

Article

Interplay of Sensor Quantity, Placement and System Dimension in POD-Based Sparse Reconstruction of Fluid Flows

Balaji Jayaraman * , S M Abdullah Al Mamun and Chen Lu

School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078, USA; smamun@ostateemail.okstate.edu (S.M.A.A.M.); luchen@ostateemail.okstate.edu (C.L.)

* Correspondence: balaji.jayaraman@okstate.edu

Received: 5 February 2019; Accepted: 10 June 2019; Published: 13 June 2019

Abstract: Sparse linear estimation of fluid flows using data-driven proper orthogonal decomposition (POD) basis is systematically explored in this work. Fluid flows are manifestations of nonlinear multiscale partial differential equations (PDE) dynamical systems with inherent scale separation that impact the system dimensionality. Given that sparse reconstruction is inherently an ill-posed problem, the most successful approaches require the knowledge of the underlying low-dimensional space spanning the manifold in which the system resides. In this paper, we adopt an approach that learns basis from singular value decomposition (SVD) of training data to recover sparse information. This results in a set of four design parameters for sparse recovery, namely, the choice of basis, system dimension required for sufficiently accurate reconstruction, sensor budget and their placement. The choice of design parameters implicitly determines the choice of algorithm as either l_2 minimization reconstruction or sparsity promoting l_1 minimization reconstruction. In this work, we systematically explore the implications of these design parameters on reconstruction accuracy so that practical recommendations can be identified. We observe that greedy-smart sensor placement, particularly interpolation points from the discrete empirical interpolation method (DEIM), provide the best balance of computational complexity and accurate reconstruction.

Keywords: sparse reconstruction; sensors; cylinder flow; singular value decomposition (SVD); proper orthogonal decomposition (POD); compressive sensing

1. Introduction

Multiscale fluid flow phenomena in engineering and geophysical settings are invariably data-sparse, i.e., there are more scales to resolve than there are sensors. A major goal is to recover more information about the dynamical system through reconstruction of the higher dimensional state. To expand on this view, in many practical fluid flow applications, accurate simulations may not be feasible for a multitude of reasons, including lack of accurate models, unknown governing equations or extremely complex boundary conditions. In such situations, measurement data represent the absolute truth and are often acquired from very few probes, and therefore offering limited scope for analysis. A common recourse is to combine such sparse measurements with underlying knowledge of the flow system, either in the form of idealized simulations or phenomenology or knowledge of a sparse basis to recover detailed information. The former approach is termed as data assimilation while we refer to the latter as Sparse Reconstruction (SR). On the other hand, simulations typically represent a data surplus setting that offer the best avenue for analysis of realistic flows as one can identify and visualize coherent structures, perform well converged statistical analysis including quantification of spatiotemporal coherence and scale content due to the high density of data probes in the form of computational grid points. With growth in computing power and generation of big data,

there is an ever growing demand for quick analytics and machine learning tools [1] to both sparsify, i.e., dimensionality reduction [2–5], and recover the full state without loss of information. Thus, tools for *encoding* information into a low-dimensional feature space (convolution) complement sparse recovery tools that help *decode* compressed information (deconvolution). This in essence provides a significant context for leveraging machine learning in fluid flow analysis [6,7].

Recent advances in compressive sensing (CS) [8–11] have opened the possibility of direct sparse sampling [6] of data in real-time without having to collect high resolution information and then downsample. Of course, for direct sampling, one requires a collection of generic basis in which the data has a high probability of being sparse in whereas when collecting high resolution information and then down sampling, one can learn optimal basis from data. Thus, reconstruction from sparse data has been popular in their various manifestations such as Gappy Proper Orthogonal Decomposition (GPOD) [12,13], Fourier-based Compressive Sensing (CS) [8–11] and Gaussian kernel-based Kriging [14–16]. A parallel application of such ideas is in the acceleration of nonlinear model order reduction using sparse sampling for hyper-reduction [17–20]. Outside of the basis-driven reconstruction approaches, there also exist alternative classes of methods based on nonlinear estimation [21] and pattern recognition ideas such as k-nearest neighbors or kNN [22]. In the former, mapping from the sparse to fine data is approximated through a nonlinear map such as a neural network or its variants. In this way, the sensor placement and the basis learning procedures are combined which can be leveraged for learning observable dictionaries [23]. In the latter, a library based look-up of snapshots is employed to map an appropriate lifted dynamic feature to the ensemble of flow fields that will be interpolated locally in the feature space.

In this study, we focus primarily on basis enabled sparse linear estimation of the high dimensional state by converting the inherently ill-posed, under-determined inverse problem into a well-posed one in the space of basis coefficients. The contribution from this work is the development of a systematic framework for characterizing the SR performance in terms of accuracy of data recovery that can inform practical applications. Secondly, we explore how these SR methods interact and potentially gain from greedy and smart sensor placement. To this end, we focus on both low-dimensional laminar wake flow as well as high-dimensional geophysical turbulence measurements, i.e., sea surface temperature data from global ocean models to mimic practical use conditions.

Sparse Reconstruction with Data-driven Basis: Data recovery techniques based on Karhunen-Loeve (K-L) procedure with least squares (l_2) error minimization, also known as Gappy POD or GPOD [12,13], was originally developed in the 1990s to recover marred faces [17] in images. The idea is to utilize the POD basis computed offline from data to *encode* the reconstruction problem into a low-dimensional feature space. This way, sparse data can be used to recover the sparse unknowns in the space of POD coefficients by minimizing the l_2 errors. If the data-driven POD basis are not known a priori, an iterative formulation [12,17] to successively approximate the POD basis and the coefficients was proposed with limited success [12,14,24], i.e., it is prone to numerical instabilities and inefficiency. Advancements in the form of a progressive iterative reconstruction framework [14] are effective, but impractical due to computational cost. In fact, all the aforementioned issues are related to the POD-basis being data-driven and, therefore, can approximate the data effectively but not generalizable. For generalization, they are required to be known a priori—a stringent requirement in practice as training data are rarely available and even when they are, they may not effectively span the prediction regime. Such limitations make data-driven basis hard to use for practical applications. Nevertheless, they find tremendous value in data-driven modeling such as those based on learning the Koopman operator [25,26] and nonlinear model order reduction [4] of statistically stationary systems where training data are available.

Sparse Reconstruction with Generic Basis: To bypass the above limitations, one uses generic basis such as wavelets [27] or Fourier functions with the underlying assumption that most practical systems are relatively sparse in the subspace spanned by the basis. This is particularly true for image processing applications but not necessarily for fluid flows whose dynamics obey PDEs and may

support sharp gradients. While avoiding the cost of computing the basis offline, such approaches run into sparsity issues, as the basis vectors do not optimally encode the underlying dynamical system with sufficiency approximation quality. Therefore, with more basis vectors needed to represent the dynamics, more sensors are needed for accurate reconstruction. Once again, the reconstruction problem is ill-posed as the subspace is not low-dimensional, resulting in the flow being under sampled. To bypass this, one needs to look for a sparse solution. The magic of Compressive Sensing (CS) [8–11] is in its ability to overcome this constraint by seeking a solution that can be less sparse than the dimension of the chosen feature space using l_1 -norm regularized reconstruction. Such methods have been successfully applied in image processing using Fourier or wavelet basis and also to fluid flows [6,7,28–31]. Compressive sensing essentially looks for a regularized sparse solution using l_1 norm minimization of the sparse coefficients by solving a computationally manageable convex optimization problem. Such sparsity promoting l_1 methods have been combined with data-driven POD basis to reconstruct sparse PIV data [6] and pressure measurements around the surface of the cylinder [7]. To bypass the limitations from generalizability of POD basis for SR, one often builds a library of modes (one-time cost [7]) from precursor training simulation or measurement data over a range of flow parameters (e.g., Reynolds numbers) to identify the suitable K -sparse solution.

To carry out a systematic analysis of interplay between the various SR design parameters, namely, desired reconstruction dimension, sensor budget and their placement, we limit ourselves to l_2 minimization approaches (except for comparative analysis where we use l_1) and four different sensor placement methods including random sampling (as used in [32]), QR with column pivoting [33,34], Discrete Empirical Interpolation Method (DEIM) [18] and explicit condition number minimization [13]. The suitability of these choices are numerically explored in a discretized parameter space of sensor budget and system dimension. The rest of manuscript is organized as follows. In Section 2, we review the theory of sparse reconstruction (Section 2.1), computation of data-driven POD basis (Section 2.2), role of measurement locations (Section 2.3), algorithms for sensor placement (Section 2.4) and sparse recovery (Section 2.5). In Section 3, we discuss how the training data are generated and summarize the different sensor placement outcomes. In Section 4, we discuss the results from our SR analysis of fluid flow data using POD basis and summarize the major conclusions in Section 5.

2. Formulating the Sparse Reconstruction Problem

Given a high-resolution data representing the state of the flow system at any given instant denoted by $x \in \mathbb{R}^N$, its corresponding sparse representation is given by $\tilde{x} \in \mathbb{R}^P$ with $P \ll N$. Then, the sparse reconstruction problem is to recover x , when given \tilde{x} along with information of the sensor locations in the form the measurement matrix $C \in \mathbb{R}^{P \times N}$, as shown in Equation (1). The measurement matrix C carries information about how the sparse data \tilde{x} are collected from x . Variables P and N are the number of sparse measurements and the dimension of the high resolution field, respectively.

$$\tilde{x} = Cx. \quad (1)$$

Naturally, when one loses the information about the system, the recovery of said information is not absolute as the reconstruction problem is ill-posed, i.e., there are more unknowns than constraints in Equation (1). The most straightforward approach to recover x is by computing the inverse of C using a least-squares error minimization procedure given by

$$C^+ \tilde{x} = x, \quad (2)$$

which is often inaccurate due to ill-posedness.

2.1. Sparse Reconstruction Theory

The theory underlying sparse reconstruction has strong foundations in the field of inverse problems [35] with applications in diverse fields of study such as a geophysics [36,37] and image

processing [38,39]. In this section, we formulate the reconstruction problem as presented in the CS literature [8,10,40–42] that deals with “compressible” signals, i.e., they are sparse in some basis Φ as shown below:

$$x = \sum_{i=1}^{N_b} \phi_i a_i \quad \text{or} \quad x = \Phi a, \tag{3}$$

where $\Phi \in \mathbb{R}^{N \times N_b}$ and $a \in \mathbb{R}^{N_b}$ with K non-zero elements (or K -sparse). In the formulation above, $\Phi \in \mathbb{R}^{N \times N_b}$ is used instead of $\Phi \in \mathbb{R}^{N \times K}$ as the K most relevant basis vectors for a given data are not usually known a priori. Consequently, a more exhaustive basis set of dimension $N_b \approx P > K$ is typically employed. To recover N -dimensional data, one needs at most N linearly independent basis vectors, i.e., $N_b \leq N$. In practice, the candidate basis dimension need not be N and can be represented by $N_b \ll N$ as only $K (\leq N_b)$ of them are needed to approximate the signal to a desired quality. This is typically the case when Φ is composed of optimal data-driven basis vectors such as POD modes. The reconstruction problem is then recast as identification of these K sparse coefficients. In this article, we focus on such vectors x that have a sparse representation in a chosen subspace spanned by Φ .

In many practical situations, Φ , K , N_b and N are user inputs. Standard *transform coding* [27] practice in image compression involves collecting a high resolution sample, projecting it onto a Fourier or wavelet basis where the data are sparse, retain the K most relevant coefficients while discarding the rest. The *sample and then compress* mechanism still requires acquisition of high resolution samples and processing them before dimensionality reduction. This is challenging due to demands on processing power, storage, and time. Compressive sensing [8,10,40–42] focuses on direct sparse sensing based inference of the K -sparse coefficients by essentially combining the steps in Equations (1) and (3) to yield

$$\tilde{x} = C\Phi a = \Theta a, \tag{4}$$

where $\Theta \in \mathbb{R}^{P \times N_b}$ is the map between the basis coefficients a that represent the data in a feature space and the sparse measurements, \tilde{x} in physical space. The challenge in solving for x using the under determined Equation (1) is that C is ill-conditioned and x in itself is not sparse. However, when x is sparse in Φ , the reconstruction using Θ in Equation (4) becomes practically feasible (for $P \gtrsim K$) by solving for a . Thus, one effectively seeks a K -sparse a with P constraints (given by \tilde{x}) using established methods from linear algebra and constrained optimization.

2.1.1. Case 1: For $K = N_b$

For the over determined system with $P > K = N_b$, a is estimated using a regularized least squares solution based on the normal equation as $a = (\Theta)^{L+} \tilde{x} = (\Theta^T \Theta + \alpha \mathbf{I})^{-1} \Theta^T \tilde{x}$ for a chosen α . This is obtained by minimizing the appropriate cost function given by $J_{cost} = \|\tilde{x} - \Theta a\|_2^2 + \alpha \|a\|_2^2$. This regularized least-squares solution procedure for the overdetermined case is nearly identical to the original GPOD algorithm developed by Everson and Sirovich [17] if Φ is chosen as the POD basis. However, \tilde{x} in GPOD contains zeros as placeholders for all the missing elements, whereas the above formulation retains only the measured data points.

When $P \leq K = N_b$, and the system is under-determined with non-unique solutions, one looks for a minimum norm reconstruction of a . This is achieved by minimizing the corresponding s -norm of a , i.e., $\|a\|_s$ (s chosen appropriately) and x is then recovered from Equation (3). A minimum l_2 norm reconstruction of x that penalizes the larger elements of a is realized by choosing s as 2 and is a solution to the optimization problem given by

$$\begin{aligned} l_2 \text{ norm minimization reconstruction : } a &= \operatorname{argmin} \|a'\|_2 \text{ such that } \Theta a' = \tilde{x}; \\ l_2 \text{ cost function to be minimized : } \min \{ &\alpha (\tilde{x} - \Theta a) + \|a\|_2^2 \}. \end{aligned} \tag{5}$$

One can solve for α and a in Equation (5) using method of Lagrange multipliers to yield a solution that is a right pseudo-inverse of Θ as

$$a = (\Theta)^{R+} \tilde{x} = \Theta^T (\Theta \Theta^T)^{-1} \tilde{x}, \tag{6}$$

provided Θ has minimum rank P .

2.1.2. Case 2: For $K < N_b$

When $K \ll N_b$, one typically looks for a sparse solution of a . The l_2 approaches discussed above do not generate sparse solutions. A natural way to enhance sparsity of a is to minimize $\|a'\|_0$, i.e., minimize the number of non-zero elements such that $\Theta a' = \tilde{x}$ is satisfied. It has been shown [43] that, with $P = K + 1$ ($P > K$ in general) *independent* measurements, one can recover the sparse coefficients with high probability using minimum l_0 norm reconstruction. This is heuristically interpreted as each measurement needing to excite a different basis vector ϕ_i so that its coefficient a_i is uniquely identified. If two or more measurements excite the same basis vector ϕ_j , then additional measurements may be needed to produce acceptable reconstruction. On the other hand, for $P \leq K$ independent measurements, the probability of recovering the sparse solution is highly diminished. Nevertheless, l_0 -reconstruction is a computationally complex, np -complete and poorly conditioned problem with no stability guarantees.

The popularity of compressed sensing arises from guarantees of near-exact reconstruction of the uncompressed information by solving for the K sparsest coefficients using l_1 norm minimization methods. The l_1 reconstruction is a relatively simpler convex optimization problem (as compared to l_0) and solvable using linear programming techniques for basis pursuit [8,40,44] and shrinkage methods [45]. Theoretically, one can perform the simplistic brute force search to locate the largest K coefficients of a that satisfy the constrained optimization problem given by

$$l_1 \text{ norm minimization reconstruction : } a = \operatorname{argmin} \|a'\|_1 \text{ such that } \Theta a' = \tilde{x}; \tag{7}$$

$$l_1 \text{ cost function to be minimized : } \min\{\alpha \|\tilde{x} - \Theta a\|_2^2 + \|a\|_1\}.$$

For these approaches, the computational effort increases nearly exponentially with dimension. To overcome this burden, a host of greedy algorithms [9,11,46] have been developed to solve the l_1 problem in Equation (7) with complexity $\mathcal{O}(N^3)$ for $N_b \approx N$. However, this approach requires $P > \mathcal{O}(K \log(N_b/K))$ measurements [8,40,47] to reconstruct the K -sparse vectors with high probability. Both l_2 - and l_1 -based formulations are schematically illustrated in Figure 1.

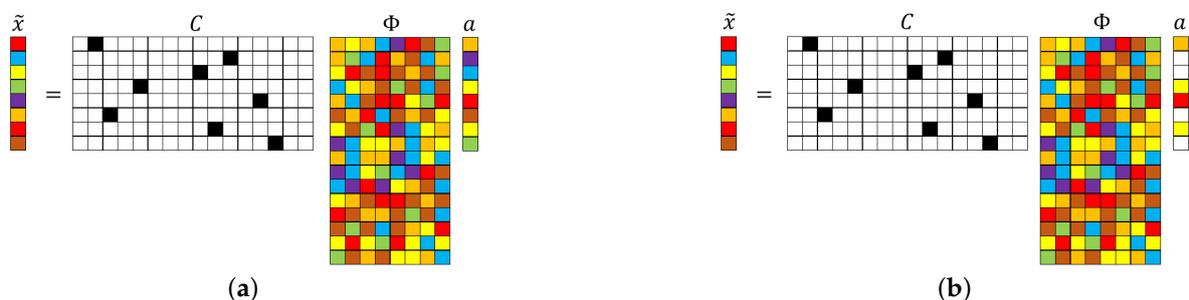


Figure 1. Schematic illustration of l_2 (a) and l_1 (b) minimization reconstruction for sparse recovery using a single-pixel measurement matrix. The numerical values in C are represented by colors: black (1) and white (0). The other colors represent numbers that are neither 0 nor 1. In the above schematics, $\tilde{x} \in \mathbb{R}^P$, $C \in \mathbb{R}^{P \times N}$, $\Phi \in \mathbb{R}^{N \times N_b}$ and $a \in \mathbb{R}^{N_b}$, where $N_b \leq N$. The number of colored cells in a represents the system sparsity K . We note that, for $K = N_b$, the method of choice is l_2 and, for K smaller than N_b , the method of choice is l_1 .

Solving the l_1 minimization problem in Equation (7) is complicated relative to the l_2 solution described in Equation (5). This is because, unlike Equation (5), the cost function in Equation (7) is not differentiable at $a_i = 0$, which necessitates an iterative solution. Further, the minimization of the l_1 cost function is also an unconstrained optimization problem that is commonly converted into a constrained optimization problem given by

$$l_1 \text{ norm constrained minimization : } \min \|\tilde{x} - \Theta a\|_2^2 \text{ such that } \|a\|_1 < t, \tag{8}$$

where t is a user defined sparsity knob to “shrink” the coefficients. The above constrained optimization problem in Equation (8) is quadratic in a and, therefore, a quadratic programming problem with the feasible region bounded by polyhedron (in the space of a). Two classes of l_1 solution methodologies exist: (i) least absolute selection and shrinkage operator (LASSO) [45]; and (ii) basis pursuit denoising [44]. LASSO and its variant essentially convert the constrained formulation into a set of linear constraints. Recently popular approaches include greedy methods such as optimal matching pursuit (OMP) [11,29] and interior point methods [48]. An intuitive iterative sequential least-squares thresholding framework was used by Brunton et al. [49], which achieves “shrinkage” by repeatedly zeroing out the coefficients smaller than a given choice of hyperparameter.

In summary, the reconstruction framework is characterized by three parameters, N_b, K and P where N_b is the candidate basis dimension, K is the desired reconstruction dimension and P is the sensor budget. The interplay of N_b, K , and P determines the choice of algorithm employed, i.e., whether the reconstruction is based on least squares minimization, l_2 norm minimization or sparsity enabling l_1 approaches, as summarized in Table 1. These different possibilities are illustrated as follows. In practical situations such as recovery of coherent structures from sparse field data, the approximation quality of the basis is not known beforehand thereby requiring a library of candidate basis from a variety of flow regimes such that $N_b > K$ from which the K best coefficients are estimated using sparse regression as in Case 3. However, when the basis approximation quality of the data is known a priori, then we need to retain only the K most significant modes for reconstruction, i.e., $K = N_b$ as in Cases 1 and 2. Such situations often come up in laboratory flows or for improving the speed of computational models, where prior simulation or PIV data can be used to build the appropriate basis vectors. If there exists sufficient sparse measurements ($P > K$) as in coarse-grained computational models, then we realize Case 1. However, in practical laboratory experiments with limited probes ($P < K$), we deal with Case 2.

Table 1. The choice of sparse reconstruction algorithm based on problem design using parameters P (sensor sparsity), K (targeted reconstruction sparsity) and N_b (candidate basis dimension).

Case	$K - N_b$ Relationship	$P - K$ Relationship	Algorithm	Reconstructed Dimension
1	$K = N_b$	$P \geq K$	least squares (l_2)	K
2	$K = N_b$	$P < K$	min. norm recons. (l_1) or (l_2)	P
3	$K < N_b$	$P > K$	min. norm recons. (l_1)	K

All of the above sparse recovery estimations are conditional upon the measurements (rows of C) being incoherent with respect to the sparse basis Φ . This is usually accomplished by using random sensor placement, especially when Φ is made up of Fourier functions or wavelets. If the basis functions Φ are orthonormal, such as wavelet or POD basis (with inherent hierarchy), one can discard the majority of the small coefficients in a (setting them as zeros) and still achieve reasonably accurate reconstruction [10]. However, incoherency is a necessary but not sufficient condition for exact reconstruction, which requires sensors optimally placed to minimize reconstruction errors.

2.2. Data-Driven Basis Computation Using POD

For the SR framework, the common basis types to map to a low-dimensional space are POD modes, Fourier functions, and wavelets [8,10]. While an exhaustive study on the effect of the different choices on reconstruction performance is useful, in this paper, we focus on POD-based SR. A comparison between discrete cosine transform and POD bases is carried out in [6].

Proper orthogonal decomposition (POD), also known as Principal Components Analysis (PCA) or Singular Value Decomposition (SVD), is a dimensionality reduction technique that computes low-dimensional basis vectors (POD modes) and coefficients from snapshots of experimental or numerical data [2,4] through eigendecomposition of the spatial (or temporal) correlation tensor of the data. It was adopted in the turbulence community by Lumley [50] to extract coherent structures in turbulent flows. The resulting singular vectors or POD modes represent an orthogonal basis that maximizes the variance capture from flow data. For this reason, such eigenfunctions are considered *energy optimal* and other optimality constraints can also be incorporated. Taking advantage of the orthogonality, one can project these POD basis onto snapshots of data in a Galerkin sense to deduce coefficients that represent evolution over time in the POD feature space. The optimality of the POD basis also allows one to effectively reconstruct the full field information with knowledge of very few coefficients, a feature that is attractive for solving sparse reconstruction problems such as in Equation (4). However, this is contingent on the spectrum of the correlation tensor of the data having sufficiently rapid decay of the eigenvalues, i.e., it supports a low-dimensional representation. This is typically not true in the case of turbulent flows where the decay of energy across singular values is gradual. Further, in such dynamical systems, the small scales with low-energy can still be dynamically important and will need to be recovered, thus requiring significant sensor budget.

The computational cost of eigendecomposition of the spatial correlation tensor depends on the full state dimension N , which is usually large. Alternative approaches based on the method of snapshots [51] is adopted in this work. Here, the eigen problem is reformulated using a temporal correlation tensor with reduced dimension (assuming the number of snapshots in time is smaller than the spatial dimension) and summarized below. Consider $X \in \mathbb{R}^{N \times M}$ (different from $x \in \mathbb{R}^N$) as the full state with only the fluctuating part (no mean) where N and M are the state and snapshot dimensions, respectively. The procedure involves eigendecomposition of the symmetric correlation tensor, $\bar{C}_M = X^T X$ ($\bar{C}_M \in \mathbb{R}^{M \times M}$) as

$$\bar{C}_M V = V \Lambda, \tag{9}$$

with $V = [v_1, v_2, \dots, v_M]$ being the matrix of eigenvectors and the diagonal elements of Λ denoting the eigenvalues $[\lambda_1, \lambda_2, \dots, \lambda_M]$. Typically, both the eigenvalues and corresponding eigenvectors are sorted in descending order such as $\lambda_1 > \lambda_2 > \dots > \lambda_M$. The POD modes Φ and coefficients a are then computed as

$$\Phi = X V \sqrt{\Lambda^{-1}}. \tag{10}$$

One can represent the field X as a linear combination (Equation (3)) of the POD modes Φ with coefficients $a \in \mathbb{R}^{M \times M}$ given by

$$a = \Phi^T X. \tag{11}$$

It is worth mentioning that subtracting the temporal mean from the input data is not critical to the success of this procedure as retaining it yields an extra mean mode in the decomposition. Using the snapshot procedure for the POD/SVD computation fixes the maximum number of POD basis vectors to at most M which is typically much smaller than the dimension of full state vector, N . Further dimension reduction is possible through truncation of the low energy modes such that the resulting dimension $K < M$.

2.3. Measurement Locations, Data Basis and Incoherence

The reconstruction performance depends on the measurement matrix, \mathbf{C} being *necessarily* incoherent with respect to the low-dimensional basis Φ [10] and this is usually accomplished through random measurements. In practice, one can adopt two types of random sampling of the data, namely single-pixel measurement [7,52] or random projections [8,10,41]. Typically, single-pixel measurement refers to measuring information at chosen spatial locations using point probes. The resulting matrix \mathbf{C} is structured as $\mathbf{C} \leftarrow [e_{q_1}, e_{q_2}, \dots, e_{q_p}]^T$, where e_{q_p} is column vector with zeros except at the index p where it assumes a value of 1. Another popular choice adopted in the compressive sensing or image processing community is to populate \mathbf{C} using random numbers from a chosen distribution (Gaussian or Bernoulli). In theory, a random \mathbf{C} is highly likely to be incoherent with respect to standard Fourier basis [10]. However, for most fluid flow and other practical sensing applications, the sparse data are usually sourced from point measurements and the basis are data-driven. Irrespective of the choice, one should have sufficient measurements distributed in space to excite the different modes relevant to the data being reconstructed. Mathematically, this is related to the accuracy of the generalized inverse (Moore–Penrose pseudo-inverse) of $\mathbf{C}\Phi$ and is quantified in the form of a *coherency* number, μ as

$$\mu(\mathbf{C}, \Phi) = \sqrt{N} \cdot \max_{i \leq P, j \leq K} |\langle c_i, \phi_j \rangle|, \quad (12)$$

where c_i is a row vector in \mathbf{C} (i.e., $c_i = e_{q_i}$) and ϕ_j is a column vector of Φ . μ typically ranges from 1 (incoherent) to \sqrt{N} (coherent). The smaller is the μ , the fewer measurements one needs to reconstruct the data. This is because the coherency parameter enters as the pre-factor in the lower-bound for the sensor quantity in l_1 -based CS for accurate recovery [53]. There exist other optimal sensor placement algorithms such as K-means clustering, the data-driven sparse online Gaussian processes [54], physics-based approaches based on extrema in modal shapes [55] and mathematical approaches [13] that minimize condition number of Θ and maximize determinant of the Fisher information matrix [56]. A thorough study on the role of sensor placement on reconstruction quality is much needed and is an active topic of research. In this paper, we assess three different greedy approaches for nearly optimal sensor placement in sparse recovery applications, namely the Discrete Empirical Interpolation Method (DEIM) [18,57], iterative condition number minimization [13] and the reduced matrix QR-factorization with column pivoting [34] and compare these approaches with outcomes from random sensor placement. To assess whether the resulting measurement matrices are incoherent with respect to the POD basis, we also estimate the coherency numbers in the following discussion.

2.4. Algorithms for Sensor Placement

Identifying optimal sensor placement for a given data, especially for a flow field that evolves over time is highly challenging and is an ongoing topic of active research. The goal of optimal point sensor placement for reconstruction is to identify and activate only a few rows of the basis matrix Φ that effectively conditions Θ (for $P = K = N_b$) or its variants, $\mathbf{M} = \Theta^T \Theta$ or $\Theta \Theta^T$ (depending on if $P > K = N_b$ or $P < K = N_b$, respectively). This is schematically illustrated in Figure 2.

To design smart sensor placement, one needs an optimization criterion, which in this case is to minimize reconstruction error when using a small number of sensors that, of course, depends on the choice of basis Φ . Since reconstruction from sparse data in general requires inversion of Θ or \mathbf{M} , most smart sensing strategies are designed to improve the condition number of Θ , \mathbf{M} for inversion purposes by optimizing their spectral content in the form of its determinant, trace, spectral radius or condition number. A direct method of optimizing such metrics require searching over the different possible sensor selections resulting in combinatorial complexity. Thankfully, there exist a variety of greedy algorithms [13,18,58] that have been shown to be successful for fluid flow data.

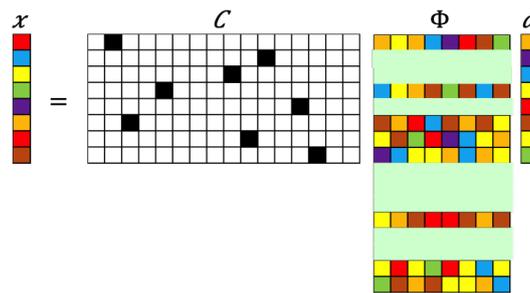


Figure 2. Schematic illustration of sparse sensor placement. The pastel colored rectangles represent rows activated by the sensors denoted in the measurement matrix through dark squares.

2.4.1. Random Sensor Placement

The most simple and efficient sensor placement approach is to sample randomly. This is commonly accomplished using a random permutation of the possible sensor locations. In this paper, we choose the first P values from this random permutation. It may be equally effective to adopt ideas such as K-means clustering as in [5]. To better assess the effectiveness of such random sensor placement methods, we generate multiple realizations to minimize bias. The outcomes are then quantified in terms of the mean as well as outliers. This particular approach is designed to serve as an inexpensive benchmark to compare against more expensive greedy sampling methods.

2.4.2. Minimization of Matrix Condition Number (MCN)

As shown in Section 2.1.1, the success of the reconstruction effort for $K = N_b$ is tied to the accuracy of the inverse computation of $\mathbf{M} = \Theta^T \Theta$ or $\Theta \Theta^T$. Therefore, if $\mathbf{M} = \Theta^T \Theta$ or $\Theta \Theta^T$ has full column or row rank, respectively, along with a reasonable condition number, then the inverse can be estimated accurately. This approach focuses on sensor placement (through the construction of \mathbf{C}) that minimizes the condition number of \mathbf{M} or $\kappa(\mathbf{M})$. The condition number is directly related to the orthogonality of Θ and the presence of significant diagonal entries in \mathbf{M} . Therefore, this algorithm can be viewed as placing sensors at locations that preserve the orthogonality of downsampled POD modes. Mathematically, the condition number represents the ratio of maximum to minimum singular values of Θ or \mathbf{M} . Therefore, for large $\kappa(\mathbf{M})$, the errors tend to be amplified with respect to the signal. As shown by Willcox [13] and Yildirim et al. [58], such a method compares favorably to more physics-based approaches [55,59] such as placing sensors at the extrema of dominant POD modes. The key steps of this MCN algorithm are:

- (i) Starting with the first sensor, consider each possible choice of sensor location to evaluate \mathbf{M} and identify the location with least $\kappa(\mathbf{M})$ as the chosen sensor placement.
- (ii) With the previous sensor location(s) set, loop over all possible remaining locations to identify the rest of the budgeted sensors as above.

A slightly more efficient version of this algorithm was presented by Willcox [13] where the first sensor location is chosen as the one that maximizes the sum of the difference between the diagonal and off-diagonal entries of \mathbf{M} . The rest of the algorithm is similar as above.

2.4.3. QR Factorization with Column Pivoting

The reduced matrix QR factorization [33] decomposes any given real matrix $A \in \mathbb{R}^{S \times T}$ with full column rank into a unitary matrix $Q \in \mathbb{R}^{S \times T}$ and an upper triangular matrix $R \in \mathbb{R}^{T \times T}$. Therefore, it follows that $|\det(A)| = |\det(Q) \cdot \det(R)| = |\det(R)| = \left| \prod_i r_{ii} \right| = \left| \prod_i \lambda_i \right|$ where r_{ii} are the diagonal entries of R and λ_i , the eigenvalues. It is easy to show that minimizing the condition number of A is related to optimizing the spectral characteristics of the matrix such as the determinant or

spectral radius, i.e., maximize $\left| \prod_i r_{ii} \right|$. In general, the R from a reduced matrix QR factorization has diagonal values, r_{ii} in no particular sequence. However, when combined with column pivoting, we have $AD = QR$, where $D \in \mathbb{R}^{T \times T}$ is a square column permutation matrix containing ones and zeros. The resulting QR decomposition outcome can be controlled through the pivoting procedure such that the diagonal values of R , r_{ii} form a decreasing sequence. Therefore, pivoting provides a smart approach for “submatrix volume maximization” and in turn maximize the absolute value of the determinant [34] by reordering the columns of A . This approach can easily be extended to sensor placement by leveraging the connections between the permutation matrix D and the point sensor measurement matrix C in Figure 2 and Equation (4).

For the case with $P = K$, the reconstruction problem in Equation (4) requires inversion of the square matrix $\Theta = C\Phi_k$. For improved reconstruction, the determinant of Θ needs to be maximized through sensor placement, which in turn is expected to reduce (and maybe minimize) the condition number. One can see that for a square matrix the following relationship

$$\det(\Theta) = \det(\Theta^T) = \det(\Phi_k^T C^T). \tag{13}$$

is true. Therefore, reduced matrix QR factorization of $\Phi^T \in \mathbb{R}^{K \times N}$ with column pivoting will yield

$$\Phi_k^T D = QR \tag{14}$$

where $D \in \mathbb{R}^{N \times N}$ is a square permutation matrix. The right-hand side of Equation (13) will be maximized for a given sensor quantity P if C is chosen as the first P rows of D^T . Note that the index locations of ones in each row of C are denoted by $[q_1, q_2, q_3, \dots, q_P]$.

For the oversampled case with $P > K$, Θ is a tall and slender matrix with a left (Moore–Penrose) pseudo-inverse requiring computation of M^{-1} (where $M = \Theta^T \Theta \in \mathbb{R}^{K \times K}$). Therefore, the sensor placement that increases the probability of accurate reconstruction should maximize $\det(\Theta^T \Theta)$ so that condition number of M is bounded. Specifically, we have the following relationships:

$$\det(M) = \det(\Theta^T \Theta) = \prod_i \sigma_i(\Theta^T \Theta) = \prod_{i=1}^K \sigma_i(\Theta \Theta^T) = \prod_{i=1}^K \sigma_i(C \Phi_K \Phi_K^T C^T). \tag{15}$$

Leveraging the above relationships, we see that maximizing the determinant of $M = \Theta^T \Theta$ can be realized by maximizing $\det(\Phi_K \Phi_K^T C^T)$ through a reduced matrix QR factorization,

$$(\Phi_K \Phi_K^T) D = QR, \tag{16}$$

and choosing C as the first P rows of $D^T \in \mathbb{R}^{N \times N}$. The index locations of ones in each row of C are denoted by $[q_1, q_2, q_3, \dots, q_P]$. The algorithm of greedy sensor selection for oversampled case using a given tailored basis Φ_K and number of sensors P is summarized in Algorithm 1.

Algorithm 1: Greedy sensor selection using QR factorization with column pivoting.

input :Data-driven basis, Φ_K
 Number of sensors, P
output: Measurement Matrix C

- 1 **if** ($P = K$) **then**
- 2 | $[q_1, q_2, \dots, q_P] \leftarrow$ Reduced Matrix QR Column Pivoting of Φ_k^T ;
- 3 **else if** ($P > K$) **then**
- 4 | $[q_1, q_2, \dots, q_P] \leftarrow$ Reduced Matrix QR Column Pivoting of $\Phi_k \Phi_k^T$;
- 5 $C \leftarrow [e_{q_1}, e_{q_2}, \dots, e_{q_P}]^T$ where $e_{q_i} = [0, \dots, 0, \underbrace{1}_{q_i}, 0, \dots, 0]^T$

2.4.4. Discrete Empirical Interpolation Method (DEIM)

All the above smart sensor placement methods focus on minimizing the condition number directly or indirectly through the determinant of the matrix whose inverse is sought. The discrete empirical interpolation method (DEIM), a discrete variant of the Empirical Interpolation Method (EIM) originally presented by Barrault et al. [60] and subsequently extended to nonlinear model order reduction applications by Chaturantabut and Sorensen [18,57] recursively learns the interpolation points (with indices q_j) at locations carrying maximum linear dependence error using previously estimated interpolation points. In this way, the DEIM sensors can be interpreted as minimizing linear dependence of the downsampled basis vectors.

The primary idea behind DEIM is to estimate a high-dimensional state using information at sparsely sampled interpolation points. Such techniques (other examples being Gappy POD [17] and missing point estimation or MPE [20]) are popular as hyper-reduction tools that bypass expensive nonlinear term computations in model order reduction. Naturally, one can adopt these interpolation points for sensor placement in sparse reconstruction applications. To illustrate this, the POD-based DEIM approximation of order M (number of interpolation points) for $u(t)$ in the space spanned by the basis $\Phi \in \mathbb{R}^{N \times M}$ is given by

$$u(t) = \Phi a(t) \tag{17}$$

where $a(t) \in \mathbb{R}^M$ is coefficient vector. When using POD bases, Φ , one can simply estimate $a(t) = \Phi^T u(t)$, but this requires dealing with the higher dimensional state vectors $\in \mathbb{R}^N$ that are computationally comparable to high-fidelity models even when using projection-based approaches for model reduction. Hyper-reduction strategies [19] bypass this issue by estimating the approximate coefficients $a(t)$ using carefully chosen set of M interpolation points instead of the full (N -dimensional) state so that computational cost scales with M . Specifically, one chooses M interpolation points corresponding to indices $[q_1, \dots, q_M]$, $q_i \in \mathbb{N}$ to form a M -by- M linear system $D^T \Phi a(t) = D^T u(t)$, where the interpolation or measurement matrix is given by $D = [e_{q_1}, \dots, e_{q_M}] \in \mathbb{R}^{N \times M}$ with columns $e_{q_i} = [0, \dots, 0, \underbrace{1}_{q_i}, 0, \dots, 0]^T \in \mathbb{R}^N$. The DEIM approximation of $u(t)$ then becomes

$$u_{DEIM}(t) = \underbrace{\Phi(D^T \Phi)^{-1}}_{N \times M} \underbrace{D^T u(t)}_{M \times 1}, \tag{18}$$

where $\Phi(D^T \Phi)^{-1}$ is typically precomputed once to yield a $N \times M$ matrix while $D^T u(t)$ represents an M -dimensional representation of the state at the interpolation points. This way, one avoids repeated computation of the high-dimensional $u(t)$. One can easily see the connections between DEIM approximation and the sparse recovered state $x_{SR} = \Phi a$ with a obtained using Equations (5)–(8) using $C = D^T$. Therefore, estimating the interpolation points (D^T) is similar to estimating the sparse measurement locations in C . The indices q_1, \dots, q_M are estimated sequentially from the input basis $\{\Phi_j\}_{j=1}^M$ using Algorithm 2 from [18].

The process starts from selecting the first interpolation index $q_1 \in \{1, 2, \dots, N\}$ using the first input POD basis $|\Phi_1|$. The remaining interpolation indices $\{q_j, j = 2, 3, \dots, M\}$ are selected such that they correspond to the largest magnitude of $|\mathbf{r}|$ where $\mathbf{r} = \Phi_j - \Phi a$ (see Line 5 of the Algorithm 2) is the residual error between current input basis and its interpolation Φa obtained using $\{\Phi_1, \Phi_2, \Phi_3 \dots \Phi_{j-1}\}$ at the indices $\{q_1, q_2, q_3 \dots q_{j-1}\}$. In Lines 1 and 6, the “max” function is the same as that available in MATLAB and $|\rho| = |r_{q_j}|$. The residual represents a measure of the linear independence of Φ_j with respect to the earlier basis vectors in the sequence and the interpolation point is at the maximum absolute value of this measure. Naturally, the realized q_j depends on the choice of basis Φ_j and their sequence, whereas the ordering of the input basis is not critical for QR-pivoting. The linear independence of the input basis ensures the above procedure is well-defined, i.e., $D^T \Phi$ is non-singular and $\rho \neq 0$ for all iterations. By using POD basis as the input, the linear independence and hierarchy

(i.e., basis ordered in terms of decreasing singular values) characteristics are guaranteed, which in turn ensures that the sparse interpolation indices are hierarchical and non-repeating.

Algorithm 2: Discrete Empirical Interpolation Method (DEIM).

input : $\{\Phi_j\}_{j=1}^M \subset \mathbb{R}^N$ linearly independent
output: $\vec{q} = [q_1, \dots, q_M]^T \in \mathbb{R}^M$

- 1 $[|\rho|, q_1] = \max\{|\Phi_1|\}$
- 2 $\Phi = [\Phi_1], D = [e_{q_1}], \vec{q} = [q_1]$
- 3 **for** $j = 2, \dots, M$ **do**
- 4 Solve $(D^T \Phi)a = D^T \Phi_j$ for a
- 5 $r = \Phi_j - \Phi a$
- 6 $[|\rho|, q_j] = \max\{|r|\}$
- 7 $\Phi \leftarrow [\Phi \ \Phi_j], D \leftarrow [D \ e_{q_j}], \vec{q} \leftarrow \begin{bmatrix} \vec{q} \\ q_j \end{bmatrix}$
- 8 **end**

2.5. Sparse Recovery (SR) Framework

The reconstruction algorithm used in this work based on the Gappy POD or GPOD framework [17] and is an l_2 minimization solution of the sparse recovery problem summarized in Equations (4)–(6) with Φ composed of $K \leq M$ basis vectors, i.e., dimension of a is $K \leq M$. At this point, we remind the reader of naming conventions adopted in this paper: the instantaneous j th full flow state is denoted by $x_j \in \mathbb{R}^N$, whereas the entire set consisting of M snapshots is assembled into a matrix form denoted by $X \in \mathbb{R}^{N \times M}$. This discussion focuses on single snapshot reconstruction as the extension to multiple snapshots is trivial, i.e., each snapshot can be reconstructed sequentially or in some cases be grouped together as a batch. This allows for such algorithms to be parallelized easily.

The primary difference between the SR framework in Equation (4) as used in compressive sensing or DEIM-based approaches and GPOD [12,13,17,61] as shown in Equation (20) is the construction of the measurement matrix C and the sparse measurement vector \tilde{x}_j . In Equation (4), the down sampled state $\tilde{x}_j \in \mathbb{R}^P$ is a *compressed* version containing only the measured data, whereas in GPOD, $\tilde{x}_j \in \mathbb{R}^N$ is a *masked* version of the full state vector, i.e., values outside of the P measured locations are zeroed out to generate a filtered version of x_j . For high-resolution data $x_j \in \mathbb{R}^N$ with chosen basis $\Phi_j \in \mathbb{R}^N$, the low-dimensional features, $a^j \in \mathbb{R}^K$ are obtained from the relationship shown below:

$$x_j = \sum_{i=1}^K \Phi_i a_i^j. \tag{19}$$

The masked (incomplete) data $\tilde{x}_j \in \mathbb{R}^N$, corresponding measurement matrix C and mask vector $m \in \mathbb{R}^N$ are related by:

$$\tilde{x}_j = \langle m \cdot x_j \rangle = C x_j, \tag{20}$$

where $C \in \mathbb{R}^{N \times N}$. Therefore, the GPOD algorithm results in a larger measurement matrix with numerous rows of zeros, as shown in Figure 3 (compare with Figure 1). To bypass the complexity of handling this $N \times N$ matrix, a mask vector, $m \in \mathbb{Z}^{N \times 1}$ (representing the diagonal elements of C) with 1s and 0s operates on x_j through a point-wise multiplication operator $\langle \cdot \rangle$. To illustrate, the point-wise multiplication is represented as $\tilde{x}_j = \langle m_j \cdot x_j \rangle$ for each snapshot $j = 1, \dots, M$ where each element of x_j multiplies with the corresponding element of m_j . This is applicable even when each data snapshot, x_j , has its own measurement mask m_j , which is a useful way to represent the evolution of sparse sensor locations over time. The SR formulation in Equation (4) can also support

time varying sensor placement, but would require a compression matrix, $C_j \in \mathbb{R}^{P \times N}$ that changes with each snapshot.

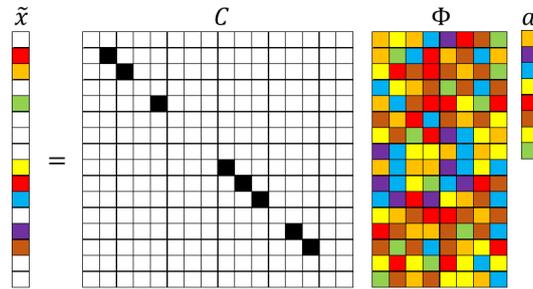


Figure 3. Schematic of GPOD formulation for sparse recovery. The numerical values represented by the colored blocks: black (1), white (0), and color (other numbers).

The goal of the SR procedure is to recover the full data from the masked data,

$$\tilde{x}_j \approx m_j \sum_{i=1}^K \bar{a}_i^j \Phi_i, \tag{21}$$

by approximating the coefficients \bar{a}^j (in the l_2 sense) with basis, Φ_K , learned offline using training data (snapshots of the full field data). The coefficient vector \bar{a}^j cannot be computed by direct projection of \tilde{x}_j onto Φ as these are not designed to optimally represent the sparse data. Instead, one needs to obtain the “best” approximation of \bar{a}^j , by minimizing the error \mathcal{E}_j in the l_2 sense as

$$\mathcal{E}_j = \left\| \tilde{x}_j - m_j \sum_{i=1}^K \bar{a}_i^j \Phi_i \right\|_2^2 = \left\| \tilde{x}_j - m_j \cdot \Phi \bar{a}^j \right\|_2^2 = \left\| \tilde{x}_j - C \Phi \bar{a}^j \right\|_2^2. \tag{22}$$

In Equation (22), we see that m_j acts on each column of Φ through a point-wise multiplication operation, which is equivalent to masking each basis vector Φ_j . The above formulation is valid for a single snapshot reconstruction when the mask vector, m_j , changes with each snapshot \tilde{x}_j for $j = 1, \dots, M$ and the error \mathcal{E}_j represents the single snapshot reconstruction error to be minimized. It can easily be seen from below that one will have to minimize the different \mathcal{E}_j s sequentially to learn the entire coefficient matrix, $\bar{a} \in \mathcal{R}^{K \times M}$ for all the M snapshots. Denoting the masked basis functions as $\tilde{\Phi}_i = \langle m_j \cdot \Phi_i \rangle$, Equation (22) is rewritten as

$$\mathcal{E}_j = \left\| \tilde{x}_j - \sum_{i=1}^K \bar{a}_i^j \tilde{\Phi}_i \right\|_2^2. \tag{23}$$

In the above, $\tilde{\Phi}$ is analogous to $\Theta = C \Phi$ in Equation (4). If $\tilde{\Phi}$ is snapshot independent, then all the different snapshots can be recovered simultaneously. To minimize \mathcal{E}_j , one sets the derivative with respect to \bar{a}_j as zero to yield a normal equation,

$$\mathbf{M} \bar{a}^j = f^j, \tag{24}$$

where $\mathbf{M}_{k1,k2} = \langle \tilde{\Phi}_{k1}, \tilde{\Phi}_{k2} \rangle$ or $\mathbf{M} = \tilde{\Phi}^T \tilde{\Phi}$ and $f_i^j = \langle \tilde{x}_j, \tilde{\Phi}_i \rangle$ or $f^j = \tilde{\Phi}^T \tilde{x}_j$. The reconstructed solution is then given by

$$\bar{x}_j = \sum_{k=1}^K \Phi_k \bar{a}_k^j. \tag{25}$$

Algorithm 3 summarizes the above steps assuming the basis functions (Φ_k) are known.

Algorithm 3: Least squares (l_2) sparse reconstruction with basis Φ .

input : Basis $\Phi \in \mathbb{R}^{N \times N_b}$
 Incomplete data vector $\tilde{X} \in \mathbb{R}^{N \times M}$
 the mask vector $m \in \mathbb{R}^N$
output: Approximated full data vector $\bar{X} \in \mathbb{R}^{N \times M}$

- 1 **for** each snapshot index $j \leq M$ **do**
- 2 Build a least squares problem: $\mathbf{M}\bar{a}^j = f^j$ (Equation (24)) as below Compute masked basis function: $\tilde{\Phi} = m\Phi$ Compute matrix $\mathbf{M} = \tilde{\Phi}^T \tilde{\Phi}$ Compute vector: $f^j = \tilde{\Phi}^T \tilde{x}_j$; Solve \bar{a}^j from the least squares problem: $\mathbf{M}\bar{a}^j = f^j$ (Equation (24)) Reconstruct the approximated solution $\bar{x}_j = \Phi \bar{a}^j$ (Equation (25))
- 3 **end**

2.6. Algorithmic Complexity

The cost of computing the POD basis is $O(N \times M^2)$ where N, M are the full state and snapshot dimensions, respectively. The cost of sparse reconstruction turns out to be $O(N \times K \times M)$ for both methods, where the $K \leq M$ is the chosen reconstruction dimension. Naturally, for many practical problems with an underlying low-dimensional structure, POD is expected to result in a smaller K than any other choice of basis. This helps reduce the sensor quantity requirement and reconstruction cost. Further, since the number of snapshots (M) is also tied to the desired basis dimension (K), a larger K requires a larger M and in turn is more expensive to generate the basis.

The complexity of the sensor placement depends on the choice of method adopted. Using QR factorization with column pivoting, the cost of sensor placement for a $N \times N$ matrix is $O(N^3)$ and $O(NM^2)$ for a $N \times M$ matrix. The DEIM method carries a complexity of $O(NM^3)$ when retaining M POD modes and identifying M sensors with a state dimension of N . The MCN algorithm requires an expensive combinatorial search to find the sensor location that minimizes the condition number of $\mathbf{M} \in \mathbb{R}^{M \times M}$ in each possible sweep. This results in a computational cost of $O(N^2M^3)$ for identifying M sensors. These estimates are consistent with our observation that the DEIM and QR-pivoting approaches yield results in very short time.

3. Data Generation for Numerical Experiments

3.1. Low-Dimensional Cylinder Wake Flows

Studies of cylinder wakes [26,62–64] have attracted considerable interest from the model reduction and dynamical systems communities for its particularly rich flow physics content, encompassing many of the complexities of nonlinear flow systems, and yet, easy to simulate accurately. In this paper, we explore data-driven sparse reconstruction for the unsteady cylinder wake flow at Reynolds number $Re = 100$. To generate two-dimensional cylinder flow data, we adopt the spectral Galerkin method [65] with a fourth-order spectral expansion within each element to solve the incompressible Navier–Stokes equations,

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0 \tag{26}$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{\partial P}{\partial x} + \nu \nabla^2 u \tag{27}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -\frac{\partial P}{\partial y} + \nu \nabla^2 v \tag{28}$$

where u and v are the horizontal and vertical velocity components, P is the pressure field, and ν is the fluid viscosity. The rectangular domain used for this flow problem is $-25D < x < 45D$ and $-20D < y < 20D$, where D is the diameter of the cylinder. For the purposes of this study, data from a

reduced domain, i.e., $-2D < x < 10D$ and $-3D < y < 3D$ consisting of 24,000 grid points, are used. The mesh was designed to sufficiently resolve the thin shear layers near the surface of the cylinder and transient wake physics downstream. We collect snapshots of data every $\Delta t = 0.2$ s.

3.1.1. Cylinder Wake Limit-Cycle Dynamics

In this section, we explore sparse reconstruction of unsteady wake flows with well-developed periodic vortex shedding behavior. The GPOD type algorithm is chosen over the traditional compressive sensing-based SR formulations to bypass the need for maintaining a separate measurement matrix, especially when employing point sensors to mimic realistic data acquisition. The time-evolution of the cylinder wake is shown in Figure 4 where the wake becomes increasingly unstable before it settles into a limit-cycle. The first three POD modes and coefficients are shown in Figure 5 for the limit-cycle regime. The dominant POD modes (mode 1 and mode 2) capture the symmetric vortex shedding patterns while the temporal evolution of POD coefficients show periodic evolution. The low dimensionality of this system is evident from the singular value spectrum for the data matrix shown in Figure 6. For this study, we used 300 snapshots collected every 0.2 units corresponding to sixty non-dimensional times, $T = \frac{ut}{D}$, that correspond to multiple (≈ 10) cycles of the periodic dynamics.

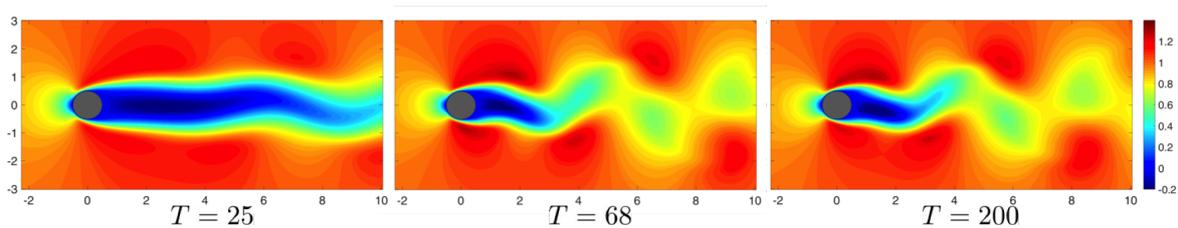


Figure 4. Isocontour plots of the stream-wise velocity component for the cylinder flow at $Re = 100$ at $T = 25, 68, 200$ show evolution of the flow field. Here, T represents the time non-dimensionalized by the advection time-scale.

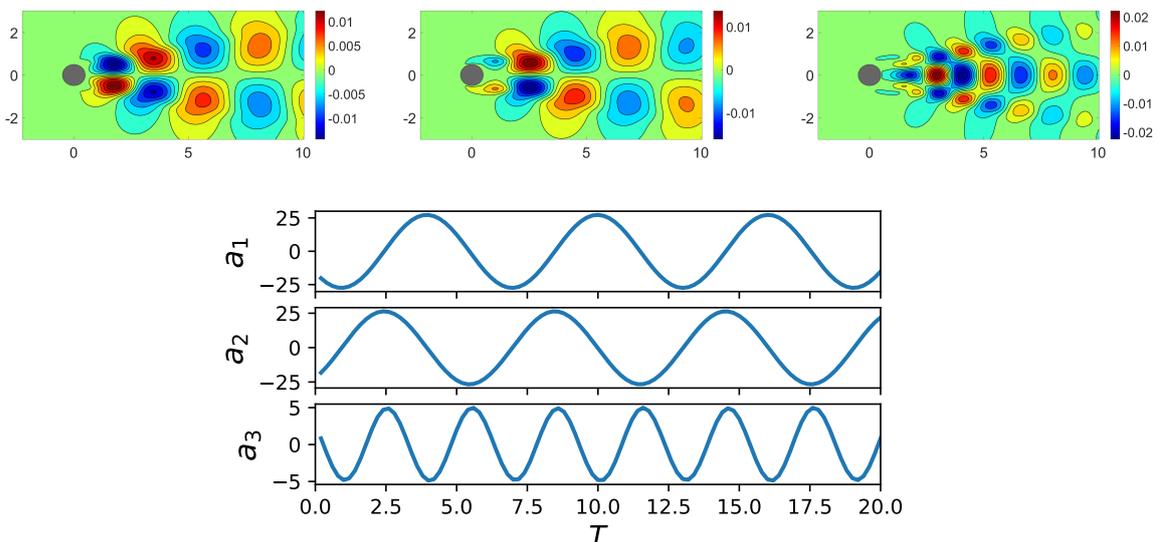


Figure 5. Isocontours of the three most energetic modes (first row from left to right) and time evolution of the first three POD coefficients (second row) for the cylinder wake flow at $Re = 100$.

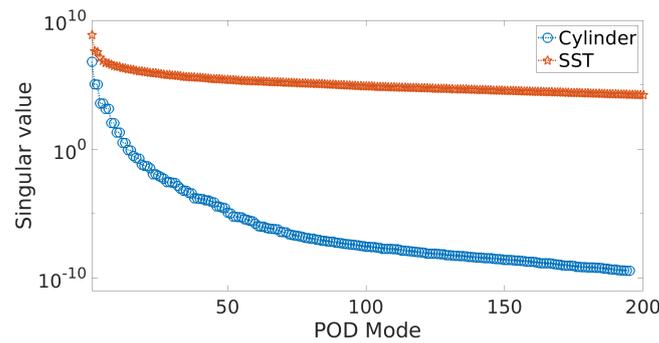


Figure 6. Singular value spectrum of the data matrix for both the cylinder wake flow at $Re = 100$ and the sea surface temperature (SST) data.

3.1.2. Smart Sensor Placement in Cylinder Wake

Figure 7 summarizes the realized sensor placement for a budget of 200 sensors using four different algorithms: (i) random placement (Section 2.4.1); (ii) QR factorization with column pivoting that maximizes the determinant capture for a given P (Section 2.4.3); (iii) discrete empirical interpolation method (DEIM) (Section 2.4.4); and (iv) explicit condition number minimization through an iterative combinatorial search (Section 2.4.2). All these sensor placement algorithms use POD bases computed with 300 snapshots of data containing both u and v velocity components. As expected, the random method in Figure 7a samples the entire domain, although biased towards regions with higher grid density such as the region close to the cylinder surface where the mesh resolution is high. On the other hand, the three “smart” sensor placement algorithms shown in Figure 7b–d identify locations either in the wake or on the cylinder surface. The QR-Pivot and the DEIM methods identify a cluster of centers about two diameters downstream of the cylinder. Further downstream, the computed centers split into two narrow “tail”-like regions on either side of the wake centerline. On the other hand, the MCN framework (Figure 7d) identifies only five sensors in the wake, mostly downstream while an overwhelming number are predicted on the cylinder surface. This may be attributed to the low-dimensional wake physics where only a few POD modes are needed to capture a significant fraction of the energy.

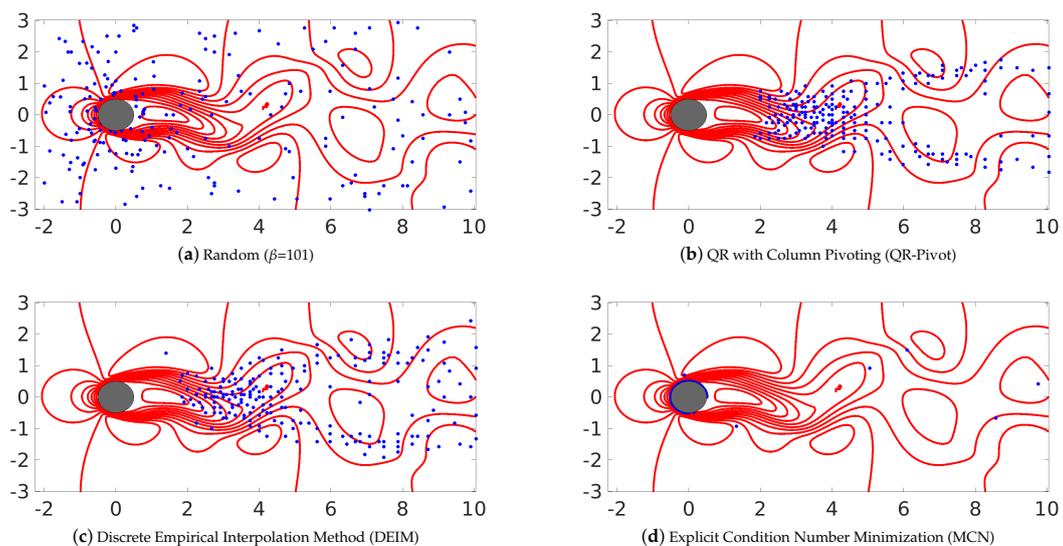


Figure 7. Cont.

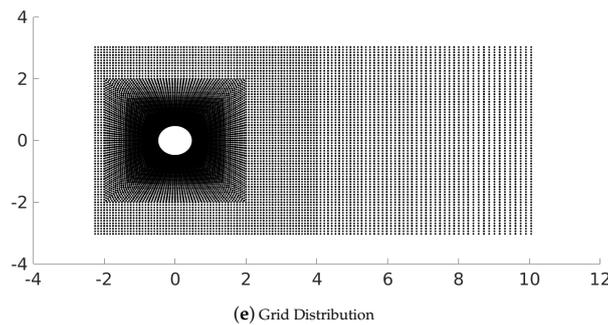


Figure 7. Sensor locations generated using the different methods considered in this work including both random as well as smart algorithms for the cylinder wake flow ($Re = 100$). These image show the most relevant 200 sensors, i.e., ($P = 200$), while (e) represents the grid distribution.

The cylinder wake flow is a useful low-dimensional system for analysis, verification and validation purposes as one can easily identify regions of critical activity for sensor placement and recovery. In particular, this case helps in understanding the limitations and accuracy of the different methodologies, even those with significant enough computational cost to be not used in practice.

3.2. Global Sea Surface Temperature (SST) Data

To showcase the practical capabilities of the algorithms presented here, we need problems that mimic real-life complexity. For this reason, we also consider sea surface temperature (SST) data representing coarse grained version of synoptic scale ocean turbulence and characterized by rich dynamics of synoptic-scale seasonal fluctuations interacting with local and day-to-day non-stationary dynamics from turbulent eddy currents. For this study, we chose a dataset consisting of daily mean quantities from high-resolution blended analysis of sea surface temperature from 2018 made available by the National Oceanic & Atmospheric Administration (NOAA) <https://www.esrl.noaa.gov/psd/>. For this year long data, we have 365 snapshots of daily mean temperature fields with a spatial resolution of 720×1440 (0.25 degree longitude \times 0.25 degree latitude global grid). This results in a total state dimension of 1,036,800 observations, of which only 691,150 correspond to ocean regions and were used in this study.

3.2.1. Sea Surface Temperature (SST) Dynamics

For this data matrix, the singular value spectra is shown in Figure 6. It is evident from the above plots that the SST data have a slower decay of singular values and will require more modes to capture the same energy fraction relative to the cylinder flow. We note that, by using filtered temperature fields (i.e., averaging over any given day), the scale separation is significantly reduced as compared to what would be observed in high Reynolds number turbulence. The dominant modes and the temporal evolution of the POD coefficients is shown in Figure 8.

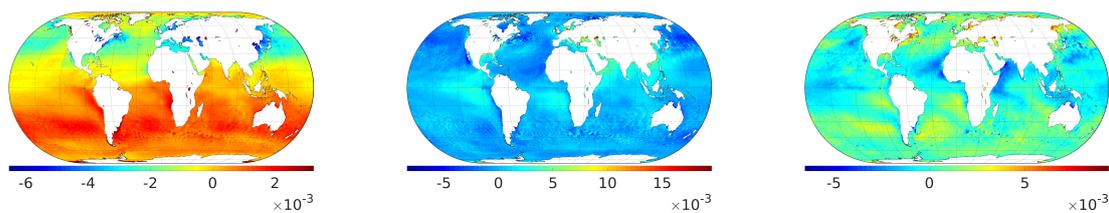


Figure 8. Cont.

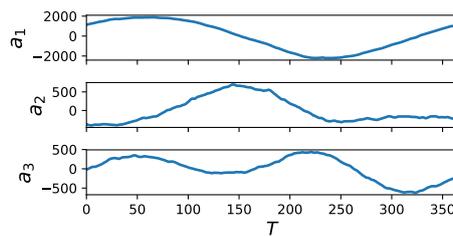
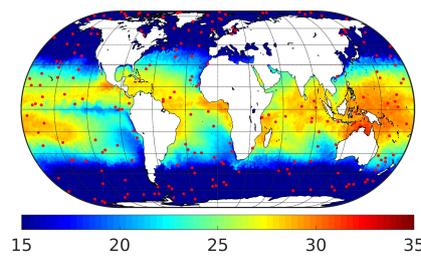


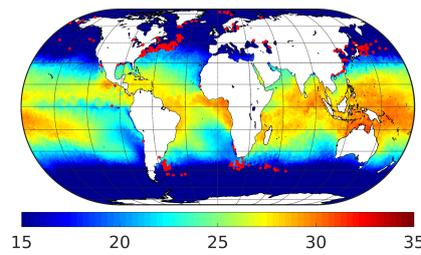
Figure 8. Visualization of the first three POD modes (top left to right) and POD coefficients (bottom) for the sea surface temperature data.

3.2.2. Smart Sensor Placement for SST Fields

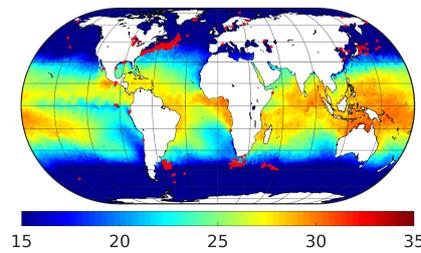
Figure 9 summarizes the realized sensor placement for a budget of 200 sensors using the three different algorithms: (i) random; (ii) QR with pivoting; and (iii) DEIM. For these data, we did not pursue the explicit condition number minimization due to its exorbitant computational cost. All these sensor placement algorithms use POD bases computed using 365 snapshots of temperature fields. As before, random sensor placement (Figure 9b) samples over the entire field, the greedy sensor placement methods (Figure 9b,c) cluster the sensors near coastal regions.



(a) Random ($P = 200$)



(b) QR ($P = 200$)



(c) DEIM ($P = 200$)

Figure 9. Illustration of the 200 most relevant sensor locations (red dots) generated using: (a) random; (b) QR-pivot; and (c) DEIM methods for the sea surface temperature (in °C) data.

4. Sparse Reconstruction of Flow Fields

In this section, we explore sparse reconstruction of both simple and complex flow fields in the form of low Reynolds number cylinder wake ($Re = 100$) and synoptic scale turbulent temperature fields from global weather models using the above SR infrastructure. Adopting both laminar wake and geophysical turbulent flows allows us to evaluate the performance of the algorithms for both interpretable low-dimensional as well as complex high-dimensional systems observed in practice. In this study, we adopted the GPOD formulation as against the traditional SR formulation. This choice was purely a matter of convenience and helped bypass the need for maintaining a separate measurement matrix. In all the cases reported in this section, Tikhonov regularization was employed to generate unique solutions.

4.1. Sparse Reconstruction (SR) Experiments and Analysis

For this a priori assessment of SR performance, we reconstructed sparse data from simulations where the full field representation is available. The sparse sensor locations were chosen as single point measurements using the different sensor placement methods discussed in Section 3.1.2 and these locations were fixed for the ensemble of snapshots used for the reconstruction (i.e., we did not consider dynamic sensor placement). Reconstruction performance was evaluated by comparison of the SR data with the simulated field at truth across the entire ensemble of numerical experiments. Of course, using such a data-driven basis requires availability of training data so that one can compute the basis vectors a priori. In practice, one would have to build a library of basis vectors from data that can in turn be used for sparse recovery. In this study, we undertook this a priori approach to narrowly focus on the relative roles of reconstruction dimension (K), sensor budget (P) and placement (C) for the POD-based SR. In particular, we aimed to accomplish the following through this study: (i) quantify the extent of oversampling relative to desired system dimension ($P > K$) needed for sufficiently accurate POD-based l_2 reconstruction of fluid flow data; and (ii) understand how sensor placement impacts reconstruction quality.

To learn the data-driven POD basis, we employed the method of snapshots [51] over the full data ensemble, which gave rise to at most M basis, i.e., a candidate basis dimension of $N_b = M$. As shown in Table 1, the choice of algorithms depend on the choice of reconstruction dimension (K), sensor budget (P) and candidate basis dimension, N_b . Recalling from before (Section 2), we see that $P \geq K$ is handled using an l_2 method as long as $P \geq N_b$. In the case of POD-based SR, the basis vectors optimize the variance capture for the training data and contain built-in ordering for representation of the system state. Therefore, the POD basis needed to be generated once and the reconstruction dimension was chosen as the first K modes to be retained in the given sequence. For generic basis with no known ordering, one needs to search for the K most significant basis amongst the maximum possible dimension of $N_b = M$ using sparsity promoting l_1 methods requiring increased computational cost.

4.2. Comparison of l_2 and l_1 Sparse Reconstruction Using Energy-Ordered POD Basis

In this subsection, we verify the equivalence between l_2 and sparsity promoting l_1 – reconstruction using such energy ordered POD basis with the same sensor measurements. This verification implies that l_1 – sparse estimates match the least squares estimates with energy-ordered basis truncation, thus paving the way to adopt the latter framework in the current study. Of course, this is not applicable in the generic case where the ability of basis in the library to approximate given data is not known a priori. In this analysis, we considered two cases of reconstruction for the limit-cycle cylinder wake at $Re = 100$: Case (a), $P > K$, $K = N_b$ and using least squares reconstruction; and Case (b), $P > K$, $K < N_b \leq M$ employing l_1 reconstruction using optimal matching pursuit or CoSAMP, as described in Needell and Tropp [11]. Figure 10 and Table 2 compare the reconstructed fields and the K predicted weights for the two cases for a single snapshot corresponding to $T = 0.2$.

Table 2. Problem design for $l_1 - l_2$ comparison.

Method	Re	K	P	N_b	β
l_2	100	10	40	10	101
l_1	100	10	40	20	101

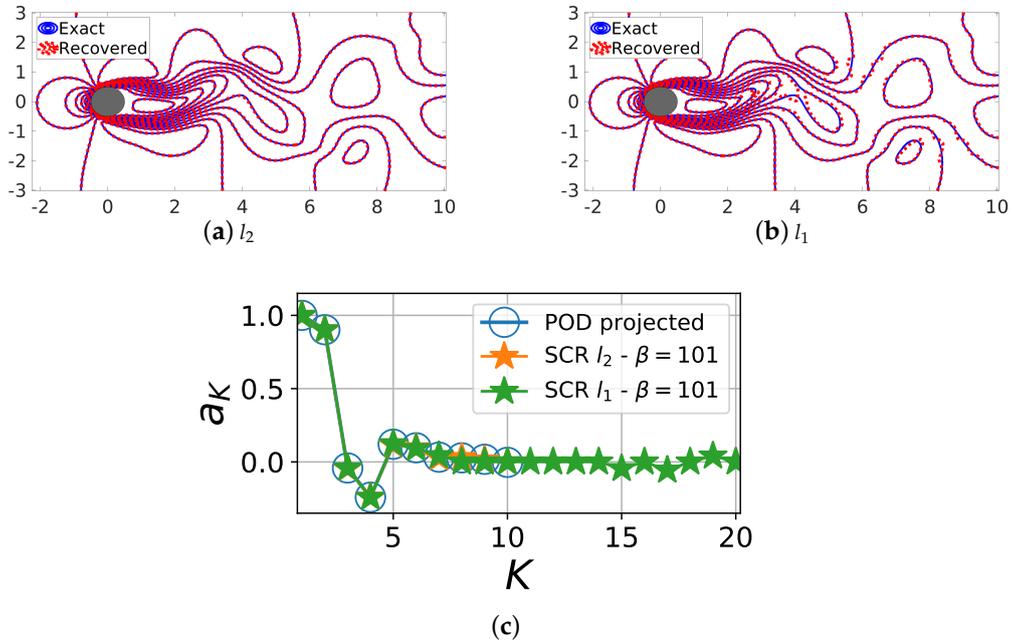


Figure 10. Comparison of the sparse reconstruction using both l_1 and l_2 minimization methods for basis that is ordered in terms of energy content. The reconstructed and actual flowfields at $T = 0.2$ are compared: (a) for l_2 ; and (b) for l_1 . The corresponding POD features from both methods are shown in (c).

The above results confirm that l_1 -based estimation of the coefficients match those from l_2 -based estimation and therefore the equivalence between data-driven sparsity and energy-sparsity. However, the smaller coefficients predicted by l_1 introduce errors that introduces noise to the reconstructed fields. For the rest of this article, we leverage energy hierarchy of POD basis and l_2 SR with $K = N_b$ such that the candidate basis set is chosen based on the desired reconstruction dimension.

4.3. Sparsity and Energy Metrics

We aimed to explore the conditions for accurate recovery of information in terms of data availability (P) and system dimensionality (K), i.e., dimension of the system for a chosen representational accuracy using a given basis. As long as the measurements are incoherent with respect to Φ and the system is overdetermined, i.e., $P > K$, one should be able to recover the higher dimensional state, X in a manner consistent with earlier discussions on l_0 minimization in Section 2. To this end, we summarize different sparsity and energy metrics so that the sensor requirement and reconstruction error expectation for a chosen dimension can be characterized. For POD, one easily defines a cumulative energy fraction captured by the K most energetic modes, \mathbb{E}_K , using the singular values (λ) of the data matrix as

$$\mathbb{E}_K = \sum_{k=1}^K \frac{\lambda_k}{(\lambda_1 + \lambda_2 + \dots + \lambda_M)} \times 100, \tag{29}$$

where M is the total number of possible eigenvalues. The energy fraction \mathbb{E}_K corresponding to dimension K for the cylinder wake flow and the global sea surface temperature data is shown

in Figure 11. For the cylinder flow, one requires two and five POD modes to capture 95% and 99% of the energy, respectively, while the SST data require 9 and 66 modes. We denote the dimension corresponding to 95% and 99% energy capture as K_{95} and K_{99} , respectively. To quantify SR performance across flow regimes with different dimensions (K, K_{95}), we define a normalized dimension metric, $K^* = K/K_{95}$, and a normalized sensor budget, $P^* = P/K_{95}$. This allows us to design an ensemble of numerical experiments in the discretized $P^* - K^*$ space so that the outcomes can be characterized effectively. In this work, the design space is populated over the range $1 < K^* < 6$ and $1 < P^* < 12$ for POD-based SR with $K \leq M$. The lower bound of one is chosen such that the minimally accurate reconstruction captures 95% of the energy. One can chose another dimension norm without loss of generality.

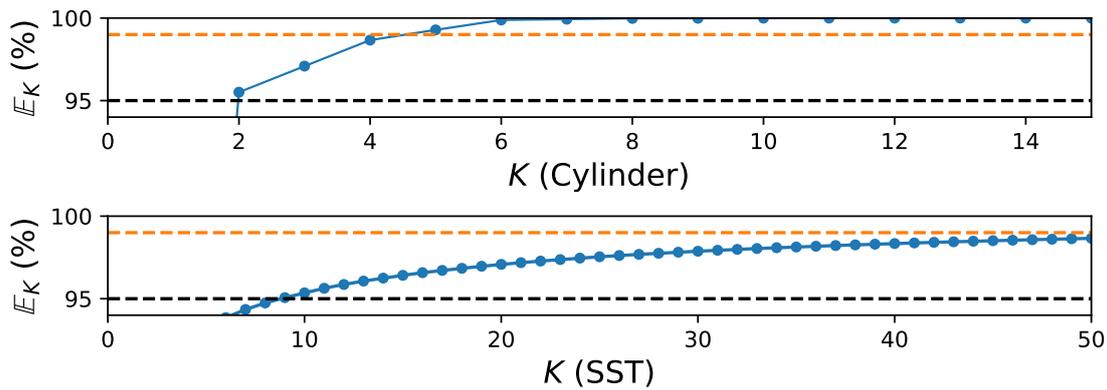


Figure 11. Schematic shows the cumulative energy capture corresponding to different system dimension, K (i.e., the number of POD modes), for cylinder flow at $Re = 100$ (**top**) and sea surface temperature (SST) data (**bottom**).

To quantify the l_2 reconstruction performance, we define the mean squared error as shown in Equation (30) below,

$$\epsilon_{K^*,P^*}^{SR} = \frac{1}{M} \frac{1}{N} \sum_{j=1}^M \sum_{i=1}^N (X_{i,j} - \bar{X}_{i,j}^{SR})^2, \tag{30}$$

where X represents the true data, and \bar{X}^{SR} is the reconstructed field from sparse measurements as per Algorithm 3. In the above, N and M represent the state and snapshot dimensions corresponding to indices i and j . Similarly, the mean squared errors $\epsilon_{K_{95}}^{FR}$ and $\epsilon_{K^*}^{FR}$ for reconstruction using full data with POD basis are computed as

$$\epsilon_{K_{95}}^{FR} = \frac{1}{M} \frac{1}{N} \sum_{j=1}^M \sum_{i=1}^N (X_{i,j} - \bar{X}_{i,j}^{FR,K_{95}^*})^2 \text{ and} \tag{31}$$

$$\epsilon_{K^*}^{FR} = \frac{1}{M} \frac{1}{N} \sum_{j=1}^M \sum_{i=1}^N (X_{i,j} - \bar{X}_{i,j}^{FR,K^*})^2, \tag{32}$$

where $\bar{X}^{FR} = \Phi \mathbf{a}$ is the full data reconstruction using exact POD coefficients, $\mathbf{a} = \Phi^T X$. The normalized dimension for 95% energy capture, K_{95}^* , is trivially seen to be unity.

Using the above definitions, we can now normalize the absolute (ϵ_1) and relative (ϵ_2) errors as

$$\epsilon_1 = \frac{\epsilon_{K^*,P^*}^{SR}}{\epsilon_{K_{95}}^{FR}}, \quad \epsilon_2 = \frac{\epsilon_{K^*,P^*}^{SR}}{\epsilon_{K^*}^{FR}}, \tag{33}$$

where ϵ_1 represents the SR error normalized by the corresponding full data reconstruction error for 95% energy capture and ϵ_2 is the normalized error relative to the full data reconstruction error up to a

desired system dimension, K . These two error metrics are devised to quantify the overall quality of the SR in a normalized sense (ϵ_1) and the best possible reconstruction accuracy for a chosen problem set-up, i.e., P and K . Therefore, if the best possible reconstruction for a given K is realized, then ϵ_2 should achieve the same value across different K^* . This error metric is used to assess relative dependence of P^* on K^* for a chosen flow and the dependence on flow physics is expected to be minimal given that we are dealing with normalized metrics. On the other hand, ϵ_1 provides an absolute estimate of the reconstruction accuracy for a given flow system so that minimal values of P^*, K^* needed to achieve a desired recovery quality can be estimated. Using these metrics, we now assess the linear sparse estimation of fine-scale fields for both low-dimensional cylinder wake as well as geophysical turbulence.

4.4. Sparse Reconstruction of Low-Dimensional Wake Flow

To bare the aspects of interplay between the SR design variables, we performed numerous experiments corresponding to different points in the $P^* - K^*$ design space and for different sensor placements (fixed in time).

4.4.1. Sparse Reconstruction Accuracy

We computed the errors ϵ_1 and ϵ_2 as described in Section 4.3 across the $K^* - P^*$ space, the contours of which are shown in Figures 12 and 13 for both random and greedy sensor placements. As the random sensor placement results in high variability between realizations, we estimated multiple sets of sensor locations corresponding to different seeds (as denoted by β in this article). Specifically, we computed the SR errors from ten different random arrangements and the corresponding reconstruction errors are presented in terms of the mean, maximum and minimum (based on the average over the $K^* - P^*$ space). For the greedy “smart” sensor placements, a single realization is representative of the method (Figure 13). For ease of interpretation, the contour levels in both these figures are made consistent.

The relative error metric ϵ_2 (the right column in Figures 12 and 13), shows that the smaller errors (predominantly blue regions) are located in the region where $P^* > K^*$ and approximately separated from the rest of the $K^* - P^*$ space with a straight line given by $P = K + 1$. This indicates that the oversampled SR problem with $P > K$ yields good results in terms of ϵ_2 while for small P^* (i.e., under-sampled), the normalized relative error can reach as high as $\mathcal{O}(10^1 - 10^2)$. Since ϵ_2 is normalized by the error contained in the exact K -dimensional POD reconstruction, this metric represents how effectively the sparse data can approximate the K -dimensional solution using l_2 minimization for the given sensor quantity and placement. In principle, the exact K -sparse POD reconstruction is the best possible outcome to expect irrespective of how much sensor data are available. We also observe that ϵ_1 contours adhere to a L -shaped structure indicating that absolute normalized error reduces as both P and K increase due to oversampling and increased system representation. In practice, ϵ_1 is the more useful metric for planning and designing the sparse recovery framework for a given flow system.

While qualitatively accurate reconstruction is almost always observed for the higher values of P^* and K^* for the different sensor placements, there appear to be exceptions in the form of higher reconstruction errors even with oversampling. This is observed for both the random as well as smart sensing approaches. In fact, for random sensor placement, marginal oversampling results in ϵ_2 values of $\mathcal{O}(10^1)$ (colored as yellow in Figure 12) as against the expected range of $\mathcal{O}(1)$ range. This trend is observed for the greedy sensor placement methods as well, particularly QR-pivoting and MCN. Overall, the greedy methods show better reconstruction performance for the marginally oversampled cases, i.e., $P^* \approx K^*$, as compared to random sensing. These trends are not surprising given that oversampled ($P^* \gg K^*$) and under-sampled ($P^* \ll K^*$) reconstruction invariably generate low and high errors while the transition regime is sensitive to sensor placement. Therefore, in the following sections, we dissect the sparse recovery performance using specific examples of over- and marginal sampling.

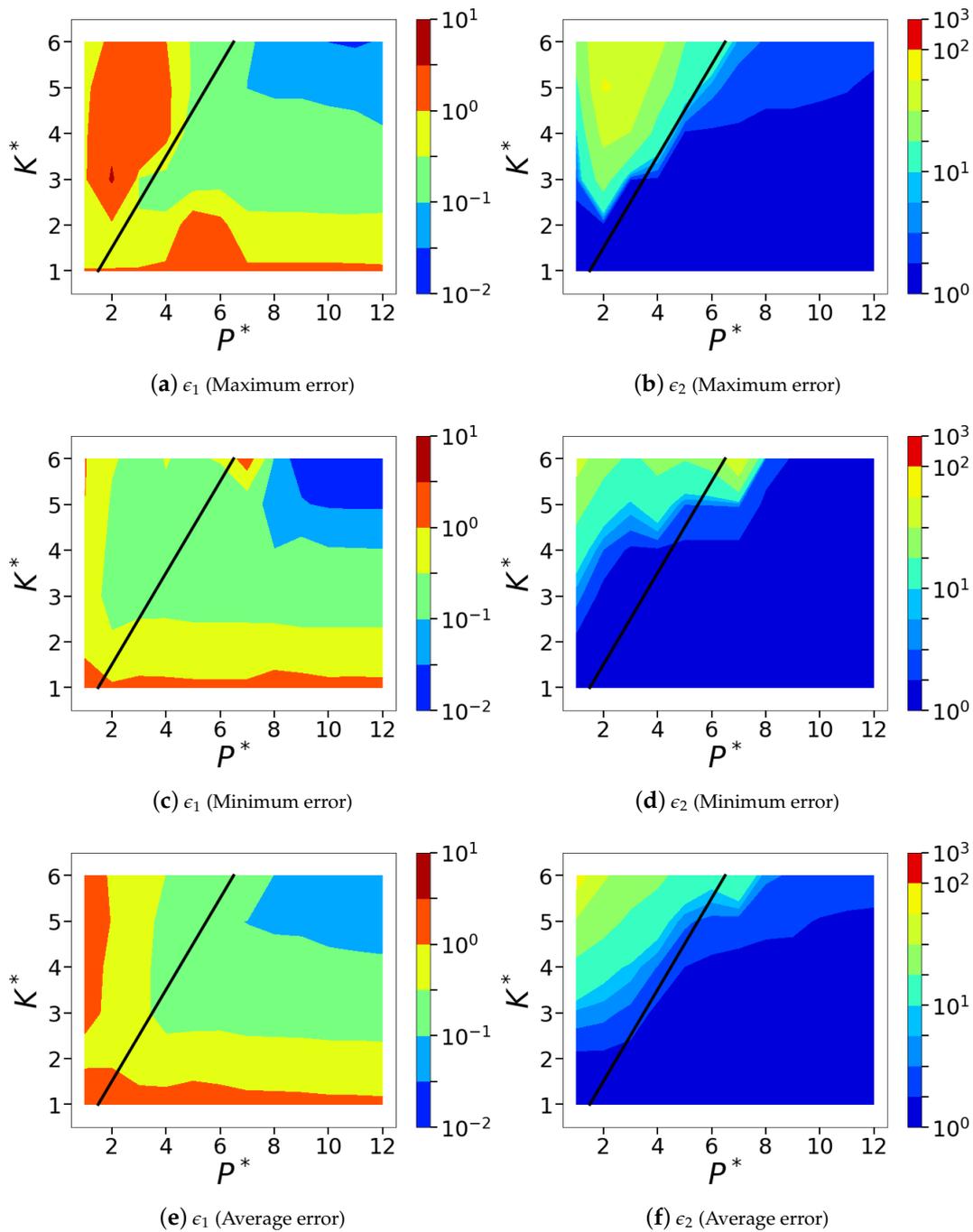


Figure 12. Isocontours of the normalized mean squared POD-based sparse reconstruction errors (l_2 norm) corresponding to the sensor placement with maximum and minimum errors from the chosen ensemble of random sensor arrangements. The average error across the entire ensemble of ten random sensor placements is also shown. Subfigures (a,c,e) show the normalized absolute error metric, ϵ_1 and subfigures (b,d,f) show the normalized relative error metric, ϵ_2 . Top row corresponds to the case with maximum error; middle row shows the case with minimum error and the bottom row shows the averaged error across different seeds.

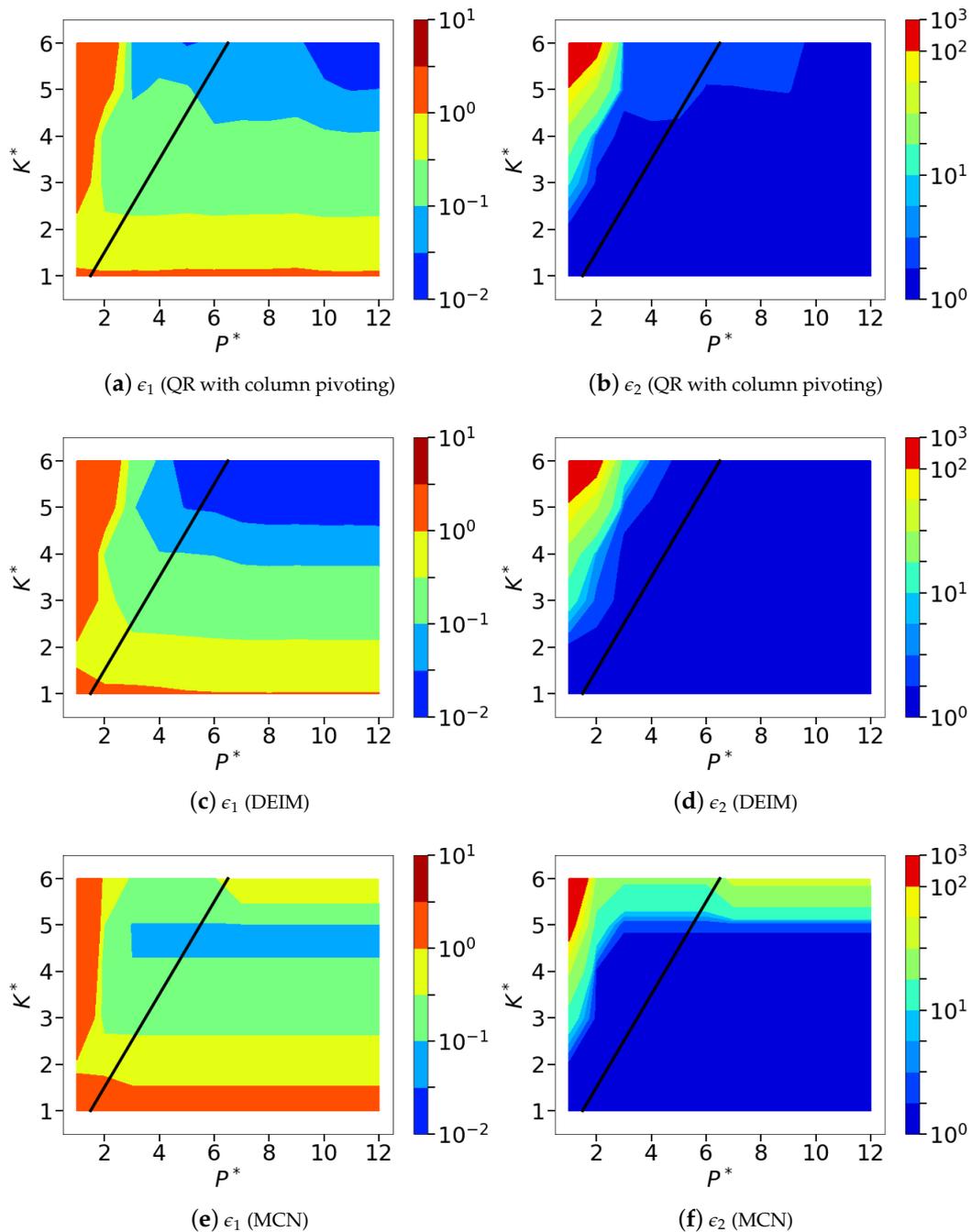


Figure 13. Isocontours of the normalized mean squared POD-based sparse reconstruction errors (l_2 norm) corresponding to the different greedy sensor placement methods, namely, QR with column pivoting (a,b), DEIM (c,d) and minimum condition number (e,f). Subfigures (a,c,e), show the normalized absolute error metric, ϵ_1 and subfigures (b,d,f) show the normalized relative error metric, ϵ_2 .

4.4.2. Assessment of Sensor Placement

Among the different greedy sensor placement methods experimented in this work, DEIM provides the most reliable reconstruction (Figure 13c,d) while closely followed by QR-pivoting (Figure 13a,b). MCN, which explicitly minimizes the condition number of M , shows good reconstruction accuracy for smaller values of $K^* \sim 4-5$ (see Figure 13e,f) that is more than sufficient for many practical estimation problems. The anomaly observed for reconstruction dimensions beyond $K^* \approx 5$ is due to very few sensors being generated in the wake downstream of the cylinder, as shown in Figure 7d.

Table 3. Sparse reconstruction performance quantification for different sensor location selection method for periodic cylinder flows at $Re = 100$. ϵ_1 is the SR error normalized by the exact reconstruction error corresponding to a dimension of K_{95} . ϵ_2 is the SR error normalized by the exact reconstruction error corresponding to a dimension of K .

Method	K	P	K^*	P^*	μ_u	μ_v	ϵ_1	ϵ_2
Random ($\beta = 101$)	2	20	1.0	10.0	2.548	2.306	1.08	1.08
	4	20	2.0	10.0	2.548	3.247	6.71×10^{-1}	1.23
	6	20	3.0	10.0	2.548	4.186	3.17×10^{-1}	1.96
QR-Pivoting	2	20	1.0	10.0	2.520	3.794	1.04	1.04
	4	20	2.0	10.0	3.917	3.794	6.05×10^{-1}	1.11
	6	20	3.0	10.0	3.917	4.506	1.88×10^{-1}	1.16
DEIM	2	20	1.0	10.0	2.323	3.720	1.00	1.00
	4	20	2.0	10.0	3.685	3.867	5.56×10^{-1}	1.02
	6	20	3.0	10.0	3.685	4.562	1.69×10^{-1}	1.05
MCN	2	20	1.0	10.0	1.090	2.213	1.15	1.15
	4	20	2.0	10.0	1.476	3.502	8.70×10^{-1}	1.59
	6	20	3.0	10.0	1.476	3.725	2.76×10^{-1}	1.71

Although the error metrics serve as a useful indicator of performance, we also compare the instantaneous sparse recovered flow field and the estimated POD weights in Figure 14. In particular, we show results for oversampled conditions with $P^* = 10$ and $K^* = 1, 2, 3$ for which the error metrics in Figures 12 and 13 are small. As expected, the accuracy of the linear estimation improves with K^* . Further, amongst the different oversampled experiments, DEIM and QR-pivoting provide the most accurate estimation of the POD coefficients (a), especially at higher K^* . The relative inaccuracy of the MCN framework is observable even for these carefully chosen design points with low error metrics. For such low-dimensional flows, small errors in a amplify the discrepancy in full field reconstruction. The relevant quantifications including sensor budget, placement method, reconstruction dimension and error metrics for these select dissection cases are summarized in Table 3. In addition, we also estimated the coherency parameter for each of these cases, which are $\mathcal{O}(1)$, indicating that the rows of the measurement matrix are sufficiently incoherent with respect to the POD basis. Careful examination shows that coherency parameters for the QR-pivoting and DEIM are higher than that for random placement, as such smart approaches leverage the underlying physical structure contained in the POD modes.

4.4.3. Sparse Reconstruction with Marginally Oversampled Sensors

We observed that sensor placement is especially critical for marginal oversampling, i.e., $P^* \gtrsim K^*$. To illustrate this, we dissect the instantaneous snapshot reconstruction at $K^* = 5$ and $P^* = 6$ in Figure 15. The left column here shows sensor locations, the middle shows reconstructed fields and the right, POD coefficient estimates. As expected, the SR estimated coefficients (Figure 15c,f,i,l) show that data- and physics-aware sensor placements perform better at reconstruction as compared to random sampling. While not all random sensor placements result in bad reconstruction, we see strong variability in the error metrics across realizations (Figure 12).

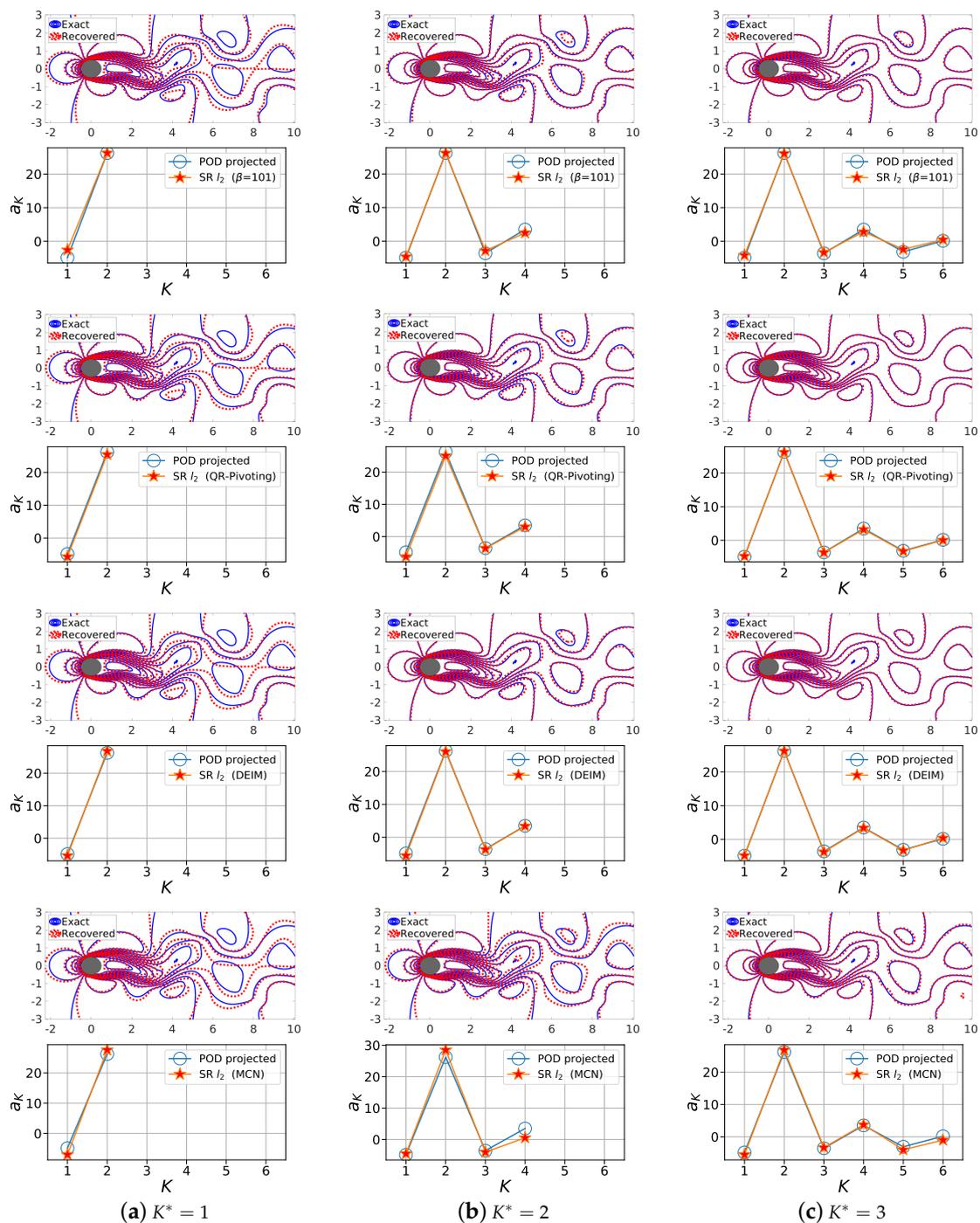


Figure 14. First row (Random $\beta = 101$), third row (QR-Pivot), fifth row (DEIM) and seventh (MCN) row: we show the line contour comparison of streamwise velocity between the actual CFD solution field (blue) and the POD-based SR reconstruction (red) for $Re = 100$ at $P^* = 10$ and (a) $K^* = 1$, (b) $K^* = 2$, (c) $K^* = 3$. Second row (Random $\beta = 101$), fourth row (QR with column pivoting), sixth row (DEIM) and eighth row (MCN) show the corresponding projected (full reconstruction) and sparse recovered coefficients a from the SR algorithm.

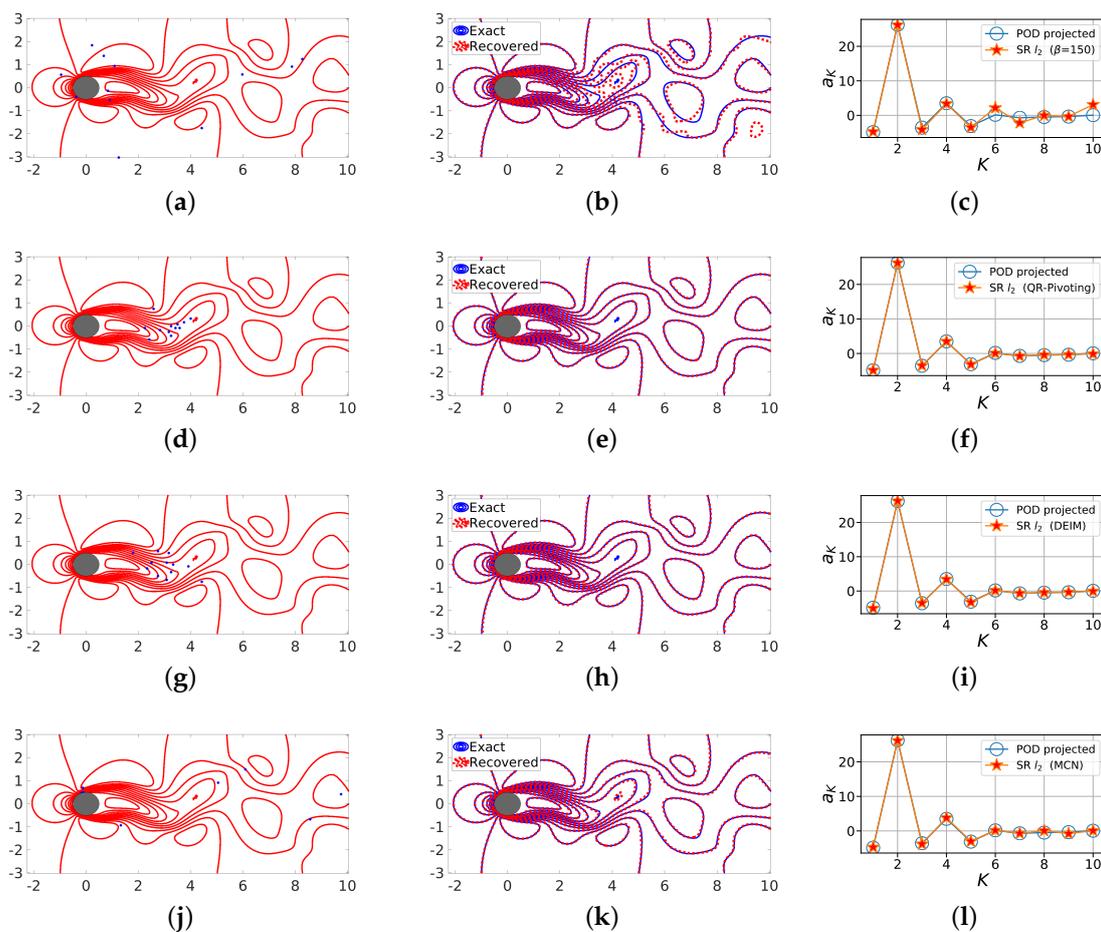


Figure 15. Dissection of instantaneous snapshot reconstruction for a marginally oversampled case ($K^* = 5, P^* = 6$). The figure shows the different sensor locations, namely random sensors with seed $\beta = 150$ (a), QR factorization with column pivoting (b), DEIM (c) and minimum condition number (MCN) (d) along with the corresponding overlaid true and reconstructed solutions (b,e,h,k), and the recovered coefficients a (c,f,i,l) using POD-based SR for $Re = 100$. The corresponding error quantifications for the different cases are as follows. First row: $\epsilon_1 = 2.46 \times 10^{-1}$ and $\epsilon_2 = 8.18$. Second row: $\epsilon_1 = 5.20 \times 10^{-2}$ and $\epsilon_2 = 2.37$. Third row: $\epsilon_1 = 4.44 \times 10^{-2}$ and $\epsilon_2 = 1.47$. Fourth row: $\epsilon_1 = 8.04 \times 10^{-2}$ and $\epsilon_2 = 2.67$.

4.4.4. Sparse Reconstruction with Highly Oversampled Sensors

As shown in Figure 14, oversampling results in reasonable accuracy for all the different sensor placement methods including random sensing which has a high probability of populating the physically significant regions of the flow. However, the exception to this is the MCN approach in the limit of large P relative to K . To highlight the severity of this issue, we considered a single design point, $K^* = 6, P^* = 10$ (Figure 16) where P^*/K^* is smaller than in Figure 14 and for which the sparse recovery should be highly accurate purely from theoretical considerations. Figure 16 shows the sensor locations for each of the different algorithms in the left column, reconstructed fields in the middle and the predicted POD coefficients in the right. We see that the placements using random, DEIM and QR-pivoting produce identically accurate results while MCN sensors generate highly erroneous outcomes due to having only a few (six) sensors in the cylinder wake as compared to a reconstruction dimension (K) of twelve. While very few sensors are sufficient to generate a good reconstruction for this low-dimensional flow, this study highlights the need for designing the SR problem with awareness of the effective sensor locations and not just their quantity.

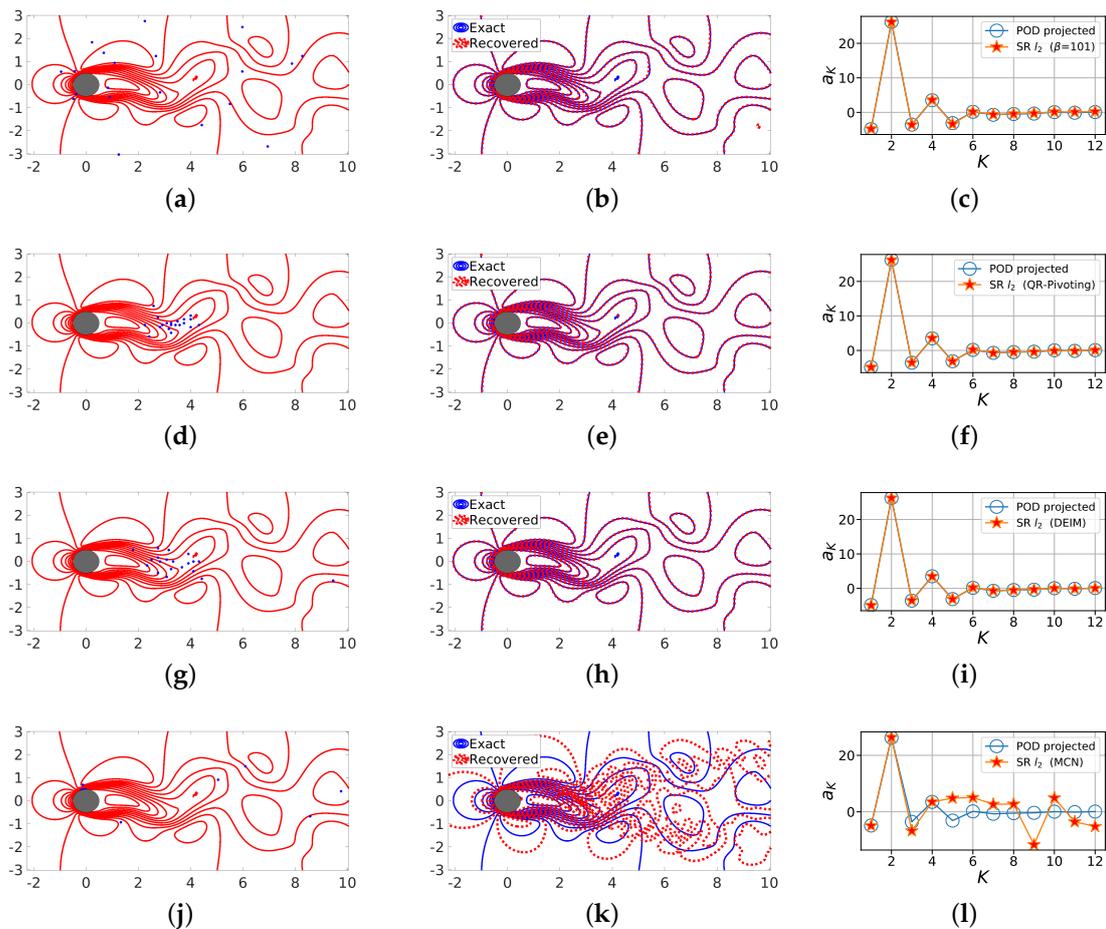


Figure 16. Dissection of instantaneous snapshot reconstruction for a highly oversampled case ($K^* = 6$, $P^* = 10$). The figure shows the different sensor locations, namely random sensors with seed $\beta = 150$ (a), QR factorization with column pivoting (b), DEIM (c) and minimum condition number (MCN) (d) along with the corresponding overlaid true and reconstructed solutions (b,e,h,k), and the recovered coefficients a (c,f,i,l) using POD-based SR for $Re = 100$. The corresponding error quantifications for the above cases are as follows. First row: $\epsilon_1 = 5.73 \times 10^{-2}$ and $\epsilon_2 = 3.43$. Second row: $\epsilon_1 = 3.01 \times 10^{-2}$ and $\epsilon_2 = 1.80$. Third row: $\epsilon_1 = 1.98 \times 10^{-2}$ and $\epsilon_2 = 1.19$. Fourth row: $\epsilon_1 = 9.77 \times 10^{-1}$ and $\epsilon_2 = 58.60$.

The low-dimensional dynamics of the cylinder wake is well understood and consequently an ideal test case for validation and performance characterization. Loiseau et al. [22] observed that the temporal dynamics of the cylinder wake is accurately characterized by amplitude and phase of the POD coefficient time history, which in turn is accurately estimated by a feature set of lift and its time-derivative. Using such domain knowledge, it is straight forward to design appropriate sensor placement. However, to truly demonstrate the effectiveness of the methods described in this work, we considered a more complex system in the form of synoptic scale ocean turbulent flows.

4.5. Sparse Reconstruction of Sea Surface Temperature Data

As before, we performed an ensemble of nearly one hundred SR experiments pertaining to different design choices as was done for the cylinder wake. The resulting error estimates ϵ_1 and ϵ_2 (as described in Section 4.3) across the $K^* - P^*$ space were generated for both random and greedy sensor placements, as shown in Figure 17. For this study, we did not include the explicit condition number minimization (MCN) approach due to its computational complexity for high-dimensional systems. Additionally, for the random sensor placement, we only considered a single realization in this analysis.

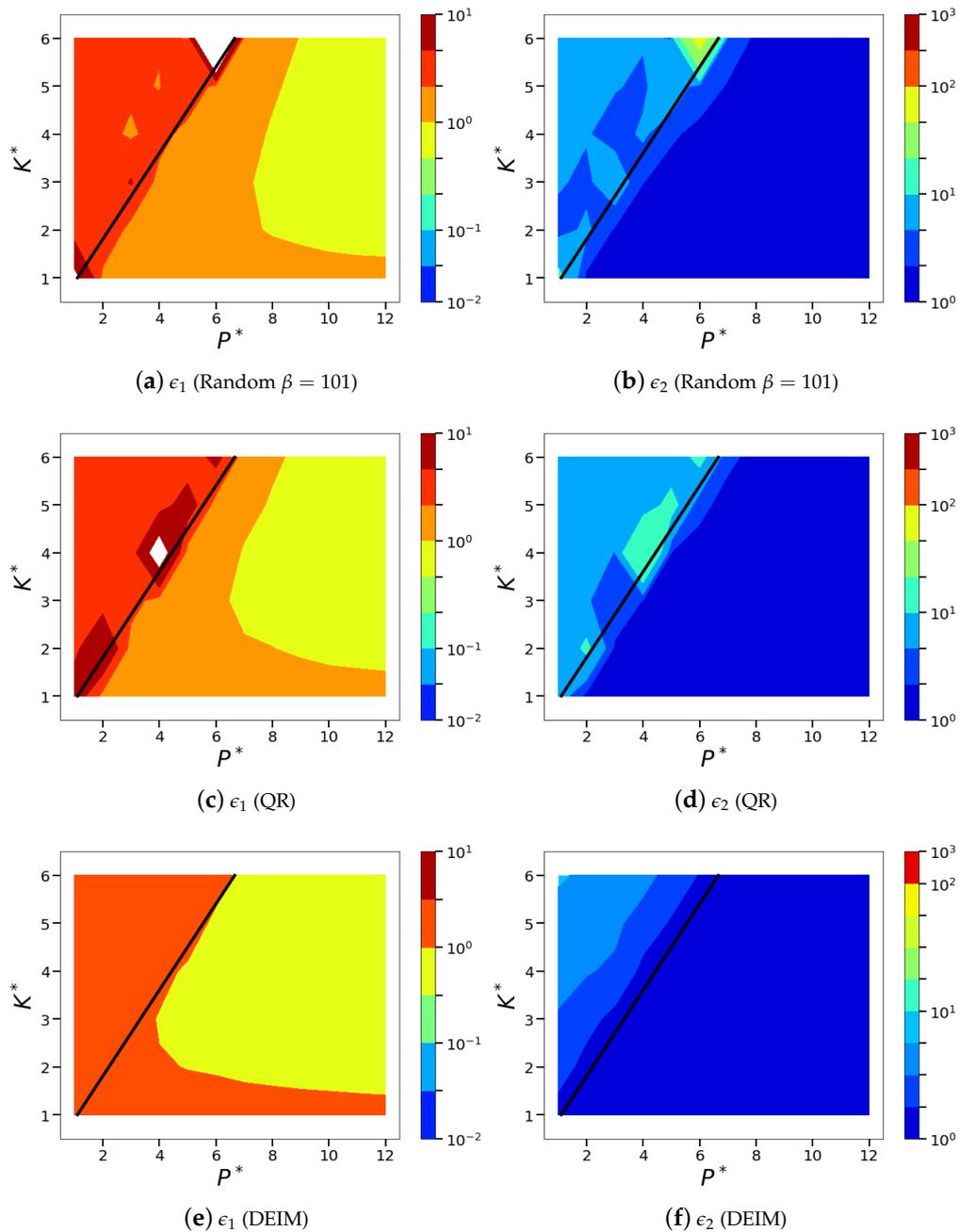


Figure 17. Isocontours of the normalized mean squared POD-based sparse reconstruction errors (l_2 norm) of sea surface temperature data corresponding to the Random (a,b), QR (c,d) and DEIM (e,f) sensor placement methods. Subfigures (a,b,e) show the normalized absolute error metric, ϵ_1 and subfigures (b,d,f) show the normalized relative error metric, ϵ_2 .

Overall, the topology of error metrics across the $P^* - K^*$ design space for the SST data (Figure 17) is similar to that observed for the low-dimensional cylinder wake (Figure 13). In particular, the smaller relative errors (ϵ_2) are located in the oversampled region with $P^* > K^*$. To remind the reader, ϵ_2 is normalized by the error contained in the exact K -dimensional POD reconstruction and represents how effectively the sparse data can approximate the K -dimensional representation of the flow field for the given budget and placement. As expected ϵ_1 contours adhere to a L -shaped structure indicating that absolute normalized error reduces as both P and K increase due to oversampling and increased system representation.

To assess the extent of similarity in the relative errors (ϵ_2) across the different systems, we compared the variation of ϵ_2 with P^* for different reconstruction dimensions K^* in Figure 18. The image to the left corresponds to DEIM sensor placement while the image to the right represents data using QR-pivoting. From these, we note the qualitative and quantitative similarity between the low-dimensional cylinder wake flow (dashed lines) and the more complex SST data errors (solid lines) across the different values of K^* . In all these different curves, the ideal reconstruction error corresponds to $\epsilon_2 = 10^0$, which is achieved only in the asymptotic limit of P^* . However, all the different curves across the different flow systems as well as varying values of K^* begin to asymptote at around $P^* \sim 8-10$ for both the sensor placement methods. However, at smaller values of P^* in the marginally oversampled regime ($P^* \approx K^*$), there exists a strong flow dependence in the error decay.

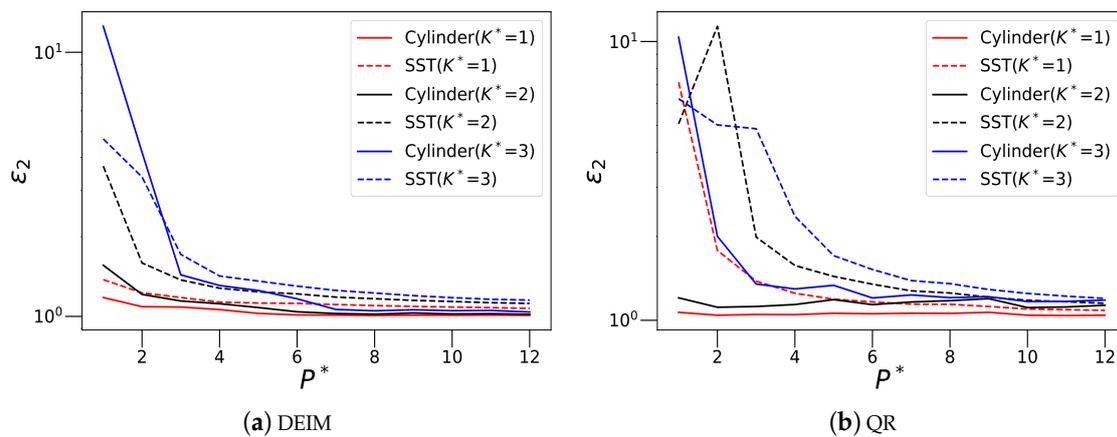


Figure 18. Comparison of relative error (ϵ_2) decrease with increasing sensor budget for both wake and SST data. The figure shows three curves for different values of $K^* = 1, 2, 3$ for both DEIM (a) and QR-pivoting (b) based sensor placement.

Amongst the different sensor placement methods, the best performance is realized for the DEIM which shows a smooth variation as one moves from under-sampled to oversampled regions (Figures 17 and 18). On the other hand, both the random and QR-pivot sensors display peaks corresponding to strong inaccuracy ($\mathcal{O}(10^1)$) in regions of marginal oversampling ($P^* \approx K^*$). This is clearly illustrated in Figure 19 which shows the reconstructed field (left) and the estimated coefficients (right) for the different sensor arrangements in the marginally sampled arrangement. We see that estimated coefficients are inaccurate for both the random and QR-pivoting based placements whereas the DEIM offers improved accuracy. The red dots in the reconstructed field denote the chosen sensor locations and DEIM places a small fraction of them off the pacific coast (top right region in Figure 19e) unlike the QR-pivoting (Figure 19c) and random (Figure 19a) sampling methods. Such shortcomings in the sensor placement is easily overcome by all the different methods in the oversampled regime as expected and is shown in Figure 20. The differences in sparse recovery across the three sensor placement methods are mostly unnoticeable with oversampling although DEIM again provides the best estimates for the coefficients.

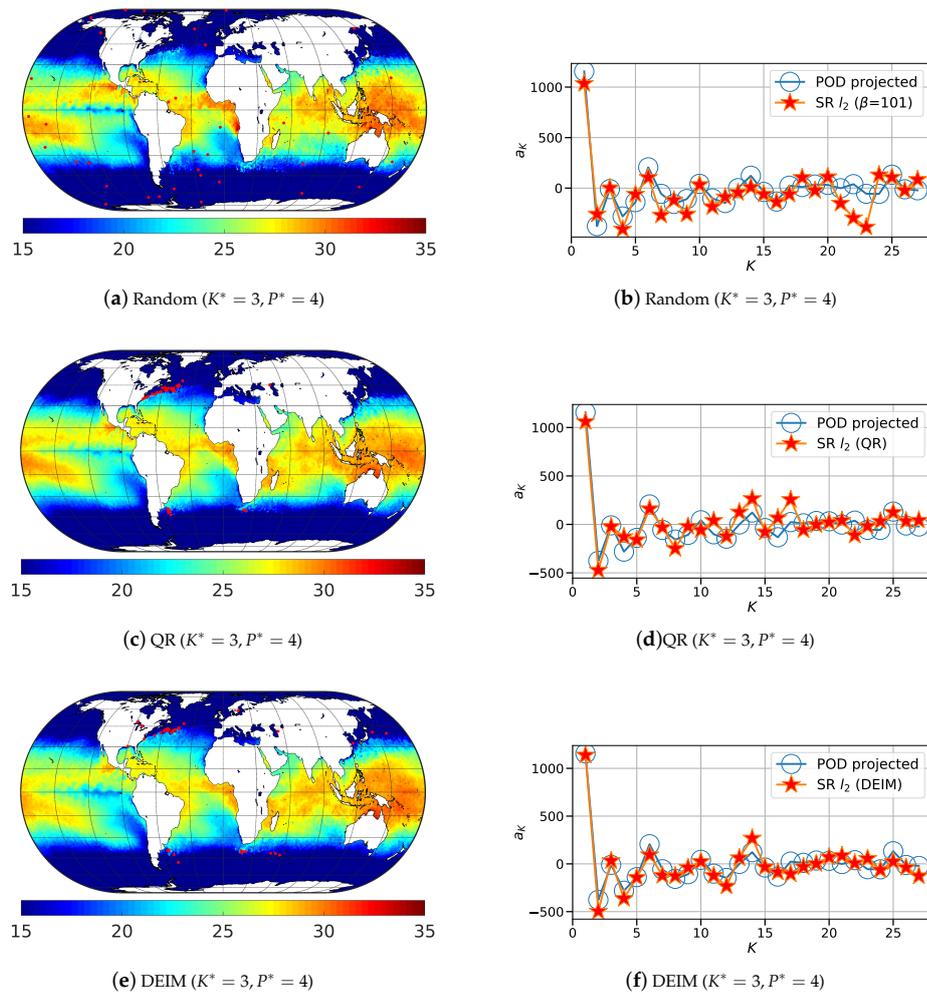


Figure 19. Comparison of the sparse reconstruction using Random, QR and DEIM sensor placement method on instantaneous snapshot for a marginally oversampled case ($K^* = 3, P^* = 4$). The left subfigures show the reconstructed solutions for different sensor placement methods namely, random placement (a), QR with column pivoting (c) and DEIM (e) whereas the corresponding reconstructed coefficients using POD-based SR are shown in subfigures (b,d,f). Red dots on the contour plots represent sensor locations.

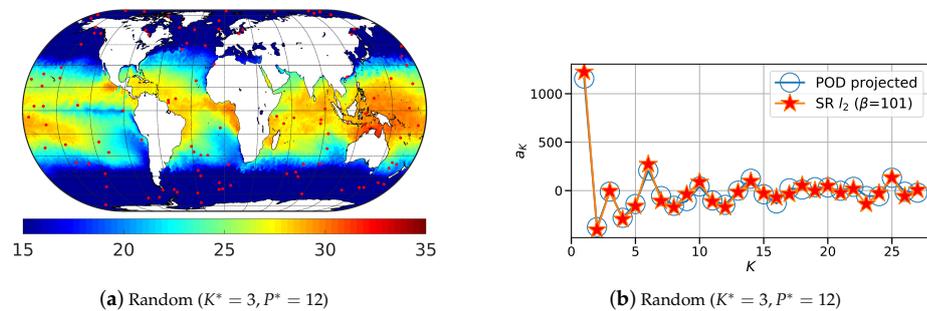


Figure 20. Cont.

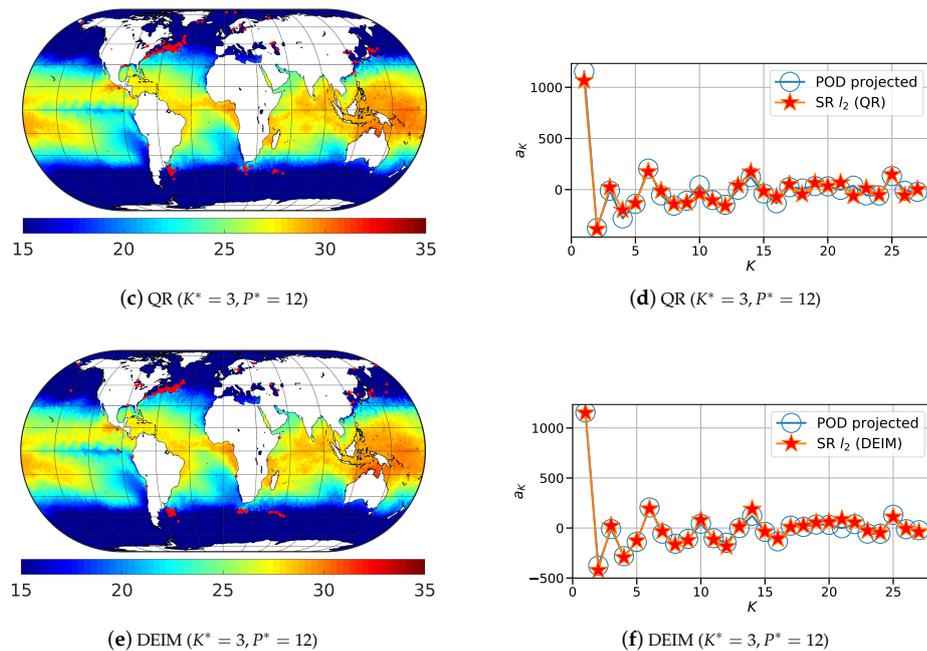


Figure 20. Comparison of the sparse reconstruction using Random, QR and DEIM sensor placement method on instantaneous snapshot for a highly oversampled case ($K^* = 3, P^* = 12$). The left subfigures show the reconstructed solutions for different sensor placement methods namely, random placement (a), QR with column pivoting (c) and DEIM (e) whereas the corresponding reconstructed coefficients using POD-based SR are shown in subfigures (b,d,f). Red dots on the contour plots represent sensor locations.

5. Conclusions

In this work, we systematically assess sparse reconstruction (SR) of fluid flows based on linear estimation principles using a chosen set of basis vectors. To this end, we characterize the accuracy of POD-based linear estimation of the full state using system normalized error metrics and basis/sensor budget to shed light on the interplay between design parameters (i.e., sensor quantity, their placement and system dimension). We demonstrate the outcomes for both low-dimensional wake flows as well as a high-dimensional sea surface temperature data generated from global ocean models.

Given that the POD basis is ordered in terms of variance capture and their ability to approximate the data is known a priori, we bypass more expensive l_1 norm minimization with l_2 approaches of complexity $\mathcal{O}(NKM)$. We verify the validity of this by explicitly demonstrating that l_1 methods using optimal matching pursuit and l_2 minimization (with the first K basis) using least squares yield the same K -sparse reconstruction from very few random measurements. In addition, we also observe from systematic analysis of error metrics over a carefully designed parameter ($P^* - K^*$) space that even marginal oversampling, i.e., $P \gtrsim K$, is sufficient for reasonably accurate full state recovery using l_2 SR with high probability akin to the requirement for l_0 minimization. We reiterate here that this study does not advocate the use l_2 over sparsity promoting l_1 methods, while acknowledging that the latter is the sensible choice in many practical situations where knowledge of the underlying system is scarce. Further, the SR accuracy from oversampling achieves its maximum possible limit only in the asymptotic sense and the rate of convergence is dependent on the choice of sensor placement and the system dimension. We further expand the $P - K$ design space to include the effect of data-driven sensor placement with the following candidates: random sensing and greedy-smart sensing algorithms such as DEIM, QR with column pivoting and explicit condition number minimization or MCN. We observe that, while random sampling shows highly variable errors for marginal oversampling, greedy-smart

sensor placement show improved recovery under these conditions. However, the best performance is realized for the DEIM-based sensor placement for which the error convergence to asymptotic behavior is rapid and systematic as against QR-pivoting, which displays error hot spots in regions of marginal oversampling.

In the limit of heavy oversampling, the computationally intensive MCN method produces diminishing returns, as seen for the low-dimensional wake flow due to its inability to place sufficient sensors in dynamically relevant regions of the flow. More research is necessary to delineate the causes for this behavior. Considering the computational complexity of data-driven sensor placement and the accuracy of sparse reconstruction, DEIM and QR factorization with column pivoting (in that order) turn out to be the best alternatives to random sampling.

Author Contributions: B.J. designed the study. C.L. and S.M.A.A.M. developed the codes with input from B.J. C.L. and S.M.A.A.M. carried out the analysis with input from B.J. C.L. and S.M.A.A.M. developed a partial draft of the manuscript. B.J. produced the final written manuscript.

Funding: This research was funded by NASA EPSCoR (Oklahoma), Baker Hughes General Electric Company and Oklahoma State University.

Acknowledgments: We acknowledge computational resources from OSU High Performance Computing.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

Matrices

\tilde{C}_M	Two-point correlation matrix
Φ	Low dimensional basis matrix
Θ	Linear map between basis coefficients and sparse measurements
a	Matrix of POD coefficients
C	Measurement matrix
Q	Unitary matrix from QR factorization
R	Upper triangular matrix from QR factorization
Λ	Eigenvalues of correlation matrix
I	Identity matrix
M	$\Theta\Theta^T$ or $\Theta^T\Theta$
V	Eigenvectors of correlation matrix

Mathematical Operators

$+$	Pseudo-inverse
∇	Laplacian operator
\prod	Multiplication operator

Scalars

α	Regularization parameter
β	Random seed
ϵ_1	Normalized absolute error
ϵ_2	Normalized relative error
κ	Condition number
λ	Singular value of correlation tensor
\mathbb{E}_K	Cumulative energy fraction
\mathcal{E}	l_2 error
μ	Coherency number
ν	Kinematic viscosity
J_{cost}	Cost function
K	Reconstruction dimension
K^*	Number of modes normalized by K_{95}
K_{95}	Number of modes needed for 95% energy capture
K_{99}	Number of modes needed 99% energy capture

M	Snapshot dimension
N	Full state dimension
N_b	Candidate basis dimension
P	Number of sparse measurements
P^*	Sensor budget normalized by K_{95}
Re	Reynolds number
T	Non-dimensional time
t	Time

Vectors

ϕ	Basis vector
\tilde{x}	Compressed data vector
a	POD coefficient vector
m	Mask vector
u	x Component of velocity
v	y Component of velocity
w	z Component of velocity
x	Data snapshot vector

References

- Friedman, J.; Hastie, T.; Tibshirani, R. *The Elements of Statistical Learning*; Springer Series in Statistics New York; Springer Publishing: New York City, NY, USA, 2001; Volume 1.
- Holmes, P. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*; Cambridge University Press: Cambridge, UK, 2012.
- Berkooz, G.; Holmes, P.; Lumley, J.L. The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid Mech.* **1993**, *25*, 539–575. [[CrossRef](#)]
- Taira, K.; Brunton, S.L.; Dawson, S.; Rowley, C.W.; Colonius, T.; McKeon, B.J.; Schmidt, O.T.; Gordeyev, S.; Theofilis, V.; Ukeiley, L.S. Modal analysis of fluid flows: An overview. *AIAA J.* **2017**, *55*, 4013–4041. [[CrossRef](#)]
- Jayaraman, B.; Lu, C.; Whitman, J.; Chowdhary, G. Sparse Convolution-based Markov Models for Nonlinear Fluid Flows. *arXiv* **2018**, arxiv:1803.08222b.
- Bai, Z.; Wimalajeewa, T.; Berger, Z.; Wang, G.; Glauser, M.; Varshney, P.K. Low-dimensional approach for reconstruction of airfoil data via compressive sensing. *AIAA J.* **2014**. [[CrossRef](#)]
- Bright, I.; Lin, G.; Kutz, J.N. Compressive sensing based machine learning strategy for characterizing the flow around a cylinder with limited pressure measurements. *Phys. Fluids* **2013**, *25*, 127102. [[CrossRef](#)]
- Candès, E.J. Compressive sampling. In Proceedings of the International Congress of Mathematicians, Madrid, Spain, 22–30 August 2006; Volume 3, pp. 1433–1452.
- Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666. [[CrossRef](#)]
- Candès, E.J.; Wakin, M.B. An introduction to compressive sampling. *IEEE Signal Proc. Mag.* **2008**, *25*, 21–30. [[CrossRef](#)]
- Needell, D.; Tropp, J.A. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmonic Anal.* **2009**, *26*, 301–321. [[CrossRef](#)]
- Bui-Thanh, T.; Damodaran, M.; Willcox, K. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA J.* **2004**, *42*, 1505–1516. [[CrossRef](#)]
- Willcox, K. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Comput. Fluids* **2006**, *35*, 208–226. [[CrossRef](#)]
- Venturi, D.; Karniadakis, G.E. Gappy data and reconstruction procedures for flow past a cylinder. *J. Fluid Mech.* **2004**, *519*, 315–336. [[CrossRef](#)]
- Gunes, H.; Sirisup, S.; Karniadakis, G.E. Gappy data: To Krig or not to Krig? *J. Comput. Phys.* **2006**, *212*, 358–382. [[CrossRef](#)]
- Gunes, H.; Rist, U. On the use of kriging for enhanced data reconstruction in a separated transitional flat-plate boundary layer. *Phys. Fluids* **2008**, *20*, 104109. [[CrossRef](#)]
- Everson, R.; Sirovich, L. Karhunen–Loeve procedure for gappy data. *JOSA A* **1995**, *12*, 1657–1664. [[CrossRef](#)]

18. Chaturantabut, S.; Sorensen, D.C. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* **2010**, *32*, 2737–2764. [[CrossRef](#)]
19. Dimitriu, G.; Ștefănescu, R.; Navon, I.M. Comparative numerical analysis using reduced-order modeling strategies for nonlinear large-scale systems. *J. Comput. Appl. Math.* **2017**, *310*, 32–43. [[CrossRef](#)]
20. Zimmermann, R.; Willcox, K. An accelerated greedy missing point estimation procedure. *SIAM J. Sci. Comput.* **2016**, *38*, A2827–A2850. [[CrossRef](#)]
21. Erichson, N.B.; Mathelin, L.; Yao, Z.; Brunton, S.L.; Mahoney, M.W.; Kutz, J.N. Shallow Learning for Fluid Flow Reconstruction with Limited Sensors and Limited Data. *arXiv* **2019**, arxiv:1902.07358.
22. Loiseau, J.C.; Noack, B.R.; Brunton, S.L. Sparse reduced-order modelling: Sensor-based dynamics to full-state estimation. *J. Fluid Mech.* **2018**, *844*, 459–490. [[CrossRef](#)]
23. Mathelin, L.; Kasper, K.; Abou-Kandil, H. Observable dictionary learning for high-dimensional statistical inference. *Arch. Comput. Methods Eng.* **2018**, *25*, 103–120. [[CrossRef](#)]
24. Saini, P.; Arndt, C.M.; Steinberg, A.M. Development and evaluation of gappy-POD as a data reconstruction technique for noisy PIV measurements in gas turbine combustors. *Exp. Fluids* **2016**, *57*, 1–15. [[CrossRef](#)]
25. Schmid, P.J. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **2010**, *656*, 5–28. [[CrossRef](#)]
26. Rowley, C.W.; Dawson, S.T. Model reduction for flow analysis and control. *Annu. Rev. Fluid Mech.* **2017**, *49*, 387–417. [[CrossRef](#)]
27. Mallet, S. *A Wavelet Tour of Signal Processing*; Academic Press: Cambridge, UK, 1998.
28. Brunton, S.L.; Proctor, J.L.; Tu, J.H.; Kutz, J.N. Compressed sensing and dynamic mode decomposition. *J. Comput. Dyn.* **2015**, *2*, 165–191.
29. Brunton, S.L.; Tu, J.H.; Bright, I.; Kutz, J.N. Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. *SIAM J. Appl. Dyn. Syst.* **2014**, *13*, 1716–1732. [[CrossRef](#)]
30. Bai, Z.; Brunton, S.L.; Brunton, B.W.; Kutz, J.N.; Kaiser, E.; Spohn, A.; Noack, B.R. Data-driven methods in fluid dynamics: Sparse classification from experimental data. In *Whither Turbulence and Big Data in the 21st Century?*; Springer: Berlin, Germany, 2017; pp. 323–342.
31. Kramer, B.; Grover, P.; Boufounos, P.; Nabi, S.; Benosman, M. Sparse sensing and DMD-based identification of flow regimes and bifurcations in complex flows. *SIAM J. Appl. Dyn. Syst.* **2017**, *16*, 1164–1196. [[CrossRef](#)]
32. Al Mamun, S.; Lu, C.; Jayaraman, B. Extreme Learning Machines as Encoders for Sparse Reconstruction. *Fluids* **2018**, *3*, 88. [[CrossRef](#)]
33. Trefethen, L.N.; Bau, D., III. *Numerical Linear Algebra*; SIAM: Bangkok, Thailand, 1997; Volume 50.
34. 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.* **2018**, *38*, 63–86.
35. Tarantola, A. *Inverse Problem Theory and Methods for Model Parameter Estimation*; SIAM: Bangkok, Thailand, 2005; Volume 89.
36. Arridge, S.R.; Schotland, J.C. Optical tomography: Forward and inverse problems. *Inverse Probl.* **2009**, *25*, 123010. [[CrossRef](#)]
37. Tarantola, A.; Valette, B. Generalized nonlinear inverse problems solved using the least squares criterion. *Rev. Geophys.* **1982**, *20*, 219–232. [[CrossRef](#)]
38. Neelamani, R. Inverse Problems in Image Processing. Ph.D. Thesis, Rice University, Houston, TX, USA, 2004.
39. Khemka, A. Inverse Problems in Image Processing. Ph.D. Thesis, Purdue University, West Lafayette, Indiana, 2009.
40. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [[CrossRef](#)]
41. Baraniuk, R.G. Compressive sensing [lecture notes]. *IEEE Signal Proc. Mag.* **2007**, *24*, 118–121. [[CrossRef](#)]
42. Baraniuk, R.G.; Cevher, V.; Duarte, M.F.; Hegde, C. Model-based compressive sensing. *IEEE Trans. Inf. Theory* **2010**, *56*, 1982–2001. [[CrossRef](#)]
43. Sarvotham, S.; Baron, D.; Wakin, M.; Duarte, M.F.; Baraniuk, R.G. Distributed compressed sensing of jointly sparse signals. In Proceedings of the Asilomar conference on signals, systems, and computers, Pacific Grove, CA, USA, 30 October–2 November 2005; pp. 1537–1541.
44. Chen, S.S.; Donoho, D.L.; Saunders, M.A. Atomic decomposition by basis pursuit. *SIAM Rev.* **2001**, *43*, 129–159. [[CrossRef](#)]

45. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **1996**, *58*, 267–288. [[CrossRef](#)]
46. Candès, E.J.; Wakin, M.B.; Boyd, S.P. Enhancing sparsity by reweighted l_1 minimization. *J. Fourier Anal. Appl.* **2008**, *14*, 877–905. [[CrossRef](#)]
47. Candès, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [[CrossRef](#)]
48. Kim, S.J.; Koh, K.; Lustig, M.; Boyd, S.; Gorinevsky, D. An Interior-Point Method for Large-Scale L1 Regularized Least Squares. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 606–617. [[CrossRef](#)]
49. Brunton, S.L.; Proctor, J.L.; Kutz, J.N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* **2016**, 201517384. [[CrossRef](#)]
50. Lumley, J. *Stochastic Tools in Turbulence*; Academic: New York, NY, USA, 1970.
51. Sirovich, L. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Q. Appl. Math.* **1987**, *45*, 561–571. [[CrossRef](#)]
52. Brunton, S.L.; Rowley, C.W.; Williams, D.R. Reduced-order unsteady aerodynamic models at low Reynolds numbers. *J. Fluid Mech.* **2013**, *724*, 203–233. [[CrossRef](#)]
53. Candès, E.; Romberg, J. Sparsity and incoherence in compressive sampling. *Inverse Probl.* **2007**, *23*, 969. [[CrossRef](#)]
54. Csató, L.; Opper, M. Sparse on-line Gaussian processes. *Neural Comput.* **2002**, *14*, 641–668. [[CrossRef](#)]
55. Cohen, K.; Siegel, S.; McLaughlin, T. Sensor placement based on proper orthogonal decomposition modeling of a cylinder wake. In Proceedings of the 33rd AIAA Fluid Dynamics Conference and Exhibit, Orlando, FL, USA, 23–26 June 2003; p. 4259.
56. Kubrusly, C.S.; Malebranche, H. Sensors and controllers location in distributed systems—A survey. *Automatica* **1985**, *21*, 117–128. [[CrossRef](#)]
57. Chaturantabut, S.; Sorensen, D.C. A state space error estimate for POD-DEIM nonlinear model reduction. *SIAM J. Numer. Anal.* **2012**, *50*, 46–63. [[CrossRef](#)]
58. Yildirim, B.; Chrysostomidis, C.; Karniadakis, G. Efficient sensor placement for ocean measurements using low-dimensional concepts. *Ocean Model.* **2009**, *27*, 160–173. [[CrossRef](#)]
59. Hanagud, S.; de Noyer, M.B.; Luo, H.; Henderson, D.; Nagaraja, K. Tail buffet alleviation of high-performance twin-tail aircraft using piezostack actuators. *AIAA J.* **2002**, *40*, 619–627. [[CrossRef](#)]
60. Barrault, M.; Maday, Y.; Nguyen, N.C.; Patera, A.T. An empirical interpolation method: Application to efficient reduced-basis discretization of partial differential equations. *Compt. Rendus Math.* **2004**, *339*, 667–672. [[CrossRef](#)]
61. Bui-Thanh, T.; Damodaran, M.; Willcox, K. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In Proceedings of the 21st AIAA Applied Aerodynamics Conference, Orlando, FL, USA, 23–26 June 2003; p. 4213.
62. Roshko, A. *On the Development of Turbulent Wakes from Vortex Streets*; NACA-TR-1191; NASA: Washington, D.C., USA, 1954.
63. Williamson, C. Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low Reynolds numbers. *J. Fluid Mech.* **1989**, *206*, 579–627. [[CrossRef](#)]
64. Noack, B.R.; Afanasiev, K.; Morzynski, M.; Tadmor, G.; Thiele, F. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.* **2003**, *497*, 335–363. [[CrossRef](#)]
65. Cantwell, C.D.; Moxey, D.; Comerford, A.; Bolis, A.; Rocco, G.; Mengaldo, G.; De Grazia, D.; Yakovlev, S.; Lombard, J.E.; Ekelschot, D.; et al. Nektar++: An open-source spectral/hp element framework. *Comput. Phys. Commun.* **2015**, *192*, 205–219. [[CrossRef](#)]

