Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

Article

Extreme Learning Machines as Encoders for Sparse Reconstruction

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

[1] School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK, USA

* Author to whom correspondence should be addressed; e-mail, telephone and fax number of the corresponding author.

Abstract: Reconstruction of fine-scale information from sparse data is often needed in practical fluid dynamics where the sensors are typically sparse and yet, one may need to learn the underlying flow structures or inform predictions through assimilation into data-driven models. Given that sparse reconstruction is inherently an ill-posed problem, the most successful approaches encode the physics into an underlying sparse basis space that spans the manifold to generate well-posedness. To achieve this, one commonly uses generic orthogonal Fourier basis or data specific proper orthogonal decomposition (POD) basis to reconstruct from sparse sensor information at chosen locations. Such a reconstruction problem is well-posed as long as the sensor locations are incoherent and can sample the key physical mechanisms. The resulting inverse problem is easily solved using l_2 minimization or if necessary, sparsity promoting l_1 minimization. Given the proliferation of machine learning and the need for robust reconstruction frameworks in the face of dynamically evolving flows, we explore in this study the suitability of non-orthogonal basis obtained from Extreme Learning Machine (ELM) auto-encoders for sparse reconstruction. In particular, we assess the interplay between sensor quantity and sensor placement for a given system dimension for accurate reconstruction of canonical fluid flows in comparison to POD-based reconstruction.

Keywords: sparse reconstruction, extreme learning machines, sensors, SVD, POD, compressive sensing

1. Introduction

Multiscale fluid flow phenomena are ubiquitous in engineering and geophysical settings. Depending on the situation, one encounters either a data-sparse or a data-rich problem. In the data-sparse cases, the

© 2018 by the author(s). Distributed under a Creative Commons CC BY license.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

goal is to recover more information about the dynamical system while in the data-surplus case, the goal is to reduce the information into a simpler form for analysis or to build evolutionary models for prediction and then recover the full state. Thus, both these situations require reconstruction of the full state. To expand on this view, for 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 represents the absolute truth and is often acquired from very few probes that limits potential for in-depth analysis. A common recourse is to combine such sparse measurement 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). In the absence of such a mechanism, the only method to identify structure information of the flow is to use phenomenology such as Taylor's frozen eddy hypothesis. 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 spatio-temporal coherence and scale content due to the high density of data probes in the form of computational grid points. With growth in computing power, they often generate *big data* contributing to an ever growing demand for quick analytics and machine learning tools [1] to both sparsify, i.e. dimensionality reduction [2–5] and reconstruct the data without loss of information. Thus, tools for *encoding* information into a low-dimensional feature space (convolution) complement sparse reconstruction tools that help *decode* compressed information (deconvolution). This in essence is a key aspect of leveraging machine learning for fluid flow analysis [6,7]. The other aspects of machine learning-driven study of fluid flows include building data-driven predictive models [5,8,9], pattern detection, and classification. This work broadly contributes to the *decoding* problem of reconstructing high resolution fields data in both data-sparse and data-rich environments.

A primary target of this work is to address the practical problems of flow sensing and control in the field or a laboratory where a few affordable probes are expected the to sense effectively. Advances in compressive sensing (CS) [10–13] have opened the possibility of direct compressive sampling [6]. Thus, sparse data-driven decoding and reconstruction ideas have been gaining popularity in their various manifestations such as Gappy Proper Orthogonal Decomposition (GPOD) [14,15], Fourier-based Compressive Sensing (CS) [10–13] and Gaussian kernel-based Kriging [16–18]. The overwhelming corpus of literature on this topic focus on theoretical expositions of the framework and demonstrations of performance. The novelty of this work is two-fold. Firstly, we combine sparse reconstruction principles with machine learning ideas for learning data-driven encoders/decoders using Extreme Learning Machines (ELMs) [19–23], a close cousin of the shallow artificial neural network architecture. Secondly, we explore the performance characteristics of such methods for nonlinear fluid flows withing the parametric space of system dimensionality, sensor quantity and to a limited extent, their placement.

Sparse reconstruction is inherently ill-posed and underdetermined inverse problem where the number of constraints (i.e., sensor quantity) are much less than the number of unknowns (i.e., high resolution field). However, if the underlying system is sparse in a feature space then the probability of recovering a unique solution increases by solving the reconstruction problem in a lower-dimensional space. The

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

core theoretical develoments of such ideas and their first practical applications happened in the realm of image compression and restoration [12,24]. Data reconstruction techniques based on Karhunen-Loeve (K-L) procedure with l_2 minimization, also known as GPOD [14,15,25], was originally developed in the nineties to recover marred faces [25] in images. The fundamental idea is to utilize the POD basis computed offline from the data ensemble to *encode* the reconstruction problem into a low-dimensional feature space. This way, the sparse data can be used to recover the sparse unknowns in the feature space (i.e., sparse POD coefficients) by minimizing the l_2 errors. If the POD basis are not known *a priori*, an iterative formulation [14,25] to successively approximate the POD basis and the coefficients was proposed. While this approach has been shown to work in principle [14,16,26], it is prone to numerical instabilities and inefficiency. Advancements in the form of a progressive iterative reconstruction framework [16] are effective, but impractical for real-time application. A major issue with POD-basis is that they are data-driven and hence cannot be generalized, but are optimally sparse for the given data. This requires that they be generated offline and be used for efficient online sparse reconstruction using little sensor data. However, if training data are unavailable or if the prediction regime is not spanned by the precomputed basis, then the econstruction becomes *untenable*.

A way to overcome the above limitations is to use generic basis such as wavelets [27] or Fourier-based kernels. Such choices are based on the assumption that most systems are sparse in the feature spaces. This is particularly true for image processing applications but may not be optimal for fluid flows whose dynamics obey the PDEs. While avoiding the cost of computing the basis offline, such approaches run into sparsity issues as the basis do not optimally encode the underlying dynamical system. Thus, once again the reconstruction problem can be ill-posed even when solving in the feature space because the number of sensors could be smaller than the system dimensionality for the choice of basis. l_2 minimization produces a solution with sparsity mathcing the dimensionality of the feature space, thus requiring sensor quantity exceeding the system dimensionality. The magic of Compressive Sensing (CS) [10–13] is in its ability to overcome this constraint by seeking a solution that can be less sparse than the dimensionality of the feature space using l_1-minimized norm reconstruction. Such methods have been successfully applied in image processing using Fourier or wavelet basis and also to fundamental fluid flows [6,7,9,28–30]. Compressive sensing essentially looks for a sparse solution through the l_1 norm minimization of the sparse coefficients by solving a convex optimization problem that is computationally tractable and thereby, avoid the tendency of l_2-based methods to overfit the data. In recent years, compressive sensing-type l_1 reconstruction using POD basis has been employed successfully for reconstruction of sparse PIV data [6] and pressure measurements around a cylinder surface [7]. Since the POD basis are data-driven, they represent an optimal basis for reconstruction and require the least quantity of sensor measurements for a given reconstruction quality. However, the downside is the requirement of highly sampled training data as a one-time cost to build a library of POD bases. Such a framework has been attempted in [7] where POD modes from simulations over a range of Reynolds (Re) numbers of a cylinder wake flow were used to populate a library of bases and then used to classify the flow regime based on sparse measurements. In order to reduce this cost, one could also downsample the measurement data and learn the POD bases as per [6]. Recent efforts also combine CS with data-driven predictive ML tools such as Dynamic Mode Decomposition (DMD) [31,32] to identify

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

flow characteristics and classify into different stability regimes [30]. In the above, SR is embedded into the analysis framework for extracting relevant dynamical information.

Both SR and CS can be viewed as generalizations of sparse regression in a higher-dimensional basis space. This way, one can relate SR to other statistical estimation methods such as Kriging. Here, the data is represented as a realization of a random process that is stationary in the first and second moments. This allows one to interpolate information from the known to unknown data locations by employing a kernel (Commonly Gaussian) in the form of a variogram model and the weights are learned under conditions of zero bias and minimal variance. The use of kriging to recover flow-field information from sparse PIV data has been reported [16–18] with encouraging results.

The underlying concept in all the above described techniques is that they solve the reconstruction inverse problem in a feature or basis space where the number of unknowns are comparable to the number of constraints (sparse sensors). This mapping is done through a convolution or projection operator that can be constructed from data or kernel functions. Hence the reason we refer to this class of methods as sparse basis reconstruction (SR) in the same vein as sparse convolution-based Markov models [5,9,33,34]. This requires existence of an optimal sparse basis space in which the physics can be represented. This exists in the form of wavelets and Fourier functions for many common applications in image and signal processing, but may not be optimally sparse for fluid flow solutions to PDEs. Hence, the reason data-driven basis such as POD/PCA [3,4] are popular. Further, since they are optimally sparse, such methods can reconstruct with very little amount of data as compared to say, Kriging that employ generic Gaussian kernels. In this article, we introduce for the first time, the use of ELM-based encoders as basis for sparse reconstruction. As part of an earlier effort [35], we explored how sensor quantity, placement and the system dimensionality impact the accuracy of the POD-based sparse reconstructed field. In this effort, we extend this analysis for two different choices of data-driven sparse basis including POD and ELM. In particular, we aim to accomplish the following: (i) explore whether the relationship between system sparsity and sensor quantity for accurate reconstruction of the fluid flow is independent of the basis employed and (ii) understand the relative influence of sensor placement on the different choice of SR basis.

The rest of manuscript is organized as follows. In section 2 we review the basics of sparse reconstruction theory, different choices of data-driven basis including POD and ELM, role of measurement locations and summarize the algorithm employed for SR. In section 3 we discuss how the training data is generated. In section 4 we discuss the results from our analysis of the SR of the cylinder wake flow using both POD and ELM basis. This is followed by a summary of major conclusions from this study in section 5.

2. The Sparse Reconstruction Problem

Given a high resolution data representing the state of the flow system at any given instant denoted by $\mathbf{x} \in \mathbb{R}^N$, its corresponding sparse representation given by $\tilde{\mathbf{x}} \in \mathbb{R}^P$ with $P \ll N$. Then, the sparse reconstruction problem is to recover $\mathbf{x}$, when given $\tilde{\mathbf{x}}$ along with information of the sensor locations in the form the measurement matrix C as shown in eqn. (1). The measurement matrix C determines how

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

the sparse data $\tilde{\mathbf{x}}$ is collected from $\mathbf{x}$. Variables P and N are the number of sparse measurements and the dimension of the high resolution field, respectively.

$$\tilde{\mathbf{x}} = C\mathbf{x}. \tag{1}$$

In this article, we focus on vectors $\mathbf{x}$ that have a sparse representation in a basis space $\mathbf{\Phi} \in \mathbb{R}^{N \times K}$ such that $K \ll N$ and yielding $\mathbf{x} = \mathbf{\Phi}a$. 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., more unknowns than equations in eqn. (1). Thus the most straightforward approach to recover the solution $\mathbf{x}$ by computing the inverse of C as shown in eqn. (2) is *not possible* as computing the inverse is equivalent to solving an under-determined system of equations.

$$C^{+}\tilde{\mathbf{x}} = \mathbf{x}. \tag{2}$$

2.1. Sparse Reconstruction Theory

Sparse reconstruction theory has strong connections to the field of inverse problems and hence finds mention directly or indirectly in diverse fields of study such as a geophysics [36,37], image processing [38,39] and broadly speaking, inverse problems [40]. In this section, we formulate the reconstruction problem which has been presented in CS literature [10,12,41–43]. Many signals tend to be "compressible", i.e., they are sparse in some K-sparse basis $\mathbf{\Phi}$ as show below:

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

where $\mathbf{\Phi} \in \mathbb{R}^{N \times N_b}$ and $a \in \mathbb{R}^{N_b}$ with K non-zero elements. In the sparse reconstruction formulation above, $\mathbf{\Phi} \in \mathbb{R}^{N \times N_b}$ is used instead of $\mathbf{\Phi} \in \mathbb{R}^{N \times K}$ as the sparsity of the system K is not known *a priori*. Consequently, a more exhaustive basis set of dimension $N_b \approx P > K$ is typically employed. To represent N-dimensional data, one can atmost use N basis vectors, i.e., $N_b \leq N$. In practice, the number of possible basis need not be N and can be represented by $N_b \ll N$ as only K of them are needed to represent the acquired signal up to a desired quality. This is typically the case when $\mathbf{\Phi}$ is composed of optimal data-driven basis vectors such as POD modes. The reconstruction problem is then recast as identification of these K coefficients. In many practical situations, the knowledge of Φ and K is not known *a priori* and N_b, N are typically user input. Standard *transform coding* [27] practice in image compression involves collecting a high resolution sample, transforming it to a Fourier or wavelet basis space where the data is sparse and retain the K-sparse structure while discarding the rest of the information. This is the basis of JPEG and JPEG-2000 compression standards [27]. The *sample and then compress* mechanism still requires acquisition of high resolution samples and processing them before reducing the dimensionality. This is highly challenging as handling large amounts of data is difficult in practice due to demands on processing power, storage, and time. Compressive sensing [10,12,41–43] focuses on direct sparse sensing based inference of the K-sparse coefficients by essentially combining the steps in equations 1 and 3 as below:

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

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

where $\mathbf{\Theta} \in \mathbb{R}^{P \times N_b}$ is the map between the basis coefficients a that represent the data in a feature space and the sparse measurements, $\tilde{\mathbf{x}}$ in physical space. The challenge in solving for $\mathbf{x}$ using the underdetermined eqn. (1) is that C is ill-conditioned and $\mathbf{x}$ in itself is not sparse. However, when $\mathbf{x}$ is sparse in $\mathbf{\Phi}$, the reconstruction using $\mathbf{\Theta}$ in eqn. (4) becomes practically feasible by solving for a that is K-sparse. Thus, one effectively solves for K unknowns using P constraints and this is typically achieved by computing a sparse solution a as per eqn. (7) by minimizing the corresponding s-norm. $\mathbf{x}$ is then recovered from eqn. (3). s chosen as 2 represents the l_2 norm reconstruction of $\mathbf{x}$ and gives the a with least energy. The l_2-based method can be solved by a minimization problem as shown in eqn. (5).

$$\min ||\tilde{\mathbf{x}} - \mathbf{\Theta} a||_2^2 \, . \tag{5}$$

Using left pesudo-inverse of $\mathbf{\Theta}$, eqn. (5) becomes:

$$a = (\mathbf{\Theta})^{+} \tilde{\mathbf{x}}, \tag{6}$$

where $\mathbf{\Theta}^{+}$ can be approximated as a solution to the normal equation as $\left(\mathbf{\Theta}^T \mathbf{\Theta}\right)^{-1} \mathbf{\Theta}^T \tilde{\mathbf{x}}$. This least-squares solution procedure is nearly identical to the original GPOD algorithm developed by Everson and Sirovich [25] if $\mathbf{\Phi}$ 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. The GPOD formulation summarized in section 2.5 is plagued by issues that are beyond the scope of this article. Unfortunately, this l_2 approach rarely, if ever finds the K-sparse solution. A natural way to enhance sparsity of a is to minimize $\|a'\|_0$, i.e., minimize the number of non-zero elements such that $\mathbf{\Theta} a' = \tilde{\mathbf{x}}$ is satisfied. It has been shown [44] that with $P = K + 1$ ($P > K$ in general) *independent* measurements, one can recover the sparse coefficients with high probability using l_0 reconstruction. This condition can be heuristically interpreted as each measurement needing to excite a different basis vector ϕ_i so that its coefficient a_i can be optimally identified. If two or more measurements excite the same basis 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-minimization is a computationally complex, np-hard and poorly conditioned problem with no stability guarantees.

$$a = \text{argmin} \quad \|a'\|_0 \quad \text{such that} \quad \mathbf{\Theta} a' = \tilde{\mathbf{x}}. \tag{7}$$

The popularity of compressed sensing arises due to the theoretical advances [45–48] guaranteeing near-exact reconstruction of the uncompressed information by solving for the K sparsest coefficients. The l_1 reconstruction is a relatively simpler convex optimization problem and solvable using linear programming techniques for basis pursuit [10,41,49]. Theoretically, one can perform the traditional brutal search to locate the largest K coefficients of a, but the computational effort increases exponentially with the dimension. To overcome this burden, a host of greedy algorithms [11,13,50] have been developed to solve the l_1 minimization problem in eqn. (7) with complexity $\mathcal{O}(N^3)$ for $N_b \approx N$. However, the price one pays here is that $P > \mathcal{O}(Klog(N/K))$ measurements are needed [10,41,45] to *exactly* reconstruct the K-sparse vectors using this approach. The schematics of both l_2 and l_1-based formulations are illustrated in Figure 1.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

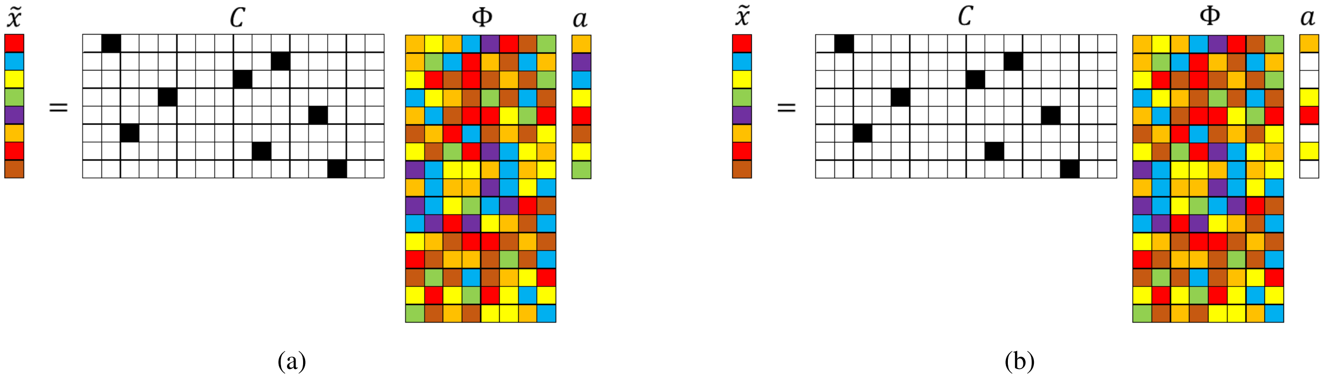


Figure 1. Schematic illustration of l_2 (left) and l_1 (right) minimization reconstruction for sparse recovery using a single-pixel measurement matrix. The numerical values in C are represented by colors: black (1), white (0). The other colors represent numbers that are neither 0 nor 1. In the above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

In summary, there are three parameters N_b, K, P that impact the reconstruction framework. N_b represents the candidate basis space dimension employed for this reconstruction and can at worst obey $N_b \approx N$. K represents the desired system sparsity and tied to the desired quality of reconstruction. That is, K is chosen such that if these features are predicted accurately, then the achieved reconstruction is satisfactory. The more sparse a system, the smaller K is for a desired reconstruction quality. P represents the available quantity of sensors provided as input to the problem. The interplay of N_b, K, and P determines the choice of the algorithm employed, i.e., whether the reconstruction is based on l_1 or l_2 minimization, and the reconstruction quality as summarized in Table 1. In general, K is not known *a priori* and is tied to the desired quality of reconstruction. N and N_b are chosen by the practitioner and depends on the feature space in which the reconstruction will happen. N is the preferred dimension of the reconstructed state, and N_b is the dimension of the candidate basis space in where the reconstruction problem is formulated. As shown in Table 1, for the case with $K = N_b$, the best reconstruction will predict the K weights correctly (using l_2 for the over determined problem) and can be as bad as P when $P < K$ (using l_1 minimization for the under determined problem). In all the cases explored in this discussion, the underlying assumption of $N_b \ll N$ is used. When $K < N_b$ and $N_b > P \geq K$, the worst case prediction will be K weights (for a desired sparsity K) as compared to P weights for the best case (maximum possible sparsity) using l_1 minimization. With $K < N_b$ and $N_b > K > P$, the best case reconstruction will be P weights using l_1. For $P \geq N_b > K$, the best reconstruction will predict N_b weights as compared to K for the worst case. Thus in cases $1, 4, 5$ the desired reconstruction sparsity is always realized whereas in cases 2 and 3, the sensor quantity determines the outcome.

All of the above sparse recovery estimations are conditional upon the measurement basis (rows of C) being incoherent with respect to the sparse basis $\mathbf{\Phi}$. In other words, the measurement basis cannot sparsely represent the elements of the “data basis." This is usually accomplished by using a random sampling for sensor placement, especially when $\mathbf{\Phi}$ is made up of Fourier functions or wavelets. If the basis functions Φ are orthonormal, such as wavelet and POD basis, one can discard the majority of the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

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

Case	$K - N_b$ Relationship	$P - K$ Relationship	Algorithm	Reconstructed Dimension
1	$K = N_b$	$P \geq K$	l_2	K
2	$K = N_b$	$P < K$	l_1	P
3	$K < N_b$	$P < K < N_b$	l_1	P
4	$K < N_b$	$N_b > P \geq K$	l_1	K or P
5	$K < N_b$	$P \geq N_b > K$	l_2	K or N_b

small coefficients in a (setting them as zeros) and still retain reasonably accurate reconstruction. The mathematical explanation of this conclusion has been previously shown in [12]. However, it should be noted that incoherency is a necessary, but not sufficient condition for exact reconstruction. Exact reconstruction requires optimal sensor placement to capture the most information for a given flow field. In other words, incoherency alone does not guarantee optimal reconstruction which also depends on the sensor placement as well as quantity.

2.2. Data-driven Sparse Basis Computation using POD

In the SR framework, basis such as POD modes, Fourier functions, and wavelets [10,12] can be used to generate low-dimensional representations for both l_2 and l_1-based methods. While an exhaustive study on the effect of different choices on reconstruction performance is potentially useful, in this study we compare SR using POD and ELM basis. A similar effort using has been reported in [6] where a comparison of POD with discrete cosine transform basis was carried out.

Proper orthogonal decomposition (POD), also known as Principal Components Analysis (PCA) or Singular Value Decomposition (SVD), is a dimensionality reduction technique that computes a linear combination of low-dimensional basis functions (POD modes) and weights (POD coefficients) from snapshots of experimental or numerical data [2,4] through eigendecomposition of the spatial correlation tensor of the data. It was introduced in the turbulence community by Lumley [51] to extract coherent structures in turbulent flows. The resulting singular vectors or POD modes represent an orthogonal basis that maximizes the energy capture from the flow field. For this reason, such eigenfunctions are considered *optimal* in terms of energy capture and other optimality constraints are theoretically possible. Taking advantage of the orthogonality, one can project these POD basis onto each snapshot 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 eqn. (4). Since the eigendecomposition of the spatial correlation tensor of the flow field requires handling a system of dimension N, it requires significant computational expense. An alternative method is to compute the POD modes using the method of snapshots [52] where the eigendecomposition problem is reformulated in a reduced dimension (assuming the number of snapshots in time is smaller than the spatial dimension) framework as summarized below. Consider that $X \in \mathbb{R}^{N \times M}$ is the full field representation with only the fluctuating part, i.e., the temporal mean is taken out of the data. N is the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

dimension of the full field representation and M is the number of snapshots. The procedure involves computation of the temporal correlation matrix $\bar{C}_M$ as:

$$\bar{C}_M = X^T X. \tag{8}$$

The resulting correlation matrix $\bar{C}_M \in \mathbb{R}^{M \times M}$ is symmetric and an eigendecomposition problem can be formulated as:

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

where the eigenvectors are given in $V = [v_1, v_2, ..., v_M]$ and the diagonal elements of Λ are the eigenvalues $[\lambda_1, \lambda_2, ..., \lambda_M]$. Typically, both the eigenvalues and corresponding eigenvectors are sorted in descending order such as $\lambda_1 > \lambda_2 > ... > \lambda_M$. The POD modes $\mathbf{\Phi}$ and coefficients $\boldsymbol{a}$ can then be computed as

$$\mathbf{\Phi} = XV\Lambda^{-1}. \tag{10}$$

One can represent the field X as a linear combination of the POD modes Φ as shown in eqn. (3) and leverage orthogonality, i.e., $\mathbf{\Phi}^{-1} = \mathbf{\Phi}^T$, to directly estimate the POD coefficients, $\boldsymbol{a} \in \mathbb{R}^{M \times M}$ as shown in eqn. (11),

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

It is worth mentioning that subtracting the temporal mean from the input data is not always necessary for the above procedure. Further, the snapshot procedure fixes the maximum number of POD basis vectors to M which is typically much smaller than the dimension of the full state vector, N. If one wants to reduce the dimension further, then a criterion based on energy capture is devised so that the modes carrying the least amount of energy *truncated* to dimension $K < M$. For many common fluid flows, using the first few POD modes and coefficients are sufficient to capture almost all the relevant dynamics. However, for turbulent flows with large-scale separation, a significant number of POD modes will need to be retained.

2.3. Data-driven Sparse Basis Computation using ELM Autoencoder

While POD represents the energy optimal basis for a given data [53,54], they tend to be highly data-specific and do not span highly evolving flows [55]. As an alternative, one typically reconstructs data from systems with no prior knowledge using generic basis such as Fourier functions [10,12]. Another alternative is to use radial basis function (RBFs) or Gaussian function regression to represent the data that are known to be more robust representations of dynamically evolving flow conditions [55]. In this work, we adopt this flavor by leveraging Extreme Learning Machines (ELM) [19,20] which are regressors employing a Gaussian prior. An extension of this framework to learn encoding-decoding maps of a given data using autoencoder formulation is proposed by Zhou et al. [21–23]. ELM is a single hidden layer feedforward neural network (SLFN) with randomly generated weights for the hidden nodes and bias terms followed by the application of an activation function before computing the output weights by constraining to the output data. If we set the number of hidden nodes smaller than the dimensionality of the input data then we can get compressed feature representations of the original data as the output weight of ELM autoencoder as show in fig. 2. Given a set of data $X \in \mathbb{R}^{N \times M}$ where, N is full field grid

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

dimension and M is the number of snapshots or more simply, in vector form $\mathbf{x}_j \in \mathbb{R}^N$ for $j = 1...M$. By mapping this data from input space to the K-dimensional hidden layer feature space, the output of ELM autoencoder can be written as:

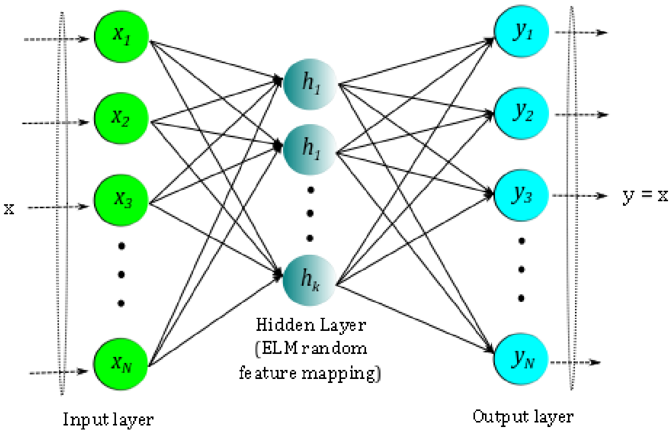


Figure 2. Schematic of ELM autoencoder network, the output is set to be the same as the input x.

$$\mathbf{x}_j = \sum_{i=1}^{K} \phi_i a_i^j = \sum_{i=1}^{K} \phi_i h_i(\mathbf{x}_j) = \sum_{i=1}^{K} \phi_i g(w_i^T \mathbf{x}_j + b_i) \tag{12}$$

where $\mathbf{x}_j \in \mathbb{R}^N$ is the input data, j is the snapshot index, N is the dimension of input data, $w_i \in \mathbb{R}^N$ are the random input weights that map the input nodes to the hidden nodes, b_i is the random bias, $g(.)$ is the activation function operating on a scalar and $\phi_i \in \mathbb{R}^N$ are the output weights that map the hidden features to the output nodes. One example for the activation function is the radial basis function as shown in eqn. (13).

$$g(z) = e^{-(z^2)} \tag{13}$$

In matrix form the linear eqn. (12) can be written as in eqn. (14) where **a** is the matrix of outputs (with elements a_i^j) from the hidden layer (eqn. (15)) and $\mathbf{h}(\mathbf{x}_j) = [h_1(\mathbf{x}_j), \cdots, h_k(\mathbf{x}_j)] = [a_1^j, \cdots, a_k^j]$ represents the output from the K hidden nodes as a row vector for the j^{th} input snapshot $\mathbf{x}_j$. $\mathbf{h}(\mathbf{x}_j)$ is also called as the feature transformation that maps the data, $\mathbf{x}_j$ from the N dimensional input space to the K dimensional hidden layer feature space **a**.

$$X = \mathbf{\Phi a} \tag{14}$$

$$\mathbf{a} = \begin{bmatrix} a_1^1 & \cdots & a_k^1 \\ \vdots & \ddots & \vdots \\ a_1^M & \cdots & a_k^M) \end{bmatrix}^T = \begin{bmatrix} \mathbf{h}(\mathbf{x}_1) \\ \vdots \\ \mathbf{h}(\mathbf{x}_M) \end{bmatrix}^T = \begin{bmatrix} h_1(\mathbf{x}_1) & \cdots & h_k(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ h_1(\mathbf{x}_M) & \cdots & h_k(\mathbf{x}_M) \end{bmatrix}^T \tag{15}$$

The output weights can be written in matrix form as in eqn. (16) and **Y** as in eqn. (17).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

$$\mathbf{\Phi} = \begin{bmatrix} \phi_1^1 & \cdots & \phi_1^N \\ \vdots & \ddots & \vdots \\ \phi_k^1 & \cdots & \phi_k^N \end{bmatrix}^T \tag{16}$$

$$X = \begin{bmatrix} \mathbf{x}_1^1 & \cdots & \mathbf{x}_N^1 \\ \vdots & \ddots & \vdots \\ \mathbf{x}_1^M & \cdots & \mathbf{x}_N^M \end{bmatrix}^T \tag{17}$$

Using eqn. (14), $\mathbf{\Phi}$ can be solved as in eqn. (18) using the Moore-Penrose pseudoinverse.

$$\mathbf{\Phi}_{train} = \mathbf{\Phi} = X\mathbf{a}^+ = X\mathbf{a}^+_{train} \tag{18}$$

The primary difference between POD and ELM autoencoder is that in the former, the basis $\mathbf{\Phi}$ is learnt first whereas in ELM, the features $\boldsymbol{a}$ are derived first followed by $\mathbf{\Phi}$. The other major difference is that the columns of POD basis vectors represent coherent structures contained in the data snapshots, whereas the interpretation of the ELM basis is not clear.

2.4. Measurement Locations, Data Basis and Incoherence

Recalling from subsection 2.1, the reconstruction performance is strongly tied to the measurement matrix, C being *necessarily* incoherent with respect to the sparse basis $\mathbf{\Phi}$ [12] and this is usually accomplished by employing a random sampling for the sensor placement. In practice, one can adopt two types of sparse representation of the data, namely, single-pixel measurement [7,35,56] or random projections [10,12,42]. Typically, single-pixel measurement refers to measuring information at the particular spatial locations such as measurements by UAS in the atmospheric fields. Another popular choice of sensing method in the compressive sensing or image processing community is random projections where the compression matrix is populated using normally distributed random numbers on to which the full state data is projected. As per theory, the random matrix is highly likely to be incoherent with any fixed basis [12], and hence efficient for sparse recovery purposes. However, for most of the fluid flow applications, the sparse data is usually sourced from point measurements and hence, the single-pixel approach is practically relevant. Irrespective of the approach adopted, the measurement matrix C and basis functions $\mathbf{\Phi}$ should be incoherent to ensure optimal sparse reconstruction. This essentially implies that one should have sufficient measurements distributed in space to excite the different modes relevant to the data being reconstructed. Mathematically, this implies that $C\mathbf{\Phi}$ is full rank and invertible. There exist metrics to estimate the extent of coherency between C and $\mathbf{\Phi}$ in the form of an *coherency* number, μ as shown in eqn. (19) [57],

$$\mu(C, \mathbf{\Phi}) = \sqrt{N} \cdot \max_{i \le P,\, j \le K} |\langle c_i, \phi_j \rangle| \,, \tag{19}$$

where c_i is a row vector in C and ϕ_j is a column vector of $\mathbf{\Phi}$. μ typically ranges from 1 (incoherent) to $\sqrt{N}$ (coherent). The smaller the μ, the less measurements one needs to reconstruct the data in an l_1 sense. This is because the coherency parameter enters as the prefactor in the lower-bound for the sensor quantity in l_1-based CS for accurate recovery. There exist optimal sensor placement algorithms such as K-means clustering, the data-driven Online sparse Gaussian Processes [58], physics-based approaches [59] and

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

mathematical approaches [15] that minimize condition number of $\mathbf{\Theta}$. A thorough study on the role of sensor placement on reconstruction quality is much needed and an active topic of research, but not considered within the scope of this work. For this analysis that focuses on the role of basis choice on sparse reconstruction, we simplify the sensor placement strategy by using the *Matlab* function randperm(N) to generate random permutation from 1 to N and the first P values are chosen as the sampling locations in the data. However, to minimize the impact of sensor placement on the conclusions of this study, we perform an ensemble of numerical experiments with different random sensor placements and explore averaged error metrics to make our interpretations robust. Further, in most practical flow measurements the sensor placement is random or based on knowledge of the flow physics. The current approach can be considered to be consistent with that strategy.

2.5. Sparse Reconstruction Algorithm The SR algorithm used here is inspired from [25] and can

be viewed as an l_2 minimization reconstruction of the sparse recovery problem summarized through eqns. (4),(5) and (6) with $\mathbf{\Phi}$ composed of $K \leq M$ basis vectors, i.e. dimension of a is $K \leq M$. The primary difference between the SR framework in eqn. (4) and GPOD [14,15,25,54] as shown in eqn. (21) is the construction of the measurement matrix C and the sparse measurement vector $\tilde{\mathbf{x}}$. For the purposes of this discussion, we will consider reconstruction of a single snapshot. In SR (eqn. (4)) $\tilde{\mathbf{x}} \in \mathbb{R}^P$ is a *compressed* version containing only the measured data, whereas in the GPOD framework, $\tilde{\mathbf{x}} \in \mathbb{R}^N$ is a *masked* version of the full state vector, i.e. the values outside of the P measured locations are zeroed out to generate a filtered version of $\mathbf{x}$. Given a complete set of data $\mathbf{x} \in \mathbb{R}^N$, its basis functions $\phi_k \in \mathbb{R}^N$ and associated coefficients $a \in \mathbb{R}^K$ can be expressed as:

$$\mathbf{x} = \sum_{k=1}^{K} \phi_k a_k. \tag{20}$$

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

$$\tilde{\mathbf{x}} =< m \cdot \mathbf{x} >= C\mathbf{x}, \tag{21}$$

where $C \in \mathbb{R}^{N\times N}$. To contrast, SR uses $C \in \mathbb{R}^{P\times N}$, whereas in GPOD, $C \in \mathbb{R}^{N\times N}$ resulting in a larger matrix with numerous rows of zeros as shown in fig. 3 (compare with fig. 1). To bypass the complexity of handling the $N \times N$ matrix, a mask vector, $m \in N \times 1$ with 1s and 0s operates on $\mathbf{x}$ through a point-wise multiplication operator $< \cdot >$. As an illustration, the point-wise multiplication is represented as $\tilde{\mathbf{x}}_i =< m_i \cdot \mathbf{x}_i >$ for each snapshot $i = 1..M$ where each element of $\mathbf{x}_i$ multiplies with the corresponding element of m_i. This is applicable to the case where each data snapshot, $\mathbf{x}_i$ can have its own measurement mask m_i which is a useful way to represent the evolution of sparse sensor locations over time. The SR formulation in eq. (4) can also support time varying sensor placement, but would require a compression matrix, C_i that is unique for each snapshot. This approach is by design much more computationally and storage intensive, but can handle situations that do not incorporate point sensor compression. The goal of the SR procedure is to recover the full data from the masked data given

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

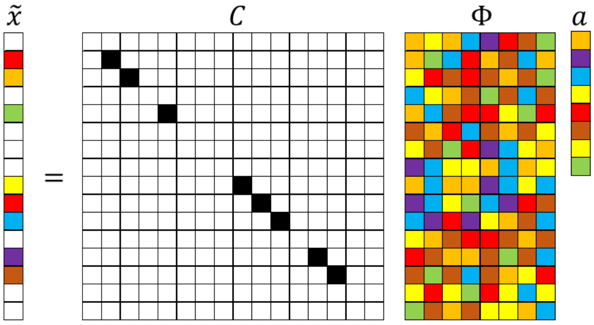


Figure 3. Schematic of GPOD-like SR formulation. The numerical values represented by the colored blocks: black (1), white (0), color(other numbers).

in eqn. (22) by approximating the coefficients $\bar{a}$ (in the l_2 sense) with basis, ϕ_K, learned offline using training data (snapshots of the full field data).

$$\tilde{\mathbf{x}} \approx m \sum_{k=1}^{K} \bar{a}_k \phi_k. \tag{22}$$

The coefficient vector $\bar{a}$ cannot be computed directly by projecting the masked data $\tilde{\mathbf{x}}$ onto the basis $\mathbf{\Phi}$ as these are not designed to optimally represent the sparse data. Instead, one needs to obtain the "best" approximation of the coefficient $\bar{a}$, by minimizing the error $\mathcal{E}$ in the l_2 sense as show in eqn. (23).

$$\mathcal{E} = \left|\left|\tilde{\mathbf{x}} - m \sum_{k=1}^{K} \bar{a}_k \phi_k\right|\right|_2^2 = ||\tilde{\mathbf{x}} - m \cdot \mathbf{\Phi}\bar{a}||_2^2 = ||\tilde{\mathbf{x}} - C\mathbf{\Phi}\bar{a}||_2^2 . \tag{23}$$

In eqn. (23) we see that m acts on each column of $\mathbf{\Phi}$ through a point-wise multiplication operator which is equivalent to masking each basis vector ϕ_k. Unless the sensor placement is constant with time, we remind that the above formulation is valid for a single snapshot reconstruction where the mask vector, m_i, changes with every snapshot $\tilde{\mathbf{x}}_i$ for $i = 1..M$ and the error $\mathcal{E}_i$ represents the single snapshot reconstruction error that will be minimized to compute the approximate features $\bar{a}^i$. It can easily be seen from below that one will have to minimize the different $\mathcal{E}_i$'s sequentially to learn the entire coefficient matrix, $\bar{\boldsymbol{a}} \in \mathcal{R}^{K\times M}$ for all the M snapshots. Denoting the masked basis functions as $\tilde{\phi}_k(z) =< m(z) \cdot \phi_k(z) >$, eq. (23) is rewritten as in eqn. (24).

$$\mathcal{E} = \left|\left|\tilde{\mathbf{x}} - \sum_{k=1}^{M} \bar{a}_k \tilde{\phi}_k\right|\right|_2^2 . \tag{24}$$

In the above formulation, $\tilde{\mathbf{\Phi}}$ is analogous to $C\mathbf{\Phi} = \mathbf{\Theta}$ in eq. (4). To minimize $\mathcal{E}$, one computes the derivative with respect to $\bar{a}$ and equated to zero as below:

$$\frac{\partial}{\partial \bar{a}}(\mathcal{E}) = 0. \tag{25}$$

The result is the linear normal equation given by eqn. (26)

$$\mathbf{M}\bar{a} = f, \tag{26}$$

where $\mathbf{M}_{k1,k2} = \langle \tilde{\phi}_{k1}, \tilde{\phi}_{k2} \rangle$ or $\mathbf{M} = \tilde{\mathbf{\Phi}}^T \tilde{\mathbf{\Phi}}$ and $f_k = \langle \tilde{\mathbf{x}}, \tilde{\phi}_k \rangle$ or $f = \tilde{\Phi}^T \tilde{\mathbf{x}}$. The reconstructed solution is given by eqn. (27) below:

$$\bar{\mathbf{x}} = \sum_{k=1}^{M} \phi_k \bar{a}_k. \tag{27}$$

Algorithm 1 summarizes the above SR algorithm assuming the basis functions (ϕ_k) are known. The above solution procedure for sparse recovery is the same as that described in section 2.1 except for the dimensions of the $\tilde{\mathbf{x}}$ and C.

Algorithm 1: l_2-based algorithm: Sparse Convolution Reconstruction with known basis, $\mathbf{\Phi}$.

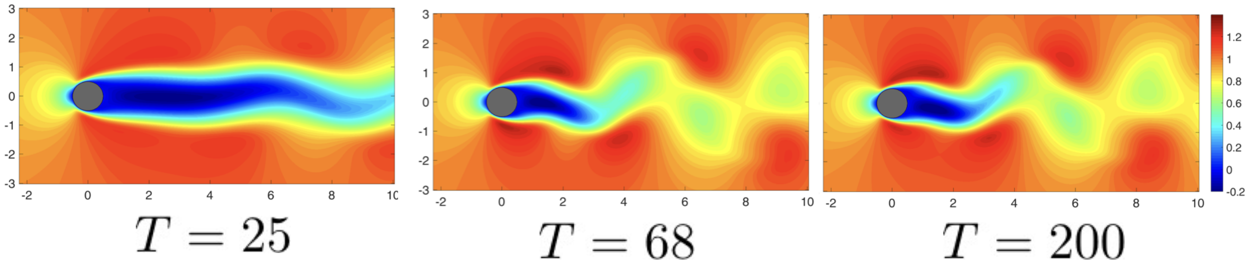
```
input : Full data ensemble $X \in \mathbb{R}^{N \times M}$
        Incomplete data vector $\tilde{X} \in \mathbb{R}^N$
        the mask vector $m \in \mathbb{R}^N$.
output: Approximated full data vector $\bar{X} \in \mathbb{R}^N$
Take out the temporal mean of the ensemble $X$.
Compute basis function $\mathbf{\Phi}$.
Decide on number of modes to retain.
Build a least square problem: $\mathbf{M}\bar{a} = f$.
Compute masked basis function: $\tilde{\mathbf{\Phi}} = m\mathbf{\Phi}$.
Compute matrix $\mathbf{M} = \tilde{\mathbf{\Phi}}^T \cdot \tilde{\mathbf{\Phi}}$.
Compute vector: $f = \tilde{\mathbf{\Phi}}^T \cdot \tilde{X}$.
Solve $\bar{a}$ from the least squares problem: $\mathbf{M}\bar{a} = f$.
Reconstruct the approximated solution $\bar{X} = \mathbf{\Phi}\bar{a}$.
Add the temporal mean back to $\bar{X}$:
  Output the approximated full data vector $\bar{X}$.
```

3. Data Generation for Canonical Cylinder Wake

Studies of cylinder wakes [33,60–62] have attracted considerable interest from the data-driven modeling community for its particularly rich flow physics content, encompassing many complexities of nonlinear fluid flow dynamical systems and ease of simulation using established CFD tools. In this study, we explore data-driven sparse reconstruction for the cylinder wake flow at $Re = 100$. It is well known that the cylinder wake is initially stable and then becomes unstable before settling into a limit-cycle attractor. To illustrate this dynamics, snapshots showing evolution of stream-wise velocity component (u) are shown in Figure 4 for $Re = 100$. For this study, we focus on reconstruction in the temporal regime with periodic limit cycle dynamics in contrast to our earlier work [35] where we focused on reconstruction in both the limit cycle and transient regimes. The two-dimensional cylinder flow data is generated using a spectral Galerkin method [63] to solve incompressible Naiver-Stokes equations, as shown in Eq. 28 below:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088



$T = 25$ $T = 68$ $T = 200$

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.

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0, \tag{28}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial P}{\partial x} + \nu\nabla^2 u, \tag{29}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{\partial P}{\partial y} + \nu\nabla^2 v, \tag{30}$$

where u and v are horizontal and vertical velocity components, P is the pressure field, and ν, the fluid viscosity. The rectangular domain used for this flow simulation 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 $-3D < y < 3D$, is used. The mesh was designed to sufficiently resolve the thin shear layers near the surface of the cylinder and transit wake physics downstream. For the case with $Re = 100$ the grid includes $24,000$ points. The computational method employed is a fourth order spectral expansion within each element in each direction. The sampling rate for each snapshot output is chosen as $\Delta t = 0.2$ seconds.

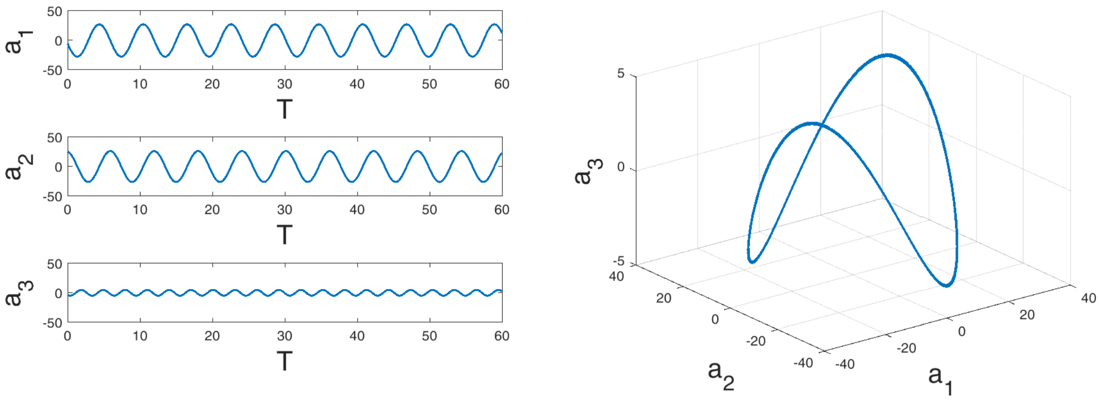
4. Sparse Reconstruction of Cylinder Wake Limit-cycle Dynamics

In this section, we explore sparse reconstruction of fluid flows at $Re = 100$ using the above SR infrastructure for the cylinder flow with well-developed periodic vortex shedding behavior. The GPOD formulation is chosen over the traditional SR formulation to bypass the need for maintaining a separate measurement matrix as we are focusing only on point sensors in this discussion. Maintaining a separate measurement matrix involves storing a lot of elements that are zeros which do not impact the matrix multiplications in both these versions of the SR algorithms. In most cases reported here, Tikhonov regularization is employed to reduce overfitting and provide uniqueness to the solution. In this study, we choose 300 snapshots of data corresponding to a non-dimensional time ($T = \frac{Ut}{D}$) of $T = 60$ with uniform temporal spacing of $dT = 0.2s$. $T = 60$ corresponds to multiple (≈ 10) cycles of periodic vortex shedding behavior for the flow with $Re = 100$ as seen from the temporal evolution of the POD coefficients as shown in fig. 5 below.

4.1. Sparse Reconstruction Experiments and Analysis

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088



(a) 2D view

(b) 3D view

Figure 5. The temporal evolution of the first three normalized POD coefficients for the limit cycle cylinder flow at $Re = 100$.

For this *a priori* assessment of SR performance we reconstruct sparse data from simulations where the full field representation is available. The sparse sensor locations are chosen as single point measurements using a random sampling of the full field data and these locations are fixed for the ensemble of snapshots used for the reconstruction. Reconstruction performance is evaluated by comparing the original simulation predicted field with those from SR using both POD and ELM basis across the entire ensemble of numerical experiments. We undertake this approach in order to assess the relative roles of system sparsity (K), sensor sparsity (P) and sensor placement (C) for both the POD and ELM-based SR. In particular, we aim to accomplish the following through this study: (i) check if $P > K$ is a necessary condition for accurate reconstruction of the fluid flow irrespective of the basis employed; (ii) dependence of estimated sparsity metric, K for desired reconstruction quality on the choice of basis; (iii) understand how sensor placement impacts reconstruction quality for different choice of basis.

To learn the data-driven basis we employ the method of snapshots [52] as shown in eqns. (8)-(11) for POD and train an autoencoder as shown in eqns. (12)-(18) for the ELM basis. For the numerical experiments described here, the data-driven basis and coefficients are obtained from the full data ensemble, i.e., $M = 300$ snapshots corresponding to $T = 60$ non-dimensional times. This gives rise to at most M basis for use in the reconstruction process in eqn. (3), i.e. a candidate basis dimension of $N_b = M$. While this is obvious for the POD case, we observe that for ELM $K \geq M$ does not improve the representational accuracy (see fig. 6). As shown in table.1, the choice of algorithms depend on the choice of system sparsity (K), data sparsity (P) and dimension of the candidate basis space, N_b. Recalling from the earlier discussion in section 2, we see that $P \geq K$ would require an l_2 method for a desired reconstruction sparsity K as long as $P \geq N_b$. In case of POD, the basis are energy optimal for the training data and hence, contain *built-in sparsity*. That is, as long as the basis is relevant for the flow to be reconstructed, retaining only the most energetic modes (basis) should generate the best possible reconstruction for the given sensor locations. Therefore, the POD basis need to be generated once and the sparsity level of the representation is determined by just choosing retain the first few modes in the sequence. On the other hand, ELM basis do not have built-in mechanism for order reduction. The underlying basis hierarchy for the given sparse data is not known *a priori* and therefore requires one to

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

search for the K most significant basis amongst the maximum possible dimension of $N_b = M$ using sparsity promoting l_1 methods. However, for this work we bypass the need l_1 algorithm in favor of the less expensive l_2 method by learning a new set of basis for each choice of $K = N_b < M$.

4.2. Sparsity and Energy Metrics

For this SR study, we explore the conditions for accurate recovery of information in terms of data sparsity (P) and system sparsity (K) which also represents the dimensionality of the system in a given basis space. In other words, sparsity represents the size of a given basis space needed to capture a desired amount of energy. As long as the measurements are incoherent with respect to the basis $\mathbf{\Phi}$ and the system is overdetermined, i.e., $P > K$, one should be able to invert $\mathbf{\Theta}$ to recover the higher dimensional state, X. From earlier discussions in section 2, we know $P > K$ is a sufficient condition for accurate reconstruction using l_0 minimization. Thus, both interpretations require a minimum quantity of sensor data for accurate reconstruction and is verified through numerical experiments in section 4.3. In this section, we describe how the different system sparsity metrics, $K = N_b$, are chosen for the numerical experiments with both POD and ELM basis. Since the basis spaces are different, a good way to compare system dimensions is through the energy captured by the respective basis representations. For POD one easily defines a cumulative energy fraction captured by the K most energetic modes, $\mathbb{E}_K$, using the corresponding singular values of the data as shown in eqn. (31).

$$\mathbb{E}_K = \sum_{k=1}^{K} \frac{\lambda_k}{(\lambda_1 + \lambda_2 + ... + \lambda_M)} \times 100, \tag{31}$$

In the above, the singular values, λ are computed from eqn. (9), and M is the total number of possible eigenvalues for M snapshots. For the cylinder flow case with $Re = 100$, one requires two and five POD modes to capture 95% and 99% of the energy content respectively, indicative of the sparsity of the dynamics in this basis space. In this case, we compute $err_K^{POD} = \left|\left|X - \mathbf{\Phi}_{1..K}^{POD} a_{1..K}^{POD}\right|\right|_2$, where $\mathbf{\Phi}_{1..K}^{POD}$, $a_{1..K}^{POD}$ represents the vector of the first K POD bases and coefficients respectively. The approach of tying the system sparsity with the energy content or alternatively, the error obtained by reconstructing the basis with full data allows one to compare different choices of basis space. Since there exists no natural hierarchy for the ELM basis, we characterize the system sparsity K through the reconstruction error (with respect to the true data) obtained during the training of the ELM network as in eqn. (18). This is computed as the 2-norm of the difference between ELM trained flow field, i.e., ELM output layer, $X_{train,K}^{ELM} = \mathbf{\Phi}_{train,K}^{ELM} a_{train,K}^{ELM}$ and the exact flow field, i.e., the ELM input layer, X_{exact}. Mathematically, we compute $err_K^{ELM} = \left|\left|X - X_{train,K}^{ELM}\right|\right|_2$. To relate the system dimensionality in the ELM and POD spaces and assess their relative sparsity, we compare the energy captured for different values of K in terms of their respective reconstruction errors as shown in fig. 6. We note that the ELM training in eqn. (12) employs random weights which produces variability in the error metric,err_K^{ELM}. To minimize this variability, we perform a set of twenty different training for each value of K_{ELM} and compute the average error as plotted in fig. 6.

From this, we observe that for ELM, $K = 99$ produces same representation error as produced by using just $K = 2$ for POD basis. It turns out that $K = 2$ for POD corresponds to the capture of 95% of the energy as per eqn. (31), i.e. $K_{95}^{ELM} = 99$ & $K_{95}^{POD} = 2$. The decay of err_K^{ELM} with K is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

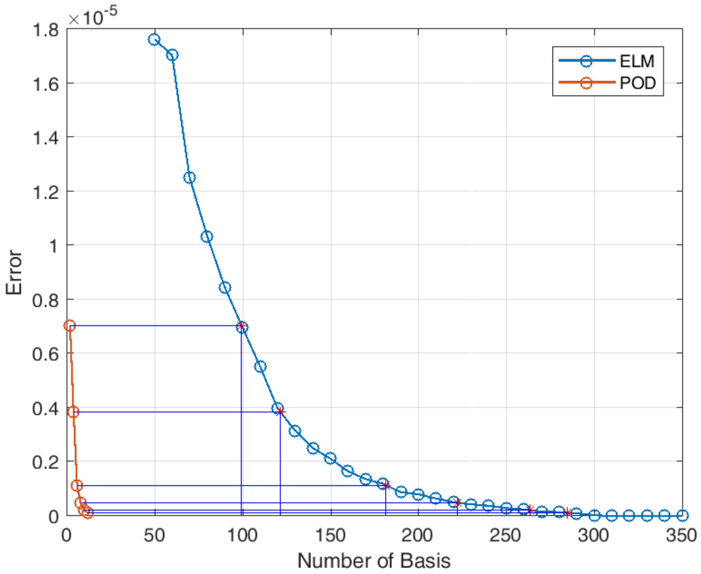


Figure 6. Error (err_K^{POD}, err_K^{ELM}) using different number of basis (K) for both POD reconstruction and ELM prediction.

nearly exponential, but slower than that observed for err_K^{POD}. Further, for $K \geq M$, the ELM training over-fits to the data which drives the reconstruction error to near zero. To assess SR performance across different flow regimes (that have different K_{95}) with different values of K we define a normalized system sparsity metric, $K^* = K/K_{95}$ and a normalized sensor sparsity metric, $P^* = P/K_{95}$. This allows us to design an ensemble of numerical experiments in the discretized $P^* - K^*$ space and the outcomes can be generalized. In this study, the design space is populated over the range $1 < K^* < 6$ and $1 < P^* < 12$ for POD SR and for ELM SR the range is $1 < K^* < 3$ and $1 < P^* < 6$ as the K is bounded by the total number of snapshots, $M = 300$. The lower bound of one is chosen such that the minimally accurate reconstruction captures 95% of the energy. If one desires a different reconstruction norm, then K_{95} can be changed to K_{xx} without loss of generality and the corresponding K-space modified accordingly. Alternately, one can choose $\mathbb{E}_K$, the normalized energy fraction metric to represent the desired energy capture as a fraction of $\mathbb{E}_{K_{95}}$, but is not used in this study.

To quantify the l_2 reconstruction performance, we define the mean squared error as shown in eqn. (32) below,

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

where X is the true data, and $\bar{X}^{SR}$ is the reconstructed field using sparse measurements as per algorithm 1. N and M represent the state and snapshot dimensions affiliated with indices i and j, respectively. Similarly, the mean squared error $\epsilon_{K_{95}^*}^{FR}$ and $\epsilon_{K^*}^{FR}$ for the *full reconstruction* from both POD and ELM based SR are computed as:

$$\epsilon_{K_{95}^*}^{FR} = \frac{1}{M}\frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N}(X_{i,j} - \bar{X}_{i,j}^{FR,K_{95}^*})^2, \tag{33}$$

$$\epsilon_{K^*}^{FR} = \frac{1}{M}\frac{1}{N}\sum_{j=1}^{M}\sum_{i=1}^{N}(X_{i,j} - \bar{X}_{i,j}^{FR,K^*})^2, \tag{34}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

where $\bar{X}^{FR}$ is the full field reconstruction using *exactly computed coefficients* for both POD and ELM cases, $K^*_{95} = K_{95}/K_{95} = 1$ is the normalized system sparsity metric (i.e. number of basis normalized by K_{95}) corresponding to 95% energy capture and $K^* = K/K_{95}$ represents the desired system sparsity. The superscript FR corresponds to the full reconstruction using *exactly computed coefficients* a. For POD this is simply $a = \mathbf{\Phi}^T X$ as per eqn. (11). However for ELM, $a = \mathbf{\Phi}^+ X$ where $\mathbf{\Phi} = \mathbf{\Phi}_{train}$ and is obtained as per eqn. (18). However, this computed a is not the same as a_{train} used in the ELM training step (eqn. (18)) as $\mathbf{\Phi}^+\mathbf{\Phi}$ is not exactly an identity matrix. Therefore, the error in the pseudoinverse computation of $\mathbf{\Phi}$ produces two sets of a, one from direct estimation using the ELM basis ($a = \mathbf{\Phi}^+ X$) and the other from ELM training step (a_{train}) through a direct estimation. This results in two types of error estimations with the a_{train} being used to estimate err^{ELM}_K and a being used to compute $\epsilon^{FR}_{K^*_{95}}$ in eqn. (33) and $\epsilon^{FR}_{K^*}$ in eqn. (34).

Using the above definitions, we can now generate normalized versions of the absolute (ϵ_1) and relative (ϵ_2) errors as shown in eqn. (35). ϵ_1 represents the SR error normalized by the corresponding full reconstruction error for 95% energy capture. ϵ_2 represents the normalized error relative to the desired reconstruction accuracy for the chosen system sparsity, K. These two error metrics are chosen so as to achieve the twin goals of assessing the overall absolute quality of the SR in a normalized sense (ϵ_1) and the best possible reconstruction accuracy for the chosen problem set-up (i.e P, K). Thus, if the best possible reconstruction for a given K is realized then ϵ_2 will take the same value across different K^*. This error metric is used to assess relative dependence of P^* on K^* for the chosen flow field. On the other hand, ϵ_1 provides an absolute estimate of the reconstruction accuracy so that minimal values of P^*, K^* needed to achieve a desired accuracy can be identified.

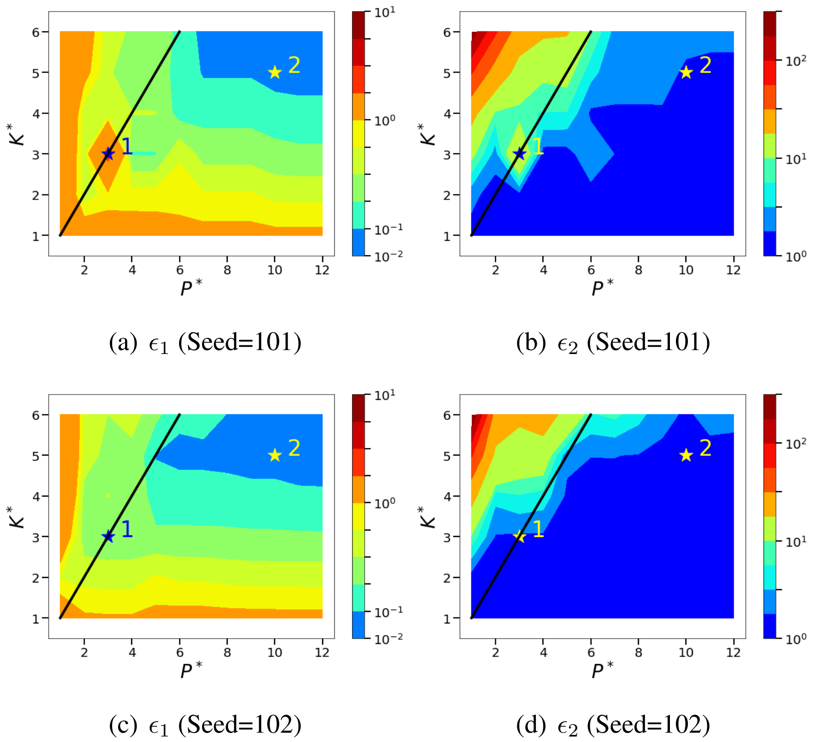
$$\epsilon_1 = \frac{\epsilon^{SR}_{K^*,P^*}}{\epsilon^{FR}_{K^*_{95}}}, \; \epsilon_2 = \frac{\epsilon^{SR}_{K^*,P^*}}{\epsilon^{FR}_{K^*}}. \tag{35}$$

4.3. Sparse Reconstruction of Limit-cycle Dynamics in Cylinder Wakes using POD Basis

As the first part of this study is designed to establish a baseline SR performance using POD basis similar to that performed in [35], we carried out a series of POD-based sparse reconstruction (SR) experiments corresponding to different points in the $P^* - K^*$ design space and spread over the different sensor placements. In these experiments, the sparse data is obtained from *a priori* high resolution flow field data with randomly placed sparse sensors that do not change with each snapshot. The randomized sensor placement in this case is controlled using seed value in *matlab* which is fixed for all the experiments within a given design space. By computing the errors as described in section 4.2 across the $K^* - P^*$ space, the contours of ϵ_1 and ϵ_2 at $Re = 100$ for a few different random sensor placements are shown in figs. 7 and 8. The relative error metric ϵ_2 (the right column in Figure 7 and 8), shows that the smaller errors (both light and dark blue regions) are predominantly located over the region where $P^* > K^*$ as separated the other region using a diagonal line corresponding to $P^* = K^*$. This indicates that the over specified SR problem with $P > K$, i.e. having more sensors than the dimensionality chosen to represent the system yields good results in terms of ϵ_2 while for small P^*, 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-sparse POD reconstruction, this metric represents how effectively the sparse sensor data can approximate the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088



(a) ϵ_1 (Seed=101)

(b) ϵ_2 (Seed=101)

(c) ϵ_1 (Seed=102)

(d) ϵ_2 (Seed=102)

Figure 7. Isocontours of the normalized mean squared POD-based sparse reconstruction errors (l_2 norms) using different random seeds (101 and 102) for sensor placements. Left: normalized absolute error metric, ϵ_1. Right: normalized relative error metric, ϵ_2.

K-sparse solution using l_2 minimization. In principle, the exact K-sparse POD reconstruction is the best possible outcome to expect irrespective of how much sensor data is available as long as $K = N_b$. We note that by constraining the SR problem by choosing the desired energy sparsity, K, the l_2 reconstruction is reconciled with the l_0 minimization solution [35]. Consistent with the observations in [35], we observe that ϵ_1 contours adhere to a L-shaped structure indicating that absolute normalized error reduces as K is increased to capture more energy contained in the full field data.

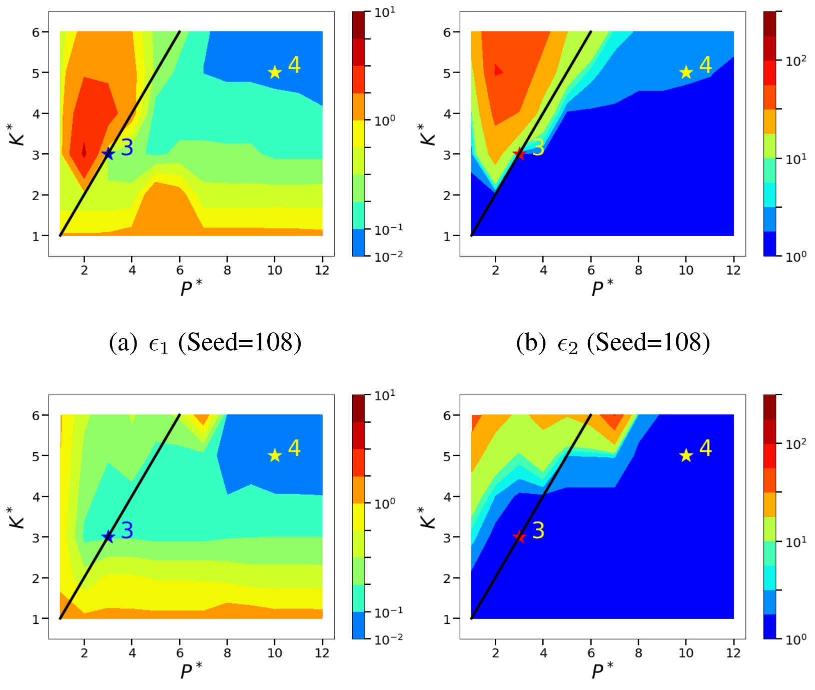
While this ‘best’ reconstruction is almost always observed for the higher values of P^* and K^* for the different sensor placements, there appear to be some exceptions. Notably, for few sensor placement choices(seeds 101 in fig. 7 and 108 in fig. 8), a small portion of ϵ_1 in the region abutting the $P^* = K^*$ line shows nearly an order of magnitude higher error, $\mathcal{O}(10^1)$ (colored as red in fig. 7 and 8) as compared to the expected values of $\mathcal{O}(1)$ observed for sensor placement using seeds 102 and 109 in figs. 7 and 8 respectively.

We probe these ‘anomalous’ points by visualizing the SR of a random snapshot along with their corresponding sensor placements to deduce possible insight into the reason for high errors in terms of sensor placement. In [35], it was shown that although the coherency number, μ was small for a given sensor placement choice, the sparse data points should still span the physics to be captured. Inspired by this, we experiment the impact of sensor placement for two regions corresponding to $P^* \simeq K^*$ and $P^* > K^*$.

At point 1 (where $P^* = K^*$) for sensor placement with seed 101 (fig. 7) both ϵ_1 and ϵ_2 indicate an order of magnitude higher error relative to its neighborhood. But for the same point with modified sensor

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088



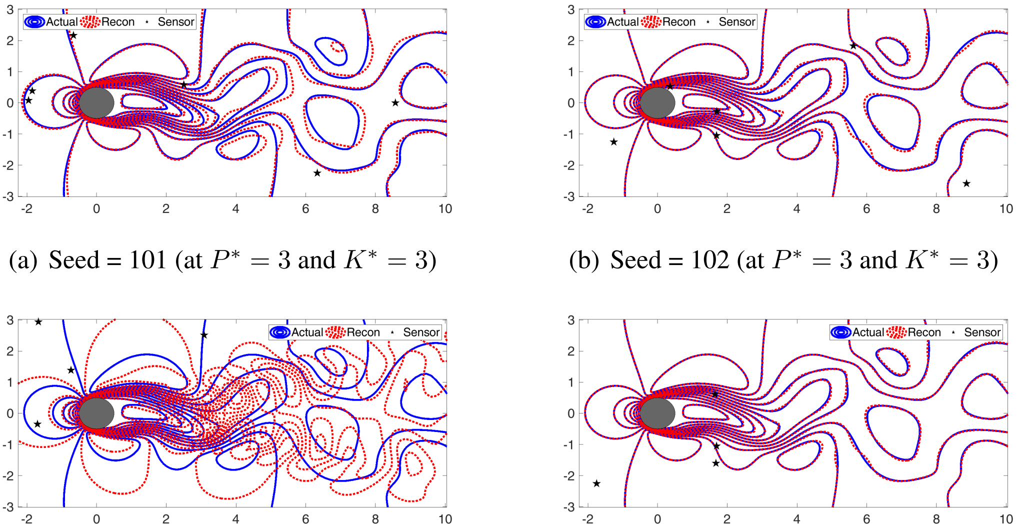
(a) ϵ_1 (Seed=108)

(b) ϵ_2 (Seed=108)

(c) ϵ_1 (Seed=109)

(d) ϵ_2 (Seed=109)

Figure 8. Isocontours of the normalized mean squared POD-based sparse reconstruction errors (l_2 norm) using different random seed (108 and 109) for sensor placements. Left: normalized absolute error metric, ϵ_1. Right: normalized relative error metric, ϵ_2.



(a) Seed = 101 (at $P^* = 3$ and $K^* = 3$)

(b) Seed = 102 (at $P^* = 3$ and $K^* = 3$)

(c) Seed = 108 (at $P^* = 2$ and $K^* = 3$)

(d) Seed = 109 (at $P^* = 2$ and $K^* = 3$)

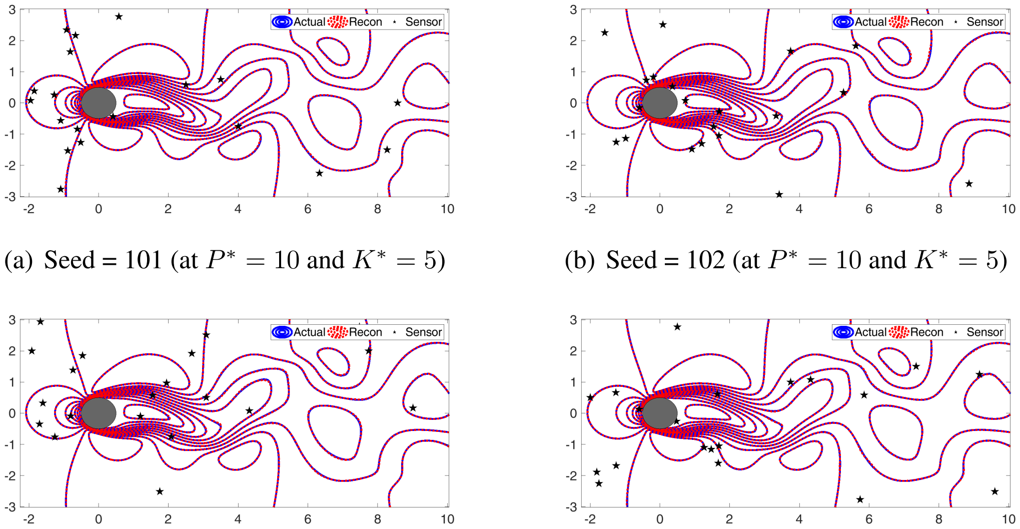
Figure 9. Comparison of SR with FR for different random sensor placement. Blue solid line: FR. Red dashed line: SR. Black Star: Sensor location.

placement using seed 102, we observe nearly an order of magnitude lower error. Comparing the SR flow fields for both the sensor placements with different seeds as shown in figure 9, we see that seed 102 places more data points in the wake of the cylinder as compared to that for seed 101 and hence better reconstruction. A needed follow on study to this effort is to explore data-driven methods for optimal

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

sensor placement and its influence on SR error. Mathematically, data points in the cylinder wake region excite the most energetic POD modes as against sensors placed elsewhere. This is clearly shown in [35] where the coefficients or features corresponding to the most energetic POD modes are erroneous when recovered using inadequate sensor placement. This trend is observed even for the under specified (ill-posed) case with $P^* < K^*$, i.e. point 3 in fig.8 for cases with sensor placement seeds 108 and 109 as shown in fig. 9. In conclusion, we observe that although the POD-based SR is very efficient in



(a) Seed = 101 (at $P^* = 10$ and $K^* = 5$)

(b) Seed = 102 (at $P^* = 10$ and $K^* = 5$)

(c) Seed = 108 (at $P^* = 10$ and $K^* = 5$)

(d) Seed = 109 (at $P^* = 10$ and $K^* = 5$)

Figure 10. Comparison of SR with FR for different random sensor placement. Blue solid line: FR. Red dashed line: SR. Black Star: Sensor location.

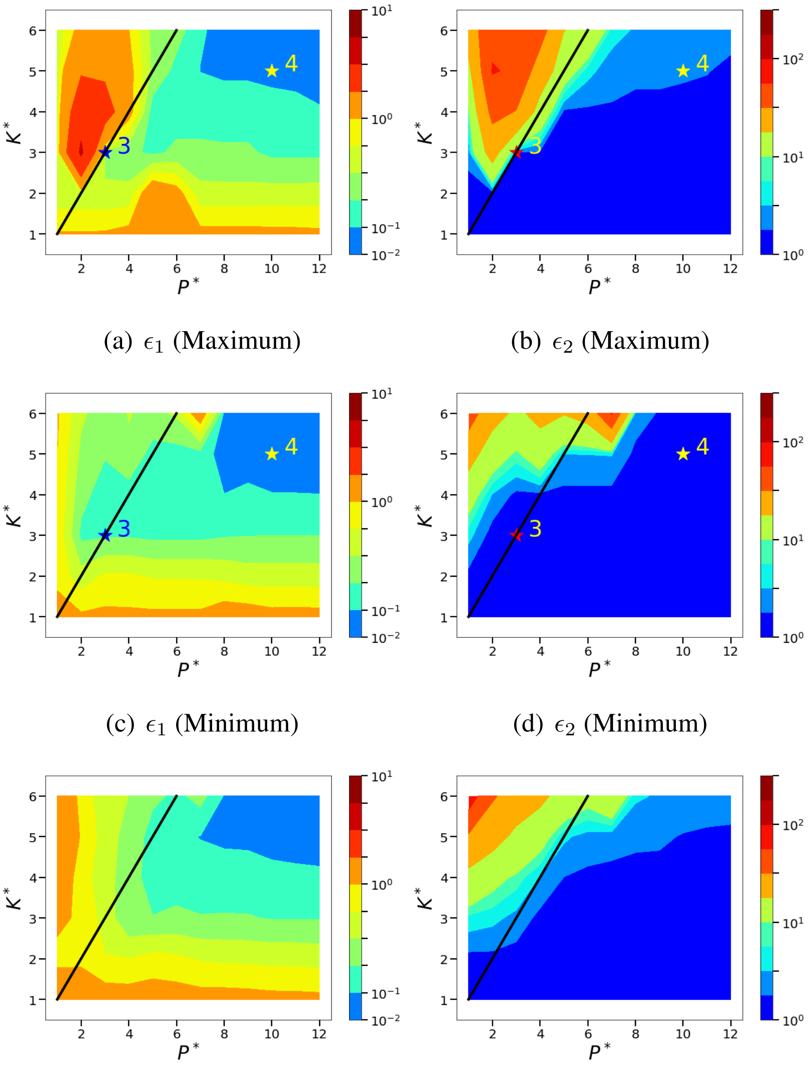
terms of sensor quantity requirement, the reconstruction errors are sensitive to sensor location even when $P^* \simeq K^*$. For cases with $P^* > K^*$ (points 2 and 4 in figs. 7 and 8), we observe that this sensitivity to sensor placement is highly minimized although small differences exist as shown in fig. 10 . This is because, increase in number of measurement points enhances the probability of locating points within the cylinder wake. In order ot generalize the error metrics computed in figs. 7 and 8, we perform an ensemble of ten different sensor placements corresponding to seeds ranging from 101 to 110 and plot the error contours corresponding to maximum and minimum errors estimated across the entire $P^* - K^*$ design space and also the average error over this ensemble as shown in fig. 11. The maximum error is found for seed 108 and minimum error for seed 109. The average error contours represent the most probable SR outcome independent of the anomalies from sensor placement.

4.4. Sparse Reconstruction using ELM Basis

The previous section focused on the SR performance of limit cycle cylinder wake dynamics using POD basis and was shown to perform well if certain conditions are met, namely, $P^* > K^*$ and reasonable sensor placement. A key issue identified with POD-based SR for low-dimensional systems such as the cylinder wake dynamics is that it requires a small number of sensors for reconstruction, but sensitive to their placement. As observed in Section 4.2, ELM requires more number of basis for the same amount

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088



(a) ϵ_1 (Maximum)

(b) ϵ_2 (Maximum)

(c) ϵ_1 (Minimum)

(d) ϵ_2 (Minimum)

(e) ϵ_1 (Average)

(f) ϵ_2 (Average)

Figure 11. 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. The average error across the entire ensemble of ten random sensor placements is also shown. Left: normalized absolute error metric, ϵ_1. Right: normalized relative error metric, ϵ_2.

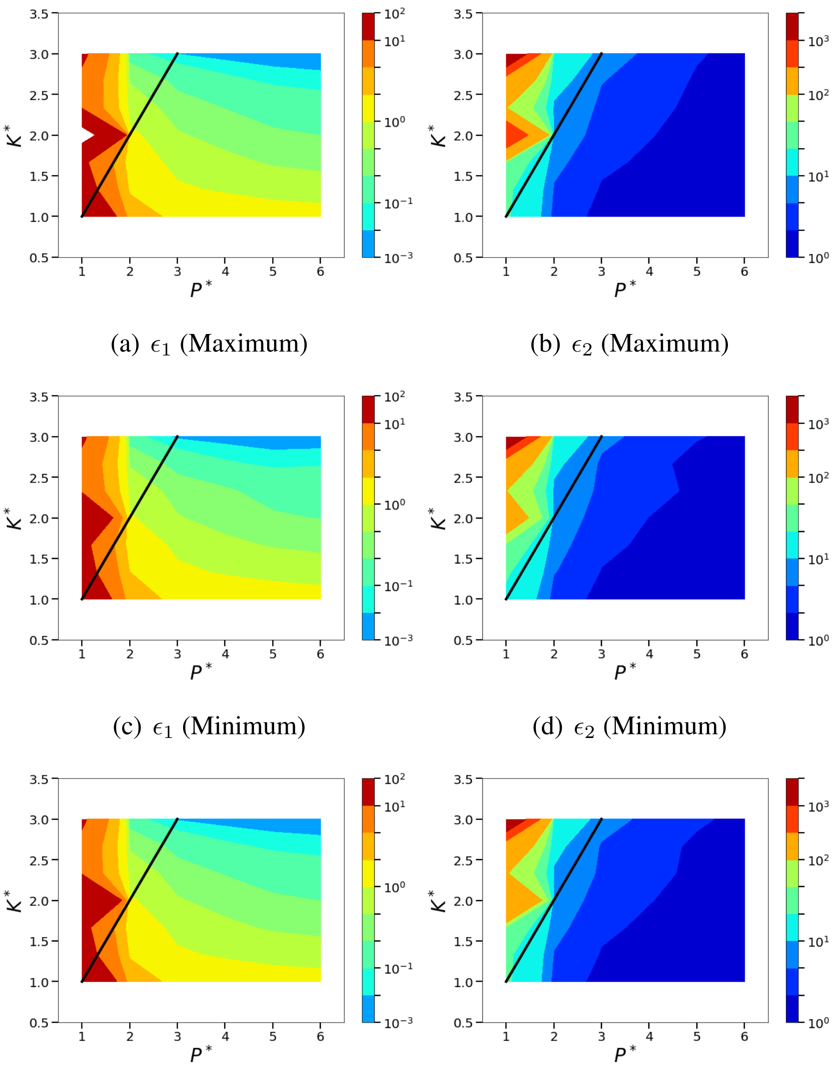
of energy capture. For example, for 95% energy capture, we will need $K = 99$ for ELM and just $K = 2$ for POD. While SR using ELM basis is similar in concept, the non-orthogonality of ELM basis and their relative lack of sparsity brings about certain differences. It was shown in section 4.2 that there exist two different kinds of errors - a training error and the reconstruction error. If the ELM basis were to be orthogonal just like the POD basis, both these errors (and the ELM features a, a_{train}) will be equivalent. Secondly, in section 2.3 the algorithm for computing basis using ELM autoencoder use randomly chosen input weights while generating the K-dimensional hidden layer features which represents the low-dimensional representation. Consequently, the realized ELM basis are not unique unlike the POD basis for a given training data set. It was observed from our numerical experiments that while the sensitivity of this randomness in the ELM basis to the reconstruction errors is not sever, it is

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

perceptible. To analyze the performance of ELM-based SR, we carry two different types of analysis. The first explores of the effect of randomness in the weights used within the ELM training while fixing the sensor placement. The second explores randomness from the choice of sensor placement for a given ELM basis.

To accomplish the first, we compute the normalized ELM-based SR error metrics ϵ_1 and ϵ_2 using ten different choices for the random input weight sets and contours corresponding to the seed with maximum, minimum and the average error contours are shown in fig. 12. These plots clearly show that there exist very little sensitivity to the choice of random weights in the ELM training. Further, as was the case for the POD-based SR, the region of good reconstruction is separated from the high-error region by the straight line corresponding to $P^* = K^*$ indicating that similar performance normalized performance bounds exist for both POD (orthogonal basis) and ELM (non-orthogonal basis).



(a) ϵ_1 (Maximum)

(b) ϵ_2 (Maximum)

(c) ϵ_1 (Minimum)

(d) ϵ_2 (Minimum)

(e) ϵ_1 (Average)

(f) ϵ_2 (Average)

Figure 12. Isocontours of the normalized mean squared l_2 ELM-based sparse reconstruction errors for maximum, minimum and average error using different choice of random input weights. Left: normalized absolute error metric, ϵ_1. Right: normalized relative error metric, ϵ_2.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088

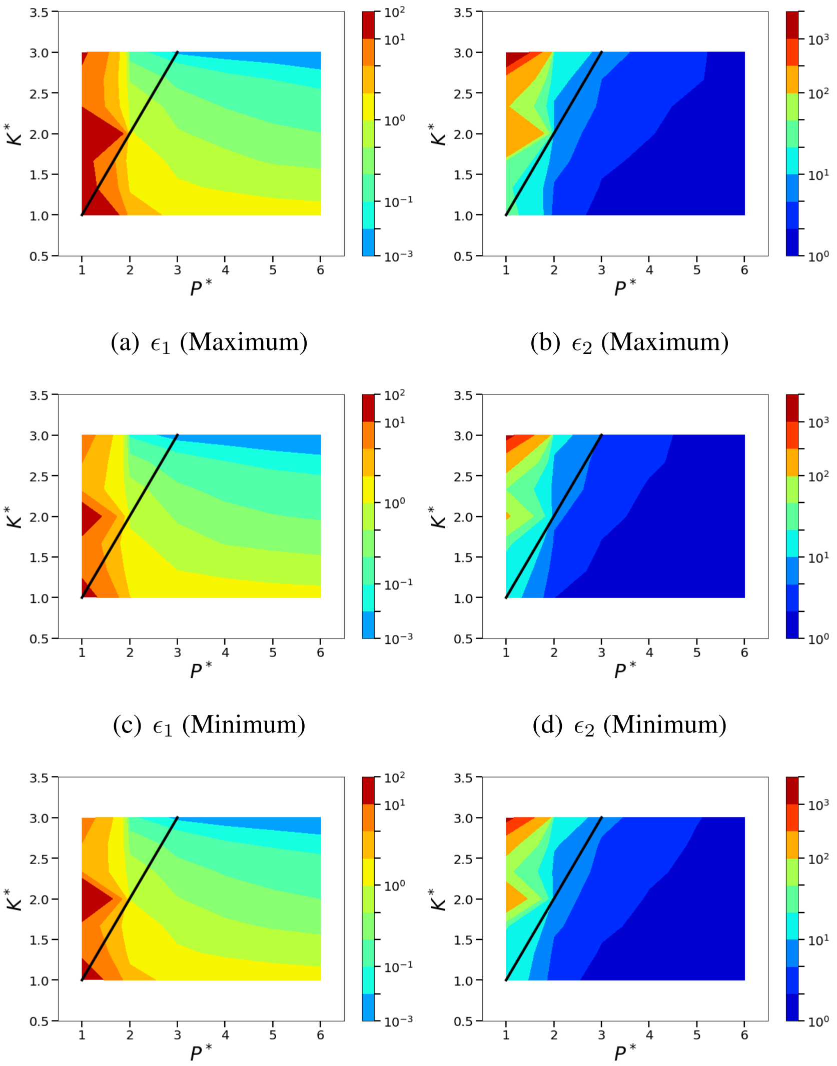
The second part of the ELM-based SR analysis is to assess the effect of the sensor placement on the reconstruction performance for a fixed choice of random weights in the ELM training. To accomplish this, we consider an ensemble of numerical simulations with ten different choices of sensor placement across the entire $P^* - K^*$ design space. Figure 13 shows the normalized error contours for the case with sensor placement resulting in te maximum error, minimum errors and the average error field over the entire ensemble. Once again, we observe accurate SR performance overall for $P^* - K^*$ which we results in a well-posed reconstruction problem. Further, we observe that the choice of sensor placement has very minimal impact on the over SR error metrics. This is not surprising given that ELM requires more sensors as compared to POD to achieve the same level of reconstruction accuracy. Therefore, not only do a significant number of these sensors find themselves in the most dynamically relevant regions of the flow, but even if a few of the sensors were to be misplaced, the contribution to the overall error metric from this is much smaller as compared to POD-based SR. In this way, we argue that although ELM-based SR requires more sensors for accurate reconstruction as compared to POD-based SR, they tend to be more robust based on the limited numerical experiments performed in this study.

5. Conclusion

In this article, we explore the interplay of sensor quantity (P), placement and system energy-based sparsity (K) using optimal data-driven POD & ELM basis on l_2 sparse reconstruction (SR) of a cylinder wake flow. Overall, we observed that the choice of sparse basis plays a crucial role in the SR performance as it determines the quantity of sensors and their placement for a desired recovery quality. Employing POD/SVD basis also allows for efficient energy-based SR as long as they span the sparse data, i.e. the most energetic POD basis for the training data are also the energetic structures in the data to be reconstructed. Unlike generic basis spaces such as Fourier or radial basis functions, the data-driven POD basis needs to be highly flow relevant as retaining the K most energetic modes for reconstruction is also the K-sparse solution for the given sensor locations. For the more generic ELM basis, relevance to the flow to be reconstructed is important, but there exists no inherent hierarchy and consequently, a new K-dimensional basis space is generated for every K by training the ELM over and again. Also, the POD basis is unique for a given data set while the ELM basis is not as it depends on the random weights to generate the hidden layer features during training of the network. The second major difference between POD and ELM basis is that while POD modes are orthogonal, ELM basis are not. Thirdly, POD represents the sparsest possible basis space to span the given data while ELM shows an order of magnitude higher sparsity for the same level of energy capture. Consistent with the outcomes reported in [35], it was observed that $P^* > K^*$ produced consistently accurate reconstructions as the SR is well-posed and over specified under these conditions for both the POD and ELM-based approaches. POD basis being sparse, the SR problem required very few sensors for accurate reconstruction of this cylinder wake dynamics, but susceptible to inaccurate predictions when the sensor placement is inadequate. To account for this sensitivity, all the error metrics reported in this article are ensemble averaged over multiple choices of sensor distribution. On the other hand, ELM-basis being relatively less sparse require more sensors for accurate reconstruction, but also turned out to be highly robust to sensor placement.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at Fluids 2018, 3, 88; doi:10.3390/fluids3040088



(a) ϵ_1 (Maximum)

(b) ϵ_2 (Maximum)

(c) ϵ_1 (Minimum)

(d) ϵ_2 (Minimum)

(e) ϵ_1 (Average)

(f) ϵ_2 (Average)

Figure 13. Isocontours of the normalized mean squared l_2 ELM-based sparse reconstruction errors for maximum, minimum and average error using different choice of random sensor placement. Left: normalized absolute error metric, ϵ_1. Right: normalized relative error metric, ϵ_2.

6. Acknowledgments

We acknowledge computational resources from HPCC and start-up research funds from the Oklahoma State University.

7. Author Contributions

BJ conceptualized the research with input from AM and CL. AM and CL developed the sparse reconstruction codes with input from BJ. BJ and AM analyzed the data. AM and CL developed the first draft of the manuscript and BJ edited the final written manuscript.

8. Conflicts of Interest

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

The authors declare no conflict of interest.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

References

1. Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.
2. Philip Holmes. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.
3. Gal Berkooz, Philip Holmes, and John L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.
4. Kunihiko Taira, Steven L Brunton, Scott Dawson, Clarence W Rowley, Tim Colonius, Beverley J McKeon, Oliver T Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S Ukeiley. Modal analysis of fluid flows: An overview. 55(12):4013–4041, 2017.
5. Balaji Jayaraman, Chen Lu, Joshua Whitman, and Girish Chowdhary. Sparse convolution-based markov models for nonlinear fluid flows. *arXiv preprint arXiv:1803.08222b*, 2018.
6. Zhe Bai, Thakshila Wimalajeewa, Zachary Berger, Guannan Wang, Mark Glauser, and Pramod K Varshney. Low-dimensional approach for reconstruction of airfoil data via compressive sensing. *AIAA Journal*, 2014.
7. Ido Bright, Guang Lin, and J Nathan Kutz. Compressive sensing based machine learning strategy for characterizing the flow around a cylinder with limited pressure measurements. *Physics of Fluids*, 25(12):127102, 2013.
8. Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.
9. Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Compressive sampling and dynamic mode decomposition. *arXiv preprint arXiv:1312.5186*, 2013.
10. Emmanuel J Candès et al. Compressive sampling. In *Proceedings of the international congress of mathematicians*, volume 3, pages 1433–1452. Madrid, Spain, 2006.
11. Joel A Tropp and Anna C Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on information theory*, 53(12):4655–4666, 2007.
12. Emmanuel J Candès and Michael B Wakin. An introduction to compressive sampling. *IEEE signal processing magazine*, 25(2):21–30, 2008.
13. Deanna Needell and Joel A Tropp. Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3):301–321, 2009.
14. Tan Bui-Thanh, Murali Damodaran, and Karen Willcox. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA journal*, 42(8):1505–1516, 2004.
15. Karen Willcox. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & fluids*, 35(2):208–226, 2006.
16. Daniele Venturi and George Em Karniadakis. Gappy data and reconstruction procedures for flow past a cylinder. *Journal of Fluid Mechanics*, 519:315–336, 2004.
17. Hasan Gunes, Sirod Sirisup, and George Em Karniadakis. Gappy data: To krig or not to krig? *Journal of Computational Physics*, 212(1):358–382, 2006.
18. Hasan Gunes and Ulrich Rist. On the use of kriging for enhanced data reconstruction in a separated transitional flat-plate boundary layer. *Physics of Fluids*, 20(10):104109, 2008.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

19. Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.
20. Guang-Bin Huang, Dian Hui Wang, and Yuan Lan. Extreme learning machines: a survey. *International journal of machine learning and cybernetics*, 2(2):107–122, 2011.
21. Liyanaarachchi Lekamalage Chamara Kasun, Hongming Zhou, Guang-Bin Huang, and Chi Man Vong. Representational learning with extreme learning machine for big data. *IEEE intelligent systems*, 28(6):31–34, 2013.
22. Hongming Zhou, Yeng Chai Soh, Chaoyang Jiang, and Xiaoying Wu. Compressed representation learning for fluid field reconstruction from sparse sensor observations. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, pages 1–6. IEEE, 2015.
23. Hongming Zhou, Guang-Bin Huang, Zhiping Lin, Han Wang, and Yeng Chai Soh. Stacked extreme learning machines. *IEEE transactions on cybernetics*, 45(9):2013–2025, 2015.
24. Justin Romberg. Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):14–20, 2008.
25. Richard Everson and Lawrence Sirovich. Karhunen–loeve procedure for gappy data. *JOSA A*, 12(8):1657–1664, 1995.
26. Pankaj Saini, Christoph M Arndt, and Adam M Steinberg. Development and evaluation of gappy-pod as a data reconstruction technique for noisy piv measurements in gas turbine combustors. *Experiments in Fluids*, 57(7):1–15, 2016.
27. S Mallet. A wavelet tour of signal processing, 1998.
28. Steven L Brunton, Jonathan H Tu, Ido Bright, and J Nathan Kutz. Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. *SIAM Journal on Applied Dynamical Systems*, 13(4):1716–1732, 2014.
29. Zhe Bai, Steven L Brunton, Bingni W Brunton, J Nathan Kutz, Eurika Kaiser, Andreas Spohn, and Bernd R Noack. Data-driven methods in fluid dynamics: Sparse classification from experimental data. In *Whither Turbulence and Big Data in the 21st Century?*, pages 323–342. Springer, 2017.
30. Boris Kramer, Piyush Grover, Petros Boufounos, Saleh Nabi, and Mouhacine Benosman. Sparse sensing and dmd-based identification of flow regimes and bifurcations in complex flows. *SIAM Journal on Applied Dynamical Systems*, 16(2):1164–1196, 2017.
31. Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.
32. Jonathan H Tu, Clarence W Rowley, Dirk M Luchtenburg, Steven L Brunton, and J Nathan Kutz. On dynamic mode decomposition: theory and applications. *arXiv preprint arXiv:1312.0041*, 2013.
33. Clarence W Rowley and Scott TM Dawson. Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49:387–417, 2017.
34. Hao Wu and Frank Noé. Variational approach for learning markov processes from time series data. *arXiv preprint arXiv:1707.04659*, 17, 2017.
35. Chen Lu and Balaji Jayaraman. Interplay of sensor quantity, placement and system dimensionality on energy sparse reconstruction of fluid flows. *arXiv preprint arXiv:1806.08428*, 2018.
36. Simon R Arridge and John C Schotland. Optical tomography: forward and inverse problems. *Inverse Problems*, 25(12):123010, 2009.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

37. Albert Tarantola and Bernard Valette. Generalized nonlinear inverse problems solved using the least squares criterion. *Reviews of Geophysics*, 20(2):219–232, 1982.
38. Ramesh Neelamani. *Inverse problems in image processing*. PhD thesis, Rice University, 2004.
39. Animesh Khemka. *Inverse problems in image processing*. PhD thesis, Purdue University, 2009.
40. Albert Tarantola. *Inverse problem theory and methods for model parameter estimation*, volume 89. siam, 2005.
41. David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.
42. Richard G Baraniuk. Compressive sensing [lecture notes]. *IEEE signal processing magazine*, 24(4):118–121, 2007.
43. Richard G Baraniuk, Volkan Cevher, Marco F Duarte, and Chinmay Hegde. Model-based compressive sensing. *IEEE Transactions on Information Theory*, 56(4):1982–2001, 2010.
44. Shriram Sarvotham, Dror Baron, Michael Wakin, Marco F Duarte, and Richard G Baraniuk. Distributed compressed sensing of jointly sparse signals. In *Asilomar conference on signals, systems, and computers*, pages 1537–1541, 2005.
45. Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.
46. Emmanuel J Candes, Justin K Romberg, and Terence Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on pure and applied mathematics*, 59(8):1207–1223, 2006.
47. Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE transactions on information theory*, 52(12):5406–5425, 2006.
48. Emmanuel J Candes and Justin K Romberg. Signal recovery from random projections. In *Computational Imaging III*, volume 5674, pages 76–87. International Society for Optics and Photonics, 2005.
49. Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM review*, 43(1):129–159, 2001.
50. Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted âĎŞ 1 minimization. *Journal of Fourier analysis and applications*, 14(5-6):877–905, 2008.
51. JL Lumley. Stochastic tools in turbulence (new york: Academic). 1970.
52. Lawrence Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.
53. Patricia Astrid, Siep Weiland, Karen Willcox, and Ton Backx. Missing point estimation in models described by proper orthogonal decomposition. In *Decision and Control, 2004. CDC. 43rd IEEE Conference on*, volume 2, pages 1767–1772. IEEE, 2004.
54. T Bui-Thanh, Murali Damodaran, and Karen Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *21st AIAA Applied Aerodynamics Conference*, page 4213, 2003.
55. Balaji Jayaraman, Chen Lu, Joshua Whitman, and Girish Chowdhary. Sparse convolution-based markov models for nonlinear fluid flows. *arXiv preprint arXiv:1803.08222*, 2018.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 21 August 2018 doi:10.20944/preprints201808.0385.v1

Peer-reviewed version available at *Fluids* **2018**, *3*, 88; doi:10.3390/fluids3040088

56. Steven L Brunton, Clarence W Rowley, and David R Williams. Reduced-order unsteady aerodynamic models at low reynolds numbers. *Journal of Fluid Mechanics*, 724:203–233, 2013.

57. Emmanuel Candes and Justin Romberg. Sparsity and incoherence in compressive sampling. *Inverse problems*, 23(3):969, 2007.

58. Lehel Csató and Manfred Opper. Sparse on-line gaussian processes. *Neural computation*, 14(3):641–668, 2002.

59. Kelly Cohen, Stefan Siegel, and Tom McLaughlin. Sensor placement based on proper orthogonal decomposition modeling of a cylinder wake. In *33rd AIAA Fluid Dynamics Conference and Exhibit*, page 4259, 2003.

60. Anatol Roshko. On the development of turbulent wakes from vortex streets. 1954.

61. CHK Williamson. Oblique and parallel modes of vortex shedding in the wake of a circular cylinder at low reynolds numbers. *Journal of Fluid Mechanics*, 206:579–627, 1989.

62. Bernd R Noack, Konstantin Afanasiev, Marek Morzynski, Gilead Tadmor, and Frank Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497:335–363, 2003.

63. Chris D Cantwell, David Moxey, A Comerford, A Bolis, G Rocco, Gianmarco Mengaldo, Daniele De Grazia, Sergey Yakovlev, J-E Lombard, D Ekelschot, et al. Nektar++: An open-source spectral/hp element framework. *Computer Physics Communications*, 192:205–219, 2015.