamical mode decomposition are respectively used as reconstruction and prediction models. Starting from an initial condition, reconstructions are frequently assimilated to update erroneuous predictions. The procedure is tested on four cases with increasing complexity and robustness is assessed through training and testing errors. Quantitative results suggests that reconstruction and prediction models **purely learnt from data** can be used for effective data assimilation, hence enabling the long-term prediction of even complex fluid flows.

1. INTRODUCTION

In fluid mechanics, each task (modelisation, closure, control or reduction) can be written as an optimization problem. However, for high Reynolds number, the convection term dominates the diffusion term, yielding a nonlinear, high-dimensional, multi-scale and nonconvex problem. Solving directly this formulation is challenging even intractable and new methods must be developed. Given the huge amount of both numerical and experimental data, a possibility is to use machine learning tools to solve optimization problems purely from data [5]. This paper investigates such data-driven procedures to estimate a fluid flow velocity field. In particular, a dynamical mode decomposition (DMD) model is used in combination with support vector machine regression (SVR) to continuously predict four fluid flows: 2D vortex shedding, a spatial mixing layer, 3D vortex shedding and an urban flow.

2. STATE OF ART

The field to estimate is denoted U_t. Two approaches are possible to obtain the estimate: the reconstruction and the prediction [6]. In the **reconstruction** problem, limited measurements y_t at time t are used to recover the velocity field at the same time (interpolation in space). In the **prediction** problem, a dynamical model is used to advance in time the velocity field U_{t-1} (extrapolation in time). For turbulent flows, the state is high-dimensional because of the complex spatio-temporal dynamics. However, low dimensional features can be extracted, making relevant the use of dimensionality reduction techniques [21]. Reconstruction and prediction problems are therefore equivalent to the estimation of the reduced (also called latent) state, as shown in figure 1.

2.1 Reconstruction

The measurement operator $\mathscr{H}$ being likely ill conditioned thus not invertible, the inverse operator $\mathscr{G} = d \circ f$ is estimated from data. Three ideas were developed in the literature. The first approach is the direct reconstruction [1], evaluating $\mathscr{G}(y)$ for each new measurement vector. The flow field is written as a linear combination of reference modes which can be generic (e.g. Fourier modes) or tailored to the considered flow (data driven modal decomposition). The second approach is the regressive reconstruction where the complete operator $\mathscr{G}$ is learned using supervised learning methods. Given a parametric

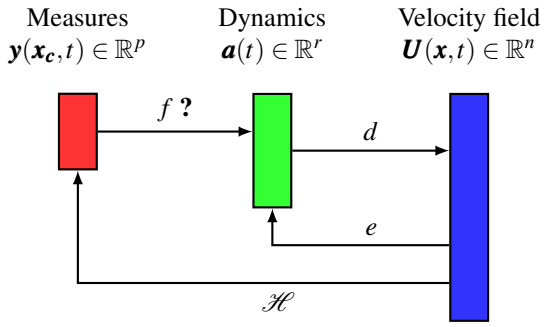


Figure 1: Representation of the reconstruction problem with dimensionality reduction: measurements are used to recover dynamics of dominant structures.

or nonparametric formulation of $\mathscr{G}$, a cost function evaluating the error between training examples (snapshots with associated measurements) and their reconstruction is minimized. First investigations of this method include the well-known stochastic estimation [2], where the reconstructed field is explained by a multi-linear function of available measurements. The third approach is data-assimilation where a dynamical model evolves the field estimate while measurements improve the forecasts [16]. The dynamical model may be a reduced-order approximation of Navier-Stokes equations, found by a Galerkin projection onto a data-driven basis or found by model identification.

2.2 Prediction

The velocity field satisfies a partial differential equation, namely Navier Stokes equations. To advance the state, the flow map is introduced, integrating the initial condition from t_0 to t [11]. Several strategies were developed to approximate this flow map in a data-driven fashion. A first approach consists in using supervised learning techniques to learn the input-output relation between past states and future states. Neural networks are particularly suitable given their high flexibility to capture nonlinearities. As an example, the reader is referred to [7] where we use a recurrent neural network with long-short term memory to continuously predict the chaotic Lorenz system. A second approach makes extensive use of the Koopman theory which introduces a linear but infinite dimensional operator to advance all possible observations of the state. Finite approximations of this so called Koopman operator give a linear dynamical model that can be used for prediction and control. Current research focus on finding a good space of observables where to learn the approximation or at least limit the spurious behaviour of identified eigenfunctions [14]. When working with latent state components as observables, the approximation of the Koopman operator is known to be the dynamical mode decomposition (DMD) model [22].

2.3 Reduction

Even highly turbulent flows exhibit low dimensional spatial directions called modes. Vortex shedding for wake flows, Kelvin Helmotz vortices for shear flows and coherent structures for boundary layers can be cited as examples. The extraction of such structures can be performed by linear or nonlinear encoding transformations e. If the decoding transformation d is known, the estimation of the latent structures dynamics $\boldsymbol{a}(t) \in \mathbb{R}^r$ is enough to infer the velocity field. The most common reduction technique is the proper orthogonal decomposition (POD) [19], which is the name given to principal components analysis applied to fluid flow data. Extracted modes are uncorrelated and hierarchically sorted by the data variability they recover. The simplicity of the implementation makes the POD a method of choice for dimensionality reduction. However, recent improvements in deep learning have made possible considerable progress in dimensionality reduction by using autoencoders. Two neural networks are therefore trained simultaneously to optimally encode and decode data. As a reference example, Xu et al. [23] took advantage of convolutional networks to leverage nested nonlinear manifolds and predict transient flows.

2.4 Work in this paper

This conference paper investigates the use of dynamical mode decomposition as a dynamical model and support vector regression as a reconstruction model to estimate four flow fields with increasing complexity: the flow in the wake of a 2D cylinder (Re = 200), a spatial mixing layer (Reynolds base on vorticity thickness Re = 500), the wake of a 3D cylinder (Re = 20000) and the flow in the viscinity of a tower placed in an atmospheric boundary layer (Reynolds base on the tower base length Re = 64000). Flow fields are reduced using a linear autoencoder and recursive forecasts of the latent state are sequentially enforced by the reconstructions. Data assimilation is performed using the models established solely from data, hence refered as data-driven assimilation. Figure 2 gives a general overview of the strategy.

3. METHODS AND DATA

This section gives details about notations, simulation data and mathematics behind the reduction, reconstruction, prediction and assimilation procedure.

3.1 Simulation data

Simulation data are written as a matrix $U \in \mathbb{R}^{n \times m}$ where n is the dimension (number of cells multiplied by the

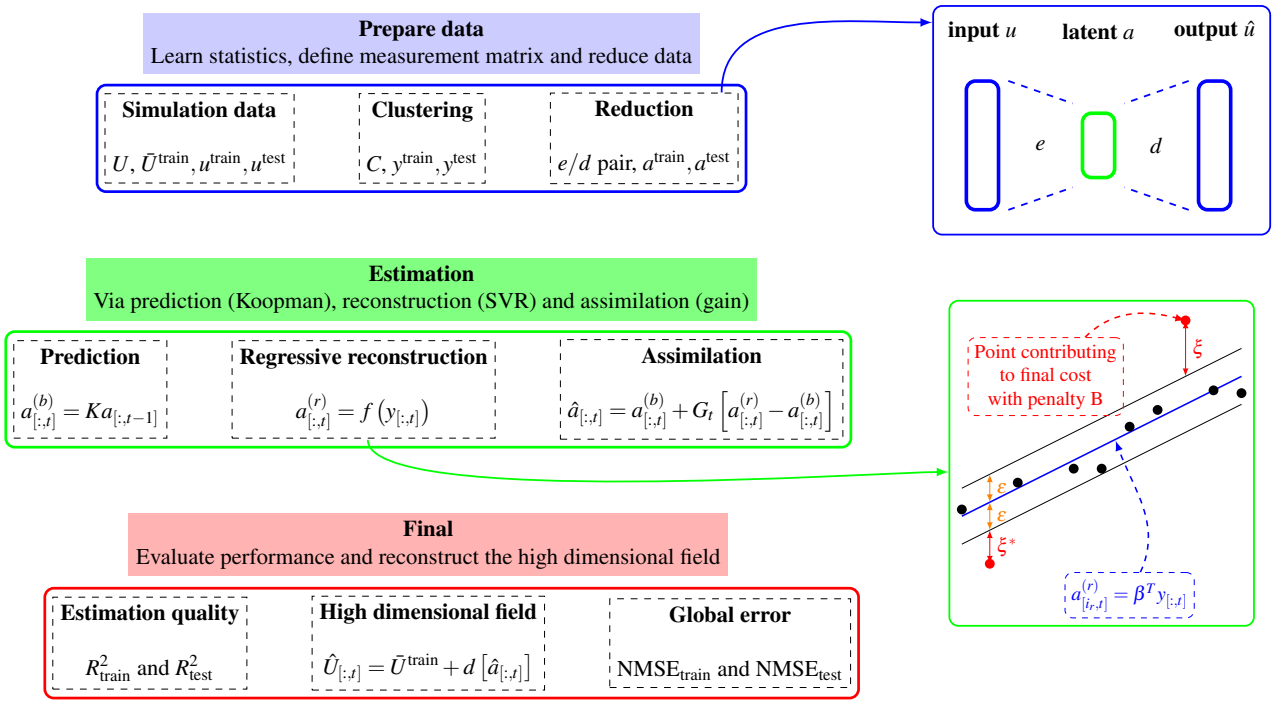


Figure 2: The proposed framework for the data-driven estimation of a fluid flow field.

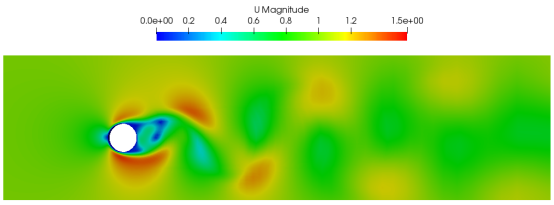
number of velocity components) and m is the number of snapshots. Considering a 70/30 split, $m^{\text{train}} = 0.7m$ and $m^{\text{test}} = 0.3m$ are respectively the number of training and testing snapshots. The fluctuant velocity field matrix is defined as $u = U - \bar{U}$ where $\bar{U}$ is the mean flow computed over all snapshots. This tall but skinny matrix is split into u^{train} and u^{test} matrices. The **case 0** corresponds to a URANS $k-\omega$ computation of a 2D cylinder placed in a uniform flow. The Reynolds number, based on the cylinder diameter, is $Re = 200$. The snapshot ensemble is composed of 125 snapshots written on a 5840 dimensional grid. The **case 1** corresponds to a direct numerical simulation of a 2D spatial mixing layer. The upper (fast) and lower (slow) stream velocities are respectively $U_1 = 30m/s$ and $U_2 = 10m/s$. The initial vorticity thickness is $\delta_{\omega_0} = 1m$ and the inlet profile is a hyperbolic tangent [12]. Stochastic perturbation is added to the inlet profile to trigger the Kelvin Helmotz instability [13]. A total number of 2700 snapshots is available, for a domain containing 3690 cells. The **case 2** corresponds to a large eddy simulation of a square cylinder wake. The Reynolds number based on the cylinder diameter is $Re = 20000$ and the snapshot ensemble contains 1312 snapshots for a domain with 48023 cells. The **case 3** corresponds to a large eddy simulation of the flow in the viscinity of a tower. A vortex method is used to reproduce the inlet turbulent velocity profile. The Reynolds number based on the tower width is $Re = 64000$. Estimations are performed in a transversal plane centered around the tower, which contains 28432 cells. A total number of 1596 snapshots is available. Figure 3 gives a visualisation of these flow fields.

3.2 Dimensionality reduction

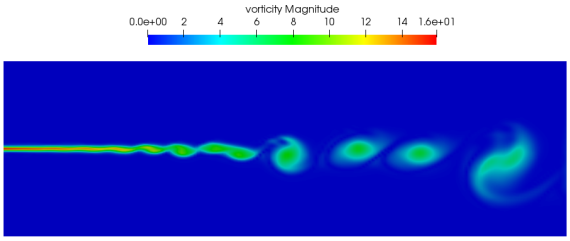
The encoder e compresses the data u from an initial space to a latent space and the decoder d decompresses encoded data. Given a family of candidate encoders E and decoders D, the best e/d pair is determined by:

$$(e^*, d^*) = \underset{(e,d)\in E\times D}{\arg\min}\ \varepsilon(u_{[:,t]}, d\left[e(u_{[:,t]})\right]) \qquad (1)$$

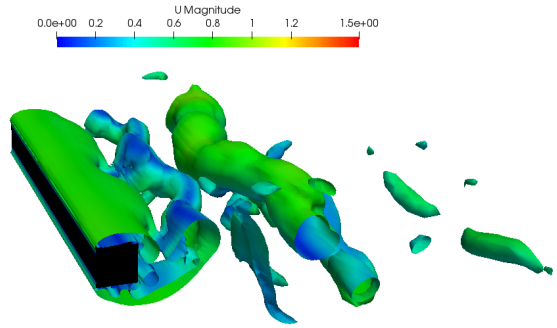
In proper orthogonal decomposition, the encoder and decoder are unitary matrices obtained from the spectral decomposition of the training covariance matrix $C_u = u^{\text{train}}[u^{\text{train}}]^T$. This decomposition yields: $C_u\Phi = \Phi\Lambda$ where the transfer matrix $\Phi \in \mathbb{R}^{n\times n}$ transforms initial basis vectors into uncorrelated directions. These modes are hierarchically sorted according to the variance Λ_{ii} they recover. When truncating the transformation to first r modes, initial data are written in the best r dimensional subspace to describe variability in u^{train}. In recent applications, encoders and decoders are neural networks. Here, we consider a linear autoencoder i.e., a neural network with one hidden layer and linear activations [18]. The weights and biases in the network are optimized by minimizing the mean square error between training samples and their autoencoding (see table 1). The dimension



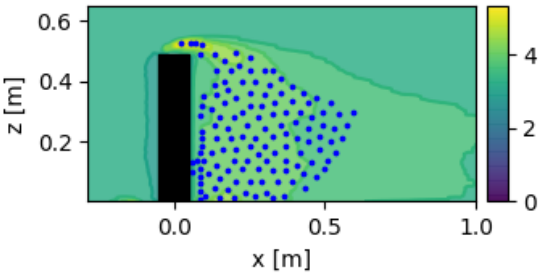
(a) Case 0 - 2D cylinder's wake colored by velocity magnitude (m/s)



(b) Case 1 - spatial mixing layer colored by vorticity magnitude (s^{-1})



(c) Case 2 - 3D cylinder with Q = 50 isosurfaces colored by velocity magnitude (m/s)



(d) Case 3 - plane in a urban flow, colored by the x componant of the root mean square velocity field (m/s). Sensors are also superimposed.

Figure 3: Visualisation of the four cases investigated in this communication.

	POD	**LAE**
Find	$\Phi_r \in \mathbb{R}^{n\times r}$	$W_1 \in \mathbb{R}^{n\times r}$ and $b_1 \in \mathbb{R}^r$ $W_2 \in \mathbb{R}^{r\times n}$ and $b_2 \in \mathbb{R}^n$
Encoding	$a_{[:,t]} = \Phi_r^T u_{[:,t]}$	$a_{[:,t]} = W_1 u_{[:,t]} + b_1$
Decoding	$\hat{u}_{[:,t]} = \Phi_r a_{[:,t]}$	$\hat{u}_{[:,t]} = W_2 a_{[:,t]} + b_2$
Noteworthy	Orthogonal modes $\Phi_r^T \Phi_r = I_{r\times r}$	Non orthogonal modes $W_1 = W_2^\dagger$
Modes	Φ_r	W_2
Method	SVD	Minimisation error $\varepsilon = \|\|u^{\text{train}} - \hat{u}^{\text{train}}\|\|_2^2$

Table 1: Autoencoding formulas

of the latent state is chosen accordingly with the number of POD modes required to recover 99% of the variance for case 0 and 80% for all other cases.

3.3 Sensor placement and reconstruction

Measurements are supposed to be p known locations in the fluctuant field to estimate. The measurement operator is a matrix $C \in \mathbb{R}^{p\times n}$ with $C_{ij} = 1$ if $y_i = u_j$ and 0 otherwise. Training and testing measurements are therefore $y^{\text{train}} = Cu^{\text{train}}$ and $y^{\text{test}} = Cu^{\text{test}}$. The spatial location of sensors is determined by enhanced clustering [10]. First, cell centres are partitioned into Voronoi cells defined by their centroids. Second, most energetic clusters are defined as sensors. In this paper, the number of clusters is set to 500 for all cases and sensors correspond to Voronoi centroids recovering 80% of the training field variance. The objective is now to learn the optimal mapping f so that $a^{(r)} = f(y)$ is a *good* estimate of the actual latent state a. This is a three-step procedure: choice of a form of a function, learning procedure to minimise a cost function and validation. In this paper, a focus is made on support vector regression (SVR). If this regression is performed in the latent state space, the estimation for the mode i_r is $a^{(r)}_{[i_r,t]} = \beta_{i_r}^T y_{[:,t]}$ i.e. a linear combination of measurements. Here, $\beta_{i_r} \in \mathbb{R}^{p\times 1}$ ensures at most an ε deviation from true targets and its optimal value is found by solving the primal formula:

$$\begin{cases} \min\limits_{\beta_{i_r},\xi,\xi^*} \dfrac{1}{2}\beta_{i_r}^T\beta + B\sum\limits_{t=1}^{m_{\text{train}}}(\xi_t + \xi_t^*) \\ a^{\text{train}}_{[i_r,t]} - \beta_{i_r}^T y^{\text{train}}_{[:,t]} \leq \varepsilon + \xi_t \ \forall t \\ \beta_{i_r}^T y^{\text{train}}_{[:,t]} - a^{\text{train}}_{[i_r,t]} \leq \varepsilon + \xi_t^* \ \forall t \\ \xi_t, \xi_t^* \geq 0 \ \forall t \end{cases} \tag{2}$$

With slack variables ξ and ξ^* to penalyze observations out of the ε tube. This regularization is controlled by the box constraint B. Instead of performing the linear regres-

sion in the latent state space, it can be perform in a higher even infinite dimensional subspace using the kernel trick. The estimation of a is therefore a nonlinear combination a measurements, yielding:

$$a^{(r)}_{[i_r,t]} = \sum_{t=1}^{m_{\text{train}}} (\alpha^*_t - \alpha_t) G(y^{\text{train}}_{[:,t]}, y_{[:,t]}) \tag{3}$$

Where G is a kernel function and (α, α^*) are Lagrange multipliers that intervene in the dual formulation of the problem. The kernel computes high dimensional interactions between variables y^{train} and y without actually transforming variables. The reader is referred to the tutorial of Smola [20] for a complete derivation of the equation and the cost function. To ensure that the model is robust on unseen data, hyperparameters of the SVR (the kernel and the box constraint) are cross-validated using a randomized grid search.

3.4 Dynamical mode decomposition

The dynamical mode decomposition finds the best one-step ahead linear dynamical model to describe the dynamics of data. The DMD matrix K is obtained by the Moore Penrose inverse:

$$K = \left[a^{\text{train}}\right]^{(+1)} \left[a^{\text{train}}\right]^{\dagger} \tag{4}$$

Where $\left[a^{\text{train}}\right]^{(+1)}$ is the time shifted version of a^{train}. This matrix can be used as a linear predictive model: starting from an initial condition a_{t_0}, the h-step ahead recursive forecast of the latent state is given by:

$$\hat{a}_{t_0+h} = K^h a_{t_0} \tag{5}$$

The K matrix is a finite approximation of the Koopman operator. If the dynamics of the latent state is nonlinear, identified eigenfunctions from left eigenvectors of K are spurious i.e. they do not evolved as predicted by their associated eigenvalues [4].

3.5 Data-driven assimilation

For nonlinear systems, the subspace spanned by latent state components is not Koopman invariant. Therefore, the dynamical model is erroneous and predictions must be updated. To make a continuous forecast of the latent state, the idea is to assimilate reconstructions. Three types of errors must be considered: the error on the initial condition, the dynamical model error and the reconstruction error. In this paper, reconstructions are assimilated each F = 5 new predictions. Covariance matrices of errors are defined as follows:

- $Q = \mathbb{E}\left[\left(a_{t_0+F} - K^F a_{t_0}\right)^2\right]$ for the recursive prediction using the dynamical mode. This matrix is estimated by a sample covariance, using training data.
- R for the covariance error on measurements. Noise is independantly applied to each component so that $R_{i_p,i_p} = \sigma^2_{i_p}$ with $\sigma_{i_p} = I_y \max\left[y_{[i_p,:]}\right]$.
- P_0 for the covariance error on the initial condition. Similarly to R, we define $P_{0_{i_r,i_r}} = \sigma^2_{i_r}$ with $\sigma_{i_r} = I_a \max\left[a_{[i_r,:]}\right]$.

The proposed assimilation scheme is based on Kalman filter [16] equations and is summarized in figure 4.

3.6 Metrics

To compare actual trajectories and estimated ones (reconstructed, predicted or assimilated), the normalized mean square error (NMSE) is used. Denoting s the true trajectory and $\hat{s}$ the estimated one, the error on component i for m_e samples is:

$$\text{NMSE}_i = \frac{\sum_{t=1}^{m_e} \left[s_{[i,t]} - \hat{s}_{[i,t]}\right]^2}{\sum_{t=1}^{m_e} \left[s_{[i,t]} - \overline{s}_{[i,:]}\right]^2} \tag{6}$$

A normalized error of zero means that on average, the error is small compared to the expected variability. The determination coefficient is also used, which is a "score" version of the NMSE metric, defined by:

$$R^2_i = 1 - \text{NMSE}_i \tag{7}$$

Metrics evaluated on training and testing data must be similar to ensure a good bias-variance trade-off, typically achieved by cross validation of hyperparameters. The global metric is determined by averaging over the number of modes. For the DMD results, training and testing data sets are split into *overlapping* trajectories with length $H = 16$. The h-step ahead prediction quality is quantified with a mean score over the number of trajectories while the global score also averages over the horizon[1]. For the assimilation results, metrics are evaluated by averaging over the ten *successive* trajectories that can be extracted from training or testing sets [2].

4. RESULTS

To reduce the dimension from n to $r \ll n$, the proper orthogonal decomposition and the linear autoencoder are used. The latent space dimension corresponds to the number of POD modes required to recover 99% of the variance for case 0 and 80% for other cases. Table 2

[1] DMD errors are evaluated on smaller sequences instead of the whole trajectory because the error would likely be 100% given the erroneous nature of the DMD model for nonlinear dynamics

[2] Assimilations are evaluated on ten smaller trajectories instead of the whole trajectory to test the procedure with different initial conditions.

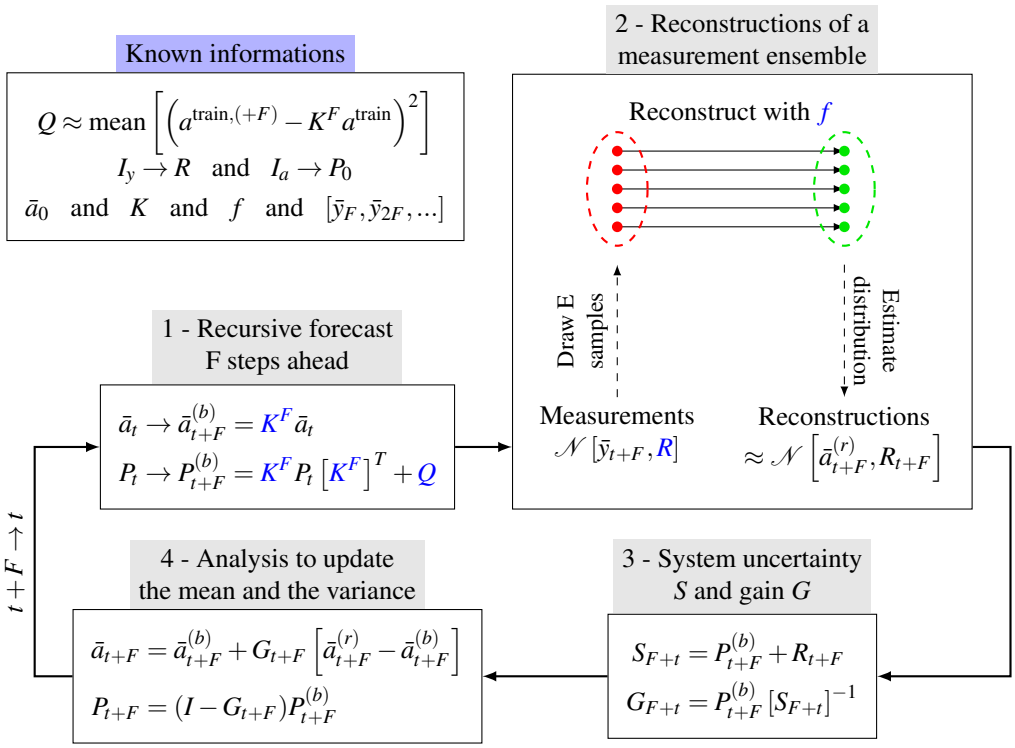


Figure 4: Data-assimilation procedure using data-driven models.

summarizes reduction results, including dimensions and autoencoding errors. As expected, the number of spatial modes and the error increases with the complexity of the flow. Testing errors are higher than training errors for case 2 and case 3 which is symptomatic of overfitted modes. To reduce this difference, robust principal components and cross validated neural network could be used, but this is out of the scope of this communication. Figure 5 gives a visualisation of the first POD mode (the most energetic) and the first LAE mode for the 2D cylinder. This comparison suggests that the nonorthogonal reduction is more interpretable than the orthogonal reduction, as supported by Erichson et al. conclusions [9].

Case	n	r	NMSE POD	NMSE LAE
0	11 680	5	[0.12 0.12]	[0.12 0.12]
1	27 360	11	[8.88 9.59]	[9.22 9.95]
2	144 069	21	[29.14 31.17]	[30.29 32.11]
3	85 296	139	[24.69 39.84]	[47.67 57.61]

Table 2: State space and latent space dimensions for each case and autoencoding errors. Results are written as [train test] errors (NMSE in %).

The clustering algorithm gives 500 centroids to optimally partition the mesh in an unsupervised fashion. Sensors are defined as the centroids that best describe the variability in training data. Considering the two (cases 0 and 1) or three (cases 2 and 3) components of the velocity field to estimate, the total number of sensors p is summarized in table 3. The choice of 500 clusters is arbitrary and could be optimized using heuristic criterions such as elbow and silouhette but this is not the scope here. In particular, the number of clusters for the 2D cylinder is undoubtedly high regarding the simple periodic behaviour of the flow. To learn the mapping between the measurement space and the latent state space with SVR, cross validation is performed. Tested hyperparameters are randomly chosen in a grid, for a total of 25 combinations and a 70% chance of hitting the optimal hyperparameter space [3]. Each mode is regressed independantly, yielding a total of 176 learned models to obtain reconstruction scores in table 3. For case 0, latent state components are perfectly estimated from measurements which is not surprising given the simplicity of the flow. Results are much more mitigated for case 3 where a strong overfit of training data is visible. To support this idea, figure 6 illustrates the reconstruction of the first latent state component, for the POD and the LAE methods. Reconstructions of training data (blue points) nearly recover all the expected variance, hence the close to unit determination score. For testing data, the determination coefficient (orange points) for each mode clearly respects the hierarchy imposed by POD reduction: first modes, corresponding to slow and coherent structures, are easier

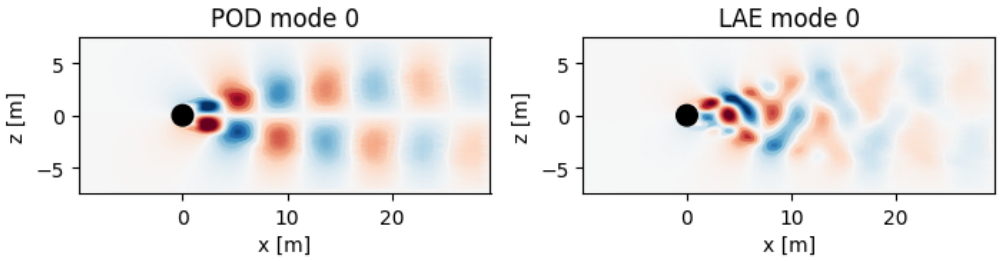


Figure 5: Comparison of a POD mode and a LAE mode for case 0. The vortex shedding is better captured with the nonorthogonal reduction.

to estimate than higher modes, corresponding to fast and small structures. At the opposite, the LAE modes are not hierarchically sorted and the estimation score is similar for each method. Overfitting being not a desirable property for robust fluid flow estimation, SVR doesn't seem to be tailored for the urban flow, and other regression techniques should be investigated.

Case	p	R^2 POD	R^2 LAE
0	92	[99.33, 99.31]	[99.5, 99.5]
1	122	[98.84 96.29]	[97.48 94.95]
2	372	[98.73 90.92]	[97.75 84.56]
3	393	[96.58 72.55]	[94.53 60.11]

Table 3: Reconstruction results for each case. Results are written as [train test] errors (R^2 in %).

Concerning the DMD model, normalized errors are summarized in table 4. The prediction error corresponds to the error integrated over the $H = 16$ horizon and the number of overlapping trajectories with length H in training or testing data. Interestingly, the latent state obtained with the LAE reduction for case 0 and case 1 is easier to predict with a linear model. Besides the global score, plotting the h-step ahead error as a function of h reveals how errors accumulate in the recursive process. Figure 7 gives an exemple when predicting testing sequences of the 3D cylinder with the POD reduction. The dynamics of the latent state being nonlinear, using the DMD matrix as a dynamical model for long term prediction is naturally erroneous, hence the bars of errors. This is confirmed by the green curves, corresponding to the h-step ahead score of each identified eigenfunctions. All of them are spurious (meaning they do not evolve as predicted by their eigenvalues) which is symptomatic of an observable subspace which is not Koopman invariant.

To correct predictions from the dynamical model, reconstructions are assimilated each $F = 5$ nondimensional time steps. Noise is applied to the initial condition a_0 and

Case	Prediction error	
	POD	LAE
0	[24.01 24.16]	[2.02 2.04]
1	[14.8 14]	[8.07 7.89]
2	[75.5 70.88]	[65.4 59.35]
3	[86.68 93.95]	[81.3 90.48]

Table 4: Errors in the recursive prediction of trajectories with length $H = 16$. NMSE values are obtained by averaging over the horizon and the number of overlapping trajectories with length $H = 16$ in training or testing data. Results are given as [train test] errors.

the measurement vector y, with intensity levels $I_a = 0.1$ and $I_y = 0.2$. Global scores are obtained by a mean over the number of modes and the one (case 0) or ten (other cases) *successive* trajectories in testing data. Results are given in table 5. It appears that using data assimilation results in a better long-term estimation of the latent state compared to the sole recursive forecast or the sole reconstruction at each time step. Figure 8 qualitatively supports this conclusion for the POD-reduced mixing layer.

Case	Dynamics	Reconstructions	Assimilation F = 5
0	[24.24 2.02]	[0.87 1.11]	[1.51 1.33]
1	[69.33 54.02]	[11.82 7.87]	[6.67 4.08]
2	[81.38 72.29]	[49.33 33.36]	[42.01 34.17]
3	[96.42 94.71]	[78.57 81.20]	[76.58 70.90]

Table 5: Comparison of errors for each estimation method (NMSE in %). These errors are averaged over the one (case 0) or ten (other cases) trajectories that can be extracted in the testing data. Results are written as [POD LAE] errors.

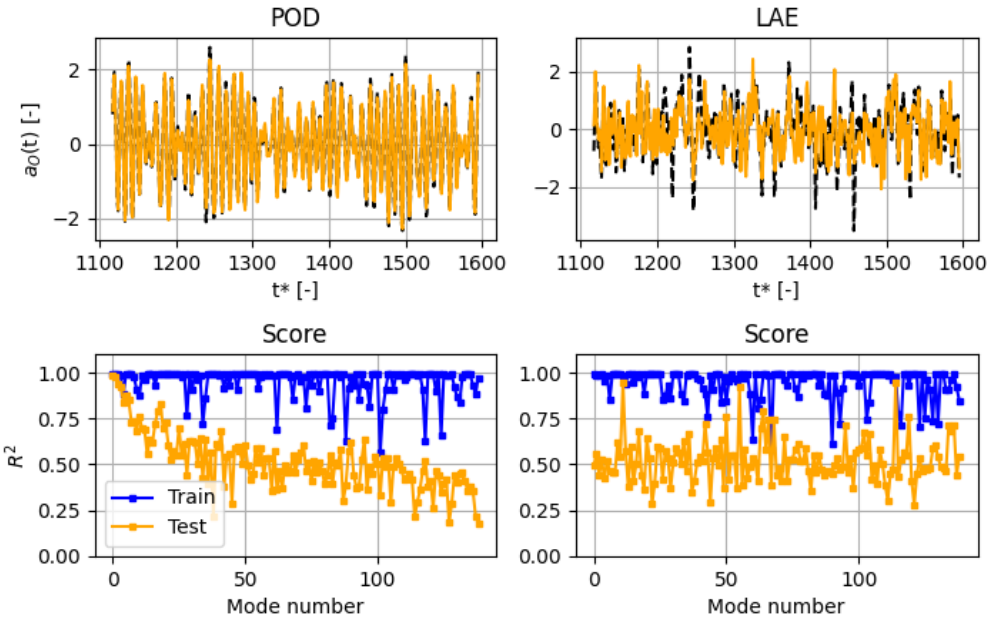


Figure 6: Reconstruction scores for case 3 and complete reconstruction of the first mode. For the POD reduction, large structures (first modes) are easier to reconstruct than small structures (last modes). For the LAE reduction, the modes are not hierarchically sorted and reconstructions scores are similar for all modes. A strong overfitting is visible, hence encouraging the use of other regression methods. Top figures: expected latent state in black and reconstructions in orange. Bottom figures: training $R^2_{i_r}$ in blue, testing in orange

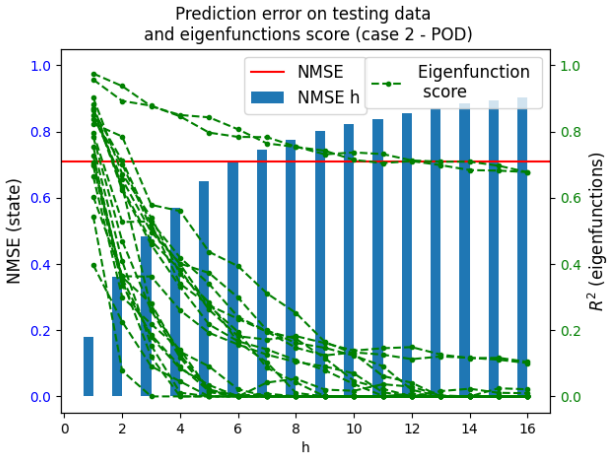


Figure 7: Bar plot corresponding to the h-step ahead error in the recursive prediction of testing data for case 2 with POD reduction. The h-step ahead score of each identified eigenfunction is also shown in green. The red line corresponds to the testing NMSE in table 4

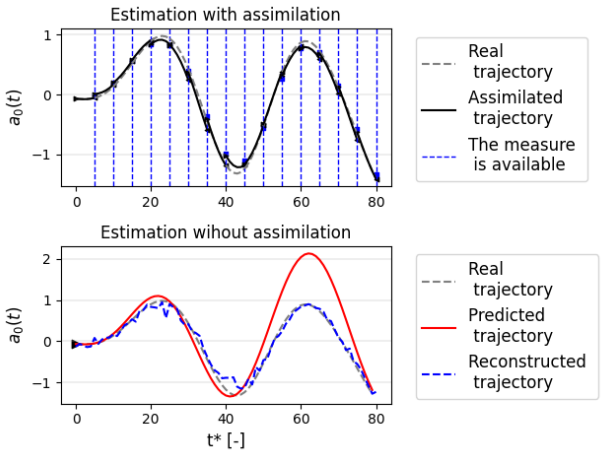


Figure 8: Exemple of the data-driven assimilation procedure on the mixing layer with POD reduction. Reconstructions are assimilated each $F = 5$ new predictions to update recursive forecasts by DMD.

Appart from the global error, the score as a function of the mode number can also be investigated. An exemple for the POD-reduced 3D cylinder is shown in figure 9. The curves quantitatively support the idea that on average, assimilated trajectories are better estimates than sole reconstructions or predictions. Concerning the POD-reduced 2D cylinder, changing the assimilation frequency

is of particular interest. This mode is badly predicted with a linear model but is perfectly reconstructed with measurements. With $F = 1$, the data assimilation procedure only gives credit to reconstructions (Kalman gain is the identity matrix) while for greater F, the assimilation is not enough frequently performed to significantly correct the mean determination coefficient. Quantitative results are given in table 6 to support this conclusion.

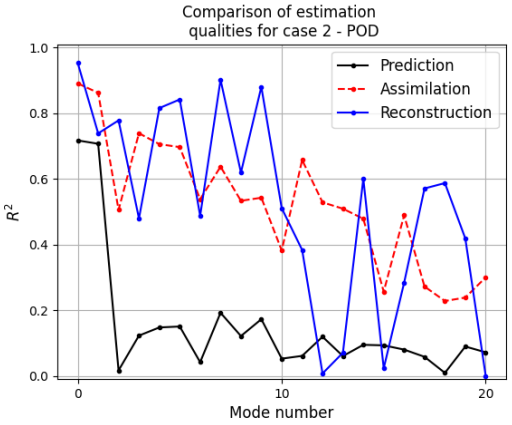


Figure 9: Comparison of estimation methods for testing trajectories of case 2 reduced with POD.

F	Reconstructions	Prediction	Assimilation
1			98.13
2	⋮	⋮	80.30
3	98.94	0.0	57.31
4			34.87
5	⋮	⋮	27.03
6			8.12

Table 6: Influence of the data assimilation frequency when estimating the fifth POD mode for case 0. Values correspond to the determination coefficient evaluated on the testing trajectory.

5. CONCLUSION

In this paper, different fluid flow estimation strategies are investigated on four increasing complexity cases. To reduce the dimension of the state to estimate, dominant spatial directions are extracted using the proper orthogonal decomposition or a linear autoencoder. Despite good results for the 2D cylinder, the mixing layer and the 3D cylinder, the reduction method clearly overfits the data for the urban flow, hence limiting the use of training modes for testing purposes. Latent states are then estimated by reconstruction and prediction. The reconstruction consists in using measurements of the fluctuant velocity field at current time to recover the latent state at the same time. This can be performed by a support vector regression, which independantly regresses each component of the latent state by available measurements. To respect a trade-off between the biais and the variance, hyperparameters are optimized by cross validation, leading to good training and testing scores for the 2D cylinder, the spatial mixing layer and the 3D cylinder. Results are much more mitigated on the urban flow, where modes dynamics are harder to reconstruct. The prediction consists in using a dynamical model to advance an initial condition in time. Dynamical mode decomposition is used for that purpose, to learn the one-step ahead linear model that optimally describes the dynamics of data. When used as a long-term predictive model, errors accumulate because the latent space is not Koopman invariant. To avoid this accumulation of errors, data assimilation is then investigated: reconstructions are sequentially used to update predictions at a frequency F. By blending the benefits of the reconstruction and the prediction models, data-assimilation enables the long-term prediction of the fluid flow field. The procedure being purely based on machine learning, it is a promising technique for estimating any fluid flow field where data is available. Further investigations on academic cases could include generative modeling [17] to account for inlet parameters (e.g. Reynolds number, turbulent intensity), the use of probabilistic models such as CROM [12] or the use of balanced truncation for sensor placement [15]. An extensive study of the reconstruction problem for the first three cases was submitted to *Journal of Computational Physics* [8]. For the reduction part, POD and variationnal autoencodeurs were considered. For the reconstruction part, linear multitask regression, SVR, neural network and gradient boosting decision trees were used. Results suggest that encoding velocity fields as distributions instead of single points improve robustness when decoding a latent state estimate. Another conclusion concerns the performance of each reconstruction method: using cross validation enable similar results for all methods so that the choice of one regression model towards another depends on the quality of the data, the interpretability and the cost of implementation/computation. Conclusions drawn from this study provide valuable informations for the development of new estimation techniques based on machine learning and their deployment on complex geometries that can be encoutered in industrial issues.

REFERENCES

[1] SM Al Mamun, Chen Lu, and Balaji Jayaraman. Extreme learning machines as encoders for sparse reconstruction. *Fluids*, 3(4):88, 2018.

[2] Anthony Arnault, Julien Dandois, J-C Monnier, Jérôme Delva, and J-M Foucaut. Analysis of the filtering effect of the stochastic estimation and accuracy improvement by sensor location optimization. *Experiments in Fluids*, 57(12):1–22, 2016.

[3] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

[4] Steven L Brunton, Bingni W Brunton, Joshua L Proctor, and J Nathan Kutz. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PloS one*, 11(2):e0150171, 2016.

[5] Steven L Brunton, Bernd R Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020.

[6] Jared L Callaham, Kazuki Maeda, and Steven L Brunton. Robust flow reconstruction from limited measurements via sparse representation. *Physical Review Fluids*, 4(10):103907, 2019.

[7] Pierre Dubois, Thomas Gomez, Laurent Planckaert, and Laurent Perret. Data-driven predictions of the lorenz system. *Physica D: Nonlinear Phenomena*, 408:132495, 2020.

[8] Pierre Dubois, Thomas Gomez, Laurent Planckaert, and Laurent Perret. Machine learning for fluid flow reconstruction from limited measurements. *Journal of Computational Physics, submitted*, 2021.

[9] N Benjamin Erichson, Lionel Mathelin, Zhewei Yao, Steven L Brunton, Michael W Mahoney, and J Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A*, 476(2238):20200097, 2020.

[10] Balaji Jayaraman, Chen Lu, Joshua Whitman, and Girish Chowdhary. Sparse feature map-based markov models for nonlinear fluid flows. *Computers & Fluids*, 191:104252, 2019.

[11] Eurika Kaiser, J Nathan Kutz, and Steven L Brunton. Data-driven discovery of koopman eigenfunctions for control. *arXiv preprint arXiv:1707.01146*, 2017.

[12] Eurika Kaiser, Bernd R Noack, Laurent Cordier, Andreas Spohn, Marc Segond, Markus Abel, Guillaume Daviller, Jan Östh, Siniša Krajnović, and Robert K Niven. Cluster-based reduced-order modelling of a mixing layer. *arXiv preprint arXiv:1309.0524*, 2013.

[13] Jordan Ko, Didier Lucor, and Pierre Sagaut. Sensitivity of two-dimensional spatially developing mixing layers with respect to uncertain inflow conditions. *Physics of Fluids*, 20(7):077102, 2008.

[14] Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature communications*, 9(1):1–10, 2018.

[15] Krithika Manohar, J Nathan Kutz, and Steven L Brunton. Optimal sensor and actuator selection using balanced model reduction. *arXiv preprint arXiv:1812.01574*, 2018.

[16] Vincent Mons, J-C Chassaing, Thomas Gomez, and Pierre Sagaut. Reconstruction of unsteady viscous flows using data assimilation schemes. *Journal of Computational Physics*, 316:255–280, 2016.

[17] Jeremy Morton, Mykel J Kochenderfer, and Freddie D Witherden. Parameter-conditioned sequential generative modeling of fluid flows. *AIAA Journal*, pages 1–17, 2021.

[18] Elad Plaut. From principal subspaces to principal components with linear autoencoders. *arXiv preprint arXiv:1804.10253*, 2018.

[19] Clarence W Rowley and Scott TM Dawson. Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49:387–417, 2017.

[20] Alex J Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[21] Kunihiko Taira, Steven L Brunton, Scott TM Dawson, Clarence W Rowley, Tim Colonius, Beverley J McKeon, Oliver T Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S Ukeiley. Modal analysis of fluid flows: An overview. *Aiaa Journal*, 55(12):4013–4041, 2017.

[22] Matthew O Williams, Ioannis G Kevrekidis, and Clarence W Rowley. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, 2015.

[23] Jiayang Xu and Karthik Duraisamy. Multi-level convolutional autoencoder networks for parametric prediction of spatio-temporal dynamics. *Computer Methods in Applied Mechanics and Engineering*, 372:113379, 2020.