



Flow Reconstruction Around a Surface-Mounted Prism from Sparse Velocity and/or Scalar Measurements Using a Combination of POD and a Data-Driven Estimator

Shengqi Lu¹ · George Papadakis¹

Received: 15 September 2022 / Accepted: 3 April 2023 / Published online: 17 April 2023
© The Author(s) 2023

Abstract

A data-driven algorithm is proposed for flow reconstruction from sparse velocity and/or scalar measurements. The algorithm is applied to the flow around a two-dimensional, wall-mounted, square prism. To reduce the problem dimensionality, snapshots of flow and scalar fields are processed to derive POD modes and their time coefficients. Then a system identification algorithm is employed to build a reduced order, linear, dynamical system for the flow and scalar dynamics. Optimal estimation theory is subsequently applied to derive a Kalman estimator to predict the time coefficients of the POD modes from sparse measurements. Analysis of the flow and scalar spectra demonstrate that the flow field leaves its footprint on the scalar, thus extracting velocity from scalar concentration measurements is meaningful. The results show that remarkably good reconstruction of the flow statistics (Reynolds stresses) and instantaneous flow patterns can be obtained using a very small number of sensors (even a single scalar sensor yields very satisfactory results for the case considered). The Kalman estimator derived at one condition is able to reconstruct with acceptable accuracy the flow fields at two nearby off-design conditions. Further work is needed to assess the performance of the algorithm in more complex, three-dimensional, flows.

Keywords Flow reconstruction · Data-driven algorithms · POD · Flow around a prism

1 Introduction

Flow reconstruction from sparse measurements has many applications. In engineering for example, these measurements can guide actuators to meet an objective, such as drag reduction or mixing enhancement. In medicine, routinely used clinical modalities (such as ultrasound) provide velocity measurements at selected planes. This information can be used to

✉ George Papadakis
g.papadakis@imperial.ac.uk

Shengqi Lu
s.lu19@imperial.ac.uk

¹ Department of Aeronautics, Imperial College London, Exhibition Rd, London SW7 2AZ, UK

reconstruct the blood flow within the whole vessel, thus provide estimates of shear stress at the wall, a parameter that is known to correlate with cardiovascular diseases, such as atherosclerosis. The literature of flow estimation is vast, see for example Brunton and Noack (2015); Sipp and Schmid (2016); Callaham et al. (2019) and references therein. Below we present only a few key approaches to place the present work in context.

In order to make the problem tractable, reduced order models (ROMs), based on Proper Orthogonal Decomposition (POD), Dynamical Mode Decomposition (DMD) or resolvent modes, are usually employed, see Taira et al. (2017, 2020); Rowley and Dawson (2017). Willcox (2006) applied the “gappy” POD method to reconstruct the unsteady flow around a subsonic airfoil. She used a systematic approach to find the best sensor locations, which resulted in the placement at the POD peaks. Yildirim et al. (2009) also used the “gappy” POD method to reconstruct the 2D flow past a circular cylinder. The authors also found that sensors placed at the POD peaks result in small errors.

Compressed sensing (CS) strategies can be combined with ROMs for flow reconstruction, see Manohar et al. (2018); Callaham et al. (2019) for application of CS to POD and Brunton et al. (2015); Bai et al. (2020) for application to DMD. In CS methods the number of sensors must be significantly larger than the rank of the system, see table S1 in Manohar et al. (2018) and Brunton et al. (2015) for the explanation. In Manohar et al. (2018), a nearly optimal sensor placement strategy based on the QR decomposition of an appropriate matrix was proposed.

In the above papers, a dynamic model that describes the evolution of the POD coefficients in time (or information on the frequency and growth/decay rate in the case of DMD modes) is not employed and thus the history effect is not accounted for. Therefore, for the same value of measurements, the same POD coefficients will be computed. Dynamic ROMs, that account for the history of the flow, can be constructed either using an equation-based approach (i.e. by Galerkin projection of the governing equations to the space spanned by the modes, see Carlberg et al. (2011); Noack et al. (2011); Xiao et al. (2015)) or a data-driven approach. Iñigo et al. (2014, 2016) employed the system identification algorithm *n4sid*, see Qin (2006); van Overschee and de Moor (1994), to extract the matrices of a dynamic ROM estimator that links the input (velocity measurement at a single point) to the output (POD coefficients). They applied the algorithm to a flat-plate boundary layer and considered linearised (i.e. infinitesimally small) perturbations. Later, the method was extended to finite perturbations around the time-average, two-dimensional, flow around an airfoil by Iñigo et al. (2019), and the three-dimensional transitional flow inside a mixing vessel by Mikhaylov et al. (2021).

More recently, Savarino and Papadakis (2022) used *n4sid* to derive directly the dynamic ROM for the POD coefficients, and then coupled it with optimal estimation theory, see Kailath et al. (2000), to derive a dynamic observer that estimates the system state from noisy sensor measurements. Gong et al. (2020) designed a Kalman filter to estimate the vortex shedding behind a cylinder with measurements from only one sensor. The authors used harmonic decomposition to build the reduced-order model for the flow field and a Kalman filter for the estimator. Gomez et al. (2019) followed the framework proposed by Surana and Banaszuk (2016) to create an observer of Koopman form to estimate the flow field around an actuated airfoil. Tu et al. (2013) also coupled a simplified ROM with Kalman filter to estimate the flow behind an elliptical-leading-edge flat plate. Habibi et al. (2021) proposed a reduced-order model Kalman filter method to obtain high fidelity blood flow velocity fields by merging multi-fidelity hemodynamic data.

A different approach to fuse available measurements with the governing equations is based on the recently proposed physics-informed neural networks or PINNs, see Raissi et al. (2019,

2020), and Karniadakis et al. (2011) for a review. A cost function is defined that quantifies the (weighted) deviation from the governing equations and any given data (for example partial measurements). The parameters of a neural network are computed so as to minimise this cost function. The method has been successfully applied to assimilate experiments in a number of challenging 3D flows. For example, Cai et al. (2021) applied PINNs to infer the 3D velocity and pressure fields from snapshots of 3D temperature fields obtained by Schlieren imaging, Yin et al. (2021) showed that PINNs can infer material properties from noisy synthetic data and Raissi et al. (2020) used PINNs to extract the wall shear stresses in a patient-specific intracranial aneurysm. This is a very promising method, but care needs to be exercised in the selection of the weights of the cost function, and the parameters of the neural network, for example the number of layers.

In the present paper, we aim to reconstruct the flow from velocity and/or scalar concentration measurements. We achieve this by extending the approach of Savarino and Papadakis (2022) to include the scalar dynamics. We develop the algorithm and apply it to the 2D flow around a surface-mounted prism. The proposed method accounts for both velocity and scalar dynamics and does not have any adjustable parameters. The ability to extract velocity from scalar measurements only has many practical applications, for example it can be used to estimate the flow and dispersion of a pollutant around a building. Atmospheric interactions with buildings add unsteady and chaotic dynamics into the system, making this reconstruction problem very challenging.

The flow and scalar dispersion around obstacles was first analyzed in 2D experiments and numerical simulations. Vincont et al. (2000) conducted experiments to analyze the dispersion of scalar emitted from a line source downstream of a 2D obstacle in a wind tunnel and a water tunnel. Fragos et al. (2012) carried out 2D simulations around a wall-mounted prism for different Reynolds numbers. Martinuzzi and Tropea (1993) elucidated the fundamental differences between nominally 2D and fully 3D flows by performing experiments in a channel flow with obstacles of different spanwise lengths. Rossi and Iaccarino (2008) performed DNS simulations to investigate the interaction between turbulent flow structures and scalar dispersion. Diaz-Daniel et al. (2017) investigated the 3D unsteady flow structures downstream of a cube immersed in a laminar boundary layer. Castro and Robins (1977) performed experiments to investigate the flow behavior behind a surface-mounted cube. Rossi et al. (2010) conducted DNS simulations to study the scalar dispersion around a wall-mounted cube. The flow structures were similar to those of Hwang and Yang (2004), where the cube was immersed in a laminar boundary layer and the channel height was of the same order of the cube.

This paper is structured as follows: Sect. 2 presents the computational set up and methodology. Sections 3 and 4 detail the extraction of POD modes for velocity and scalar and present the evolution equations for the time-coefficients respectively. The process to obtain a data-driven dynamic estimator is explained in Sect. 5. Results are presented in Sect. 6 for velocity-only and scalar-only measurements at the design conditions. We also explore the robustness of the estimator by considering two nearby off-design conditions. We conclude in Sect. 7.

2 Computational Set-up and Numerical Methodology

We consider the 2D flow around a surface-mounted prism with height h . The computational domain, shown in Fig. 1, has dimensions $19h \times 10h$. The inlet is located at $x/h = -6$ in the streamwise direction, and the outlet at $x/h = 13$. In the wall-normal direction, the domain extends to $y/h = 10$. The origin of the coordinate system is located at the bottom

left corner of the prism. Uniform velocity $U_\infty = 1$ is prescribed at the inlet and a convective boundary condition at the outlet. No-slip conditions are imposed on the prism surfaces and bottom wall, while symmetry conditions are applied on the top boundary. Scalar is released from a circular source centred at $(-2, 0.2)h$, with radius $0.1h$. The source strength (amount of scalar released per unit volume) is equal to 10. Snapshots of velocities and scalar fields are collected within a sub-region, defined by the dash lines in Fig. 1, for further processing.

The flow domain is discretized using a Cartesian finite volume mesh. The cells are clustered close to the prism surfaces and the bottom wall. Three different grids, coarse, medium and fine, are created. The medium mesh, with a zoomed-in view close to the prism, is shown in Fig. 2. Mesh details are provided in Table 1. The time step Δt is selected to satisfy $CFL < 0.5$.

The Reynolds number, defined as $Re = U_\infty h/\nu$, is set to 1000. To remove the transients due to the initial condition, the flow is first advanced for 3 flow-through times, and the simulation is then restarted and advanced for 10 more flow-through times. In total 4750 flow and scalar field snapshots are recorded synchronously during the last 10 flow-through times. The time separation between successive snapshots is $0.04 \frac{h}{U_\infty}$.

The simulations are performed using our in-house code PANTARHEI (Xiao and Papadakis 2019; Mikhaylov et al. 2021; Yao et al. 2022). The code employs the Finite Volume Method to discretise the incompressible Navier–Stokes equations. Convection and diffusion terms are approximated with a second order central scheme in space. A fractional step method is implemented to enforce the continuity equation and extract pressure. A third-order accurate scheme is used for the transient term. Convection terms are treated explicitly using extrapolation from the 3 previous time instants, while diffusion terms are treated implicitly. To ensure a bounded solution for the scalar, the Gamma TVD scheme of Jasak et al. (1999) is applied to the convection terms. The code is parallelised with the aid of the PETSc library, Balay et al. (2022). The linear systems of equations for velocity and pressure are solved using the GMRES iterative

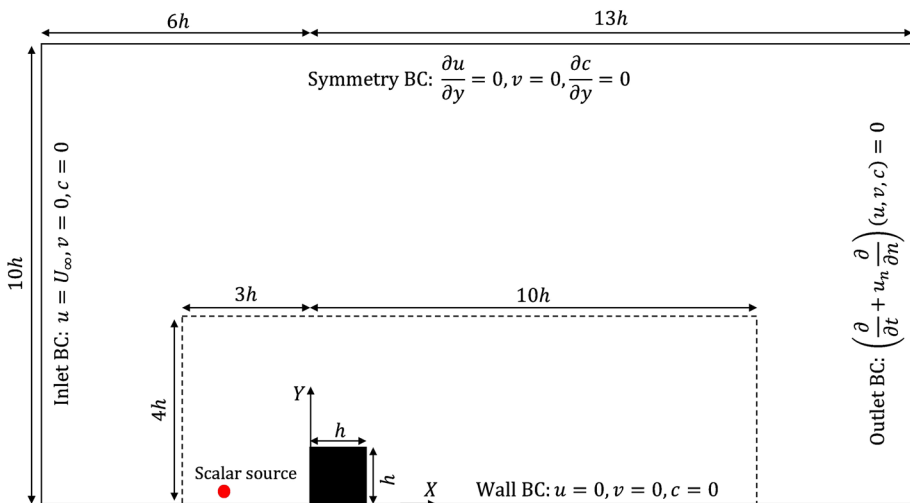


Fig. 1 Computational domain and boundary conditions. The dash lines mark the boundary of the region where snapshot data are collected

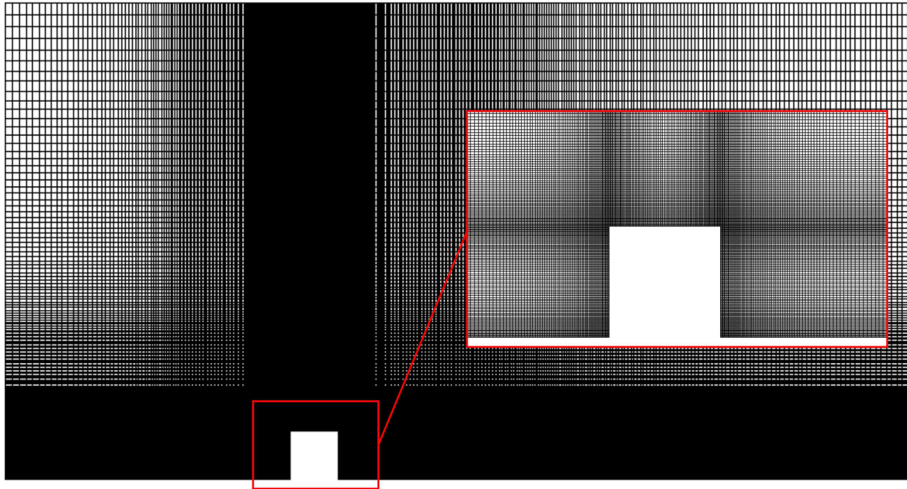


Fig. 2 The medium mesh with a zoomed-in view close to the prism

Table 1 Details for the three meshes employed for grid convergence study

Total No. of cells	$N_x \times N_y$	N_{prism}	$\delta_{1\text{st}}/h$	$\Delta t / \left(\frac{h}{U_\infty} \right)$
79,671 (coarse)	408×212	70	0.01	0.01
102,019 (medium)	474×234	82	0.0077	0.005
141,415 (fine)	576×268	102	0.006	0.005

N_x, N_y are the number of cells in the streamwise and wall-normal directions respectively, N_{prism} is the number of subdivisions along the prism edge, $\delta_{1\text{st}}$ is the thickness of the first layer of cells near the no-slip walls, and Δt the time step

algorithm. Convergence is accelerated using an algebraic multigrid preconditioner from the Hypre library, Falgout and Yang (2002).

We apply Reynolds decomposition to write an instantaneous quantity as the sum of a time-average (denoted by an overbar) and a fluctuation (denoted by a prime), for example $u = \bar{u} + u'$. Also for velocity components, we use separate variables (such as u, v), or different indices, u_i ($i = 1, 2$); the two notations are interchangeable, for example $u_1 = u, u_2 = v$. The scalar concentration is denoted by c .

Vertical profiles of velocity and scalar statistics (mean and variance) at streamwise locations $x/h = 2, 4, 6$ and 8 are shown in Figs. 3 and 4 respectively. All 3 meshes give very similar results for the time-average velocity, while there are small deviations for the Reynolds stress, $\langle u'u' \rangle$. The effect of mesh resolution is more pronounced in the scalar, especially the scalar variance, $\langle c'c' \rangle$. Overall the medium mesh gives results that are very close to those of the fine mesh and results from the former are processed and presented in the following sections.

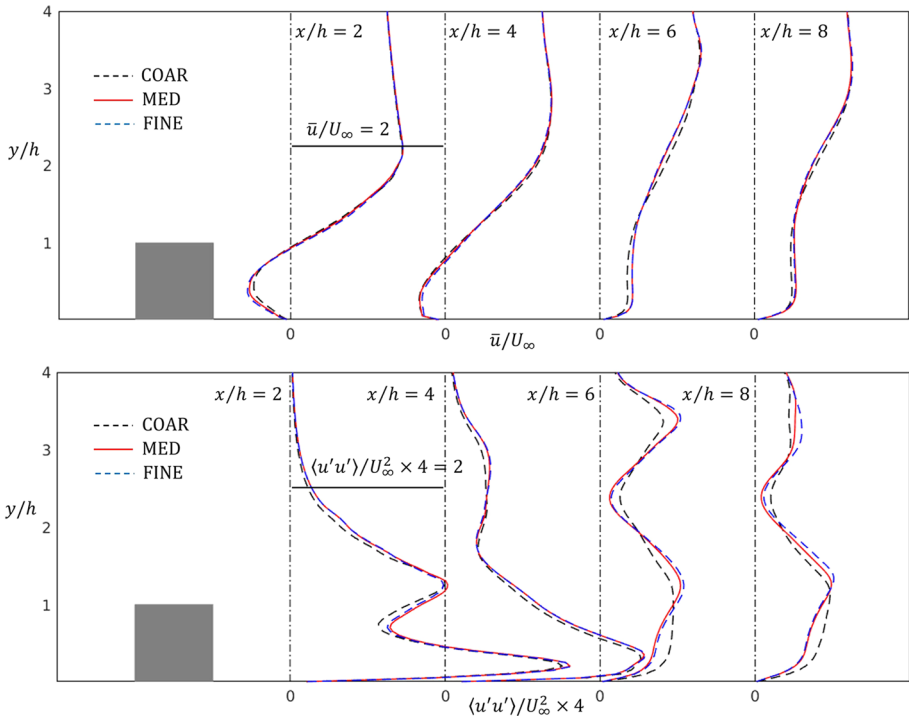


Fig. 3 Comparison of vertical profiles of mean horizontal velocity (top) and Reynolds stress (bottom) at $x/h = 2, 4, 6$ and 8 for three meshes

3 Extraction of POD Modes for Velocity and Scalar Fields

We use the methods of snapshots, see Sirovich (1987), to obtain the dominant POD modes, see Lumley (1967), for the velocity and scalar fields. The snapshot matrix $\mathbf{Y}(\mathbf{x}, t_1 : t_K)$ for the velocity fluctuations u' and v' is

$$\mathbf{Y}(\mathbf{x}, t_1 : t_K) = \begin{bmatrix} u'(\mathbf{x}_1, t_1) & u'(\mathbf{x}_1, t_2) & \dots & u'(\mathbf{x}_1, t_K) \\ \vdots & \vdots & \dots & \vdots \\ u'(\mathbf{x}_N, t_1) & u'(\mathbf{x}_N, t_2) & \dots & u'(\mathbf{x}_N, t_K) \\ v'(\mathbf{x}_1, t_1) & v'(\mathbf{x}_1, t_2) & \dots & v'(\mathbf{x}_1, t_K) \\ \vdots & \vdots & \dots & \vdots \\ v'(\mathbf{x}_N, t_1) & v'(\mathbf{x}_N, t_2) & \dots & v'(\mathbf{x}_N, t_K) \end{bmatrix} \in \mathbb{R}^{2N \times K}, \quad (1)$$

where $\mathbf{x}_i = [x_i, y_i]$, ($i = 1, 2, \dots, N$) is the location vector for the i -th spatial location, N is the number of cell centroids, K is the number of snapshots (for our case, $N = 72, 674$ and $K = 4750$). Singular value decomposition is performed on the weighted matrix,

$$\mathcal{V}^{1/2} \mathbf{Y} = \mathbf{\Phi} \mathbf{\Sigma} \mathbf{\Psi}^T, \quad (2)$$

where $\mathcal{V} = \text{diag}(V_1, V_2 \dots V_N, V_1, V_2, \dots, V_N)$ is a diagonal matrix with the cell volumes V_i in the main diagonal, $\mathbf{\Phi} \in \mathbb{R}^{2N \times K}$ contains the left singular vectors, $\mathbf{\Sigma} \in \mathbb{R}^{K \times K}$ is a diagonal

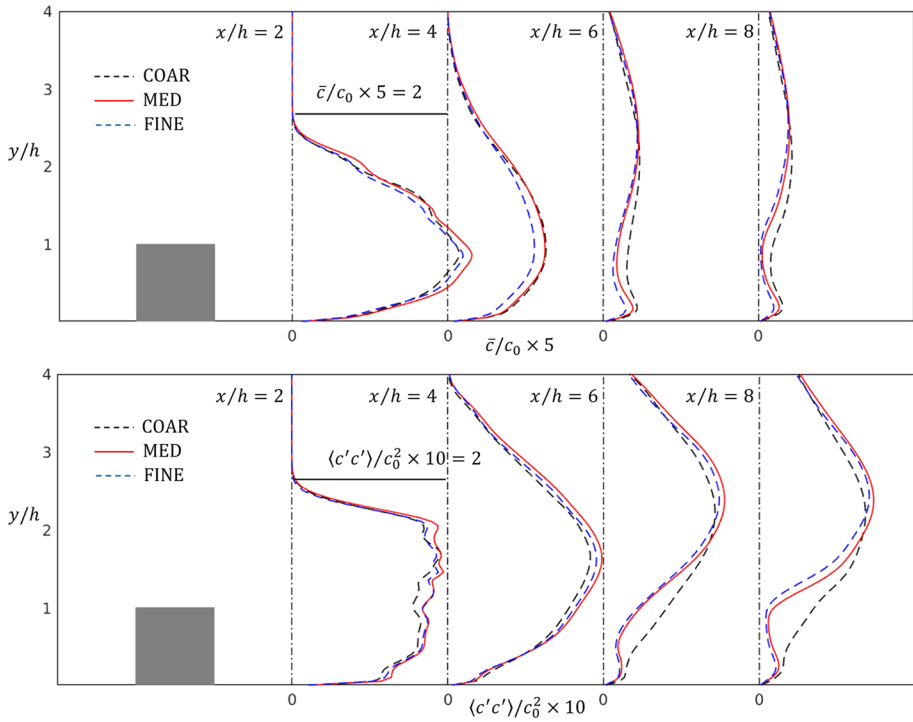


Fig. 4 Comparison of vertical profiles of mean scalar (top) and scalar variance (bottom) at $x/h = 2, 4, 6$ and 8 for three meshes. The reference value for scalar is $c_0 = 1$

matrix that stores K singular values, and $\Psi \in \mathbb{R}^{K \times K}$ contains the right singular vectors. The scaled POD eigenmodes $\phi_k(\mathbf{x})$ are extracted from the columns of Φ using,

$$\phi(\mathbf{x}) = \mathcal{V}^{-1/2} \Phi, \tag{3}$$

and satisfy the orthonormality condition,

$$\langle \phi_l, \phi_n \rangle = \iint_V \phi_l(x, y) \cdot \phi_n(x, y) dx dy = \begin{cases} 1, & l = n \\ 0, & l \neq n \end{cases} \tag{4}$$

The singular values σ_k are ranked in descending order along the diagonal of matrix Σ . The eigenvalues are computed from,

$$\lambda_k = \frac{\sigma_k^2}{K}, \tag{5}$$

and the time coefficients from,

$$\mathbf{a}(t) = \mathbf{Y}^T \mathcal{V}^{1/2} \Phi \tag{6}$$

The fluctuating velocity field can be written as

$$u'_i(x, y, t) = \sum_{k=1}^K a_k(t)\phi_k^{(i)}(x, y) \approx \sum_{k=1}^p a_k(t)\phi_k^{(i)}(x, y), \quad (i = 1, 2) \tag{7}$$

where p is the number of retained POD modes, $a_k(t)$ is the time coefficient of the k -th POD mode, and $\phi_k^{(i)}$ is the k -th POD eigenvector of the i -th velocity component.

We follow the same approach to obtain the scalar POD modes. The snapshot matrix $\mathbf{Z}(x, t_1 : t_K)$ for the scalar fluctuations c' is,

$$\mathbf{Z}(x, t_1 : t_K) = \begin{bmatrix} c'(\mathbf{x}_1, t_1) & c'(\mathbf{x}_1, t_2) & \dots & c'(\mathbf{x}_1, t_K) \\ \vdots & \vdots & \dots & \vdots \\ c'(\mathbf{x}_N, t_1) & c'(\mathbf{x}_N, t_2) & \dots & c'(\mathbf{x}_N, t_K) \end{bmatrix} \in \mathbb{R}^{N \times K}. \tag{8}$$

Note that the scalar field data are synchronized with the velocity data, thus the time instants t_i ($i = 1 \dots K$) in Eqs. (1) and (8) are the same. As before, we apply singular value decomposition to the weighted matrix $\mathcal{V}^{1/2}\mathbf{Z}$ (where now $\mathcal{V} = \text{diag}(V_1, V_2 \dots V_N)$) and obtain the scalar POD modes, $\psi_l(x, y)$, and time coefficients, $b_l(t)$ in a similar way. Thus we can write,

$$c'(x, y, t) = \sum_{l=1}^K b_l(t)\psi_l(x, y) \approx \sum_{l=1}^q b_l(t)\psi_l(x, y), \tag{9}$$

where q is the number of retained scalar POD modes.

4 Evolution Equations of the POD Coefficients

Before proceeding with the derivation of the dynamic estimator, it is very instructive to consider the form of the evolution equations of the POD time coefficients, $a_k(t)$ and $b_l(t)$. Details on the derivation of these equations for non-linear systems can be found in numerous references, Carlberg et al. (2011); Noack et al. (2011); Baiges et al. (2013); Xiao et al. (2013, 2015), to cite just a few.

The equations of the velocity fluctuations take the form,

$$\begin{aligned} \frac{\partial u'_i}{\partial x_i} &= 0 \\ \frac{\partial u'_i}{\partial t} + \frac{\partial u'_j \bar{u}_i}{\partial x_j} + \frac{\partial u'_i \bar{u}_j}{\partial x_j} + \frac{\partial p'}{\partial x_i} - \frac{1}{Re} \frac{\partial}{\partial x_j} \left(\frac{\partial u'_i}{\partial x_j} \right) &= \underbrace{\frac{\partial (\bar{u}'_i \bar{u}'_j - u'_i u'_j)}{\partial x_j}}_{=f_i}, \end{aligned} \tag{10}$$

where all the linear terms are moved in the left hand side and f_i contains the non-linear terms. Substituting (7) to (10), applying Galerkin projection, and taking into account that modes $\phi_k^{(i)}(x, y)$ are orthonormal and divergence-free, we get,

$$\begin{aligned} & \frac{da_n(t)}{dt} + \sum_{k=1}^p a_k(t) \left\langle \frac{\partial \bar{u}_i \phi_k^{(j)}}{\partial x_j}, \phi_n^{(i)} \right\rangle + \sum_{k=1}^p a_k(t) \left\langle \frac{\partial \bar{u}_j \phi_k^{(i)}}{\partial x_j}, \phi_n^{(i)} \right\rangle \\ & - \frac{1}{Re} \sum_{k=1}^p a_k(t) \left\langle \frac{\partial}{\partial x_j} \left(\frac{\partial \phi_k^{(i)}}{\partial x_j} \right), \phi_n^{(i)} \right\rangle = \langle f(t), \phi_n^{(i)} \rangle \end{aligned} \tag{11}$$

This can be written in matrix form as,

$$\frac{d\mathbf{a}(t)}{dt} + \mathcal{A}\mathbf{a}(t) = \mathbf{f}(t), \tag{12}$$

where $\mathbf{a}(t) = [a_1(t) \dots a_p(t)]^\top$, matrix \mathcal{A} incorporates the linear terms and $\mathbf{f}(t)$ includes for the non-linear terms and the error from the truncation of POD modes, for details see Iñigo et al. (2019).

We now apply a similar approach to the scalar equation. We provide all steps in the derivation of the evolution equation of $b_l(t)$ in order to make clear the influence of the velocity fluctuations. We start from the scalar transport equation that takes the form,

$$\frac{\partial c}{\partial t} + \frac{\partial u_j c}{\partial x_j} = \frac{1}{Pe} \frac{\partial}{\partial x_j} \left(\frac{\partial c}{\partial x_j} \right) + m, \tag{13}$$

where Pe is the Peclet number, $Pe \equiv ReSc$, Sc the Schmidt number (here set to 0.7) and m is the source term (per unit volume). Applying Reynolds decomposition and time-averaging, we get the Reynolds-averaged scalar equation,

$$\frac{\partial(\bar{u}_j \bar{c} + \overline{u'_j c'})}{\partial x_j} = \frac{1}{Pe} \frac{\partial}{\partial x_j} \left(\frac{\partial \bar{c}}{\partial x_j} \right) + m, \tag{14}$$

and subtracting (14) from (13), we obtain the evolution equation for the scalar fluctuations,

$$\frac{\partial c'}{\partial t} + \frac{\partial \bar{u}_j c'}{\partial x_j} + \frac{\partial u'_j \bar{c}}{\partial x_j} - \frac{1}{Pe} \frac{\partial}{\partial x_j} \left(\frac{\partial c'}{\partial x_j} \right) = \underbrace{\frac{\partial(\overline{u'_j c'} - u'_j c')}{\partial x_j}}_{=g} \tag{15}$$

Substituting the POD expansions (7) and (9) to (15), we get

$$\begin{aligned} & \sum_{l=1}^q \frac{db_l(t)}{dt} \psi_l + \sum_{l=1}^q b_l(t) \frac{\partial \bar{u}_j \psi_l}{\partial x_j} + \sum_{k=1}^p a_k(t) \frac{\partial \bar{c} \phi_k^{(j)}}{\partial x_j} \\ & - \frac{1}{Pe} \sum_{l=1}^q b_l(t) \frac{\partial}{\partial x_j} \left(\frac{\partial \psi_l}{\partial x_j} \right) = g \end{aligned} \tag{16}$$

We now apply Galerkin projection, i.e. multiply the scalar evolution equation with ψ_n and integrate over the whole domain to get,

$$\sum_{l=1}^q \frac{db_l(t)}{dt} \langle \psi_l, \psi_n \rangle + \sum_{l=1}^q b_l(t) \left\langle \frac{\partial \bar{u}_j \psi_l}{\partial x_j}, \psi_n \right\rangle + \sum_{k=1}^p a_k(t) \left\langle \frac{\partial \bar{c} \phi_k^{(j)}}{\partial x_j}, \psi_n \right\rangle - \frac{1}{Pe} \sum_{l=1}^q b_l(t) \left\langle \frac{\partial}{\partial x_j} \left(\frac{\partial \psi_l}{\partial x_j} \right), \psi_n \right\rangle = \langle g, \psi_n \rangle \quad (17)$$

Due to the orthonormality property of the scalar POD modes, (17) can be simplified as,

$$\frac{db_n(t)}{dt} + \sum_{l=1}^q b_l(t) \left\langle \frac{\partial \bar{u}_j \psi_l}{\partial x_j}, \psi_n \right\rangle + \sum_{k=1}^p a_k(t) \left\langle \frac{\partial \bar{c} \phi_k^{(j)}}{\partial x_j}, \psi_n \right\rangle - \frac{1}{Pe} \sum_{l=1}^q b_l(t) \left\langle \frac{\partial}{\partial x_j} \left(\frac{\partial \psi_l}{\partial x_j} \right), \psi_n \right\rangle = \langle g, \psi_n \rangle \quad (18)$$

We can write the above equation in matrix form as,

$$\frac{d\mathbf{b}(t)}{dt} + \mathcal{M}\mathbf{b}(t) + \mathcal{Q}\mathbf{a}(t) = \boldsymbol{\xi}(t), \quad (19)$$

where $\mathbf{b}(t) = [b_1(t) \dots b_q(t)]^T$. This equation shows that the evolution of $\mathbf{b}(t)$ is driven by the flow (effect encapsulated in $\mathcal{Q}\mathbf{a}(t)$) and also from the non-linear forcing $\boldsymbol{\xi}(t)$.

Putting (12) and (19) together we get,

$$\frac{d}{dt} \begin{bmatrix} \mathbf{a}(t) \\ \mathbf{b}(t) \end{bmatrix} + \underbrace{\begin{bmatrix} \mathcal{A} & 0 \\ \mathcal{Q} & \mathcal{M} \end{bmatrix}}_{=\tilde{\mathcal{A}}} \begin{bmatrix} \mathbf{a}(t) \\ \mathbf{b}(t) \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{f}(t) \\ \boldsymbol{\xi}(t) \end{bmatrix}}_{=\tilde{\mathbf{h}}(t)}. \quad (20)$$

or

$$\frac{d}{dt} \begin{bmatrix} \mathbf{a}(t) \\ \mathbf{b}(t) \end{bmatrix} = \tilde{\mathcal{A}} \begin{bmatrix} \mathbf{a}(t) \\ \mathbf{b}(t) \end{bmatrix} + \tilde{\mathbf{h}}(t). \quad (21)$$

In discrete form, the previous equation becomes

$$\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} [k+1] = \hat{\mathcal{A}} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} [k] + \hat{\mathbf{h}}[k]. \quad (22)$$

Matrix $\hat{\mathcal{A}}$ is unknown, but it exists. In the following section, we obtain the matrices of a discrete time, linear dynamical system describing the evolution of vector $[\mathbf{a}; \mathbf{b}]^T$ directly from data.

5 Data-Driven Dynamic Estimation from Sparse Measurements

5.1 System Identification

Following Eq. (22), we seek a system of the form,

$$\begin{aligned}
 \mathbf{x}[k + 1] &= \underbrace{\mathbf{A}}_{\text{System matrix}} \mathbf{x}[k] + \underbrace{\mathbf{w}[k]}_{\text{noise}} \\
 \underbrace{\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}}_{\text{Output}} [k] &= \underbrace{\mathbf{C}}_{\text{Output matrix}} \mathbf{x}[k] + \underbrace{\mathbf{v}[k]}_{\text{noise}},
 \end{aligned}
 \tag{23}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the state vector of size n (also known as order of the linear system), $\mathbf{A} \in \mathbb{R}^{n \times n}$ and $\mathbf{C} \in \mathbb{R}^{m \times n}$ where $m = p + q$, $\mathbf{w}[k] \in \mathbb{R}^n$ and $\mathbf{v}[k] \in \mathbb{R}^m$ are Gaussian white noise sequences. To this end, we use a system identification algorithm to obtain the system and output matrices \mathbf{A} and \mathbf{C} respectively, as well the noise covariance matrices. The unknown matrices are computed so that the output optimally matches with the true values of the POD coefficients, see also Savarino and Papadakis (2022).

System (23) is a generalised form of (22). More specifically, we have introduced the internal state $\mathbf{x}[k]$, with size n which is independent of the total number of modes, $m = p + q$. This form gives additional flexibility, as it allows one to adjust n , and thus the size of the matrices \mathbf{A} and \mathbf{C} , so that (23) returns a vector sequence $\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} [k]$ that optimally fits the one produced by DNS.

In this paper we use the `n4sid` algorithm, see van Overschee and de Moor (1994); Qin (2006) for details. Using the training dataset (see details in Sect. 6.3), the `n4sid` command of MATLAB estimates a discrete-time, state-space model of the following form,

$$\begin{aligned}
 \mathbf{x}[k + 1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{K}\mathbf{e}[k] \\
 \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} [k] &= \mathbf{C}\mathbf{x}[k] + \mathbf{e}[k]
 \end{aligned}
 \tag{24}$$

where $\mathbf{K} \in \mathbb{R}^{n \times m}$ is the gain matrix and $\mathbf{e}[k] \in \mathbb{R}^m$ is the output noise. The command provides the pair $\{\mathbf{A}, \mathbf{C}\}$, the gain matrix \mathbf{K} and the sequence $\mathbf{e}[k]$. The process noise covariance $\mathbf{Q} \in \mathbb{R}^{n \times n}$ is calculated from,

$$\mathbf{Q} = \text{cov}(\mathbf{K}\mathbf{e}) = \mathbf{K} \text{cov}(\mathbf{e})\mathbf{K}^T = \mathbf{K} \mathbb{E}(\mathbf{e}\mathbf{e}^T)\mathbf{K}^T,
 \tag{25}$$

where \mathbb{E} is the expectation operator and $\mathbb{E}(\mathbf{e}\mathbf{e}^T) = \left(\sum_{k=1}^{k=K} \mathbf{e}[k]\mathbf{e}^T[k] \right) / K$.

5.2 Construction of the Estimator

Suppose now that r velocity and/or scalar measurements are available. We collect these in the measurement vector $\mathbf{s}[k]$ of size r , and we can write,

$$\mathbf{s}[k] = \mathbf{S} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} [k] + \mathbf{g}[k],
 \tag{26}$$

where matrix $\mathbf{S} \in \mathbb{R}^{r \times m}$ is obtained by choosing the rows of eigenvectors $\phi(\mathbf{x})$ and $\psi(\mathbf{x})$ corresponding to the spatial locations of the sensors and quantity measured (velocity and/or scalar). The error vector $\mathbf{g}[k]$ arises from the truncation of the POD expansion and the inherent noise of any measuring device (here we consider that the latter is negligible).

In the training dataset, both $s[k]$ and $\begin{bmatrix} a \\ b \end{bmatrix} [k]$ are available, so the sequence $g[k]$ can be computed from,

$$g[k] = s[k] - S \begin{bmatrix} a \\ b \end{bmatrix} [k] \quad (27)$$

and its covariance $R \in \mathbb{R}^{r \times r}$ from $R = \mathbb{E}(gg^T)$.

We can now construct an estimator for the vector $\begin{bmatrix} a \\ b \end{bmatrix} [k]$ when new (i.e. unseen) measurements $s[k]$ from the validation data set are available. We denote the estimated quantities using a caret symbol, for example $\begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} [k]$ denotes the estimated vector of POD coefficients. The estimator known as Kalman observer, see Kailath et al. (2000), takes the form

$$\begin{aligned} \hat{x}[k+1] &= A \underbrace{\hat{x}[k]}_{\text{State estimate}} + \underbrace{\mathcal{L}}_{\text{Kalman gain}} \underbrace{(s[k] - \hat{s}[k])}_{\text{Estimation error}} \\ \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} [k] &= C\hat{x}[k] \\ \hat{s}[k] &= S \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} [k] = SC\hat{x}[k]. \end{aligned} \quad (28)$$

The estimator is driven by the error $s[k] - \hat{s}[k]$, i.e. the difference between the true, $s[k]$, and the estimated measurement, $\hat{s}[k]$. This difference is multiplied by the Kalman gain matrix \mathcal{L} , which is obtained by solving a Riccati equation, see Kailath et al. (2000)

$$\begin{aligned} P &= APA^T - AP(SC)^T(SCP(SC)^T + R)^{-1}SCPA^T + Q \\ \mathcal{L} &= AP(SC)^T(SCP(SC)^T + R)^{-1} \end{aligned} \quad (29)$$

We close this section by stressing the flexibility of the proposed approach. Vector $s[k]$ can contain velocity measurements only, scalar measurements only, or both. For all combinations, we can estimate the coefficients of both the velocity and scalar modes. In Sects. 6.3 and 6.4 below, we consider the estimation of velocity field using velocity only, or scalar only, measurements respectively. Of course, the scalar field is estimated at the same time, but in this paper we show results only for the velocity field.

6 Results and Discussion

6.1 Flow Patterns

Streamlines and contours of the time-averaged u and v velocities as well as vorticity are presented in Fig. 5. There are four recirculation zones. One is in front of the prism (marked with the letter F) and extends about $X_F/h = 2.4$ upstream. A small zone of slow reverse flow, denoted by T, can be detected between the high shear region and the top face of the prism; this zone is also reported in Vincont et al. (2000). Just behind the prism, there is a third small zone with positive vorticity, denoted by N, that extends about $0.77h$

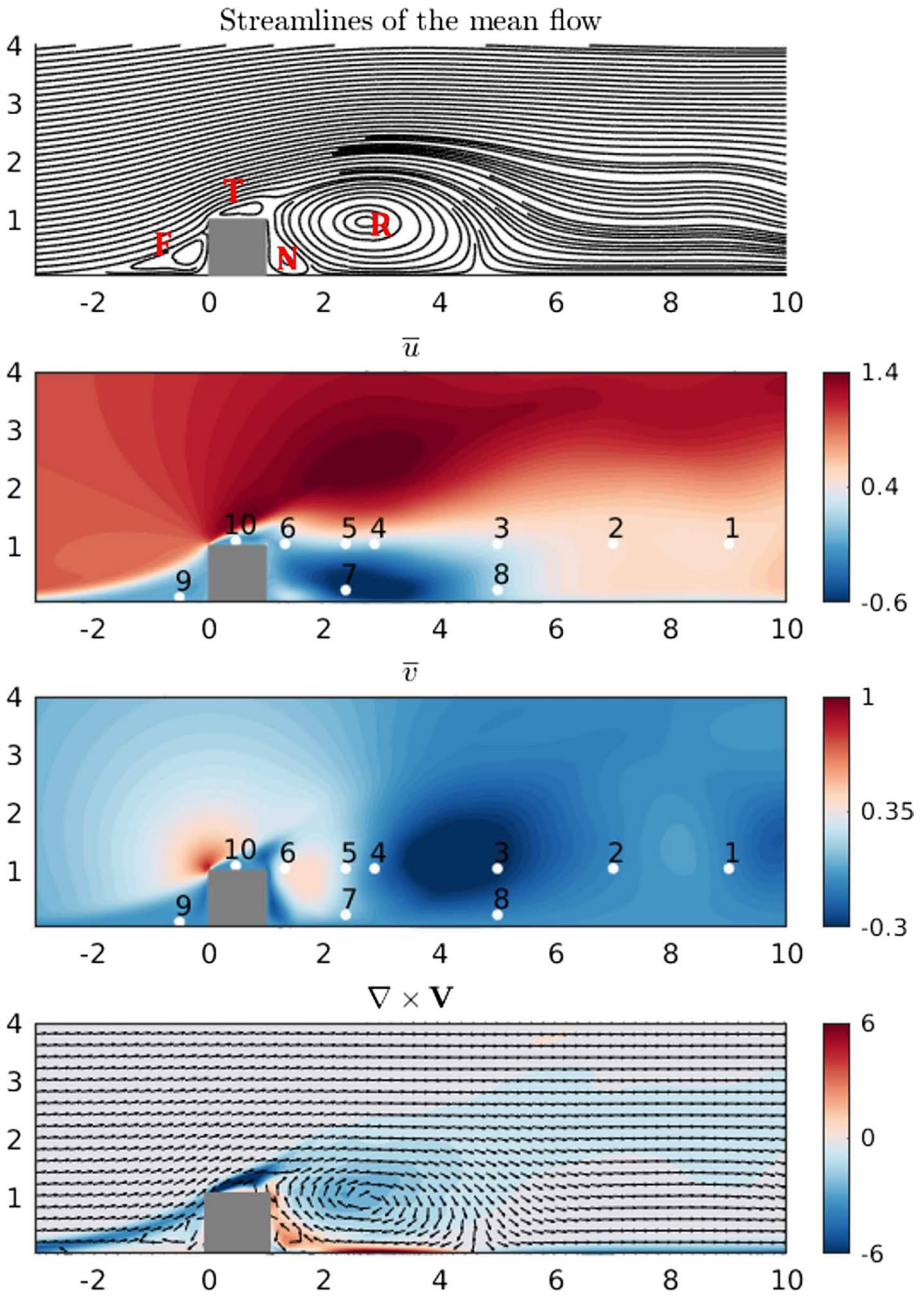


Fig. 5 Streamlines (top), contours of mean streamwise (second from top), cross-stream velocities (second from bottom) and vorticity (bottom). Velocity and scalar spectra at the probe locations marked in the middle plots are shown in Fig. 8

downstream. A much larger recirculation zone, denoted by R, forms the main wake of the prism. The flow reattaches at $X_R/h = 4.97$, i.e. the length is $L_R/h = 3.97$. In Fragos et al. (2012), the time-average recirculation length behind a 2D prism is found to be approximately $4.1h$ for $Re = 1304$ and uniform inlet flow. On the other hand, Diaz-Daniel et al. (2017) report a smaller value of $2.19h$, but this is for a 3D simulation around a cube flow at a close Reynolds number, $Re = 1100$.

Contour plots of the Reynolds stresses are shown in Fig. 6. The maximum $\langle u'^2 \rangle$ is located in the wake, near the wall, behind the prism. The red region (marking high values) starts from the boundary of the corner vortex (N) and the large recirculation zone (R). The interaction between these flow structures and the wall result in large u' fluctuations. Large values of $\langle v'^2 \rangle$ are found inside the large recirculation zone (R). The positive peak of shear stress $\langle u'v' \rangle$ is found behind the trailing edge of the prism, while the negative peak occurs inside the large recirculation zone (R). The region of large TKE values is a combination of the regions with large $\langle u'^2 \rangle$ and $\langle v'^2 \rangle$.

Contours of the mean scalar and its variance are shown in Fig. 7. The scalar concentration attains the largest value at the location of the source, as expected. The dispersion behind the prism can be seen from the variance plot; its shape shows similarities with the $\langle v'^2 \rangle$ and TKE plots. This confirms that the velocity field is imprinted on the scalar fluctuations. The task of the estimator is to extract the velocity fluctuation from scalar measurements.

In total 10 probes were placed in the computational domain to record the velocity and scalar fluctuations; the locations are shown in Figs. 5, 6 and 7 superimposed to contours of different quantities. Probes 1, 2, 3, 4 are separated by spacing $x/h = 2$ but have the same y coordinates. Probe 4 is at the peak of $\langle v'^2 \rangle$. Probe 7 is located at the peak of $\langle u'^2 \rangle$. The y -coordinate of probe 5 is the same as probe 6, which is placed at the peak of $\langle u'v' \rangle$. Probe 8 is mounted on the same wall-normal distance as probe 7 to track the fluctuations of flow and scalar near the wall. Probe 9 is mounted in front of the prism to track the flow dynamics of the front vortex, while probe 10 is mounted on top of the prism to extract information on the separated shear layer.

The frequency spectra of streamwise velocity fluctuations at 6 probes are shown in Fig. 8. At the same distance $y/h = 1.025$ above the wall, probes 1, 2, 3 have the same frequencies at $St = 0.063, 0.131, 0.194$ (only results from probe 3 are shown). The second and third harmonics at $St = 0.131, 0.194$ appear only at points further away from the prism. This is confirmed by inspecting the spectra of probes 4, 6 and 7; they contain only the dominant frequency $St = 0.063$ due to their proximity to the prism. Interestingly, point 7

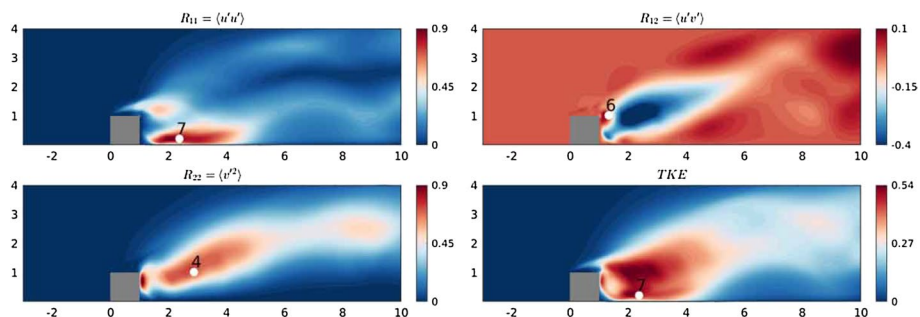


Fig. 6 Contour plots of the Reynolds stresses and kinetic energy of the fluctuating flow

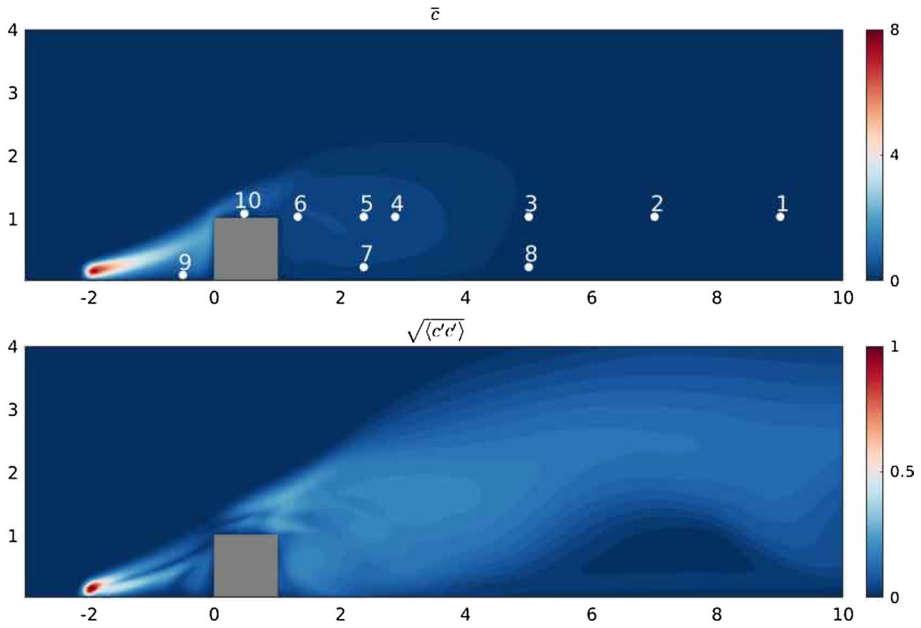


Fig. 7 Contour plots of the mean scalar field (top) and scalar variance (bottom)

has higher peak compared to 6, most likely because of the interaction of the unsteady recirculation zone R with the wall. The footprint of the dominant frequency at probes 9 and 10 is weak (the power spectral density is at least an order of magnitude smaller compared to the other points) indicating that strong unsteadiness is confined downstream of the prism.

The frequency spectra of the scalar fluctuations at the same probe points are shown in Fig. 9. The same dominant frequency $St = 0.063$ appears at all points, but contrary to the velocity spectra, higher harmonics do not have a strong presence at point 3. This is probably because the scalar concentration is diluted further away from the prism, leaving a weak footprint at higher frequencies. On the other hand, stronger harmonics appear in points 4, 6 and 7 that are closer to the prism. Probes 9 and 10 have again low spectral content because of weak unsteadiness at their location. The spectra shown in Figs. 8 and 9 demonstrate that the frequency content of flow field is imprinted on the scalar field, making the reconstruction of the former from sparse scalar measurements a meaningful exercise.

6.2 POD Modes

6.2.1 Velocity Modes

Figure 10 shows the energy content of the first 30 modes. The left plot shows the energy fraction of each mode, where three pairs of modes can be detected, and the members of each pair have similar energy content. The right plot displays the cumulative energy; the first 7 modes account for 97% of the total energy.

The shape of the dominant modes and spectra of the time coefficients are shown in Fig. 11. The first two modes share the same dominant shedding frequency at $St = 0.063$.

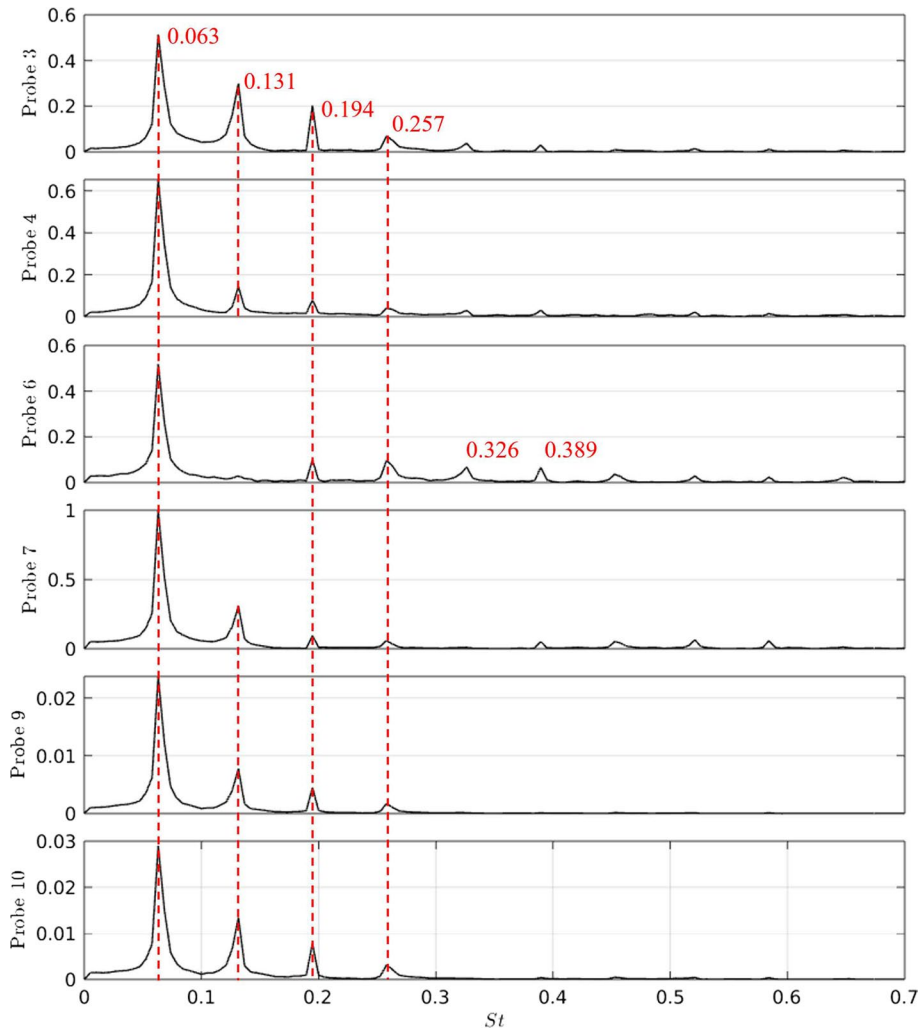


Fig. 8 Spectra of streamwise velocity fluctuations at selected probe points

There is a spatial shift between the modes which, combined with the sharp spectral peak, indicates propagating structures. The next two modes peak at the second and third harmonics $St = 0.131, 0.194$, while the third pair introduces the fourth harmonic $St = 0.257$. The higher harmonics are generated by the nonlinear interactions between lower harmonics, and result in smaller sized structures; this can be more easily seen by inspecting the v' modes. The first 6 modes share a similar spatial distribution of the eigenvectors (again more evident in v' modes). Starting from mode 7, small-scale structures can be observed at the trailing edge of the prism. They also introduce higher frequency peaks at $St = 0.326$. It is reasonable to discard the modes beyond the seventh, since they contribute very little to the fluctuating kinetic energy. Overall, the POD decomposition of the velocity field has succeeded in capturing the important flow dynamics and the structures associated with the dominant frequency.

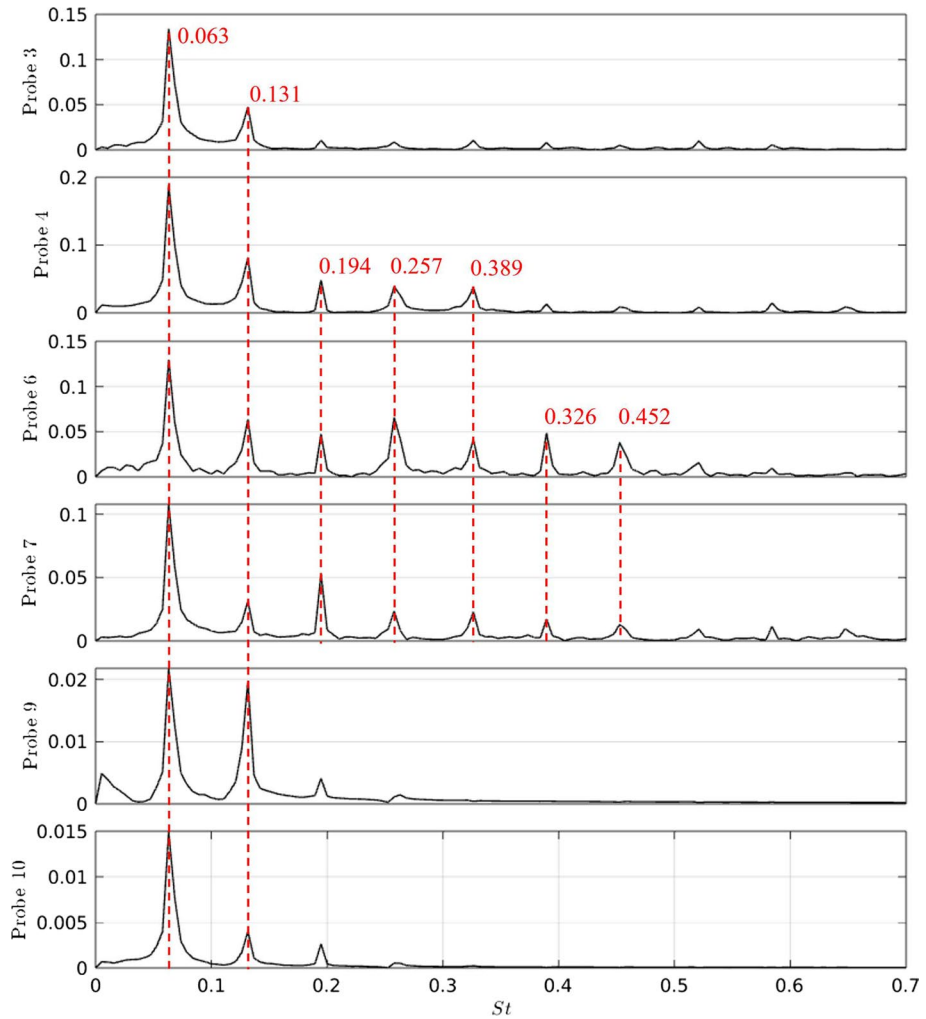


Fig. 9 Spectra of scalar fluctuations at selected probe points

6.2.2 Scalar Modes

Figure 12 shows the variance content of the scalar POD modes.

The first 9 modes (shown in red-shaded area) account for 97% of scalar variance. Thus more modes are needed for the scalar compared to the flow field to capture the same percentage of the energy. It is interesting to note that the dominant modes also appear in pairs, but there is significant energy difference between the members of each pair, larger compared to the velocity modes.

Figure 13 shows that the spectra of modes 1 and 2 contain the dominant frequency $St = 0.063$. The spatial structures are also shifted in space. The second set of modes peak at the second and third harmonics, $St = 0.131, 0.194$. The next pair of modes introduce the fourth harmonic frequency at $St = 0.257$. As the mode number increases, smaller-size structures appear in the spatial distribution. It is interesting to note that while modes 1–6

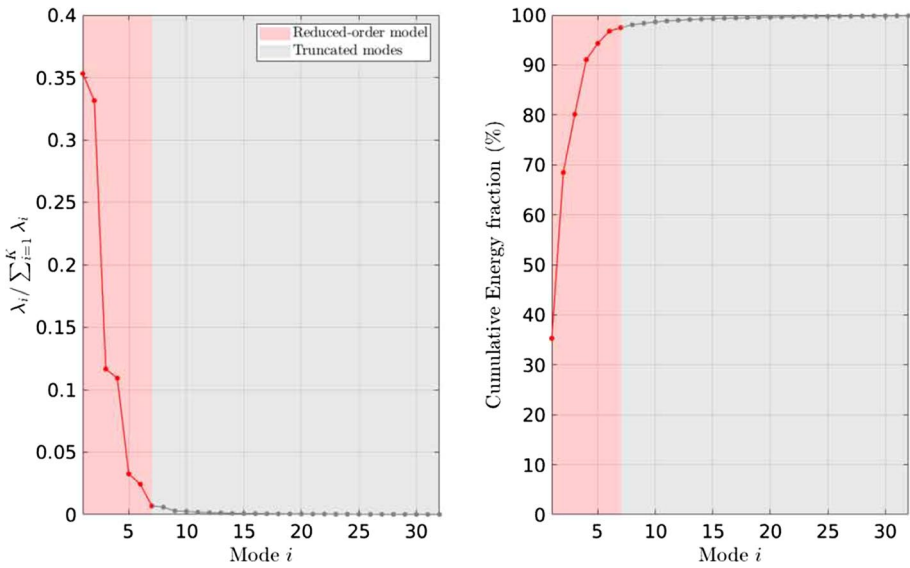


Fig. 10 Energy content of the first 30 velocity POD modes (left) and cumulative content (right). The $p = 7$ modes retained for the ROM in Sect. 6.3 are shown inside the red area, while the other modes in the grey area are discarded

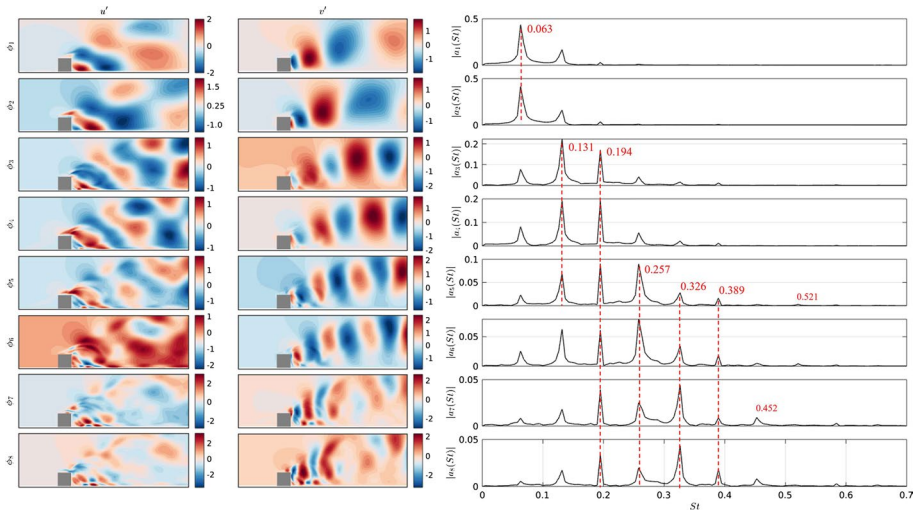


Fig. 11 Spatial distributions of $\phi_i^{(u)}$ (left column), $\phi_i^{(v)}$ (middle column) and spectra of time coefficients $a_i(t)$ (right column) of the dominant velocity POD modes

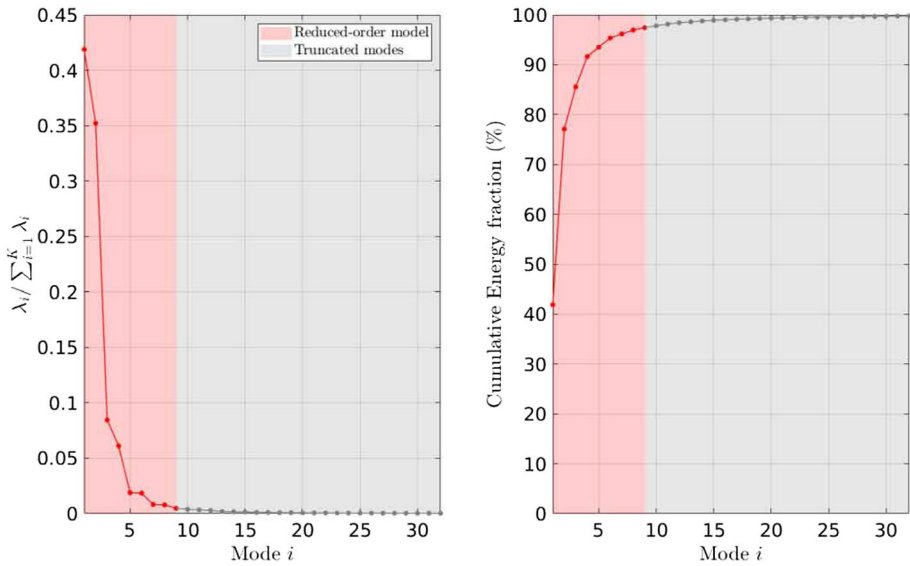


Fig. 12 Variance content of the first 30 scalar POD modes (left) and cumulative content (right). The $q = 9$ modes retained for the ROM in Sect. 6.3 are shown inside the red area, while the other modes in the grey area are discarded

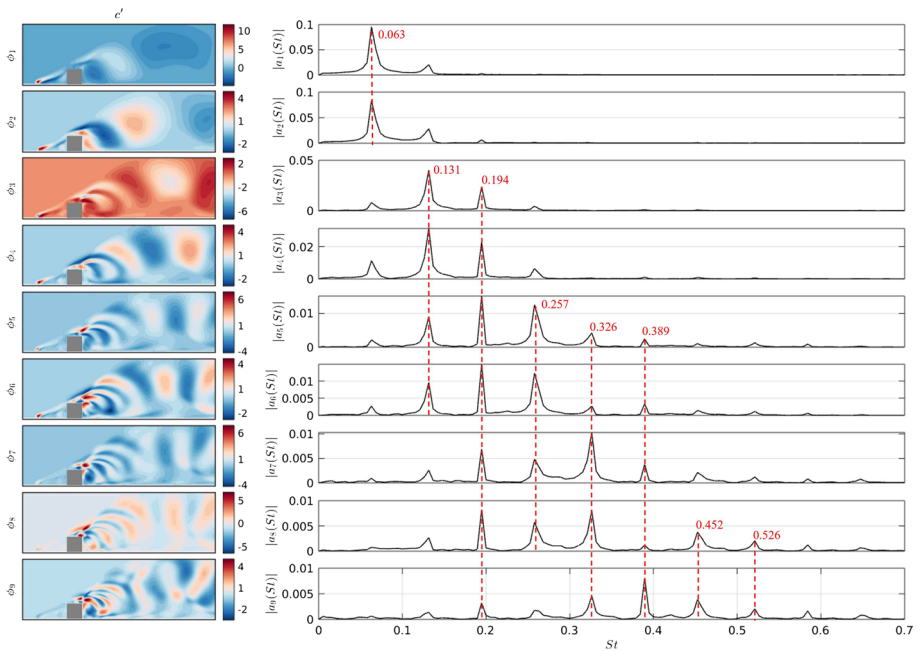


Fig. 13 Spatial distribution of the dominant scalar POD modes (left column) and spectra of the time coefficients $b_i(t)$ (right column)

have presence upstream of the prism, modes 7–9 contain small-scale structures that are spatially localised downstream of the prism.

6.3 Velocity Field Reconstruction from Velocity Measurements Only

In this section, we reconstruct the velocity field only from the velocity measurements, and in the following Sect. 6.4 from scalar measurements. In both cases, the sensors are placed at the velocity POD mode peaks; 5 locations are shown in Fig. 14. Modes 1 & 2 peak at $x/h \approx 2.5$, while modes 3–5 further downstream. At these locations, either both $u'(t)$ and $v'(t)$ are measured, or only $c'(t)$.

The time coefficients of the first 7 velocity POD modes are used in the model (23), thus $p = 7$, $q = 0$ and $m = p + q = 7$. We use half the snapshots (i.e. 2375) to extract the model matrices (this is known as training dataset) and the rest of the snapshots to validate the estimator (validation dataset). The reconstruction quality is quantified with the percentage FIT_i between the estimated $\hat{a}_i[k]$ and the true $a_i[k]$ coefficient of the i -th velocity mode,

$$FIT_i [\%] = 100 \left(1 - \frac{\|a_i[k] - \hat{a}_i[k]\|}{\|a_i[k] - \overline{a_i[k]}\|} \right), \quad (30)$$

where in our case $\overline{a_i[k]} = 0$. This parameter is equal to 100% for perfect matching and can become negative in case of large deviations between the true and estimated values. In the above equation, $\|\cdot\|$ denotes the \mathcal{L}_2 norm of a time signal.

For each mode, we compute the FIT value for different numbers of sensors, r , and model orders, n . The results are plotted in Fig. 15. For all model orders, and for a fixed number of sensors, the reconstruction quality is gradually reduced for higher POD modes. As the number of sensors increases, the quality increases significantly, and the improvement is more substantial for the higher order modes. It is interesting to examine the effect of model order for a fixed number of sensors. The reconstruction quality tends to get worse for model order n significantly larger than $m (= 7)$. For example, the FIT values are negative for modes 6, 7 with one or two sensors at model order $n = 30$. The most likely reason is that when increasing n , the number of elements in the model matrices becomes very large (it grows quadratically with n), but the training data set does not provide rich enough information to compute the matrices accurately. The reconstruction quality thus suffers, especially for large POD modes. It should be mentioned that the `n4sid` algorithm can

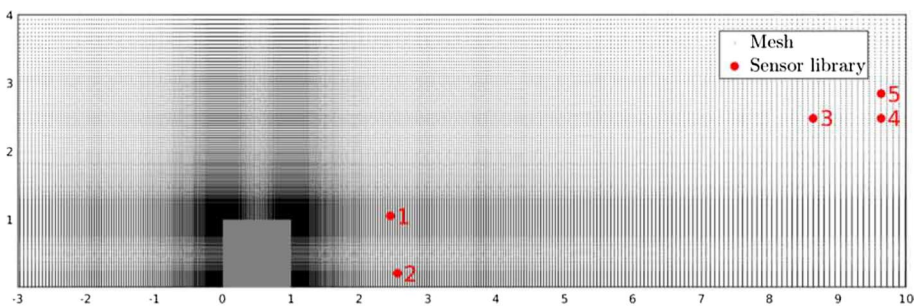


Fig. 14 Locations of velocity magnitude peaks for modes 1–5 (shown in red dots)

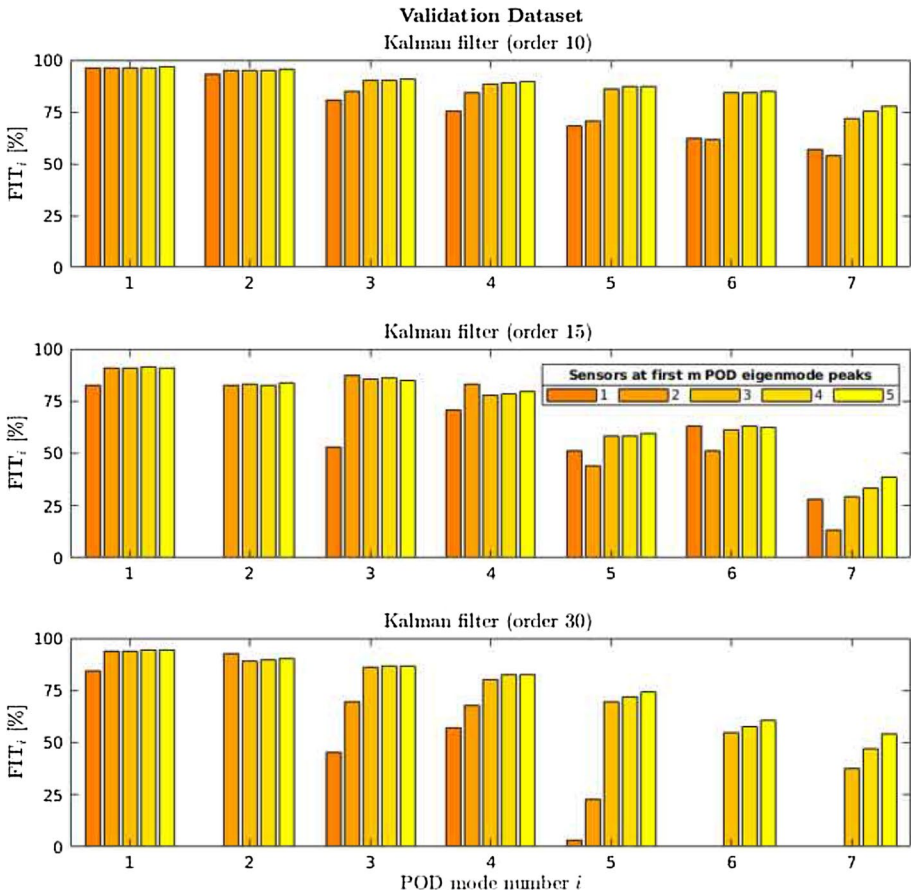


Fig. 15 FIT[%] of the first 7 POD coefficients for three different model orders, $n = 10, 15, 30$

provide the optimal value of n , but in the current MATLAB implementation the optimal value is restricted to less or equal 10, we thus explored larger n values manually.

A model order $n = 10$ is chosen to further assess the reconstruction of the flow statistics and the reproduction of the instantaneous velocity and vorticity fields.

Figure 16 presents the reconstructed and the true vorticity fields at two time instants $t = 190, 199$. The time difference is nearly half of one flow-through time. The large-scale vortical structures are captured well by the reconstruction. However, some localised structures, for example the thin shear layers at $x/h = 4$ ($t = 190$) and between $x/h = 1 - 3$ ($t = 199$), are smeared out. There are not significant differences between 1 and 5 sensors, at least visually.

Figure 17 compares the true and reconstructed flow statistics (Reynolds stresses and kinetic energy of the fluctuating field) using 1 and 5 sensors. Even if a single sensor point is used (middle column), the reconstruction of all statistics is very good. Adding more sensors does not provide additional accuracy.

This is confirmed by Fig. 18 that shows quantitative comparison of $\langle u'u' \rangle$ and $\langle u'v' \rangle$ profiles at streamwise locations $x/h = 2, 4, 6, 8$ against the DNS results. It can be

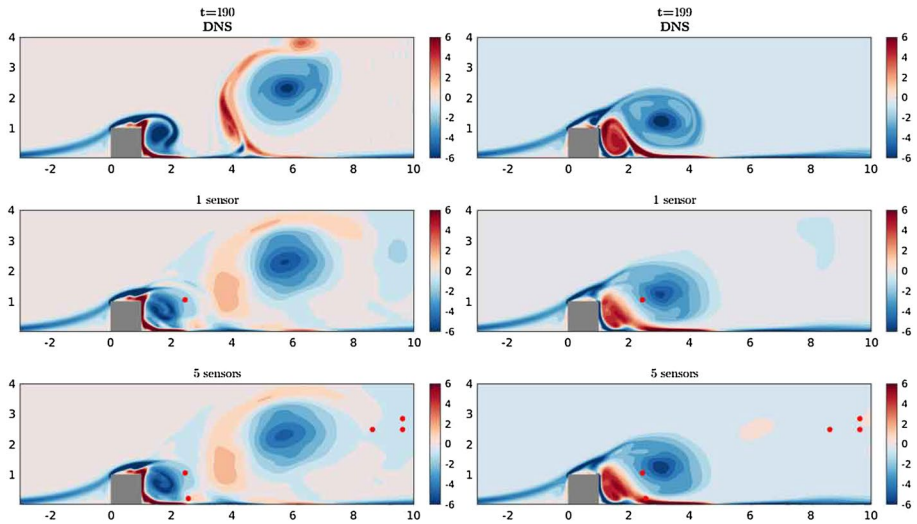


Fig. 16 Reconstruction of instantaneous vorticity magnitude with 1 and 5 sensors (shown as red dots in the middle and bottom rows) and comparison with DNS data (top row)

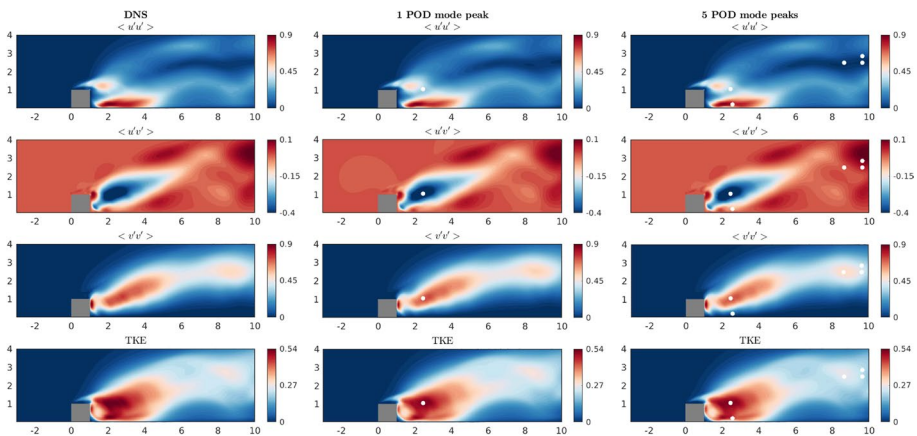


Fig. 17 Reconstruction of flow statistics with 1 and 5 sensors (shown as white dots) and comparison with DNS data

seen that the quality of reconstruction is excellent and more sensors do not offer any improvement. This is most likely due to the fact that more than 90% of the energy can be captured by 4 modes only (see Fig. 10), and all have strong footprint at the location of sensor 1. Intuitively, we expect that as the Reynolds number increases and the modes become more spatially compact, adding more sensors will improve accuracy.

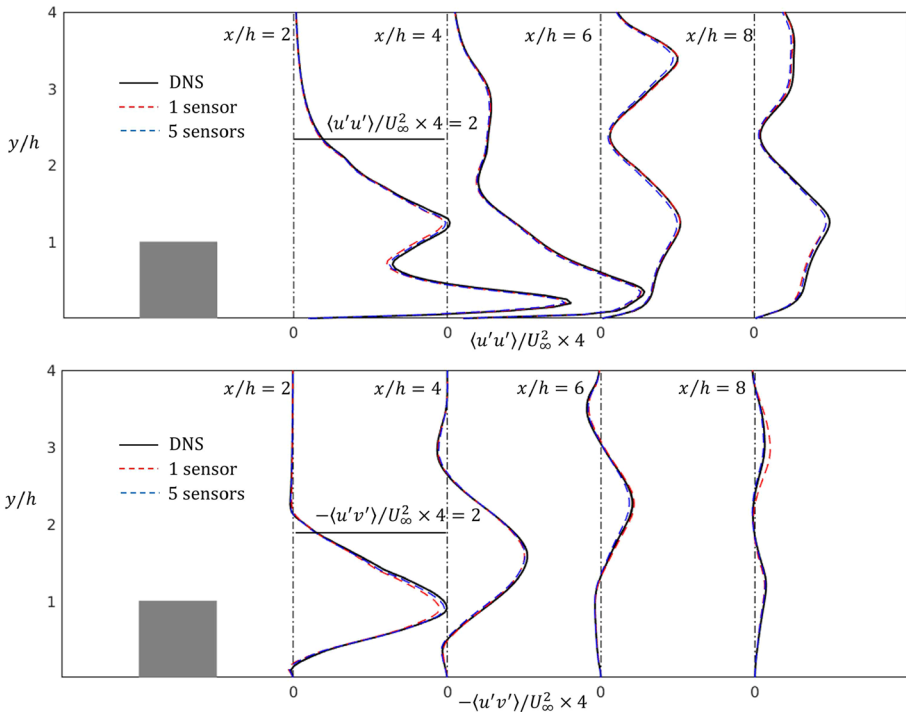


Fig. 18 Comparison between the true and reconstructed $\langle u'u' \rangle$ (top) and $\langle u'v' \rangle$ (bottom) profiles at $x/h = 2, 4, 6, 8$

6.4 Velocity Field Reconstruction from Scalar Measurements Only

In the previous section, we used only velocity measurements to reconstruct the velocity field. In this section, we test the potential to reconstruct the velocity field from scalar measurements only. To this end, $p = 7$ modes of velocity and $q = 9$ modes of scalar (that capture 97% of the fluctuating energy and scalar variance respectively) are used, thus $m = p + q = 16$. The scalar sensors are mounted at the peaks of the velocity POD modes, as already mentioned. The number of sensors is varied from 1 to 5. The effect of model order n is assessed by considering values between 20 and 40.

Figure 19 compares the reconstruction quality of different estimators for velocity (left column) and scalar modes (right column). Comparison with Fig. 15 shows that the FIT values for all velocity modes are significantly lower. The degradation in the reconstruction quality is expected, since now only one variable (scalar concentration) is measured at each sensor point, while in the previous section both velocity components were measured at each location. Velocity modes 1 and 2 are better reconstructed compared to the others. Again, increasing the model order to high values results in poorer performance (compare results with $n = 30$ and $n = 40$). The number of sensors has a mixed effect on the reconstruction quality, but recall that these sensors are placed on the peaks of velocity magnitude, not scalar, modes. Also sensors 3-5 are located away from the prism, in a region where very small values of scalar are recorded because of the dilution effect. Comparison between the left and right columns indicates that the scalar POD modes are generally

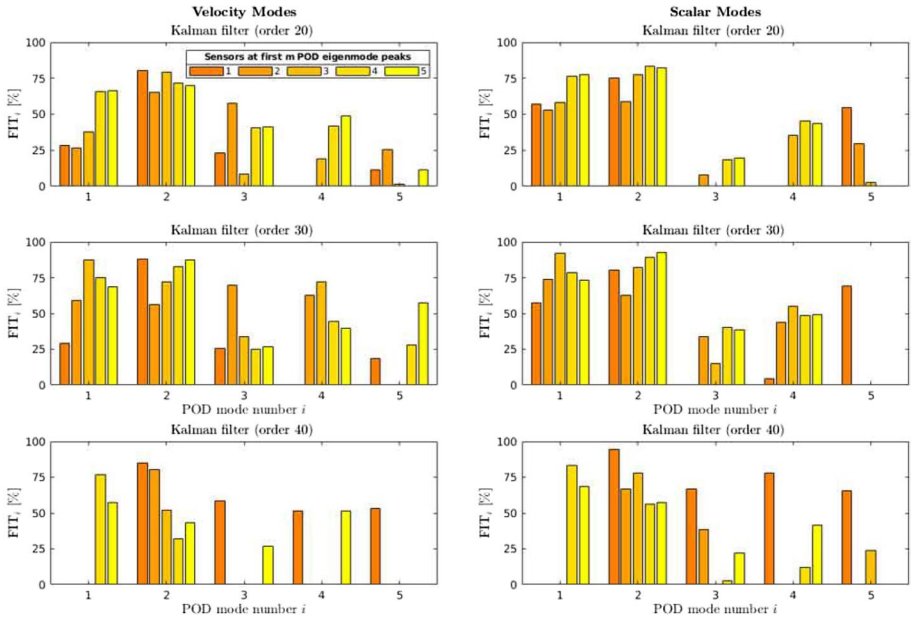


Fig. 19 FIT [%] of the first 5 velocity (left column) and scalar (right column) POD coefficients for model orders $n = 20, 30, 40$

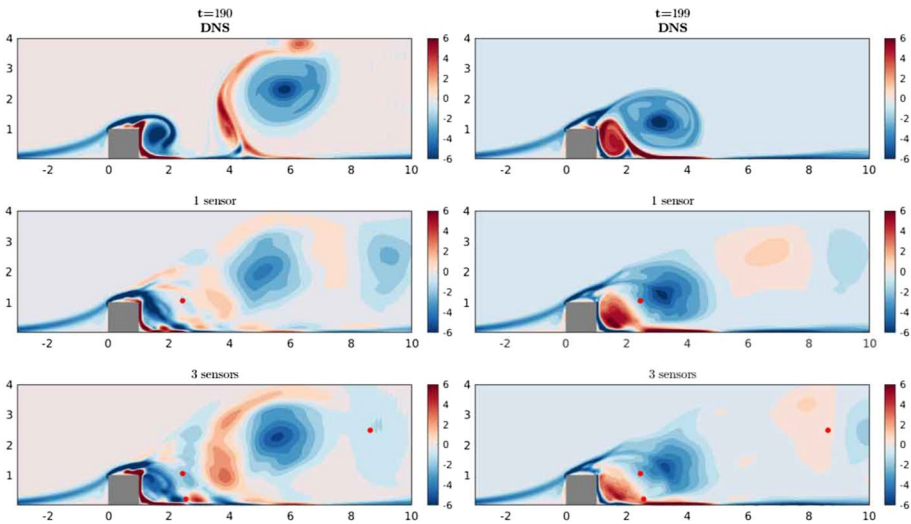


Fig. 20 Instantaneous vorticity fields at two time instants, true fields (top row) and reconstructed fields using 1 (middle row) and 3 scalar sensors (bottom row)

better reconstructed than the velocity modes. In the following we consider in more detail the results with $n = 30$.

Figure 20 shows the true and reconstructed vorticity fields (with 1 and 3 sensors) at the same time instants as in Fig. 16. The main features, such as the large scale vortices, are

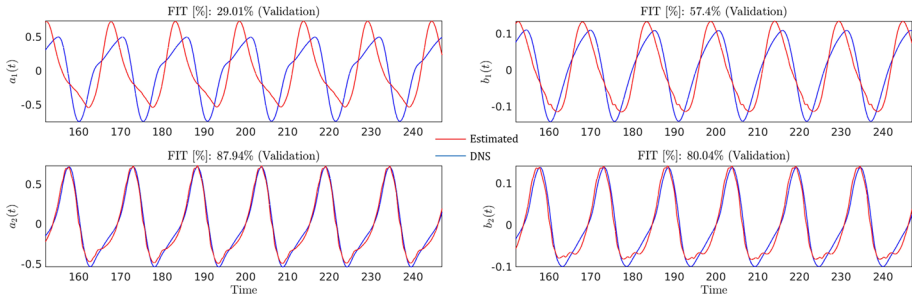


Fig. 21 Time coefficients $a_1(t)$, $a_2(t)$ for velocity modes (left column) and $b_1(t)$, $b_2(t)$ for scalar modes (right column) using a single sensor and $n = 30$. Red lines: estimated coefficients, Blue lines: DNS. Velocity coefficient variances, $\overline{a_1^2(t)} = 0.167$ (DNS), 0.171 (est), $\overline{a_2^2(t)} = 0.154$ (DNS), 0.153 (est). Scalar coefficient variances $\overline{b_1^2(t)} = 0.0073$ (DNS), 0.0075 (est), $\overline{b_2^2(t)} = 0.0059$ (DNS), 0.006 (est)

well reproduced at both time instants, but finer details (such as the shear layers close to the prism) are not so well captured. Note also a small difference (delay) in the location of the large vortex; in DNS the core is slightly further downstream at $t = 190$. In general, the instantaneous reconstruction is acceptable, but not as good as with velocity sensors.

The above features can be explained by inspecting the time coefficients for the first two velocity and scalar modes shown in Fig. 21. It can be seen that mode 2 is better estimated compared to mode 1; this is in agreement with the FIT values of Fig. 19. For mode 1 there is a small deviation in the peak values and a phase shift that explains the small discrepancy in the location of the main vortex in Fig. 20. The variances however of both coefficients are well predicted, see figure caption.

Figure 22 shows the reconstructed flow statistics. Using just a single sensor point (middle column) provides very satisfactory results. From (7), for the streamwise Reynolds stress for example we have $\overline{u'^2(t)}(x, y) = \sum_{k=1}^p \overline{a_k^2(t)} [\phi_k^{(1)}(x, y)]^2$ because $\overline{a_i(t)a_j(t)} = 0$ ($i \neq j$) and

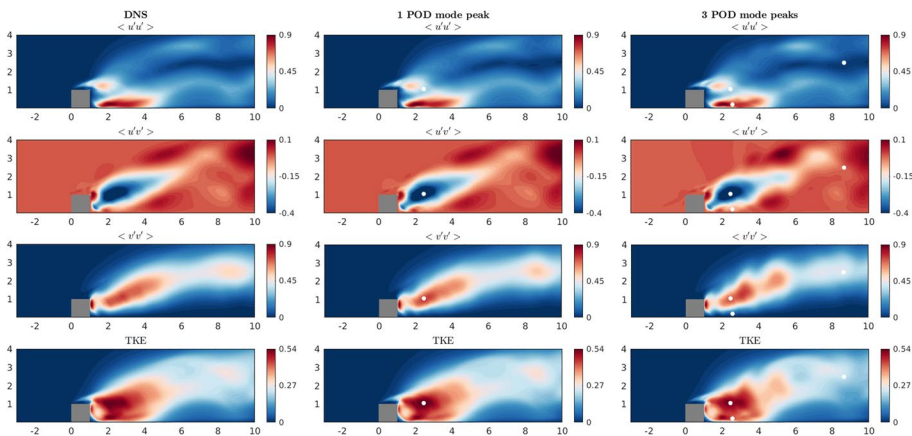


Fig. 22 Reconstruction of flow statistics from Kalman filter with different number of c' sensors and comparison with DNS data

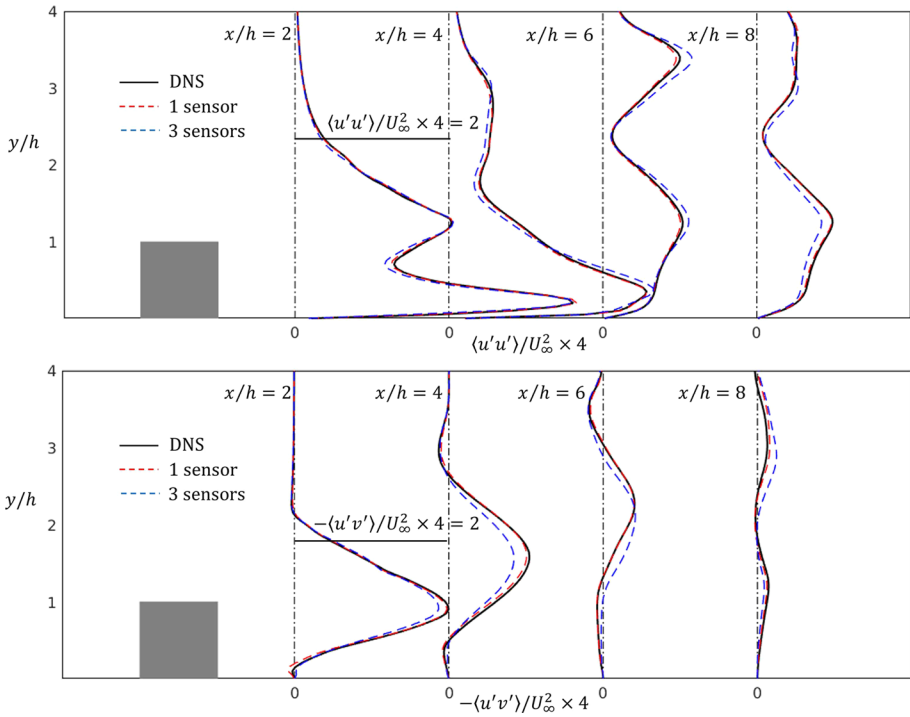


Fig. 23 Comparison of the true and reconstructed $\langle u'u' \rangle$, $\langle u'v' \rangle$ profiles at $x/h = 2, 4, 6, 8$ using 1 and 3 scalar sensors

since the variances of the POD coefficients are well captured, so are the stresses. Increasing the number of sensor points to 3 (right column) doesn't guarantee better quality for all statistics. This result is consistent with the previous Fig. 19.

Quantitative comparison at 4 streamwise locations is shown in Fig. 23. The results are very satisfactory, and there are small differences between 1 and 3 sensors. These figures clearly demonstrate that, despite the complexity of the flow, the proposed algorithm is remarkably successful in reconstructing the flow statistics using just a few sparse scalar measurements.

6.5 Flow Reconstruction at Off-Design Conditions from Scalar Measurements Only

In this section, the robustness of model identified at $Re = 1000$ is assessed at two off-design conditions, $Re = 800, 1200$. To this end, we performed two additional simulations (using the medium mesh) and collected velocity and scalar snapshot data. The data were then projected to the POD modes obtained at $Re = 1000$ and the time coefficients at the new conditions were obtained. Scalar measurements at the same locations as in the previous section were used as inputs to the estimator. We employed $p = 7$ velocity and $q = 9$ scalar modes and model order $n = 30$. Derivation of the estimator at one Re and application to another offers true predictive ability.

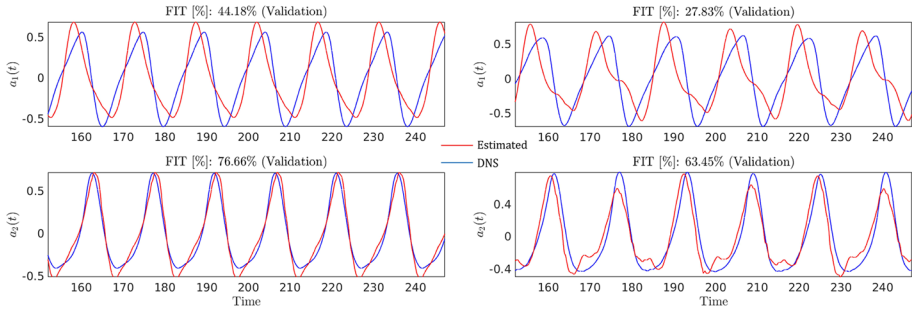


Fig. 24 Time coefficients for modes 1 & 2 at two off-design conditions using a single sensor: $Re = 800$ (left), $Re = 1200$ (right). Red lines: estimated coefficients, Blue lines: DNS. Coefficient variances, for $Re = 800$, $a_1^2(t) = 0.142$ (DNS), 0.156 (est), $a_2^2(t) = 0.141$ (DNS), 0.158 (est), and for $Re = 1200$, $a_1^2(t) = 0.185$ (DNS), 0.158 (est), $a_2^2(t) = 0.167$ (DNS), 0.133 (est)

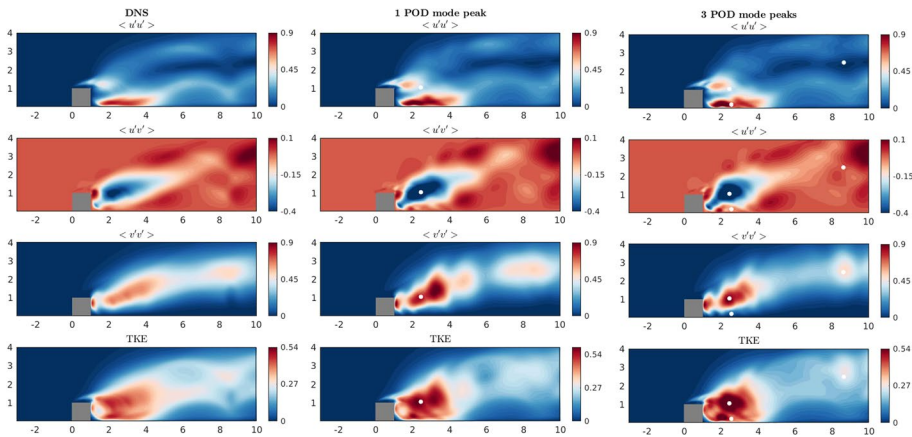


Fig. 25 Reconstruction of flow statistics with different number of c' sensors (shown as white dots) and comparison with DNS data at the off-design condition of $Re = 800$

The reconstructed temporal coefficients for modes 1 & 2 are compared with the true data in Fig. 24 for $Re = 800$ (left column) and $Re = 1200$ (right column). There are some deviations (especially for mode 1 at $Re = 1200$) but the results are very satisfactory. The variances of the coefficients are also well reproduced, see figure caption.

Figure 25 compares the reconstructed flow statistics with the true statistics at the off-design condition $Re = 800$. The general spatial distribution of the Reynolds stresses is captured by the algorithm, but there are some deviations from the true data. In particular, the peak values of $\langle v'v' \rangle$ and TKE are over-estimated in the reconstruction (darker region). Adding more sensors doesn't provide better reconstruction results.

A quantitative comparison between the true and reconstructed profiles of $\langle u'u' \rangle$ and $\langle u'v' \rangle$ is shown in Fig. 26. The shape is correctly captured but the peak values at the first streamwise location $x/h = 2$ are overpredicted. Since the variances of the coefficients are well reproduced, the most likely explanation is that the retained $p = 7$ POD modes

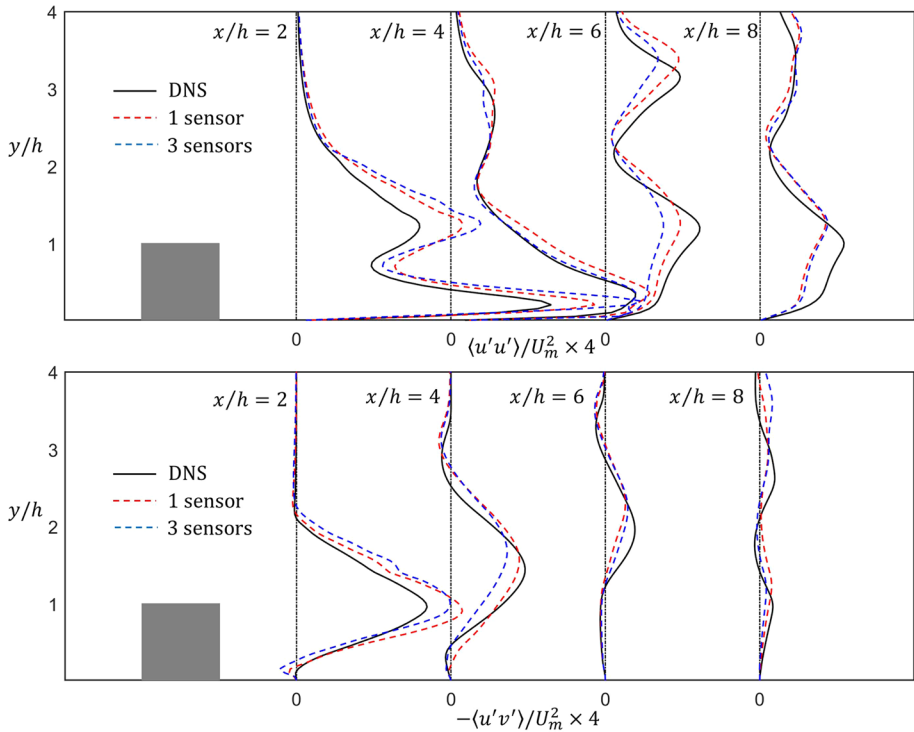


Fig. 26 Comparison of the true and reconstructed $\langle u'u' \rangle$ and $\langle u'v' \rangle$ profiles at $x/h = 2, 4, 6, 8$ at the off-design condition of $Re = 800$

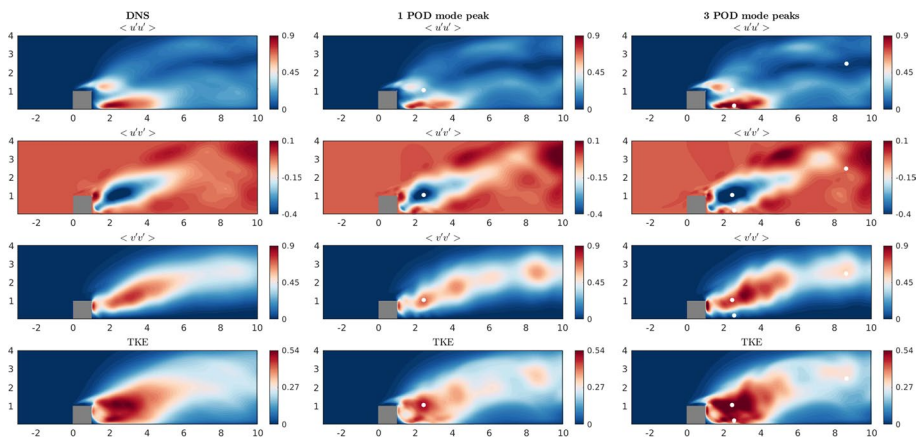


Fig. 27 Reconstruction of flow statistics with different number of c' sensors (shown in white dots) and comparison with DNS data at the off-design condition of $Re = 1200$

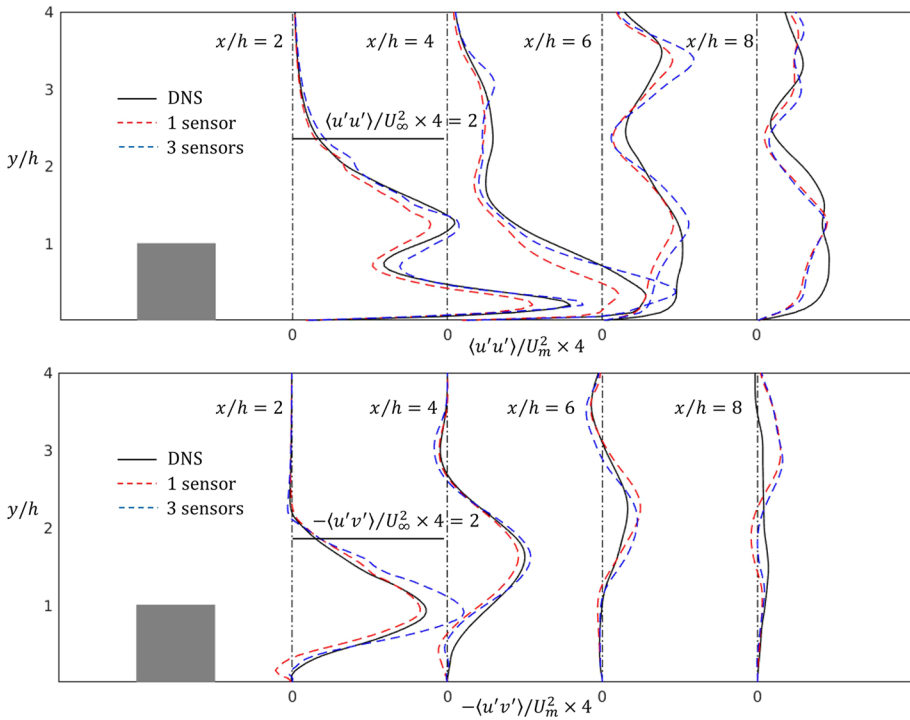


Fig. 28 Comparison of the true and reconstructed $\langle u'u' \rangle, \langle u'v' \rangle$ profiles at $x/h = 2, 4, 6, 8$ at the off-design condition of $Re = 1200$

evaluated at $Re = 1000$ cannot provide an accurate representation of the fluctuating velocity field at $Re = 800$, especially close to the wall and the separating shear layer.

The reconstructed flow statistics at $Re = 1200$ are shown in Fig. 27. Again the general shape is well captured, while the maximum values are closer to the true ones. A quantitative comparison is shown in Fig. 28. It can be seen that the reproduction of the statistics is better for this higher Re compared to the previous one.

It is remarkable that with a few scalar sensors satisfactory results can be obtained at off-design conditions. This opens the proposed method to many practical applications. A more accurate approach would be to obtain the POD modes and the estimator at a few Re numbers and then perform an interpolating procedure to derive a new estimator at an unseen operating point.

7 Conclusions

A data-driven algorithm was presented that allows the reconstruction of velocity and scalar fields from sparse velocity and/or scalar measurements. POD was applied to reduce the dimensionality of the problem and then a system identification algorithm was employed to extract the matrices of a linear, time-invariant, dynamical system that reproduces the behaviour of the time coefficients of the POD modes.

The algorithm was applied to the 2D flow around a wall-mounted square prism. The spectra of the velocity and scalar at selected probe points confirm that the velocity field is imprinted on the scalar dynamics; this indicates that extracting velocity from scalar measurements is a meaningful exercise. To provide a base case for comparison, the algorithm was first tested using velocity-only sensors placed at the peaks of the POD modes. Then it was applied using scalar-only sensors that were mounted at the same points. In both cases remarkably good flow reconstruction was achieved using a very small number of sensors; even one scalar sensor offered very satisfactory results for the case examined. Reconstruction with velocity sensors was slightly more accurate, because both velocity components were measured at each sensor location, thus more information is provided to the estimator.

The method was further applied to two off-design conditions at nearby Reynolds numbers using the trained model at the reference Reynolds number. The reconstruction results are satisfactory even with a single scalar sensor. This makes the proposed method suitable for real world applications. It is expected that more accurate results can be obtained by training different estimators at a few operating conditions and then applying an interpolation procedure to get the estimator at a new, unseen, operating point.

The proposed method is entirely data-driven and applicable to both experimental and computational data. Future work includes application to complex three-dimensional flows around surface-mounted cubes, and evaluation of optimal locations for sensor placement that will maximise the quality of reconstruction.

Author Contributions Conceptualization: GP; Formal analysis and investigation: SL; Writing - original draft preparation: SL; Writing - review and editing: GP, SL; Supervision: GP.

Funding GP is supported by EPSRC grants EP/X017273/1, EP/W001748/1 and EP/W00481X/1.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bai, Z., Kaiser, E., Proctor, J.L., Kutz, J.N., Brunton, S.L.: Dynamic mode decomposition for compressive system identification. *AIAA J.* **58**, 561–574 (2020)
- Baiges, J., Codina, R., Idelsohn, S.: Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier- Stokes equations. *Int. J. Numer. Mech. Fluids* **72**, 1219–1243 (2013)
- Balay, S., Abhyankar, S., Adams, M.F., Benson, S., Brown, J., Brune, P., Zhang, J.: PETSc/TAO users manual (Tech. Rep. No. ANL- 21/39 - Revision 3.18). Argonne National Laboratory (2022)
- Brunton, S., Noack, B.: Closed-loop turbulence control: progress and challenges. *Appl. Mechan. Rev.* **67**(5), 050801 (2015)
- Brunton, S.L., Proctor, J.L., Tu, J.H., Kutz, J.N.: Compressed sensing and dynamic mode decomposition. *J. Comput. Dyn.* **2**(2), 165–191 (2015)

- Cai, S., Wang, Z., Fuest, F., Jeon, Y.J., Gray, C., Karniadakis, G.: Flow over an espresso cup: inferring 3-D velocity and pressure fields from tomographic background oriented Schlieren via physics-informed neural networks. *J. Fluid Mech.* **915**, A10 (2021)
- Callahan, J., Maeda, K., Brunton, S.: Robust flow reconstruction from limited measurements via sparse representation. *Phys. Rev. Fluids* **4**, 103907 (2019)
- Carlberg, K., Bou-Mosleh, C., Farhat, C.: Efficient non-linear model reduction via a least-squares Petrov-galerkin projection and compressive tensor approximations. *Int. J. Numer. Meth. Eng.* **86**(2), 155–181 (2011)
- Castro, I.P., Robins, A.G.: The flow around a surface-mounted cube in uniform and turbulent streams. *J. Fluid Mech.* **79**, 307–335 (1977)
- Diaz-Daniel, C., Laizet, S., Vassilicos, J.C.: Direct numerical simulations of a wall-attached cube immersed in laminar and turbulent boundary layers. *Int. J. Heat Fluid Flow* **68**, 269–280 (2017)
- Falgout, R.D., Yang, U.M.: Hypre: a library of high performance preconditioners. In: Sliot, P.M.A., Hoekstra, A.G., Tan, C.J.K., Dongarra, J.J. (eds.) *Computational science - iccs 2002*, pp. 632–641. Springer, Berlin Heidelberg, Berlin, Heidelberg (2002)
- Fragos, V., Psychoudaki, S., Malamataris, N.: Two-dimensional numerical simulation of vortex shedding and flapping motion of turbulent flow around a rib. *Comput. Fluids* **69**, 108–121 (2012)
- Gomez, D.F., Lagor, F.D., Kirk, P.B., Lind, A.H., Jones, A.R., Paley, D.A.: Data-driven estimation of the unsteady flowfield near an actuated airfoil. *J. Guid. Control Dyn.* **42**(10), 2279–2287 (2019)
- Gong, J., Monty, J.P., Illingworth, S.J.: Model-based estimation of vortex shedding in unsteady cylinder wakes. *Phys. Rev. Fluids* **5**(023901), 5–32 (2020)
- Habibi, M., D’Souza, R.M., Dawson, S.T., Arzani, A.: Integrating multi-fidelity blood flow data with reduced-order data assimilation. *Comput. Biol. Med.* **135**, (2021)
- Hwang, J.Y., Yang, K.S.: Numerical study of vortical structures around a wall-mounted cubic obstacle in channel flow. *Phys. Fluids* **16**, 2382–2394 (2004)
- Íñigo, J.G., Sipp, D., Schmid, P.J.: A dynamic observer to capture and control perturbation energy in noise amplifiers. *J. Fluid Mech.* **758**, 728–753 (2014)
- Íñigo, J.G., Sipp, D., Schmid, P.J.: Recovery of the inherent dynamics of noise-driven amplifier flows. *J. Fluid Mech.* **797**, 130–145 (2016)
- Íñigo, J.G., Sodar, M.A., Papadakis, G.: Data-based, reduced-order, dynamic estimator for reconstruction of nonlinear flows exhibiting limitcycle oscillations. *Phys. Rev. Fluids* **30**, 114703 (2019)
- Jasak, H., Weller, H.G., Gosman, A.D.: High resolution NVD differencing scheme for arbitrarily unstructured meshes. *Int. J. Numer. Meth. Fluids* **31**(2), 431–449 (1999)
- Kailath, T., Hasibi, B., Sayed, A.H.: *Linear estimation*. Prentice-Hall (2000)
- Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L.: Physics-informed machine learning. *Nat. Rev. Phys.* **3**, 422–440 (2011)
- Lumley, J.L.: The structure of inhomogeneous turbulent flows. *Atmos. Turbul. Radio Wave Propag.* 166–178 (1967)
- Manohar, K., Brunton, B.W., Kutz, J.N., Brunton, S.L.: Data-driven sparse sensor placement for reconstruction: demonstrating the benefits of exploiting known patterns. *IEEE Control Syst. Mag.* **38**(3), 63–86 (2018)
- Martinuzzi, R., Tropea, C.: The flow around surface-mounted, prismatic obstacles placed in a fully developed channel flow. *J. Fluids Eng.* **115**, 85–92 (1993)
- Mikhaylov, K., Rigopoulos, S., Papadakis, G.: Reconstruction of largescale flow structures in a stirred tank from limited sensor data. *AIChE J.* e17348 (2021)
- Noack, B., Morzynski, M., Tadmor, G.: *Reduced-Order Modelling for Flow Control*. CISM series (vol 528)-Springer (2011)
- Qin, S.J.: An overview of subspace identification. *Comput. Chem. Eng.* **30**, 1502–1513 (2006)
- Raissi, M., Perdikaris, P., Karniadakis, G.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **378**, 686–707 (2019)
- Raissi, M., Yazdani, A., Karniadakis, G.: Hidden fluid mechanics: learning velocity and pressure fields from flow visualizations. *Science* **367**(6481), 1026–1030 (2020)
- Rossi, R., Iaccarino, G.: Numerical simulation of scalar dispersion downstream of a square obstacle using gradient-transport type models. *Atmos. Environ.* **43**(16), 2518–2531 (2008)
- Rossi, R., Philips, D.A., Iaccarino, G.: A numerical study of scalar dispersion downstream of a wall-mounted cube using direct simulations and algebraic flux models. *Int. J. Heat Fluid Flow* **31**, 805–819 (2010)
- Rowley, C.W., Dawson, S.T.M.: Model reduction for flow analysis and control. *Annu. Rev. Fluid Mech.* **49**, 387–417 (2017)

- Savarino, F., Papadakis, G.: Reconstruction of irregular flow dynamics around two square cylinders from sparse measurements using a datadriven algorithm. (2022) <https://arxiv.org/abs/2209.03290> (under review)
- Sipp, D., Schmid, P.: Linear closed-loop control of fluid instabilities and noise-induced perturbations: A review of approaches and tools I. *Appl. Mech. Rev.* **68**(2), 020801 (2016)
- Sirovich, L.: Turbulence and the dynamics of coherent structures. *Q. Appl. Maths* **45**, 561–571 (1987)
- Surana, A., Banaszuk, A.: Linear observer synthesis for nonlinear systems using koopman operator framework. *Int. Federation Automatic Control* **10**(250), (2016)
- Taira, K., et al.: Modal analysis of fluid flows: an overview. *AIAA J.* **55**(12), (2017)
- Taira, K., et al.: Modal analysis of fluid flows: applications and outlook. *AIAA J.* **58**(3), (2020)
- Tu, J.H., Griffin, J., Hart, A., Rowley, C.W., Cattafesta, L.N., Ukeiley, L.S.: Integration of non-time-resolved piv and time-resolved velocity point sensors for dynamic estimation of velocity fields. *Exp. Fluids* **54**, (2013)
- van Overschee, P., de Moor, B.: N4sid: subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica* **30**(1), 75–93 (1994)
- Vincont, J.Y., Simoens, S., Ayrault, M., Wallace, J.M.: Passive scalar dispersion in a turbulent boundary layer from a line source at the wall and downstream of an obstacle. *J. Fluid Mech.* **424**, 127–167 (2000)
- Willcox, K.: Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Comput. Fluids* **35**, 208–226 (2006)
- Xiao, D., Papadakis, G.: Nonlinear optimal control of transition due to a pair of vortical perturbations using a receding horizon approach. *J. Fluid Mech.* **861**, 524–555 (2019)
- Xiao, D., Fang, F., Du, J., Pain, C., Navon, L., Buchan, A., Hu, G.: Non-linear Petrov-Galerkin methods for reduced order modelling of the Navier-Stokes equations using a mixed finite element pair. *Comput. Methods Appl. Mech. Eng.* **255**, 147–157 (2013)
- Xiao, D., Fang, F., Buchan, A., Pain, C., Navon, I., Muggeridge, A.: Non-intrusive reduced order modelling of the Navier-Stokes equations. *Comput. Methods Appl. Mech. Eng.* **293**, 522–541 (2015)
- Yao, H., Mollicone, J.-P., Papadakis, G.: Analysis of interscale energy transfer in a boundary layer undergoing bypass transition. *J. Fluid Mech.* **941**, A14 (2022)
- Yildirim, B., Chrysostomidis, C., Karniadakis, G.E.: Efficient sensor placement for ocean measurements using low-dimensional concepts. *Ocean Model.* **27**, 160–173 (2009)
- Yin, M., Zheng, X., Humphrey, J., Karniadakis, G.: Non-invasive inference of thrombus material properties with physics-informed neural networks. *Comput. Methods Appl. Mech. Eng.* **375**, 113603 (2021)