Model Reduction for Flow Analysis and Control

Clarence W. Rowley and Scott T.M. Dawson

Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, New Jersey 08544; email: cwrowley@princeton.edu

Annu. Rev. Fluid Mech. 2017. 49:387–417

First published online as a Review in Advance on August 22, 2016

The *Annual Review of Fluid Mechanics* is online at fluid.annualreviews.org

This article's doi: 10.1146/annurev-fluid-010816-060042

Keywords

proper orthogonal decomposition, Galerkin projection, balanced truncation, dynamic mode decomposition, Koopman operator, kernel method

Abstract

Advances in experimental techniques and the ever-increasing fidelity of numerical simulations have led to an abundance of data describing fluid flows. This review discusses a range of techniques for analyzing such data, with the aim of extracting simplified models that capture the essential features of these flows, in order to gain insight into the flow physics, and potentially identify mechanisms for controlling these flows. We review well-developed techniques, such as proper orthogonal decomposition and Galerkin projection, and discuss more recent techniques developed for linear systems, such as balanced truncation and dynamic mode decomposition (DMD). We then discuss some of the methods available for nonlinear systems, with particular attention to the Koopman operator, an infinite-dimensional linear operator that completely characterizes the dynamics of a nonlinear system and provides an extension of DMD to nonlinear systems.

1. INTRODUCTION

> Truth . . . is much too complicated to allow anything but approximations.
>
> —John von Neumann (1947)

The field of fluid mechanics is in the enviable position of having a model that apparently describes the motion of a fluid to near-perfect accuracy: the Navier-Stokes equations. However, this model turns out to be very complex—even the existence and smoothness of solutions in three dimensions are not understood, and numerical simulations of turbulent flows challenge our largest computers.

This review discusses methods for obtaining simplified, approximate models for fluid flows, either given access to a numerical simulation or given data measured from an experiment. The goal of these approximate models is to provide insight about the fundamental mechanisms that describe a particular flow (e.g., a particular instability or energy transfer mechanism). Such a model is particularly important if one wishes to use flow control to alter the flow in some way: A suitable model enables the use of many well-developed tools from control theory (e.g., Skogestad & Postlethwaite 2005).

The field of reduced-order modeling is large, and new techniques are developing rapidly. Readers are referred to Benner et al. (2015) for a recent review of model reduction methods for linear systems and to Brunton & Noack (2015) for a comprehensive overview of methods used for the control of turbulence. Our aim is to review some well-developed techniques, such as proper orthogonal decomposition (POD; Lumley 1970), as well as more recent techniques, such as balanced truncation (Moore 1981) and dynamic mode decomposition (DMD; Schmid 2010) and to make comparisons between the various methods. For nonlinear systems, general techniques are not nearly as well developed as they are for linear systems; still there is progress, and we describe in some detail a particular recent line of research that makes use of the so-called Koopman operator (Koopman 1931) to represent a nonlinear system as a higher-dimensional (often infinite-dimensional) linear system.

The review is organized as follows. Section 2 reviews the fundamentals of projection onto a set of basis functions, or modes, including POD and Galerkin projection. Section 3 discusses methods for model reduction for linear systems, including balanced truncation, balanced proper orthogonal decomposition (BPOD), the eigensystem realization algorithm (ERA), and DMD, and compares the methods using an example of linearized channel flow. Section 4 presents a few methods deemed promising for model reduction of nonlinear systems, focusing on methods involving the Koopman operator, and extensions of DMD to nonlinear systems.

2. MODAL DECOMPOSITIONS

In this review, we focus on model reduction methods that involve decomposing a flow field into a set of modes. Although we are primarily interested in fluid quantities such as a velocity field defined everywhere in a spatial domain, here we assume that the quantities of interest at a particular time t have been discretized, and assembled into a vector $\mathbf{x}(t)$, called the state vector. For instance, $\mathbf{x}(t)$ might contain the values of velocity at a set of grid points, or it might contain the corresponding coefficients of Fourier modes or Chebyshev polynomials. The methods discussed here apply equally well to any of these situations. In model reduction, we are interested in the case in which the vector $\mathbf{x}(t)$ has a large number of components (e.g., containing values at thousands or millions of grid or measurement points), and we wish to obtain a lower-dimensional description of the same flow field.

In order to do this, suppose the state $\mathbf{x}(t)$ lies in $\mathbb{R}^n$, so it is a vector with n real components. Let us consider a (linearly independent) set of modes $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$, where $\mathbf{v}_j \in \mathbb{R}^n$. These modes span an r-dimensional subspace, which we call S, and if $\mathbf{x}(t)$ lies in S, then we may write it as a linear combination of these modes:

$$\mathbf{x}(t) = \sum_{j=1}^{r} a_j(t)\mathbf{v}_j. \tag{1}$$

Vectors: we use the convention that $\mathbf{x} = (x_1, \ldots, x_n)$ denotes a column vector, with the corresponding row vector denoted $\mathbf{x}^T$

We can then reconstruct the flow field $\mathbf{x}(t)$ just by knowing the r values $a_1(t), \ldots, a_r(t)$; if r is small compared with n, then this can result in significant computational savings.

It is often convenient to write this expansion in matrix form. Stacking the vectors $\mathbf{v}_j$ as columns of a matrix $\mathbf{V}$, and defining the vector $\mathbf{a} = (a_1, \ldots, a_r)$, one may equivalently write Equation 1 as

$$\mathbf{x}(t) = \mathbf{V}\mathbf{a}(t). \tag{2}$$

We note that here $\mathbf{V}$ is a rectangular matrix, of dimension $n \times r$, with $n \gg r$.

Now suppose we know that the state evolves in time, with dynamics given by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t)). \tag{3}$$

(For instance, this equation could be the discretized Navier-Stokes equations.) If $\mathbf{x}(t)$ is given by Equation 1 or 2, then it must lie in the subspace S. However, the right-hand side $\mathbf{f}(\mathbf{x}(t))$ might not lie in S. A natural approach to define dynamics on the subspace is to project the right-hand side onto the subspace: The vector in S that is closest to $\mathbf{f}(\mathbf{x})$ is given by the orthogonal projection $\mathbf{P}\mathbf{f}(\mathbf{x})$, with

$$\mathbf{P} = \mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T, \tag{4}$$

where $\mathbf{V}^T$ denotes the transpose of $\mathbf{V}$. If the modes $\mathbf{v}_j$ are orthonormal, then $\mathbf{V}^T\mathbf{V}$ is the identity, and the orthogonal projection is simply $\mathbf{P} = \mathbf{V}\mathbf{V}^T$.

However, this is not the only way to project the dynamics onto the subspace S: One may also choose a nonorthogonal projection (**Figure 1**). To specify such a projection, we choose another subspace (of the same dimension as S), spanned by $\{\mathbf{w}_1, \ldots, \mathbf{w}_r\}$. Stacking these vectors as columns of a matrix $\mathbf{W}$, we may define a projection

$$\hat{\mathbf{P}} = \mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T. \tag{5}$$

If the sets $\{\mathbf{v}_j\}$ and $\{\mathbf{w}_j\}$ form a bi-orthogonal set (i.e., the inner product $\langle \mathbf{v}_j, \mathbf{w}_k \rangle = \delta_{jk}$), then $\mathbf{W}^T\mathbf{V}$ is the identity, so the projection becomes $\mathbf{P} = \mathbf{V}\mathbf{W}^T$.

Thus, in general, specifying a projection involves two choices: the subspace S, which is determined by the set $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$, and the direction of the projection $\hat{\mathbf{P}}$, which is determined by the set $\{\mathbf{w}_1, \ldots, \mathbf{w}_r\}$. Inserting Equation 2 into the projected dynamics $\dot{\mathbf{x}} = \hat{\mathbf{P}}\mathbf{f}(\mathbf{x})$, assuming the modes are bi-orthogonal, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{a}(t) = \mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{a}(t)). \tag{6}$$

Equation 6 is a reduced-order model of the original dynamics in Equation 3: It consists of r equations that describe the evolution of $\mathbf{a}(t)$, from which we can reconstruct $\mathbf{x}(t) = \mathbf{V}\mathbf{a}(t)$. When $r \ll n$, this can represent a potentially enormous computational savings.

If $\mathbf{W} = \mathbf{V}$ (i.e., the projection is orthogonal), then this procedure is called Galerkin projection; if $\mathbf{W}$ is different from $\mathbf{V}$, then this is called Petrov-Galerkin projection. For more details, readers are referred to standard references, such as Antoulas (2005, section 9.1).

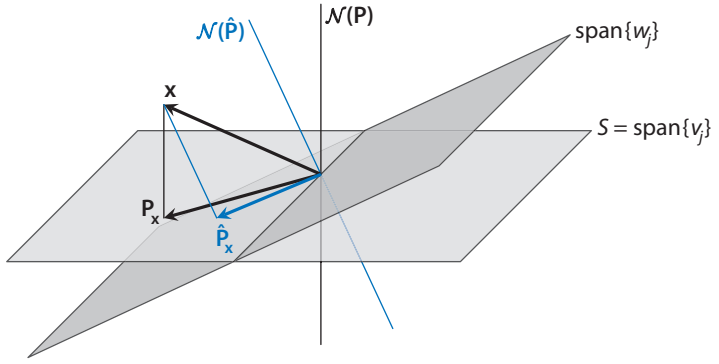

Figure 1

Projection of a vector $\mathbf{x}$ onto a subspace S, spanned by vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$. The direction of the projection $\hat{\mathbf{P}}$ is specified by vectors $\{\mathbf{w}_1, \ldots, \mathbf{w}_r\}$ and differs (in general) from the orthogonal projection $\mathbf{P}$.

2.1. Proper Orthogonal Decomposition

We have not yet specified how to choose the subspace S to be used in constructing the reduced-order model. One popular method that often works well is POD: In this method, one chooses a subspace that optimally describes a given set of data. For instance, one measures the flow field $\mathbf{x}(t)$ at different times t (these measurements are called snapshots) and then finds a subspace of a given dimension that best fits the data. This method goes by many different names in different communities (e.g., principal component analysis, Karhunen-Loève expansion) but was first introduced to the fluid mechanics community by Lumley (1970). Below, we discuss the method of snapshots, as described by Sirovich (1987).

One begins with a data set of snapshots $\mathbf{x}_1, \ldots, \mathbf{x}_m$, elements of $\mathbb{R}^n$, for instance, obtained by sampling a flow field at different times. One then forms a matrix $\mathbf{X}$, whose columns are the snapshots $\mathbf{x}_j$, and computes the reduced singular value decomposition (SVD) of this matrix:

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T = \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^T. \tag{7}$$

Here, r is the rank of $\mathbf{X}$, $\mathbf{U}$ is an $n \times r$ matrix with orthonormal columns $\mathbf{u}_j$, $\mathbf{V}$ is an $m \times r$ matrix with orthonormal columns $\mathbf{v}_j$, and $\boldsymbol{\Sigma}$ is an $r \times r$ diagonal matrix with diagonal entries $\sigma_j \geq 0$, ordered such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r$. Because the columns of $\mathbf{U}$ and $\mathbf{V}$ are orthonormal, these matrices satisfy $\mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{I}$, the $r \times r$ identity matrix.

The vectors $\mathbf{u}_j$ are the POD modes of the data set in $\mathbf{X}$. These modes have the remarkable property that, for any $d \leq r$, the subspace spanned by $\{\mathbf{u}_1, \ldots, \mathbf{u}_d\}$ is the subspace of dimension d that optimally represents the data, in the sense that $\sum_{k=1}^{d} \|\mathbf{P}\mathbf{x}_k - \mathbf{x}_k\|^2$ is minimized, where $\mathbf{P}$ again denotes the orthogonal projection onto the subspace. The singular value σ_j describes the energy contained in POD mode $\mathbf{u}_j$. In particular, letting $\mathbf{P}_j := \mathbf{u}_j\mathbf{u}_j^T$ denote the (rank-1) projection onto POD mode $\mathbf{u}_j$, one can straightforwardly show that

$$\sum_{k=1}^{m} \|\mathbf{P}_j\mathbf{x}_k\|^2 = \sigma_j^2, \qquad j = 1, \ldots, r,$$

which is thus the total energy contained in POD mode $\mathbf{u}_j$ for the data given in $\mathbf{X}$ (where the energy in a vector $\mathbf{x}$ is defined simply as the square of the norm $\|\mathbf{x}\|^2 = \mathbf{x}^T\mathbf{x}$).

It is often more natural to represent the data by an affine space (a shifted subspace), which can be advantageous, for instance, if the mean of the snapshots $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ is large. If this is the case, then one subtracts the mean from the snapshots before assembling them into the matrix $\mathbf{X}$.

Because the POD modes form a subspace that optimally represents the data, one might think that this is the end of the story: One now has a method for choosing the subspace S, so one can then use the reduced-order model from Equation 6, with $\mathbf{V} = \mathbf{W} = [\mathbf{u}_1 \ \cdots \ \mathbf{u}_d]$. In fact, this procedure does work well for some problems. However, it is often unreliable, and as shown below, it can result in very poor low-dimensional models, even when the POD modes capture nearly all the energy in a data set of interest. Although the subspace spanned by the POD modes is optimal for describing a given data set, it is usually not optimal for Galerkin projection, as in Equation 6. In fact, certain low-energy states can have a large influence on the dynamics yet not contribute substantively to low-order POD modes.

Many examples have shown that POD performs particularly poorly for non-normal systems with large transient growth, as often arise in shear flows (Chomaz 2005). For these systems, the methods described in the following section typically perform much better than POD.

Non-normal systems: a linear system of the form of Equation 8 is called normal if the matrix $\mathbf{A}$ commutes with its transpose: $\mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A}$; otherwise, it is called non-normal; eigenvectors of normal matrices may be chosen to be orthogonal

3. LINEAR MODELS

In this section, we consider linear models, which can closely approximate the dynamics of fluids in certain regimes. For these systems, techniques for model reduction are well developed. In particular, we consider linear systems with a vector of inputs, denoted $\mathbf{u}(t)$, and outputs, denoted $\mathbf{y}(t)$. The inputs describe external disturbances or, in the setting of flow control, any actuators we can use to control the flow. The outputs describe the quantities we are interested in representing with a reduced-order model; in a flow control setting, the outputs are typically the sensor measurements. A linear input-output system then has the form

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \end{aligned} \tag{8}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is the state vector, as considered above. The goal in determining a reduced-order model is to obtain an alternative linear model that represents nearly the same dependence of $\mathbf{y}(t)$ on $\mathbf{u}(t)$, but with a state vector with many fewer components than $\mathbf{x}(t)$.

Many techniques are available for the model reduction of linear systems, for instance, as discussed by Antoulas (2005). Here, we focus on methods that involve projection onto a subspace, as described in Section 2, because this approach has been successfully used to describe flow mechanics and because it generalizes to nonlinear systems (discussed in Section 4).

3.1. Balanced Truncation

Balanced truncation is a model reduction procedure based on a trade-off between controllability and observability, concepts defined below. Introduced by Moore (1981), the method is widely used in control theory and has a priori error bounds that are close to the minimum error achievable by any reduced-order model (see, e.g., Dullerud & Paganini 1999, section 4.6).

Controllability refers to the ability of the input $\mathbf{u}$ to control the evolution of the state $\mathbf{x}$. If the state $\mathbf{x}(t)$ can be driven from zero (e.g., quiescent flow, or no perturbation from a base flow) to some nonzero state $\mathbf{x}_0$ with an input $\mathbf{u}$ that is smaller than the input required to drive it to a different state $\hat{\mathbf{x}}_0$, then $\mathbf{x}_0$ is more controllable than $\hat{\mathbf{x}}_0$. For model reduction, it seems intuitively plausible to ignore states that are the least controllable, as these will not be excited (much) by disturbances or actuator inputs.

However, there is another aspect that is important to consider: Observability measures the effect a given initial state has on future outputs. This characteristic is important given that determining the evolution of the future outputs and ultimately controlling them are typically the objectives of a reduced-order model. If a nonzero initial state $\mathbf{x}_0$ results in a larger output $\mathbf{y}(t)$ than a different initial state $\hat{\mathbf{x}}_0$, then $\mathbf{x}_0$ is more observable than $\hat{\mathbf{x}}_0$. For model reduction, it then makes intuitive sense to ignore the least observable states, as these have little effect on the outputs $\mathbf{y}(t)$ that we wish to represent in our model.

The difficulty is that these two characteristics typically do not coincide: There are often states that are weakly controllable though strongly observable, or vice versa, and in this case, it is not clear that any of these states can be ignored. The approach taken in balanced truncation is to transform to a coordinate system in which the controllability and observability properties are balanced: That is, the states that are the most controllable are also the most observable. In these coordinates, then, one simply truncates the states that are least controllable/observable.

Controllability and observability are quantified by defining the following $n \times n$ matrices (where n is the full state dimension), called controllability and observability Gramians:

$$\mathbf{W}_c = \int_0^\infty e^{\mathbf{A}t}\mathbf{B}\mathbf{B}^T e^{\mathbf{A}^T t}\,\mathrm{d}t, \qquad \mathbf{W}_o = \int_0^\infty e^{\mathbf{A}^T t}\mathbf{C}^T\mathbf{C}e^{\mathbf{A}t}\,\mathrm{d}t. \tag{9}$$

These integrals are defined as long as the system is stable (i.e., all eigenvalues of $\mathbf{A}$ have a negative real part) and are usually evaluated by solving the Lyapunov equations

$$\mathbf{A}\mathbf{W}_c + \mathbf{W}_c\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = 0, \qquad \mathbf{A}^T\mathbf{W}_o + \mathbf{W}_o\mathbf{A} + \mathbf{C}^T\mathbf{C} = 0. \tag{10}$$

[To see that solutions of Equation 10 are given by Equation 9, let $\mathbf{Z}(t)$ be one of the integrands in Equation 9, integrate $\mathrm{d}\mathbf{Z}/\mathrm{d}t$ from zero to infinity, and use the fundamental theorem of calculus.] Controllability of a state $\mathbf{x}$ is then quantified by making the following (nontrivial) observation: If $\mathbf{u}(t)$ is the optimal (minimum-norm) input that drives the state from zero to state $\mathbf{x}$ (given infinite time), then $\|\mathbf{u}\|^2 = \mathbf{x}^T\mathbf{W}_c^{-1}\mathbf{x}$ (see, e.g., Dullerud & Paganini 1999, proposition 4.5). Thus, the scalar $\mathbf{x}^T\mathbf{W}_c\mathbf{x}$ quantifies how controllable a state $\mathbf{x}$ is: If this quantity is large, then a small input is required to drive the state to $\mathbf{x}$, so the state is strongly controllable.

Conversely, observability is quantified by observing from Equations 8 and 9 that, if the system starts at state $\mathbf{x}$ at time $t = 0$, with $\mathbf{u}(\mathbf{t}) = 0$, then the resulting output has norm $\|\mathbf{y}\|^2 = \mathbf{x}^T\mathbf{W}_o\mathbf{x}$. Hence, if this quantity is large, the state $\mathbf{x}$ is strongly observable.

A remarkable fact is that, as long as the system is both controllable and observable (i.e., the Gramians $\mathbf{W}_c$ and $\mathbf{W}_o$ have rank n), there exists a change of coordinates in which $\mathbf{W}_c$ and $\mathbf{W}_o$ are equal, and even diagonal. In balanced truncation, one transforms to these coordinates and then truncates the states that are least controllable and observable.

As described by Laub et al. (1987), this change of coordinates may be computed from the Cholesky factorizations

$$\mathbf{W}_c = \mathbf{L}_c\mathbf{L}_c^T \qquad \text{and} \qquad \mathbf{W}_o = \mathbf{L}_o\mathbf{L}_o^T \tag{11}$$

by computing the SVD

$$\mathbf{L}_o^T\mathbf{L}_c = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{12}$$

from which one determines the balancing transformation $\mathbf{T} = \mathbf{L}_c\mathbf{V}\mathbf{\Sigma}^{-1/2}$, with $\mathbf{T}^{-1} = (\mathbf{L}_o\mathbf{U}\mathbf{\Sigma}^{-1/2})^T$. Defining $\mathbf{x}(t) = \mathbf{T}\mathbf{z}(t)$, then in the transformed coordinates $\mathbf{z}(t)$, we have $\mathbf{W}_c = \mathbf{W}_o = \mathbf{\Sigma}$. The diagonal entries of $\mathbf{\Sigma}$ are called the Hankel singular values: They are real and non-negative, and normally ordered in decreasing order.

Example 1. Consider the following linear system with a single state:

$$\dot{x} = -\frac{1}{2}x + \varepsilon u, \qquad y = \frac{1}{\varepsilon}x.$$

The Gramians are computed from Equation 10 as $\mathbf{W}_c = \varepsilon^2$ and $\mathbf{W}_o = 1/\varepsilon^2$, so if ε is small, we see that state x is weakly controllable, but strongly observable. The balancing transformation is computed as $\mathbf{T} = \varepsilon$, and setting $x = \mathbf{T}z$, the transformed system is

$$\dot{z} = -\frac{1}{2}z + u, \qquad y = x.$$

In these new coordinates, the Gramians are $\mathbf{W}_c = \mathbf{W}_o = 1$, so the realization is balanced.

There is, unfortunately, some inconsistency in the fluid mechanics literature about the calculation of the balancing transformation. For instance, Willcox & Peraire (2002) and Willcox (2007) define the balancing transformation as the matrix of eigenvectors of the product $\mathbf{W}_c\mathbf{W}_o$. The main problem with this definition is that the scaling of the eigenvectors is arbitrary, so although such a transformation can simultaneously diagonalize the Gramians, they will not in general be equal (and this is the whole point of balancing, to make the Gramians equal). In fact, it is straightforward to see that the balancing transformation cannot be computed solely from the product $\mathbf{W}_c\mathbf{W}_o$. For instance, in the preceding example, one obtains the product $\mathbf{W}_c\mathbf{W}_o = 1$, which is independent of ε, so it is impossible to recover the balancing transformation $\mathbf{T} = \varepsilon$ from this product. Intuitively, the difficulty is that the product $\mathbf{W}_c\mathbf{W}_o$ alone does not distinguish between modes that are strongly controllable but weakly observable, and modes that are strongly observable but weakly controllable.

Recall that the integrals in Equation 9 converge only if the system is stable (i.e., the eigenvalues of $\mathbf{A}$ lie in the left half plane). If the system is unstable, one may first decompose the system into an interconnection of stable and unstable components and then perform balanced truncation only on the stable part (Hsu & Hou 1991), or one may consider alternative definitions of the Gramians given in the frequency domain (Zhou et al. 1999). Variants of these approaches have been used in examples by Barbagallo et al. (2009) and Ahuja & Rowley (2010).

3.2. Balanced Proper Orthogonal Decomposition

For systems with very large state dimension, the procedure above is computationally intractable: For instance, if the state dimension is $n = 10^6$, the Gramians are $n \times n$ matrices that are symmetric, and not sparse (even if $\mathbf{A}$ is sparse), so even storing such a matrix in double precision would require 3.6 terabytes of memory. Balanced POD provides an approximation of balanced truncation, using an algorithm that is tractable for high-dimensional systems. It is based on defining empirical Gramians, which approximate $\mathbf{W}_c$ and $\mathbf{W}_o$ from Equation 9 using data from simulations. One then computes an approximate balancing transformation using an SVD as in Equation 12.

To determine the empirical controllability Gramian, one first calculates solutions of

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t), \qquad \mathbf{x}(0) = \mathbf{b}_j, \tag{13}$$

where $\mathbf{b}_j$ are the columns of the matrix $\mathbf{B}$ in Equation 8. Thus, this step requires one simulation for each input (i.e., each component of $\mathbf{u}$, corresponding to the columns $\mathbf{b}_j$). Suppose snapshots are taken from these simulations at time steps Δt apart and arranged as columns of a matrix $\mathbf{X}$. The Gramian $\mathbf{W}_c$ in Equation 9 may then be approximated as $\mathbf{W}_c \approx \mathbf{X}\mathbf{X}^T\,\Delta t$.

Similarly, the observability Gramian may be approximated from solutions of the adjoint system

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{z}(t) = \mathbf{A}^T\mathbf{z}(t), \qquad \mathbf{z}(0) = \mathbf{c}_j, \tag{14}$$

where $\mathbf{c}_j$ are columns of the matrix $\mathbf{C}^T$ in Equation 8. This step then requires one adjoint simulation for each output (each component of $\mathbf{y}$, corresponding to the columns $\mathbf{c}_j$). If these snapshots are arranged as columns of a matrix $\mathbf{Z}$, then the Gramian $\mathbf{W}_o$ may be approximated as $\mathbf{W}_o \approx \mathbf{Z}\mathbf{Z}^T\,\Delta t$.

To compute the balancing transformation, one forms the reduced SVD of the matrix

$$\mathbf{Z}^T\mathbf{X} = \hat{\mathbf{W}}\boldsymbol{\Sigma}\hat{\mathbf{V}}^T \tag{15}$$

(note the similarity with Equation 12). Letting $\hat{\mathbf{v}}_j$ and $\hat{\mathbf{w}}_j$ denote the columns of $\hat{\mathbf{V}}$ and $\hat{\mathbf{W}}$, one then defines the balancing modes $\mathbf{v}_j$ and adjoint modes $\mathbf{w}_j$ by

$$\mathbf{v}_j = \sigma_j^{-1/2}\mathbf{X}\hat{\mathbf{v}}_j, \qquad \mathbf{w}_j = \sigma_j^{-1/2}\mathbf{Z}\hat{\mathbf{w}}_j. \tag{16}$$

These modes form a bi-orthogonal set ($\mathbf{w}_k^T\mathbf{v}_j = \delta_{jk}$), so we form a reduced-order model as in Section 2. For a model with r states, we arrange the modes $\mathbf{v}_1, \ldots, \mathbf{v}_r$ as columns of a matrix $\mathbf{V}$ and the adjoint modes $\mathbf{w}_1, \ldots, \mathbf{w}_r$ as columns of a matrix $\mathbf{W}$. We then write $\mathbf{x}(t) = \mathbf{V}\mathbf{a}(t)$, where $\mathbf{a} = (a_1, \ldots, a_r)$, and project as in Equation 6, to obtain the reduced-order model

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{a}(t) &= \mathbf{W}^T\mathbf{A}\mathbf{V}\mathbf{a}(t) + \mathbf{W}^T\mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{V}\mathbf{a}(t) + \mathbf{D}\mathbf{u}(t). \end{aligned} \tag{17}$$

The method discussed above was originally presented by Rowley (2005), and the idea of empirical Gramians was provided by Lall et al. (1999, 2002). The above method is sometimes confused with a related method given by Willcox & Peraire (2002): The methods are indeed closely related, as both involve calculating empirical Gramians, but there are important differences. The method of Willcox & Peraire (2002) involves calculating POD modes of the individual Gramians $\mathbf{W}_c$ and $\mathbf{W}_o$, and determining separate low-rank approximations for each. Thus, the procedure truncates the least controllable and least observable modes before balancing, so if there are modes that are weakly controllable yet strongly observable, they will be truncated prematurely. One would therefore expect the procedure of Willcox & Peraire (2002) to perform poorly if there are modes that are weakly controllable and strongly observable (and vice versa)—but these are precisely the systems for which one would like to use balanced truncation in the first place. Indeed, the separate reduction of Gramians as proposed by Willcox & Peraire (2002) can result in significantly less accurate reduced-order models than the above approach, as shown in an example in Rowley (2005, section 4.2). Several examples of the application of balanced POD to fluid flows are listed in **Table 1**.

3.3. Eigensystem Realization Algorithm

The ERA is a technique that seeks to extract a linear state-space model from impulse response data. The ERA was first conceived by Juang & Pappa (1985) to analyze the structural dynamics of spacecraft, but it shares close similarities with a number of previously proposed techniques (e.g., Ho & Kalman 1965, Kung 1978). More generally, the ERA is an example of a subspace system identification method. More details about this class of methods and their potential applications can be found in Verhaegen & Dewilde (1992), Viberg (1995), and Qin (2006).

Unlike POD and BPOD models, the ERA identifies a model directly from data and does not require any a priori knowledge of the governing equations. Because the data are collected in

Table 1 Examples of the use of balanced proper orthogonal decomposition (BPOD) for fluid systems

Study	Applications, findings, and variants
Ilak & Rowley (2008)	Three-dimensional channel flow; BPOD models outperform POD models due to dynamically important, low-energy POD modes, and nonorthogonal projection in BPOD
Barbagallo et al. (2009)	Control of cavity flow; balancing modes are more efficient than POD modes, but can be less robust
Dergham et al. (2011)	Flow over a backward-facing step and square cavity; modified version of BPOD that uses harmonically forced data
Semeraro et al. (2011, 2013)	Modeling and control of Tollmien-Schlichting waves in a boundary layer to delay transition to turbulence, for both linearized and nonlinear systems
Flinois et al. (2015)	Linearized flow past a cylinder; direct application of BPOD to unstable systems

discrete time, we identify a discrete-time model of the form

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}_d \mathbf{x}_k + \mathbf{B}_d \mathbf{u}_k, \\ \mathbf{y}_k &= \mathbf{C}_d \mathbf{x}_k + \mathbf{D}_d \mathbf{u}_k, \end{aligned} \tag{18}$$

which captures the impulse response data. Note that an impulse response from Equation 18 will be of the form

$$\left\{ \mathbf{D}_d, \mathbf{B}_d \mathbf{C}_d, \mathbf{B}_d \mathbf{A}_d \mathbf{C}_d, \mathbf{B}_d \mathbf{A}_d^2 \mathbf{C}_d, \ldots \right\}. \tag{19}$$

In practice, we may choose to sample less frequently than every step of the dynamics, so below we assume we sample a pair of snapshots every P steps. With such data, the ERA algorithm proceeds as follows:

1. Collect output data from an impulse response of the form $\{\mathbf{y}_0, \mathbf{y}_P, \mathbf{y}_{2P}, \ldots, \mathbf{y}_{mP}\}$ and $\{\mathbf{y}_1, \mathbf{y}_{P+1}, \mathbf{y}_{2P+1}, \ldots, \mathbf{y}_{mP+1}\}$.
2. Assemble the block Hankel matrices

$$\mathbf{H} = \begin{bmatrix} \mathbf{y}_0 & \mathbf{y}_P & \mathbf{y}_{2P} & \cdots & \mathbf{y}_{m_c} \\ \mathbf{y}_P & \mathbf{y}_{2P} & \mathbf{y}_{3P} & \cdots & \mathbf{y}_{(m_c+1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_{m_o P} & \mathbf{y}_{(m_o+1)P} & \mathbf{y}_{(m_o+2)P} & \cdots & \mathbf{y}_{(m_o+m_c)P} \end{bmatrix},$$

$$\mathbf{H}^{\#} = \begin{bmatrix} \mathbf{y}_1 & \mathbf{y}_{P+1} & \mathbf{y}_{2P+1} & \cdots & \mathbf{y}_{m_c P+1} \\ \mathbf{y}_{P+1} & \mathbf{y}_{2P+1} & \mathbf{y}_{3P+1} & \cdots & \mathbf{y}_{(m_c+1)P+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{y}_{m_o P+1} & \mathbf{y}_{(m_o+1)P+1} & \mathbf{y}_{(m_o+2)P+1} & \cdots & \mathbf{y}_{(m_o+m_c)P+1} \end{bmatrix},$$

   where m_c and m_o are chosen such that $m_c + m_o \leq m$.
3. Compute the (reduced) SVD $\mathbf{H} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.
4. Truncate the SVD by considering only the first r columns of $\mathbf{U}$ and $\mathbf{V}$, and the first r rows and columns of $\mathbf{\Sigma}$ (with the singular values ordered by size), to obtain $\mathbf{U}_r$, $\mathbf{\Sigma}_r$, and $\mathbf{V}_r$, where r is the desired model order.

Table 2 Examples of the use of the eigensystem realization algorithm for fluid systems

Study	Applications, findings, and variants
Cabell et al. (2006)	Modeling and control of cavity tones (experiment) with a synthetic jet actuator
Ahuja & Rowley (2010)	Modeling and control of unstable flow over an inclined plate (DNS); model obtained by projecting out and treating separately the unstable modes; used Kalman filter to estimate full state
Illingworth et al. (2011, 2012)	Modeling and control of cavity resonances (compressible, DNS), and combustion oscillations (experiment); closed-loop stability achieved over a range of operating conditions
Brunton et al. (2013, 2014)	Modeling (and control) of a pitching airfoil at low Reynolds number (DNS and experiment); stability derivatives subtracted before performing ERA; observer/Kalman filter identification utilized to experimentally determine Markov parameters for ERA
Belson et al. (2013)	Model and control instabilities in a Blasius boundary layer (DNS); appropriate choice of sensor and actuator locations allows for effective and robust control
Illingworth (2016)	Modeling and control of unstable flow past a cylinder (DNS); ERA applied to an unstable system that has been feedback stabilized, allowing for more effective controllers to be designed
Flinois & Morgans (2016)	Flow over a D-shaped body (DNS); ERA applied directly to unstable system

Abbreviations: DNS, direct numerical simulation; ERA, eigensystem realization algorithm.

5. The matrices of the reduced-order model of a system with p inputs and q outputs are given by

$$\begin{aligned} \mathbf{A}_r &= \boldsymbol{\Sigma}_r^{-1/2}\mathbf{U}_r^T\mathbf{H}^{\#}\mathbf{V}_r\boldsymbol{\Sigma}_r^{1/2}, \\ \mathbf{B}_r &= \text{the first } p \text{ columns of } \boldsymbol{\Sigma}_r^{1/2}\mathbf{V}_r^T, \\ \mathbf{C}_r &= \text{the first } q \text{ rows of } \mathbf{U}_r\boldsymbol{\Sigma}_r^{1/2}, \\ \mathbf{D}_r &= \mathbf{y}_0. \end{aligned} \tag{20}$$

Note that when $p = 1$, the data pairs in step 1 can just be taken from an impulse response sequence with its last and first entries removed. This more general formulation allows for the skipping of data when assembling $\mathbf{H}$, which can reduce computational costs, while still allowing data to be used across a large total time window.

In general, input-output data might not be available in the form of an impulse response, in which case other more general subspace methods may be used (e.g., Verhaegen & Dewilde 1992). Another approach is to use a technique such as observer/Kalman filter identification (Juang et al. 1991) to compute an impulse response from input-output data with random inputs, before applying the ERA.

In typical fluids systems, the data collected might be of very large dimension, which poses a problem for the computational feasibility of the ERA given that the dimension of the block Hankel matrices can become prohibitively large. One method to mitigate this issue is described by Ma et al. (2011), in which POD can first be performed on the outputs (which could be the velocity field of a fluid flow), and then some relatively small number of POD coefficients can be used as outputs for the ERA. Ma et al. (2011) also showed that, assuming a sufficient quantity of exact data, reduced-order models obtained via the ERA should be identical to those identified using BPOD. One advantage of the ERA is that, unlike balanced truncation (and BPOD), no modifications are needed to identify an unstable system. For instance, Illingworth et al. (2014) determined a model of the unstable equilibrium in the flow past a cylinder. Conversely, a balanced truncation of a stable system is guaranteed to be stable (Pernebo & Silverman 1982), but no such guarantee exists for ERA models determined from data. Some successful demonstrations of the use of ERA to model fluid flows are listed in **Table 2**.

3.4. Dynamic Mode Decomposition

DMD is a technique for determining, directly from data, both a low-dimensional subspace (i.e., the space S in **Figure 1**) and a description of the dynamics on that subspace. In particular, each mode can be associated with a single frequency and growth/decay rate, so if $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ are DMD modes, the decomposition given in Equation 1 becomes

Pseudoinverse: if $\mathbf{Y} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is the reduced SVD of $\mathbf{Y}$, then the pseudoinverse is $\mathbf{Y}^+ = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$

$$\mathbf{x}(t) = \sum_{j=1}^{r} c_j e^{\lambda_j t}\mathbf{v}_j, \tag{21}$$

where $\lambda_j \in \mathbb{C}$ determines the growth/decay rate and oscillation frequency of the DMD mode $\mathbf{v}_j$. Because data are usually available at discrete times, DMD is usually considered in the discrete-time setting: If Δt is the time step and k is an integer, with $t = k\Delta t$ and $\mu_j = e^{\lambda_j \Delta t}$, the above expansion becomes

$$\mathbf{x}(k) = \sum_{j=1}^{r} c_j \mu_j^k \mathbf{v}_j. \tag{22}$$

DMD was introduced by Schmid & Sesterhenn (2008) and Schmid (2010). In this original formulation, one collects snapshots of data $\{\mathbf{y}_1, \ldots, \mathbf{y}_{m+1}\}$, equally spaced in time, and assumes a linear relation between them:

$$\mathbf{y}_{k+1} = \mathbf{A}\mathbf{y}_k, \qquad k = 1, \ldots, m, \tag{23}$$

where the matrix $\mathbf{A}$ is not known explicitly. The DMD modes $\mathbf{v}_j$ and eigenvalues μ_j are then determined as approximations of the eigenvectors and eigenvalues of $\mathbf{A}$ using a variant of an Arnoldi algorithm (in particular, considering the Krylov subspace spanned by the known vectors $\{\mathbf{y}_1, \mathbf{A}\mathbf{y}_1, \mathbf{A}^2\mathbf{y}_1, \ldots, \mathbf{A}^m\mathbf{y}_1\}$).

We present a slightly more general formulation of DMD, as given by Tu et al. (2014b), in which we consider pairs of snapshots $(\mathbf{y}_k, \mathbf{y}_k^\#)$, for $k = 1, \ldots, m$, and assume there is a linear relation

$$\mathbf{y}_k^\# = \mathbf{A}\mathbf{y}_k, \qquad k = 1, \ldots, m. \tag{24}$$

Clearly, the former is a special case of this (with $\mathbf{y}_{k+1} = \mathbf{y}_k^\#$), but this generalization allows one to consider multiple initial conditions and time series, instead of just a single sequential time series. We then form a matrix $\mathbf{Y}$, whose columns are the snapshots $\mathbf{y}_k$, and a corresponding matrix $\mathbf{Y}^\#$, whose columns are $\mathbf{y}_k^\#$. The DMD modes and eigenvalues are then defined to be eigenvectors and eigenvalues of the matrix

$$\mathbf{A} = \mathbf{Y}^\#\mathbf{Y}^+, \tag{25}$$

where $\mathbf{Y}^+$ denotes the pseudoinverse of $\mathbf{Y}$ (Trefethen & Bau 1997, lecture 11). As shown by Tu et al. (2014b), the eigenvalues obtained from this definition are identical to those obtained from the original definition of DMD, and the eigenvectors are nearly identical, as discussed below.

Under mild conditions on the data (e.g., if the snapshots $\{\mathbf{y}_k\}$ are linearly independent), Equation 24 holds exactly. If the problem is overconstrained, so that there is no matrix $\mathbf{A}$ such that Equation 24 is exactly satisfied, then Equation 25 gives the least-squares solution that minimizes $\sum_{k=1}^{m} \|\mathbf{y}_k^\# - \mathbf{A}\mathbf{y}_k\|^2$. In this sense, DMD is related to least-squares approximation methods (Gugercin & Antoulas 2006) and indeed is a close cousin of Prony's method (de Prony 1795; see Hildebrand 1974, section 9.4), which is probably the first model reduction method ever proposed. (For a detailed discussion of Prony's method and its many modern variants, readers are referred to Hokanson 2013.)

In order to actually compute the DMD modes and eigenvalues, one does not need to form the matrix $\mathbf{A}$ explicitly. Instead, it is usually more practical to form a low-rank approximation of it, as follows:

1. Compute the reduced SVD of $\mathbf{Y} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.
2. (Optional) Truncate the SVD by only considering the first r columns of $\mathbf{U}$ and $\mathbf{V}$, and the first r rows and columns of $\mathbf{\Sigma}$, to obtain $\mathbf{U}_r$, $\mathbf{\Sigma}_r$, and $\mathbf{V}_r$.
3. Let $\tilde{\mathbf{A}} = \mathbf{U}_r^T \mathbf{A} \mathbf{U}_r = \mathbf{U}_r^T \mathbf{Y}^{\#} \mathbf{V}_r \mathbf{\Sigma}_r^{-1}$, an $r \times r$ matrix.
4. Find the (discrete-time) eigenvalues μ_j and eigenvectors $\tilde{\mathbf{v}}_j$ of $\tilde{\mathbf{A}}$, with $\tilde{\mathbf{A}}\tilde{\mathbf{v}}_j = \mu_j \tilde{\mathbf{v}}_j$. Every nonzero μ_j is a DMD eigenvalue.
5. The DMD mode corresponding to μ_j is given by $\mathbf{v}_j = \mu_j^{-1} \mathbf{Y}^{\#} \mathbf{V}_r \mathbf{\Sigma}_r^{-1} \tilde{\mathbf{v}}_j$.
6. Letting $\mathbf{P} = \mathbf{U}_r \mathbf{U}_r^T$ denote the projection onto the range of $\mathbf{U}_r$, the projected DMD mode $\mathbf{P}\mathbf{v}_j$ is given by $\mathbf{P}\mathbf{v}_j = \mathbf{U}_r \tilde{\mathbf{v}}_j$.

It is straightforward to show that the resulting DMD modes $\mathbf{v}_j$ are eigenvectors of the matrix $\mathbf{A}$ in Equation 25 (with $\mathbf{Y}^+$ given by $\mathbf{V}_r \mathbf{\Sigma}_r^{-1} \mathbf{U}_r^T$). It is common (e.g., Schmid 2010) to define the DMD modes as simply $\mathbf{U}_r \tilde{\mathbf{v}}_j$; we refer to these as projected DMD modes, as they are projections of the eigenvectors $\mathbf{v}_j$ onto the range of $\mathbf{U}_r$. In practice, $\mathbf{v}_j$ and $\mathbf{P}\mathbf{v}_j$ are usually nearly identical because the range of $\mathbf{Y}^{\#}$ is usually close to the range of $\mathbf{Y}$.

For very large data sets, the algorithm presented here may be unfeasible given that it requires all data to be loaded into memory at once. Hemati et al. (2014) provided a modified DMD algorithm, in which data are loaded and processed in a streaming manner. A parallelized algorithm, in which only two snapshots need to be loaded into memory on a given node at once, was presented by Belson et al. (2014).

There are algorithmic similarities between DMD and other data-driven methods for system identification. In particular, the optional step 2 of truncating small singular values frames DMD as a linear model reduction technique. More precisely, Tu et al. (2014b) showed that DMD is equivalent to the ERA, in the sense that $\tilde{\mathbf{A}}$ from DMD is related to $\mathbf{A}_r$ from the ERA by a similarity transform, if the data matrices $\mathbf{Y}$ and $\mathbf{Y}^{\#}$ from DMD are taken to be the block Hankel matrices $\mathbf{H}$ and $\mathbf{H}^{\#}$ from the ERA. As mentioned in Section 3.3, the ERA in turn can be shown to give reduced-order models that are identical to those produced by BPOD (Ma et al. 2011). In this sense, for the case of linear systems, all data-driven system identification algorithms discussed here are closely related in the dynamics that they identify.

Taking the SVD of the data matrix $\mathbf{X}$ is almost equivalent to performing POD on the data, with the columns of $\mathbf{U}$ being the orthogonal spatial modes (see Equation 7). (Note that to make the modes orthonormal with respect to the usual spatial inner product, some additional scaling may be required.) One important difference between POD and the decomposition in step 1 of the DMD algorithm is that the mean is not first subtracted for DMD. This is important to note, as it can be shown that subtracting the mean before applying DMD gives results identical to a temporal discrete Fourier transform (Chen et al. 2011), if Equation 23 is satisfied exactly (e.g., if the first m snapshots are linearly independent). This equivalence is generally undesirable, given that the DMD eigenvalues become roots of unity (in particular, complex solutions of $\mu^{m+1} = 1$) and no longer contain useful information about the data themselves.

There have been several modifications of DMD that have been proposed since its introduction, which aim to address certain shortcomings, or make it more suitable to certain applications. Many modal decomposition and system identification techniques have the common goal of obtaining a reduced-order representation of the dynamics. For the case of DMD, one might achieve this by selecting a subset of DMD mode/eigenvalue pairs that represent the dynamics of interest. To this end, there are several ways in which the magnitudes of DMD modes can be compared

(Tu et al. 2014b). Alternatively, one can seek a sparse representation of the dynamics, in the form of a small number of modes with nonzero amplitude, by penalizing the ℓ_1 norm of the vector of mode amplitudes (Jovanović et al. 2014). By identifying the dynamics after first projecting onto a energetically optimal subspace (i.e., when applying step 2 of the given DMD algorithm), one also is restricting the number of modes that will be identified. Such a projection is not necessarily dynamically optimal (for reasons discussed in Section 3.1), and it is possible to seek an optimal subspace for projection along with the identification of the dynamics (Goulart et al. 2012, Wynn et al. 2013). Chen et al. (2011) also provided an algorithm that gives a decomposition of a given dimension that minimizes the error in the predicted trajectory. Proctor et al. (2014) proposed an algorithm for using DMD on data from linear systems with (known) inputs, as in Equation 8. One issue that can impede the extraction of accurate dynamical quantities and models from data is the presence of noise, which is particularly an issue for experimental data. Although it has been observed empirically that DMD can be sensitive to noisy data (Duke et al. 2012b, Pan et al. 2015), recent work has explained this sensitivity and presented modifications to the algorithm to make its outputs more robust to noisy data (Hemati et al. 2015, Dawson et al. 2016). Some examples of the successful application of DMD to fluid flows are listed in **Table 3**.

Optimal transient growth: for a linear system, the optimal transient growth mode for a given time horizon τ can be found from the first column of $\mathbf{V}$ in the SVD $e^{\mathbf{A}\tau} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$

3.5. Example: Linearized Channel Flow

To illustrate the methods discussed above, we demonstrate and compare them on a linearized flow in a plane channel (plane Poiseuille flow). This flow exhibits the large transient growth seen in many shear flows (Schmid 2007) and illustrates the difficulties POD methods have with such systems. The three-dimensional incompressible Navier-Stokes equations are linearized about an equilibrium velocity profile $U(y) = 1-y^2$, in a domain that is periodic in the spanwise (z) direction, with period 2π, considering perturbations that are streamwise constant. (Velocities are normalized by the centerline velocity, and lengths by the channel half width.) The linearized equations in terms of the wall-normal velocity $v(y,z)$ and wall-normal vorticity $\eta(y,z)$ are as follows (e.g., Schmid & Henningson 2001, section 3.1):

$$\frac{\partial}{\partial t}\begin{bmatrix} -\Delta & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} v \\ \eta \end{bmatrix} = \begin{bmatrix} -\Delta^2/Re & 0 \\ -U'\partial_z & \Delta/Re \end{bmatrix}\begin{bmatrix} v \\ \eta \end{bmatrix}, \qquad \begin{matrix} v(\pm 1,z) = \partial_y v(\pm 1,z) = 0, \\ \eta(\pm 1,z) = 0, \end{matrix} \tag{26}$$

where $\Delta = \partial_y^2 + \partial_z^2$ denotes the Laplacian and Re is the Reynolds number. For numerical approximation, we use a spectral method with 16 Fourier modes in z and 32 Chebyshev polynomials in y, using the formulation described by Trefethen (2000) to obtain a system of the form $d\mathbf{x}/dt = \mathbf{A}\mathbf{x}$, where the state vector $\mathbf{x}$ has $2 \times 31 \times 16 = 992$ components.

In simulating this system, we choose an initial condition that maximizes the transient growth of energy over a given time. The initial condition giving the largest transient growth across all time horizons is shown in **Figure 2*a***, which corresponds to a time horizon $\tau = 32.9$. Note that starting this system from this initial condition is equivalent to considering the impulse response of a linear state space system (i.e., a system in the form given by Equation 8). We collect data (wall-normal velocity and vorticity fields) for a total of 1,000 time units, with snapshots collected every 0.05 time units.

We apply the various model reduction techniques discussed in this section to these data, with **Figure 2*b*** showing the H_∞ error norms of the resulting models of various orders. This system is sufficiently small that balanced truncation can be performed directly. We observe that BPOD performs similarly to exact balanced truncation. By contrast, projecting the governing

Table 3 Examples of the application of dynamic mode decomposition to fluid systems

Study	Applications, findings, and variants
Rowley et al. (2009)	Jet in crossflow (DNS)
Schmid (2010)	Plane Poiseuille flow; linearized two-dimensional flow over a square cavity; wake of a flexible membrane (PIV); jet between two cylinders (PIV)
Chen et al. (2011)	Transitional cylinder flow (DNS)
Nastase et al. (2011)	Lobed jet from three-dimensional diffusers (experiment)
Pan et al. (2011)	Wake of a NACA 0015 airfoil with Gurney flap (PIV)
Schmid et al. (2011)	Schlieren snapshots of a helium jet; PIV snapshots of an acoustically forced jet
Schmid (2011)	Passive tracer in flame simulation and axisymmetric water jet experiment
Seena & Sung (2011)	Turbulent cavity flow (DNS)
Duke et al. (2012a)	Annular liquid sheet instabilities (experiment)
Grilli et al. (2012)	Shockwave turbulent boundary layer interaction (DNS)
Jardin & Bury (2012)	Flow past a cylinder, with forcing near the mean separation point (DNS)
Lee et al. (2012)	Developing turbulent boundary layers over roughened walls (DNS)
Muld et al. (2012a)	Wake of high-speed train model (detached eddy simulation)
Muld et al. (2012b)	Flow over a surface-mounted cube (detached eddy simulation)
Schmid et al. (2012)	Transitional water jet with tomographic PIV
Semeraro et al. (2012)	Confined turbulent jet with coflow (PIV)
Bagheri (2013)	Cylinder wake approaching limit cycle (DNS)
Ghommem et al. (2013)	Flows in high-contrast porous media (DNS)
He et al. (2013)	Boundary layer and cylinder configuration (experiment)
Meslem et al. (2013)	Impinging circular jet (PIV)
Motheau et al. (2013)	Gas turbine combustion instability (LES)
Sarkar et al. (2013)	Nanofluid flow past a square cylinder (DNS)
Tu et al. (2013, 2014b)	Wake of a cylinder (DNS) and finite-thickness flat plate (PIV)
Wynn et al. (2013)	Flow over a backward-facing step (PIV) using optimal mode decomposition
Carlsson et al. (2014)	Flow-flame interactions (LES)
Gómez et al. (2014)	Turbulent pipe flow (DNS)
Jovanović et al. (2014)	Sparsity-promoting DMD applied to two-dimensional plane Poiseuille flow; screeching supersonic jet (LES); jet between two cylinders (PIV)
Ma & Liu (2014)	Flow over high angle of attack, slender bodies (DNS)
Markovich et al. (2014)	Swirling, confined flames and jets (PIV)
Tu et al. (2014a)	Flow past a cylinder (PIV), temporally sparse data
Sarmast et al. (2014)	Wind turbine wakes (LES)
Sayadi et al. (2014)	Flat plate boundary layer transition to turbulence (DNS and LES)
Subbareddy et al. (2014)	Transition of Mach 6 boundary layer with roughness element (DNS)
Thompson et al. (2014)	Flow past elliptic cylinders (DNS)
Tissot et al. (2014)	Flow past a cylinder (experiment), mode extraction for reduced-order modeling
Dunne & McKeon (2015)	Dynamic stall on a pitching and surging airfoil (PIV)
Kramer et al. (2015)	Flow in a two-dimensional differential heated cavity (DNS), for identification of flow regimes
Roy et al. (2015)	Reacting flows behind bluff bodies (experiment)
Sayadi et al. (2015)	Thermo-acoustic instabilities in ducted and bifurcating flames (numerical and experimental), using parameterized DMD

Abbreviations: DMD, dynamic mode decomposition; DNS, direct numerical simulation; LES, large-eddy simulation; PIV, particle image velocimetry.

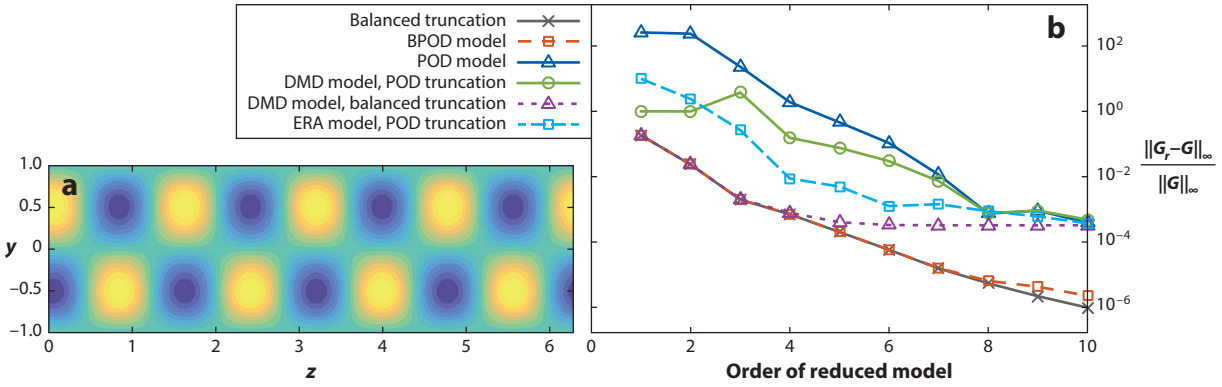

Figure 2

(*a*) Vorticity contours of optimal transient growth mode for $R = 1{,}000$ and a time horizon $\tau = 32.9$, which is used as an initial condition/input. (*b*) H_∞ errors of models of orders 1–10 in comparison to the full system, identified from a variety of model reduction algorithms described in Section 3. Abbreviations: BPOD, balanced proper orthogonal decomposition; DMD, dynamic mode decomposition; ERA, eigensystem realization algorithm; POD, proper orthogonal decomposition.

equations onto POD modes gives models that are significantly less accurate across all model orders.

Both POD and BPOD projection models inherently require knowledge of the underlying governing equations. By contrast, DMD and the ERA obtain models directly from the data themselves. When using such methods to obtain a reduced-order model, there are several choices that can be made. With DMD, perhaps the most naïve method to obtain a reduced-order model is to simply truncate the SVD that is computed within the DMD algorithm to the desired order (which is equivalent to keeping only high-energy POD modes). This is step 2 in the DMD algorithm presented in Section 3.4. **Figure 2** shows that this approach (i.e., DMD with POD truncation) does not work very well, especially for small model orders. For the ERA, direct truncation (i.e., applying step 4 in the ERA algorithm given in Section 3.3 with r taken to be the model order) gives similar results, although the stacking of data into a Hankel matrix before truncating generally results in more accurate models than DMD gives (which would be equivalent to the ERA with only one set of outputs per column of the Hankel matrix). Note that ERA models with POD projection use output projection data of the same order as the final model. Increasing the number of outputs to include more POD modes did not improve performance.

An alternative, and better, approach when using DMD and ERA models is to first truncate models to retain a larger number of modes, and then perform a balanced truncation on the resulting model to obtain the final desired model order. This method is sometimes referred to as overspecification. In this way, one can both obtain a balanced model and ensure that the subspace and direction of projection are not discounting important dynamics, without needing to know the full system equations. **Figure 2*b*** shows the results of this approach for DMD (labeled as DMD with balanced truncation), with the ERA producing identical models. We note that the saturation of the error with increasing model order largely results from the conversion of the identified discrete-time model to continuous time for comparison with the true model.

We illustrate the performance of order-three models in more detail in **Figure 3**, in both the time and frequency domain. Again, we see that balanced truncation, BPOD, and DMD or the ERA combined with balanced truncation give very accurate models, whereas models based on POD truncation are quite inaccurate, despite that the first three modes contain 99.65% of the energy of

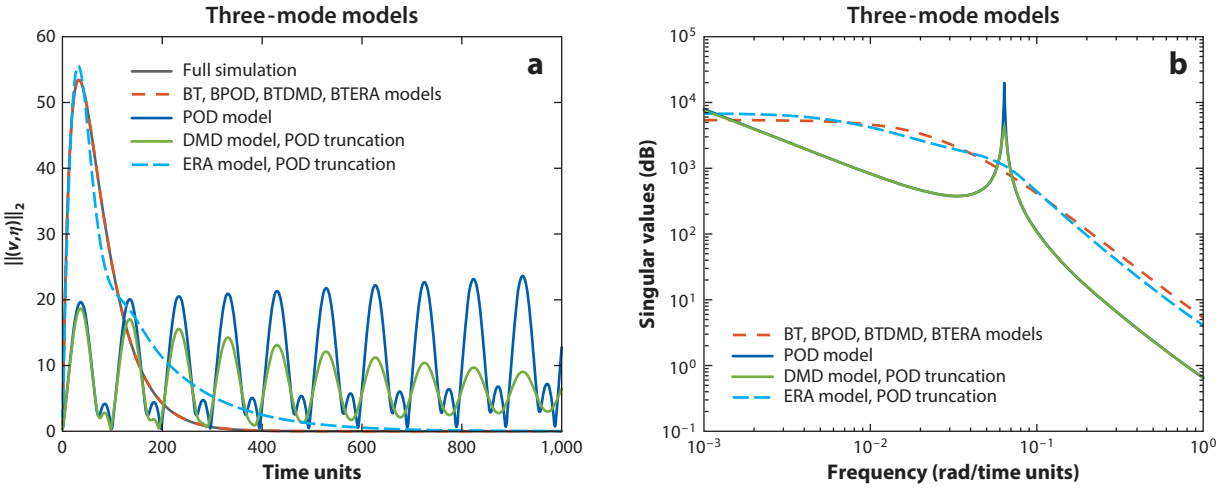

Figure 3

(*a*) Norm of the solution from different reduced-order models using three states. Models using BT, BPOD, and higher-order DMD or ERA models reduced to three states with BT (BTDMD and BTERA) are indistinguishable from the exact solution and are shown on one line. (*b*) Singular values of the frequency response of third-order models identified using various techniques. Abbreviations: BPOD, balanced proper orthogonal decomposition; BT, balanced truncation; BTDMD, balanced truncation dynamic mode decomposition; BTERA, balanced truncation eigensystem realization algorithm; DMD, dynamic mode decomposition; ERA, eigensystem realization algorithm; POD, proper orthogonal decomposition.

the data. This emphasizes the importance of accounting for both observability and controllability when performing model reduction.

For this relatively small system, balanced truncation can be computed directly, allowing for comparison with the other model reduction methods; as the state dimension increases, however, balanced truncation becomes intractable, as mentioned in Section 3.2. For the other methods discussed here, the major computational expense (aside from simulation time) is in taking the relevant SVD of the data as described in the algorithms. For example, the computational cost of the BPOD and DMD algorithms is dictated by the size of the matrices $\mathbf{Z}^T\mathbf{X}$ and $\mathbf{Y}$, respectively. Techniques such as BPOD and ERA have been successfully applied to fluid systems with tens of inputs and outputs, and state sizes in the millions (Semeraro et al. 2013). From a computational perspective, there is no reason why such algorithms could not be applied to larger, more complex systems.

4. NONLINEAR MODELS

We now turn to nonlinear models, for which the techniques for model reduction are much less well developed. We again consider representing the state $\mathbf{x}(t)$ as a linear combination of modes, as in Section 2. Although many of the methods of the previous section (e.g., balanced truncation) do not apply directly to nonlinear systems, the general approach of Galerkin projection still applies and is discussed first. It turns out that DMD (Section 3.4) does extend to nonlinear systems, as discussed in Section 4.2, using an infinite-dimensional linear operator called the Koopman operator. This formalism then leads to various extensions of the DMD algorithm, as discussed in Section 4.3. Although this section focuses on systematic methods for model reduction of nonlinear systems, other approaches specific to a particular application can often be quite useful (see the sidebar Goman-Khrabrov Models, which describes unsteady separated flows over airfoils).

GOMAN-KHRABROV MODELS

The nonlinear modeling techniques discussed here are by no means a comprehensive list. In particular, nonlinear models taking a particular form are often used for specific problems, taking advantage of known physical insights. For example, Goman-Khrabrov models (Goman & Khrabrov 1994) have been used successfully to model the lift forces on a variety of lifting surfaces, such as delta wings (Goman & Khrabrov 1994), constant cross-section airfoils (Williams et al. 2015a), and aircraft models (Luchtenburg et al. 2015). Goman-Khrabrov models have an internal variable, x, which has dynamics governed by

$$\tau_1 \dot{x} + x = x_0 \left(\alpha(t) - \tau_2 \dot{\alpha}(t)\right),$$

where τ_1 and τ_2 are time constants that can be calibrated for a particular configuration, $\alpha(t)$ is the angle of attack of the airfoil, and x_0 is a nonlinear function that notionally represents the fraction of flow separation on the suction surface of the airfoil. For an ideal airfoil, the time-varying lift coefficient is then computed from

$$c_l(\alpha, x) = \frac{\pi \alpha}{2}\left(1 + \sqrt{x}\right)^2,$$

which can be modified to account for other airfoil shapes and characteristics.

4.1. Galerkin Projection

The method of (Petrov-)Galerkin projection, as presented in Section 2, applies to nonlinear systems. However, in many cases, it is helpful to exploit the specific nature of the nonlinearity. To see this, recall the procedure in Section 2, in which we define a set of modes $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$, which are columns of $\mathbf{V}$, and adjoint modes $\{\mathbf{w}_1, \ldots, \mathbf{w}_r\}$, columns of $\mathbf{W}$, which satisfy $\langle \mathbf{v}_j, \mathbf{w}_k \rangle = \delta_{jk}$. Letting $\mathbf{x}(t) = \mathbf{V}\mathbf{a}(t)$, where $\mathbf{a} = (a_1, \ldots, a_r)$, one finds that the projected equations are

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{a}(t) = \mathbf{W}^T \mathbf{f}(\mathbf{V}\mathbf{a}(t)). \tag{27}$$

The modes $\{\mathbf{v}_j\}$ and $\{\mathbf{w}_j\}$ may be obtained as POD modes of a particular data set, or even using BPOD for the nonlinear equations linearized about an equilibrium point, as in Ilak et al. (2010).

The above expression represents r ordinary differential equations, so in principle, these should not be difficult to solve numerically, as long as r is relatively small. However, to evaluate the right-hand side, we need to evaluate $\mathbf{f}$ at the point $\mathbf{x} = \mathbf{V}\mathbf{a}$: That is, we need to do an evaluation in the high-dimensional space to compute the right-hand side of Equation 27.

Fortunately, in many situations, the situation is not so dire. For instance, the incompressible Navier-Stokes equations are of course nonlinear, but the form of the nonlinearity is only quadratic: $\mathbf{u} \cdot \nabla \mathbf{u}$. Thus, when the Navier-Stokes equations are discretized and written in the form $\mathrm{d}\mathbf{x}/\mathrm{d}t = \mathbf{f}(\mathbf{x})$ [where $\mathbf{x}(t) \in \mathbb{R}^n$ denotes the discretized velocity field], the function $\mathbf{f}$ is quadratic and may be written

$$\mathbf{f}(\mathbf{x}) = L(\mathbf{x}) + Q(\mathbf{x}, \mathbf{x}), \tag{28}$$

where $L : \mathbb{R}^n \to \mathbb{R}^n$ is linear, and $Q : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is bilinear (linear in each argument). [Let us assume that the variables have been chosen such that $\mathbf{x} = 0$ is a steady solution of the Navier-Stokes equations; otherwise, there will be an additional constant term in $\mathbf{f}(\mathbf{x})$.] For this specific form of the nonlinearity, the right-hand side of Equation 27 may be evaluated explicitly in terms of $\mathbf{a}$, as

$$\frac{\mathrm{d}}{\mathrm{d}t}a_k(t) = \sum_{i=1}^{r} L_k^i a_i(t) + \sum_{i,j=1}^{r} Q_k^{ij} a_i(t) a_j(t), \tag{29a}$$

where

$$L_k^i = \langle L(\mathbf{v}_i), \mathbf{w}_k \rangle, \qquad Q_k^{ij} = \langle Q(\mathbf{v}_i, \mathbf{v}_j), \mathbf{w}_k \rangle, \qquad i, j, k = 1, \ldots, r. \tag{29b}$$

Measure preserving: the map $\mathbf{F}$ is measure preserving if $\mu(\mathbf{F}^{-1}(A)) = \mu(A)$ for all measurable sets $A \subset X$

These terms may be calculated explicitly, given the modes $\mathbf{v}_k$ and $\mathbf{w}_k$, so one does not need to evaluate $\mathbf{f}$ while solving Equation 29a.

In the discussion of the Navier-Stokes equations above, we have ignored the pressure term. In some cases, one does need to consider a pressure term (e.g., Noack et al. 2005; Holmes et al. 2011, section 4.3), but in many cases, the pressure drops out altogether, as a consequence of the divergence theorem. For a flow evolving in a three-dimensional volume Ω, taking the inner product of ∇p and a divergence-free mode $\mathbf{w}_k$ gives

$$\langle \nabla p, \mathbf{w}_k \rangle = \iiint_\Omega \nabla p \cdot \mathbf{w}_k \, \mathrm{d}V = \iiint_\Omega \nabla \cdot (p \mathbf{w}_k) \, \mathrm{d}V = \iint_{\partial\Omega} p \mathbf{w}_k \cdot \mathbf{n} \, \mathrm{d}A,$$

where $\mathbf{n}$ denotes the outward-facing unit vector normal to the boundary $\partial\Omega$. The boundary integral vanishes under many common cases, for instance, if the fluid evolves in a periodic domain, or a finite domain with no inflow or outflow (in which case $\mathbf{w}_k \cdot \mathbf{n}$ must be zero on the boundary).

For incompressible flows, it suffices to consider quadratic nonlinearities as in Equation 28. For more complicated nonlinearities, other methods are available, such as the discrete empirical interpolation method (Chaturantabut & Sorensen 2010), although this method does not give an explicit model as in Equation 29.

4.2. The Koopman Operator

We now turn to an alternative approach to nonlinear systems, with roots in ergodic theory. The Koopman operator is a linear operator that completely characterizes the dynamics of a nonlinear system. If this sounds too good to be true, there is indeed a catch: This operator is infinite dimensional, even if the dynamical system is finite dimensional. However, we can learn something from finite-dimensional approximations of this infinite-dimensional operator.

Let us consider a discrete-time dynamical system evolving on a set X, called the state space. For a fluid flow, the state space would typically consist of divergence-free velocity fields, and the dynamical system would arise from the unsteady Navier-Stokes equations. Let $\mathbf{F} : X \to X$ be a transformation that defines dynamics on X according to

$$\mathbf{x}(k+1) = \mathbf{F}(\mathbf{x}(k)), \tag{30}$$

where $\mathbf{x}(k) \in X$. The Koopman operator $\mathcal{K}$ acts on scalar-valued functions of $\mathbf{x}$ as

$$\mathcal{K}g(\mathbf{x}) := g(\mathbf{F}(\mathbf{x})). \tag{31}$$

That is, $\mathcal{K}$ takes a scalar-valued function g and produces a new scalar-valued function $g \circ \mathbf{F}$. To be rigorous, one needs to define the function space that $\mathcal{K}$ acts on: For instance, if we have a measure μ on X (e.g., the Lebesgue measure, or an invariant measure of the dynamical system), we may consider the space $L^2(X, \mu)$. If the map $\mathbf{F}$ is measure preserving, then $\mathcal{K}$ is a unitary operator. For a fluid flow, a measure would define a notion of volume in the (infinite-dimensional) space of velocity fields, and measure-preserving dynamics would preserve this notion of volume. Here, however, we do not restrict ourselves to measure-preserving dynamical systems.

The Koopman operator is linear [e.g., $\mathcal{K}(g_1 + g_2) = \mathcal{K}g_1 + \mathcal{K}g_2$], so it can have eigenfunctions, which satisfy

$$\mathcal{K}\varphi(\mathbf{x}) = \lambda\varphi(\mathbf{x}). \tag{32}$$

To appreciate the potential of this approach, we observe that if the Koopman operator has enough eigenfunctions, then one can use them to determine a (nonlinear) change of coordinates in which

the dynamics are linear, and even uncoupled. Let φ_j be eigenfunctions of $\mathcal{K}$ with corresponding eigenvalues λ_j and consider the new variables $z_j = \varphi_j(\mathbf{x})$. Then the dynamics in these coordinates are described by

$$z_j(k+1) = \varphi_j(\mathbf{x}(k+1)) = \varphi_j(\mathbf{F}(\mathbf{x}(k))) = \mathcal{K}\varphi_j(\mathbf{x}(k)) = \lambda_j \varphi_j(\mathbf{x}(k)) = \lambda_j z_j(k), \tag{33}$$

where k is a step in the discrete advancement, as in a time integration scheme. Thus, if we can reconstruct $\mathbf{x}$ from $\mathbf{z} = (z_1, \ldots, z_n)$ (i.e., if the map $\mathbf{x} \mapsto \mathbf{z}$ is injective), then we have determined coordinates in which the dynamics are linear. This may be viewed as a nonlinear version of diagonalization. In practice, this procedure may fail, as $\mathcal{K}$ may not have enough eigenfunctions—for instance, this happens for chaotic systems (Koopman & von Neumann 1932)—but there are many systems for which it does work.

4.2.1. Connection with dynamic mode decomposition. In order to explain the connections between DMD and the Koopman operator, we first define the notion of an observable, which is simply a scalar-valued function of the state $\mathbf{x}$ [e.g., a continuous function in $L^2(X)$]. Intuitively, we may think of observables as the measurements we make: At a particular value of the state (e.g., velocity field), the observable gives us a real number, which is the measurement. We typically have more than one scalar measurement, so we let $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_n)$ denote a vector of observables [where each $\psi_k \in L^2(X)$]. Suppose we have a set of data as in Section 3.4, with initial states $\mathbf{x}_j \in X$, along with their images $\mathbf{x}_j^{\#} = \mathbf{F}(\mathbf{x}_j)$, though we do not necessarily have access to the entire state $\mathbf{x}_j$ (e.g., the entire flow field); rather, we have access only to the vector of observables evaluated at that state, $\boldsymbol{\psi}(\mathbf{x}_j)$ (e.g., several point measurements of the flow). We let $\mathbf{y}_j = \boldsymbol{\psi}(\mathbf{x}_j)$ and $\mathbf{y}_j^{\#} = \boldsymbol{\psi}(\mathbf{x}_j^{\#})$ denote these vectors of observables, and consider performing DMD on this data set, as in Section 3.4. If the governing dynamics in Equation 30 are nonlinear, it is not clear what the meaning of this DMD calculation is: The DMD procedure is performing a linear fit to the data, which arose from a nonlinear system. However, under certain conditions, the DMD eigenvalues are eigenvalues of the Koopman operator; furthermore, the eigenvectors may be used to find the corresponding Koopman eigenfunctions. The following theorem, proven in Tu et al. (2014b, section 4.1), gives precise conditions under which this relationship holds.

Theorem 1 (connection between DMD and the Koopman operator). *Suppose φ is an eigenfunction of $\mathcal{K}$ with eigenvalue λ, and suppose φ lies in the span of the observables $\{\psi_1, \ldots, \psi_n\}$, so that*

$$\varphi(\mathbf{x}) = \bar{w}_1\psi_1(\mathbf{x}) + \cdots + \bar{w}_n\psi_n(\mathbf{x}) = \mathbf{w}^*\boldsymbol{\psi}(\mathbf{x}) \tag{34}$$

for some constants $(w_1, \ldots, w_n) = \mathbf{w} \in \mathbb{C}^n$. Suppose further that $\mathbf{w}$ lies in the range of the data matrix $\mathbf{Y}$, whose columns are $\mathbf{y}_m = \boldsymbol{\psi}(\mathbf{x}_m)$, $m = 1, \ldots, M$. Then λ is an eigenvalue of $\mathbf{A} = \mathbf{Y}^{\#}\mathbf{Y}^{+}$, and $\mathbf{w}$ is a left eigenvector of $\mathbf{A}$: that is, $\mathbf{w}^*\mathbf{A} = \lambda\mathbf{w}^*$.

The notation $\mathbf{w}^*$ denotes the complex-conjugate transpose of $\mathbf{w}$. Note that here we need to consider complex vectors because, although $\mathbf{A}$ is a real matrix, its eigenvectors may be complex. The proof of this theorem is brief and illuminating.

Proof. Because $\mathcal{K}\varphi = \lambda\varphi$, we have $\mathbf{w}^*\boldsymbol{\psi}(\mathbf{F}(\mathbf{x})) = \lambda\mathbf{w}^*\boldsymbol{\psi}(\mathbf{x})$ for all $\mathbf{x} \in X$. In particular, this is true for $\mathbf{x} = \mathbf{x}_j$, so because $\mathbf{y}_j = \boldsymbol{\psi}(\mathbf{x}_j)$, this becomes $\mathbf{w}^*\mathbf{y}_j^{\#} = \lambda\mathbf{w}^*\mathbf{y}_j$. Arranging these scalar equations into a row vector gives $\mathbf{w}^*\mathbf{Y}^{\#} = \lambda\mathbf{w}^*\mathbf{Y}$, and thus $\mathbf{w}^*\mathbf{A} = \lambda\mathbf{w}^*\mathbf{Y}\mathbf{Y}^{+} = \lambda\mathbf{w}^*$, where the last equality holds because $\mathbf{w}$ is in the range of $\mathbf{Y}$. □

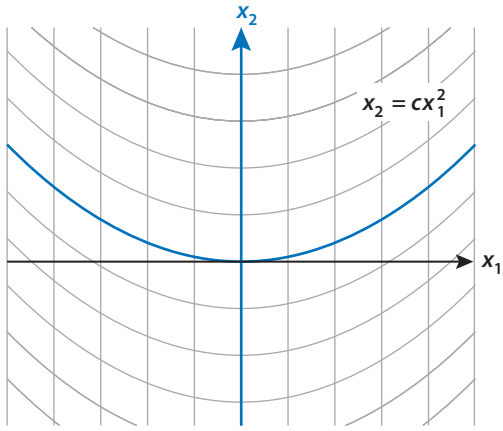

Figure 4

Phase plane for Example 2. The light gray lines denote level sets of Koopman eigenfunctions φ_1 (*vertical*) and φ_2 (*curved*); the thick lines denote invariant manifolds, which also correspond to zero level sets of φ_1 and φ_2.

In particular, one needs to have a sufficiently rich set of observables (for the Koopman eigenfunction φ to lie in the span of $\{\psi_j\}$), as well as a sufficiently rich set of data (so that $\mathbf{w}$ is in the range of $\mathbf{Y}$). The following example (originally presented in Tu et al. 2014b) illustrates how DMD can be used to construct Koopman eigenvalues and eigenfunctions, and what can go wrong if the observables are not chosen judiciously.

Example 2 (planar ordinary differential equation). Consider the ordinary differential equation given by

$$\begin{aligned}\dot{x}_1 &= \lambda_1 x_1,\\ \dot{x}_2 &= \lambda_2 x_2 + (2\lambda_1 - \lambda_2) c x_1^2.\end{aligned}$$

This system has an equilibrium at the origin, and this equilibrium is stable if both λ_1 and λ_2 have a negative real part. Furthermore, the system has invariant manifolds given by $x_1 = 0$ and $x_2 = c x_1^2$, as shown in **Figure 4**: If the state lies on one of these curves at some time, it remains on it for all time.

We may convert this differential equation to discrete time by integrating from time 0 to Δt. Letting $x_j(k)$ denote the value of x_j at time $k\Delta t$, and defining $\mu_j = e^{\lambda_j \Delta t}$, the discrete-time equations become

$$\begin{aligned}x_1(k+1) &= \mu_1 x_1(k),\\ x_2(k+1) &= \mu_2 x_2(k) + (\mu_1^2 - \mu_2) c\, x_1(k)^2.\end{aligned}$$

This discrete-time system is stable if both μ_1 and μ_2 have a magnitude less than 1 (consistent with λ_1 and λ_2 having a negative real part). It is straightforward to check that μ_1 and μ_2 are eigenvalues of the Koopman operator, with corresponding eigenfunctions

$$\begin{aligned}\varphi_1(\mathbf{x}) &= x_1,\\ \varphi_2(\mathbf{x}) &= x_2 - c x_1^2.\end{aligned}$$

In addition, φ_1^k is an eigenfunction with eigenvalue μ^k, the product $\varphi_1\varphi_2$ is an eigenfunction with eigenvalue $\mu_1\mu_2$, etc.

Let us apply DMD to this example, with initial states $\mathbf{x}$ given by $(1,1)$, $(5,5)$, $(-1,1)$, and $(-5,5)$, with $\mu_1 = 0.9$ and $\mu_2 = 0.5$, comparing different choices of observables.

First, let us use the observables $\boldsymbol{\psi}(\mathbf{x}) = (x_1, x_2)$. If $c = 0$, so that the problem is linear, then DMD eigenvalues are 0.9 and 0.5, which match the values of μ_1 and μ_2. If $c = 1$, however, then the DMD eigenvalues are 0.9 and 2.002. The latter value does not correspond to a Koopman eigenvalue, and because it has a magnitude >1, one might even presume that the equilibrium is unstable!

Next, let us consider the observables $\boldsymbol{\psi}(\mathbf{x}) = (x_1, x_2, x_1^2)$. At first glance, it might seem redundant to include the measurement x_1^2, because if we measure x_1, of course we can deduce x_1^2, but the point is that we need such a function in order for the Koopman eigenfunction $\varphi_2(\mathbf{x}) = x_2 - cx_1^2$ to lie in the span of the observables. With these three observables, the DMD eigenvalues are 0.9, 0.5, and $0.81 = 0.9^2$, which agree with Koopman eigenvalues. In addition, the Koopman eigenfunctions may be constructed from the left eigenvectors of the DMD matrix $\mathbf{A}$.

Finally, let us consider the observables $\boldsymbol{\psi}(\mathbf{x}) = (x_1, x_2, x_2^2)$. Now, the DMD eigenvalues are 0.9, 0.822, and 4.767; the latter two do not correspond to Koopman eigenvalues, and once again, one might be tempted to conclude that the equilibrium is unstable. In this case, there is still an exact linear relationship between the snapshots (i.e., $\mathbf{y}_j^{\#} = A\mathbf{y}_j$ is satisfied exactly), but the eigenvalues do not correspond to Koopman eigenvalues because the Koopman eigenfunction φ_2 is not in the span of the observables.

4.2.2. Koopman modes. The DMD modes are right eigenvectors of $\mathbf{A}$, but the Koopman eigenfunctions are determined from the left eigenvectors of $\mathbf{A}$. It is natural to consider how the DMD modes are related to the Koopman operator.

Let us assume that the DMD matrix $\mathbf{A}$ has a full set of right eigenvectors $\mathbf{v}_j$ (the DMD modes), with eigenvalues λ_j, and corresponding left eigenvectors $\mathbf{w}_j$, normalized so that $\langle \mathbf{v}_j, \mathbf{w}_k \rangle := \mathbf{w}_k^* \mathbf{v}_j = \delta_{jk}$. Then any vector $\mathbf{q} \in \mathbb{R}^n$ may be expanded as

$$\mathbf{q} = \sum_{j=1}^{n} (\mathbf{w}_j^* \mathbf{q}) \mathbf{v}_j. \tag{35}$$

In particular, the vector of observables $\boldsymbol{\psi}(\mathbf{x})$ may be expanded in this way. Defining $\varphi_j(\mathbf{x}) = \mathbf{w}_j^* \boldsymbol{\psi}(\mathbf{x})$, we then have

$$\boldsymbol{\psi}(\mathbf{x}) = \sum_{j=1}^{n} \varphi_j(\mathbf{x}) \mathbf{v}_j. \tag{36}$$

If all the φ_j in this sum correspond to Koopman eigenfunctions (i.e., if the conditions of Theorem 1 are satisfied), then the terms $\mathbf{v}_j$ in this sum are called Koopman modes. Note that the Koopman modes $\mathbf{v}_j$ depend on the observables $\boldsymbol{\psi}$, whereas the Koopman eigenfunctions φ_j do not. With the dynamics given by Equation 30, we have

$$\boldsymbol{\psi}(\mathbf{x}(k)) = \sum_{j=1}^{n} \lambda_j^k \varphi_j(\mathbf{x}(0)) \mathbf{v}_j. \tag{37}$$

Comparing this expression with Equation 22, we observe that the Koopman modes define the vectors $\mathbf{v}_j$ in this expression (the DMD modes), and the Koopman eigenfunctions determine the constants $c_j = \varphi_j(\mathbf{x}(0))$.

Example 3 (linear system). Let us consider a linear system, with $\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k)$. For such a system, we show that if we measure the full state, $\boldsymbol{\psi}(\mathbf{x}) = \mathbf{x}$, the Koopman modes

are simply the (right) eigenvectors of $\mathbf{A}$. Assume $\mathbf{A}$ has a full set of eigenvectors $\mathbf{v}_j$ that satisfy $\mathbf{A}\mathbf{v}_j = \lambda_j \mathbf{v}_j$, with corresponding left eigenvectors $\mathbf{w}_j$ that satisfy $\mathbf{w}_j^* \mathbf{A} = \lambda_j \mathbf{w}_j^*$, normalized so that $\mathbf{w}_j^* \mathbf{v}_k = \delta_{jk}$. First, observe that the function $\varphi_j(\mathbf{x}) = \mathbf{w}_j^* \mathbf{x}$ is a Koopman eigenfunction:

$$\mathcal{K}\varphi_j(\mathbf{x}) = \varphi_j(\mathbf{A}\mathbf{x}) = \mathbf{w}_j^* \mathbf{A}\mathbf{x} = \lambda_j \mathbf{w}_j^* \mathbf{x} = \lambda_j \varphi_j(\mathbf{x}).$$

Next, observe that if we measure the full state [corresponding to the observable $\boldsymbol{\psi}(\mathbf{x}) = \mathbf{x}$], then we may write

$$\mathbf{x} = \sum_{j=1}^{n} (\mathbf{w}_j^* \mathbf{x}) \mathbf{v}_j = \sum_{j=1}^{n} \varphi_j(\mathbf{x}) \mathbf{v}_j.$$

Comparing the above equation with Equation 36, we see that the $\mathbf{v}_j$ are the Koopman modes.

The connection between DMD and the Koopman operator was first presented in Rowley et al. (2009). This paper also introduced the terminology Koopman modes, although this concept was considered earlier by Mezić (2005). Bagheri (2013) gave a nice example of the application of these methods, presenting Koopman modes for the flow past the cylinder and comparing numerical approximations of DMD eigenvalues with Koopman eigenvalues calculated analytically from a Stuart-Landau equation.

4.3. Extended Dynamic Mode Decomposition

The results of the previous section (in particular, Theorem 1) demonstrate a connection between DMD and the Koopman operator, which provides an interpretation of what DMD eigenvalues/modes mean for a nonlinear system. However, Theorem 1 applies only under fairly restrictive conditions: The Koopman eigenfunction must lie within the space spanned by the observables $\{\psi_j\}$. What happens if the conditions of this theorem are not met? Does any connection between the Koopman operator and DMD still hold? As we might expect, the answer is yes, although some approximation is necessary.

Williams et al. (2015b) showed that the above DMD procedure may be viewed as a spectral collocation method for approximating the Koopman operator $\mathcal{K}$. One considers a subspace of functions spanned by the observables $\{\psi_j\}$ and writes a matrix approximation of $\mathcal{K}$ with respect to this basis. This technique is referred to as extended dynamic mode decomposition (EDMD) because DMD (as presented in Section 3.4) is a special case with the observables $\boldsymbol{\psi}(\mathbf{x}) = \mathbf{x}$.

Similar to any spectral method, the quality of the approximation depends on the basis functions chosen (in this case, the observables), and Williams et al. (2015b) suggested several different possibilities, such as polynomials, indicator functions on small boxes (Ulam's method), radial basis functions, and spectral elements. The appropriate choice depends on the nature of the data. For example, if the Koopman eigenfunctions are relatively smooth, then polynomials might be a good choice; if they have discontinuities, then spectral elements might be a better choice.

We now give an example of how this EDMD framework can be applied specifically to fluid systems, using knowledge of the governing equations to guide our selection of observables. Expressing the POD coefficients as before by $\mathbf{a} = \mathbf{U}_r^T \mathbf{x}$, we let

$$\mathbf{q} = \boldsymbol{\psi}(\mathbf{x}) = \begin{bmatrix} \mathbf{a} \\ \mathrm{vec}\,(\mathbf{a} \otimes \mathbf{a}) \end{bmatrix}, \tag{38}$$

where vec $(\mathbf{a} \otimes \mathbf{a})$ indicates a vector of all nonredundant quadratic couplings between POD coefficients (i.e., a_1^2, $a_1 a_2$, a_2^2). If we retain r POD modes, then the total number of observables is $n = r + r(r+1)/2$. From Equation 24, we obtain a model for the dynamics:

$$\mathbf{q}(k+1) = \mathbf{A}\mathbf{q}(k). \tag{39}$$

When using such a model for prediction, one can also take advantage of the known relationships between the linear and quadratic states to improve accuracy. (Note that the evolution of the quadratic terms would include higher-degree polynomials.) Additionally, although Equation 39 is a linear model in the expanded state $\mathbf{q}$, it can be written as a nonlinear model in $\mathbf{a}$. That is, within the matrix $\mathbf{A}$ are the terms required to construct a model of the form

$$\mathbf{a}(k+1) = \mathbf{L}(\mathbf{a}(k)) + \mathbf{Q}(\mathbf{a}(k), \mathbf{a}(k)), \tag{40}$$

where $\mathbf{L} : \mathbb{R}^r \to \mathbb{R}^r$ is linear, and $\mathbf{Q} : \mathbb{R}^r \times \mathbb{R}^r \to \mathbb{R}^r$ is bilinear. Aside from the possible addition of a pressure term, this is the same form of nonlinear model that would be arrived at through Galerkin projection of the governing equations onto POD modes (Equation 28). Although Galerkin projection uses data for identification of a spatial reduced-order basis and the governing equations to identify a reduced-order model, here we are using the governing equations only to pick an appropriate transformation ψ and are using temporally resolved data to identify the dynamics.

4.3.1. Example: flow past a cylinder. We apply this particular form of EDMD to the well-studied case of flow past a circular cylinder. Beyond a critical Reynolds number of approximately 47 (Provansal et al. 1987), the equilibrium becomes unstable, and there is a stable limit cycle that corresponds to vortex shedding. We perform a numerical simulation for $Re = 60$, with the initial condition close to the unstable equilibrium, and gather data that capture the initial growth near the unstable equilibrium, through convergence to the limit cycle. This is an example in which regular DMD will fail to capture the appropriate dynamics (in the sense of identifying an appropriate reduced-order model) given that the dynamics are fundamentally nonlinear. The data were generated using an immersed boundary projection method (Taira & Colonius 2007, Colonius & Taira 2008), with the unstable equilibrium solution computed by employing selective frequency damping (Åkervik et al. 2006).

Figure 5 shows the performance of both Galerkin projection and the EDMD approach outlined above in identifying a model that can predict the evolution of the first three POD coefficients. These evolve on a paraboloid, as observed by Noack et al. (2003), and here the third POD mode resembles the shift mode in Noack et al. (2003). Here the mean of the data (rather than the mean of the limit cycle) was subtracted from the data before performing POD to use a data-driven perspective, and to be consistent with the subspace used for both procedures. We observe that the EDMD model is more accurate than the Galerkin projection model in terms of obtaining both the correct transient and limit-cycle behavior. We note that the accuracy of the Galerkin projection model is sensitive to the extent and resolution of the data used.

Figure 6 compares both model procedures again, but with access to only a subset of the data, in this case only the first 400 snapshots. Even though these data have not reached the limit cycle, the EDMD model still accurately predicts the location of the limit cycle. This contrasts with the Galerkin model, in which the limited data available modify the spatial structure of the computed POD modes to an extent that renders the model qualitatively incorrect. This example highlights the advantages that can be gained from using data-driven dynamical modeling, particularly when the form of the nonlinear model is known a priori.

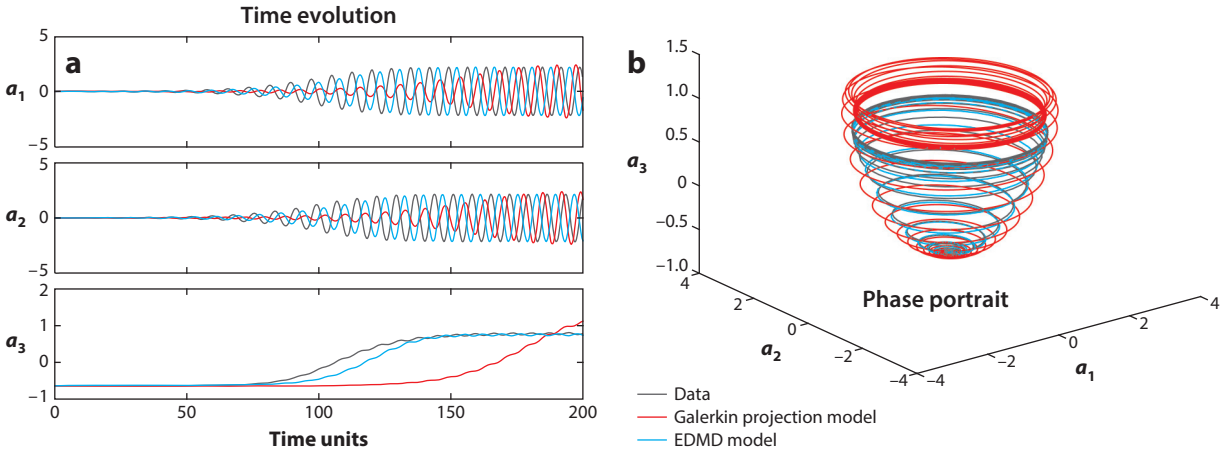

Figure 5

Performance of third-order Galerkin and extended dynamic mode decomposition (EDMD) nonlinear models in predicting the evolution of proper orthogonal decomposition coefficients for transitional flow past a cylinder, showing (*a*) time evolution and (*b*) phase portrait plots. The Galerkin and EDMD phase portrait models are allowed to evolve for 800 dimensionless time units to confirm limit-cycle behavior.

4.3.2. Kernel method. A drawback of the EDMD approach is that the number of basis functions can grow rapidly with the state dimension. For instance, we saw above that if the state dimension is n and one considers quadratic functions of the state, then the number of observables is $O(n^2)$. If one wishes to include polynomials of higher degree, the situation gets much worse: For polynomials of degree d in n variables, the number of observables is $\binom{n+d}{n}$ (Bishop 2007), so for $n = 100$ states, including polynomials up to degree $d = 10$ gives $\sim 10^{13}$ observables, far too many for practical

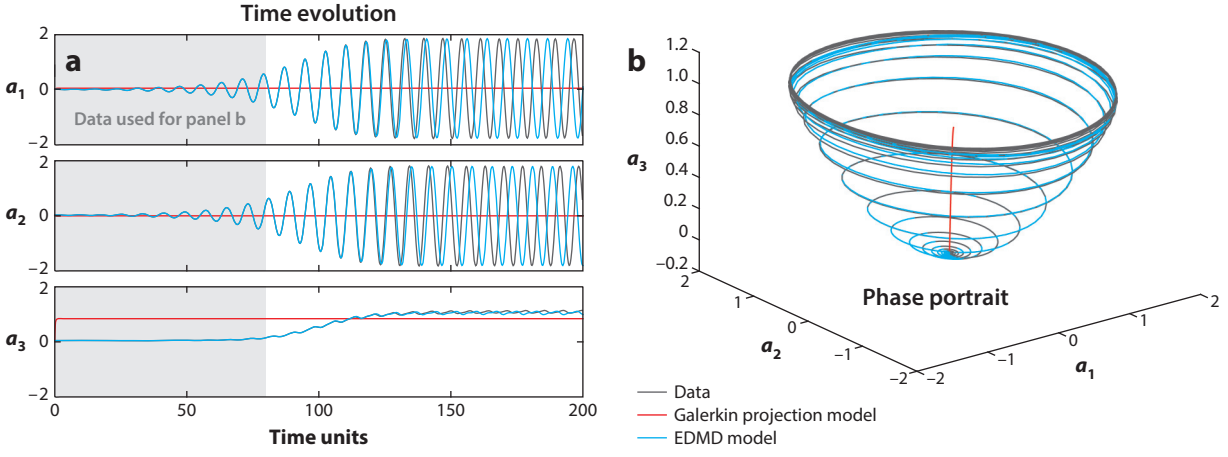

Figure 6

Performance of third-order Galerkin and extended dynamic mode decomposition (EDMD) nonlinear models in predicting the evolution of proper orthogonal decomposition coefficients for transitional flow past a cylinder, showing (*a*) time evolution and (*b*) phase portrait plots. Models are identified using only the first 400 snapshots of data, as shown in panel *a*.

computation. Williams et al. (2014) proposed using a kernel method from machine learning to reduce these calculations to $O(n)$.

One begins by rearranging the expression for the DMD matrix in Equation 25 as follows. Let $\mathbf{Y}$ and $\mathbf{Y}^{\#}$ be matrices of snapshots as in Section 3.4: If q denotes the number of observables and m the number of snapshots, then $\mathbf{Y}$ and $\mathbf{Y}^{\#}$ have dimension $q \times m$. We are interested in the case in which $q \gg m$. We define the $m \times m$ matrices

$$\mathbf{G} = \mathbf{Y}^T\mathbf{Y}, \qquad \mathbf{M} = \mathbf{Y}^T\mathbf{Y}^{\#}, \tag{41}$$

and let

$$\tilde{\mathbf{A}} = \mathbf{G}^{+}\mathbf{M} = (\mathbf{Y}^T\mathbf{Y})^{+}\mathbf{Y}^T\mathbf{Y}^{\#} = \mathbf{Y}^{+}\mathbf{Y}^{\#}. \tag{42}$$

The DMD matrix $\mathbf{A} = \mathbf{Y}^{\#}\mathbf{Y}^{+}$ from Equation 25 is $q \times q$, but we may calculate its nonzero eigenvalues and the corresponding eigenvectors from those of the smaller $m \times m$ matrix $\tilde{\mathbf{A}}$ as follows: If $\tilde{\mathbf{A}}\tilde{\mathbf{v}} = \lambda\tilde{\mathbf{v}}$, then letting $\mathbf{v} = \mathbf{Y}^{\#}\tilde{\mathbf{v}}$, we see that

$$\mathbf{A}\mathbf{v} = (\mathbf{Y}^{\#}\mathbf{Y}^{+})(\mathbf{Y}^{\#}\tilde{\mathbf{v}}) = \mathbf{Y}^{\#}\lambda\tilde{\mathbf{v}} = \lambda\mathbf{v}. \tag{43}$$

So this approach provides an alternative to the algorithm presented in Section 3.4 for calculating DMD/Koopman modes.

Now, we observe that

$$\mathbf{G}_{jk} = \langle \boldsymbol{\psi}(\mathbf{x}_j), \boldsymbol{\psi}(\mathbf{x}_k) \rangle, \qquad \mathbf{M}_{jk} = \langle \boldsymbol{\psi}(\mathbf{x}_j), \boldsymbol{\psi}(\mathbf{x}_k^{\#}) \rangle, \tag{44}$$

so these matrices depend only on the inner products of observables. If the number of observables q is very large, however, these inner products will be cumbersome to evaluate. The key feature of a kernel method is to evaluate these inner products more efficiently using a kernel function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, defined by

$$k(\mathbf{x}, \hat{\mathbf{x}}) = \left\langle \boldsymbol{\psi}(\mathbf{x}), \boldsymbol{\psi}(\hat{\mathbf{x}}) \right\rangle. \tag{45}$$

The idea is then to choose a kernel function that is easy to evaluate yet corresponds to the desired inner product in Equation 45. For example, the kernel $k(\mathbf{x}, \hat{\mathbf{x}}) = (1 + \mathbf{x}^T\hat{\mathbf{x}})^d$ corresponds to a set of observables $\boldsymbol{\psi}$ consisting of all monomials in components of $\mathbf{x}$ up to degree d (Bishop 2007, section 6.2). For instance, for $\mathbf{x} \in \mathbb{R}^2$ and $d = 2$, we see that

$$\begin{aligned}(1 + \mathbf{x}^T\hat{\mathbf{x}})^2 &= 1 + 2x_1\hat{x}_1 + 2x_2\hat{x}_2 + x_1^2\hat{x}_1^2 + 2x_1x_2\hat{x}_1\hat{x}_2 + x_1^2\hat{x}_2^2 \\ &= \langle \boldsymbol{\psi}(\mathbf{x}), \boldsymbol{\psi}(\hat{\mathbf{x}}) \rangle,\end{aligned}$$

with $\boldsymbol{\psi}(\mathbf{x}) = (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, \sqrt{2}x_1x_2, x_2^2)$. Thus, using the kernel function k, we can compute the inner products in Equation 44 without ever explicitly writing down the observables $\boldsymbol{\psi}$. Many other choices of kernel functions are possible besides the polynomial kernel considered above (e.g., exponential kernels, Gaussian kernels), each of which corresponds to a different implicit choice of observables $\boldsymbol{\psi}$. This idea can lead to dramatic computational savings: For the example with $n = 100$ states, with observables consisting of polynomials up to degree $d = 10$, we may evaluate each inner product in Equation 44 with only $n = 100$ operations, instead of $q \sim 10^{13}$ operations. This then enables one to apply EDMD for bases of very large dimension, with no more computational effort than one uses to compute DMD as in Section 3.4. Williams et al. (2014) demonstrated that the kernel method identifies physically relevant Koopman eigenvalues (and modes) much more effectively than standard DMD in examples including simulations of the Fitzhugh-Nagumo equation, and experimental data of flow past a circular cylinder.

5. CONCLUDING REMARKS

This review discusses a number of methods suitable for extracting the essential dynamical features from data describing fluid flows. This is a large and rapidly evolving field, and many techniques are available. Above we focus on several methods that have been effective in examples in fluid mechanics; most of these techniques (although not all) may be applied to data from either simulations or experiments. We highlight the similarities and connections between the various methods, many of which were developed in disciplines other than fluid mechanics.

SUMMARY POINTS

1. POD and Galerkin projection are good, general-purpose tools for model reduction. However, they can perform poorly, particularly for non-normal systems or other systems that have large sensitivity to small changes in the state.
2. For linear systems, there are several effective techniques for model reduction that are computationally tractable even for high-dimensional fluid problems. Some techniques, such as balanced truncation and BPOD, rely on accurate full-order models or simulation data, whereas other techniques, such as the ERA and DMD, can identify dynamics solely from data.
3. In an example of linearized channel flow, we found that the most effective model reduction technique based solely on data (i.e., without knowledge of governing equations) was to first compute a model using DMD, retaining more modes than ultimately desired, and then perform balanced truncation to determine a reduced-order model.
4. For nonlinear systems, general techniques are not as well developed. POD/Galerkin models can work well but often require careful tuning to give satisfactory results.
5. The Koopman operator can be used to study nonlinear systems, and its eigenfunctions and corresponding Koopman modes may be computed using a version of DMD. These eigenfunctions may be used, for instance, to determine coordinates in which the system is linear and uncoupled (a nonlinear version of diagonalization), and the Koopman modes can be used to understand coherent structures that occur at a particular temporal frequency.

FUTURE ISSUES

1. For flow control problems, one would like to know where to place sensors and actuators. Although some preliminary work has been done on model problems (Chen & Rowley 2011), for practical applications this is still a challenging problem for which reduced-order models may be able to contribute.
2. All of the reduced-order models discussed above involve some approximation and corresponding error. Incorporating uncertainty quantification into these models would be a valuable contribution. Some promising directions are given by Sapsis & Lermusiaux (2009) and Sapsis & Majda (2013).

3. When applying (E)DMD to a nonlinear system, one needs to specify a set of basis functions (observables). As shown in Section 4.2 (Example 2), the effectiveness of DMD depends critically on the basis chosen. This choice is presently more of an art than a science, and a better understanding of suitable basis functions is needed.
4. For linear systems, there are reliable methods for incorporating the effect of a control input, using many of the methods discussed in Section 3. For nonlinear systems, however, it is much more challenging to model the effect of inputs. One approach is provided by Lall et al. (2002), but this is limited to relatively mild nonlinearities (e.g., a globally stable equilibrium). Perhaps the Koopman-based ideas in Section 4.2 can be adapted to nonlinear systems with inputs, but this area remains to be explored.
5. Challenging eigenvalue problems and other computational tasks constitute the core of many of the techniques reviewed. Advancing computer power will continue to push the envelope of what is computationally feasible.

DISCLOSURE STATEMENT

The authors are not aware of any biases that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

The authors thank D.M. Luchtenburg for helpful conversations and gratefully acknowledge support from the Air Force Office of Scientific Research, grants FA9550-12-1-0075 and FA9550-14-1-0289.

LITERATURE CITED

Ahuja S, Rowley CW. 2010. Feedback control of unstable steady states of flow past a flat plate using reduced-order estimators. *J. Fluid Mech.* 645:447–78

Åkervik E, Brandt L, Henningson DS, Hœpffner J, Marxen O, Schlatter P. 2006. Steady solutions of the Navier-Stokes equations by selective frequency damping. *Phys. Fluids* 18:068102

Antoulas AC. 2005. *Approximation of Large-Scale Dynamical Systems*. Philadelphia: SIAM

Bagheri S. 2013. Koopman-mode decomposition of the cylinder wake. *J. Fluid Mech.* 726:596–623

Barbagallo A, Sipp D, Schmid PJ. 2009. Closed-loop control of an open cavity flow using reduced-order models. *J. Fluid Mech.* 641:1–50

Belson BA, Semeraro O, Rowley CW, Henningson DS. 2013. Feedback control of instabilities in the two-dimensional Blasius boundary layer: the role of sensors and actuators. *Phys. Fluids* 25:054106

Belson BA, Tu JH, Rowley CW. 2014. A parallelized model reduction library. *ACM Trans. Math. Softw.* 40:30

Benner P, Gugercin S, Willcox K. 2015. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Rev.* 57:483–531

Bishop CM. 2007. *Pattern Recognition and Machine Learning*. New York: Springer

Brunton SL, Dawson STM, Rowley CW. 2014. State-space model identification and feedback control of unsteady aerodynamic forces. *J. Fluids Struct.* 50:253–70

Brunton SL, Noack BR. 2015. Closed-loop turbulence control: progress and challenges. *Appl. Mech. Rev.* 67:050801

Brunton SL, Rowley CW, Williams DR. 2013. Reduced-order unsteady aerodynamic models at low Reynolds numbers. *J. Fluid Mech.* 724:203–33

Cabell RH, Kegerise MA, Cox DE, Gibbs GP. 2006. Experimental feedback control of flow-induced cavity tones. *AIAA J.* 44:1807–16

Carlsson H, Carlsson C, Fuchs L, Bai XS. 2014. Large eddy simulation and extended dynamic mode decomposition of flow-flame interaction in a lean premixed low swirl stabilized flame. *Flow Turbul. Combust.* 93:505–19

Chaturantabut S, Sorensen DC. 2010. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* 32:2737–64

Chen KK, Rowley CW. 2011. H_2 optimal actuator and sensor placement in the linearized complex Ginzburg-Landau equation. *J. Fluid Mech.* 681:241–60

Chen KK, Tu JH, Rowley CW. 2011. Variants of dynamic mode decomposition: boundary condition, Koopman, and Fourier analyses. *J. Nonlinear Sci.* 22:887–915

Chomaz J. 2005. Global instabilities in spatially developing flows: non-normality and nonlinearity. *Annu. Rev. Fluid Mech.* 37:357–92

Colonius T, Taira K. 2008. A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Comput. Methods Appl. Mech. Eng.* 197:2131–46

Dawson STM, Hemati MS, Williams MO, Rowley CW. 2016. Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition. *Exp. Fluids* 57:42

de Prony R. 1795. Essai expérimentale et analytique. *J. École Polytech.* 1:24–76

Dergham G, Sipp D, Robinet JC, Barbagallo A. 2011. Model reduction for fluids using frequential snapshots. *Phys. Fluids* 23:064101

Duke D, Honnery D, Soria J. 2012a. Experimental investigation of nonlinear instabilities in annular liquid sheets. *J. Fluid Mech.* 691:594–604

Duke D, Soria J, Honnery D. 2012b. An error analysis of the dynamic mode decomposition. *Exp. Fluids* 52:529–42

Dullerud GE, Paganini F. 1999. *A Course in Robust Control Theory: A Convex Approach*. Berlin: Springer-Verlag

Dunne R, McKeon BJ. 2015. Dynamic stall on a pitching and surging airfoil. *Exp. Fluids* 56:157

Flinois TL, Morgans AS. 2016. Feedback control of unstable flows: a direct modelling approach using the eigensystem realisation algorithm. *J. Fluid Mech.* 793:41–78

Flinois TLB, Morgans AS, Schmid PJ. 2015. Projection-free approximate balanced truncation of large unstable systems. *Phys. Rev. E* 92:023012

Ghommem M, Presho M, Calo VM, Efendiev Y. 2013. Mode decomposition methods for flows in high-contrast porous media: global-local approach. *J. Comput. Phys.* 253:226–38

Goman M, Khrabrov A. 1994. State-space representation of aerodynamic characteristics of an aircraft at high angles of attack. *J. Aircr.* 31:1109–15

Gómez F, Blackburn HM, Rudman M, McKeon BJ, Luhar M, et al. 2014. On the origin of frequency sparsity in direct numerical simulations of turbulent pipe flow. *Phys. Fluids* 26:101703

Goulart PJ, Wynn A, Pearson D. 2012. Optimal mode decomposition for high dimensional systems. *Proc. Conf. Decis. Control*, pp. 4965–70. New York: IEEE

Grilli M, Schmid PJ, Hickel S, Adams NA. 2012. Analysis of unsteady behaviour in shockwave turbulent boundary layer interaction. *J. Fluid Mech.* 700:16–28

Gugercin S, Antoulas AC. 2006. Model reduction of large-scale systems by least squares. *Linear Algebra Appl.* 415:290–321

He G, Wang J, Pan C. 2013. Initial growth of a disturbance in a boundary layer influenced by a circular cylinder wake. *J. Fluid Mech.* 718:116–30

Hemati MS, Rowley CW, Deem EA, Cattafesta LN. 2015. De-biasing the dynamic mode decomposition for applied Koopman spectral analysis. arXiv:1502.03854 [physics.flu-dyn]

Hemati MS, Williams MO, Rowley CW. 2014. Dynamic mode decomposition for large and streaming datasets. *Phys. Fluids* 26:111701

Hildebrand FB. 1974. *Introduction to Numerical Analysis*. New York: McGraw-Hill. 2nd ed.

Ho BL, Kalman RE. 1965. Effective construction of linear state-variable models from input/output data. *Proc. 3rd Annu. Allerton Conf. Circuit Syst. Theory*, pp. 449–59

Hokanson JM. 2013. *Numerically stable and statistically efficient algorithms for large scale exponential fitting*. PhD Thesis, Rice Univ., Houston, TX

Holmes PJ, Lumley JL, Berkooz G, Rowley CW. 2011. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge, UK: Cambridge Univ. Press. 2nd ed.

Hsu CS, Hou D. 1991. Reducing unstable linear control systems via real Schur transformation. *Electron. Lett.* 27:984–86
Ilak M, Bagheri S, Brandt L, Rowley CW, Henningson DS. 2010. Model reduction of the nonlinear complex Ginzburg-Landau equation. *SIAM J. Appl. Dyn. Syst.* 9:1284–302
Ilak M, Rowley CW. 2008. Modeling of transitional channel flow using balanced proper orthogonal decomposition. *Phys. Fluids* 20:034103
Illingworth SJ. 2016. Model-based control of vortex shedding at low Reynolds numbers. *Theor. Comput. Fluid Dyn.* 30:429–48
Illingworth SJ, Morgans AS, Rowley CW. 2011. Feedback control of flow resonances using balanced reduced-order models. *J. Sound Vib.* 330:1567–81
Illingworth SJ, Morgans AS, Rowley CW. 2012. Feedback control of cavity flow oscillations using simple linear models. *J. Fluid Mech.* 709:223–48
Illingworth SJ, Naito H, Fukagata K. 2014. Active control of vortex shedding: an explanation of the gain window. *Phys. Rev. E* 90:043014
Jardin T, Bury Y. 2012. Lagrangian and spectral analysis of the forced flow past a circular cylinder using pulsed tangential jets. *J. Fluid Mech.* 696:285–300
Jovanović MR, Schmid PJ, Nichols JW. 2014. Sparsity-promoting dynamic mode decomposition. *Phys. Fluids* 26:024103
Juang JN, Pappa RS. 1985. An eigensystem realization algorithm for modal parameter identification and model reduction. *J. Guid. Control Dyn.* 8:620–27
Juang JN, Phan M, Horta LG, Longman RW. 1991. *Identification of observer/Kalman filter Markov parameters: theory and experiments*. Tech. Memo. 104069, NASA Langley Res. Cent., Hampton, VA
Koopman BO. 1931. Hamiltonian systems and transformations in Hilbert space. *PNAS* 17:315–18
Koopman BO, von Neumann J. 1932. Dynamical systems of continuous spectra. *PNAS* 18:255–63
Kramer B, Grover P, Boufounos P, Benosman M, Nabi S. 2015. Sparse sensing and DMD based identification of flow regimes and bifurcations in complex flows. arXiv:1510.02831 [math.DS]
Kung SY. 1978. A new identification and model reduction algorithm via singular value decomposition. *Proc. 12th Asilomar Conf. Circuits Syst. Comput.*, pp. 705–14. New York: IEEE
Lall S, Marsden JE, Glavaški S. 1999. Empirical model reduction of controlled nonlinear systems. *Proc. IFAC World Congr.*, Vol. F, pp. 473–78. Laxenburg, Austria: Int. Fed. Autom. Control
Lall S, Marsden JE, Glavaški S. 2002. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *Int. J. Robust Nonlinear Control* 12:519–35
Laub AJ, Heath MT, Page CC, Ward RC. 1987. Computation of system balancing transformations and other applications of simultaneous diagonalization algorithms. *IEEE Trans. Autom. Control* 32:115–22
Lee JH, Seena A, Hyun Lee S, Sung HJ. 2012. Turbulent boundary layers over rod- and cube-roughened walls. *J. Turbul.* 13:N40
Luchtenburg DM, Rowley CW, Lohry MW, Martinelli L, Stengel RF. 2015. Unsteady high-angle-of-attack aerodynamic models of a generic jet transport. *J. Aircr.* 52:890–95
Lumley JL. 1970. *Stochastic Tools in Turbulence*. New York: Academic
Ma BF, Liu TX. 2014. Low-frequency vortex oscillation around slender bodies at high angles-of-attack. *Phys. Fluids* 26:091701
Ma Z, Ahuja S, Rowley CW. 2011. Reduced order models for control of fluids using the eigensystem realization algorithm. *Theor. Comput. Fluid Dyn.* 25:233–47
Markovich DM, Abdurakipov SS, Chikishev LM, Dulin VM, Hanjalić K. 2014. Comparative analysis of low- and high-swirl confined flames and jets by proper orthogonal and dynamic mode decompositions. *Phys. Fluids* 26:065109
Meslem A, Sobolik V, Bode F, Sodjavi K, Zaouali Y, et al. 2013. Flow dynamics and mass transfer in impinging circular jet at low Reynolds number: comparison of convergent and orifice nozzles. *Int. J. Heat Mass Transf.* 67:25–45
Mezić I. 2005. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dyn.* 41:309–25
Moore BC. 1981. Principal component analysis in linear systems: controllability, observability, and model reduction. *IEEE Trans. Autom. Control* 26:17–32

Motheau E, Mery Y, Nicoud F, Poinsot T. 2013. Analysis and modeling of entropy modes in a realistic aeronautical gas turbine. *J. Eng. Gas Turbines Power* 135:092602

Muld TW, Efraimsson G, Henningson DS. 2012a. Flow structures around a high-speed train extracted using proper orthogonal decomposition and dynamic mode decomposition. *Comput. Fluids* 57:87–97

Muld TW, Efraimsson G, Henningson DS. 2012b. Mode decomposition on surface-mounted cube. *Flow Turbul. Combust.* 88:279–310

Nastase I, Meslem A, Hassan ME. 2011. Image processing analysis of vortex dynamics of lobed jets from three-dimensional diffusers. *Fluid Dyn. Res.* 43:065502

Noack B, Afanasiev K, Morzyński M, Tadmor G, Thiele F. 2003. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.* 497:335–63

Noack BR, Papas P, Monkewitz PA. 2005. The need for a pressure-term representation in empirical Galerkin models of incompressible shear flow. *J. Fluid Mech.* 523:339–65

Pan C, Xue D, Wang J. 2015. On the accuracy of dynamic mode decomposition in estimating instability of wave packet. *Exp. Fluids* 56:164

Pan C, Yu D, Wang J. 2011. Dynamical mode decomposition of Gurney flap wake flow. *Theor. Appl. Mech. Lett.* 1:012002

Pernebo L, Silverman LM. 1982. Model reduction via balanced state space representations. *IEEE Trans. Autom. Control* 27:382–87

Proctor JL, Brunton SL, Kutz JN. 2014. Dynamic mode decomposition with control. arXiv:1409.6358 [math.OC]

Provansal M, Mathis C, Boyer L. 1987. Bénard–von Kármán instability: transient and forced regimes. *J. Fluid Mech.* 182:1–22

Qin SJ. 2006. An overview of subspace identification. *Comput. Chem. Eng.* 30:1502–13

Rowley CW. 2005. Model reduction for fluids using balanced proper orthogonal decomposition. *Int. J. Bifurc. Chaos* 15:997–1013

Rowley CW, Mezić I, Bagheri S, Schlatter P, Henningson DS. 2009. Spectral analysis of nonlinear flows. *J. Fluid Mech.* 641:115–27

Roy S, Hua JC, Barnhill W, Gunaratne GH, Gord JR. 2015. Deconvolution of reacting-flow dynamics using proper orthogonal and dynamic mode decompositions. *Phys. Rev. E* 91:013001

Sapsis TP, Lermusiaux PFJ. 2009. Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Physica D* 238:2347–60

Sapsis TP, Majda A. 2013. Blended reduced subspace algorithms for uncertainty quantification of quadratic systems with a stable mean state. *Physica D* 258:61–76

Sarkar S, Ganguly S, Dalal A, Saha P, Chakraborty S. 2013. Mixed convective flow stability of nanofluids past a square cylinder by dynamic mode decomposition. *Int. J. Heat Fluid Flow* 44:624–34

Sarmast S, Dadfar R, Mikkelsen RF, Schlatter P, Ivanell S, et al. 2014. Mutual inductance instability of the tip vortices behind a wind turbine. *J. Fluid Mech.* 755:705–31

Sayadi T, Schmid PJ, Nichols JW, Moin P. 2014. Reduced-order representation of near-wall structures in the late transitional boundary layer. *J. Fluid Mech.* 748:278–301

Sayadi T, Schmid PJ, Richecoeur F, Durox D. 2015. Parametrized data-driven decomposition for bifurcation analysis, with application to thermo-acoustically unstable systems. *Phys. Fluids* 27:037102

Schmid PJ. 2007. Nonmodal stability theory. *Annu. Rev. Fluid Mech.* 39:129–62

Schmid PJ. 2010. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* 656:5–28

Schmid PJ. 2011. Application of the dynamic mode decomposition to experimental data. *Exp. Fluids* 50:1123–30

Schmid PJ, Henningson DS. 2001. *Stability and Transition in Shear Flows*. Berlin: Springer-Verlag

Schmid P, Li L, Juniper M, Pust O. 2011. Applications of the dynamic mode decomposition. *Theor. Comput. Fluid Dyn.* 25:249–59

Schmid PJ, Sesterhenn J. 2008. *Dynamic mode decomposition of numerical and experimental data*. Presented at Annu. Meet. APS Div. Fluid Dyn., 61st, San Antonio, TX

Schmid PJ, Violato D, Scarano F. 2012. Decomposition of time-resolved tomographic PIV. *Exp. Fluids* 52:1567–79

Seena A, Sung HJ. 2011. Dynamic mode decomposition of turbulent cavity flows for self-sustained oscillations. *Int. J. Heat Fluid Flow* 32:1098–110

Semeraro O, Bagheri S, Brandt L, Henningson DS. 2011. Feedback control of three-dimensional optimal disturbances using reduced-order models. *J. Fluid Mech.* 677:63–102

Semeraro O, Bagheri S, Brandt L, Henningson DS. 2013. Transition delay in a boundary layer flow using active control. *J. Fluid Mech.* 731:288–311

Semeraro O, Bellani G, Lundell F. 2012. Analysis of time-resolved PIV measurements of a confined turbulent jet using POD and Koopman modes. *Exp. Fluids* 53:1203–20

Sirovich L. 1987. Turbulence and the dynamics of coherent structures, parts I–III. *Q. Appl. Math.* 45:561–90

Skogestad S, Postlethwaite I. 2005. *Multivariable Feedback Control: Analysis and Design*. New York: Wiley. 2nd ed.

Subbareddy PK, Bartkowicz MD, Candler GV. 2014. Direct numerical simulation of high-speed transition due to an isolated roughness element. *J. Fluid Mech.* 748:848–78

Taira K, Colonius T. 2007. The immersed boundary method: a projection approach. *J. Comput. Phys.* 225:2118–37

Thompson MC, Radi A, Rao A, Sheridan J, Hourigan K. 2014. Low-Reynolds-number wakes of elliptical cylinders: from the circular cylinder to the normal flat plate. *J. Fluid Mech.* 751:570–600

Tissot G, Cordier L, Benard N, Noack BR. 2014. Model reduction using dynamic mode decomposition. *C. R. Méc.* 342:410–16

Trefethen LN. 2000. *Spectral Methods in MATLAB*. Philadelphia: SIAM

Trefethen LN, Bau DI. 1997. *Numerical Linear Algebra*. Philadelphia: SIAM

Tu JH, Griffin J, Hart A, Rowley CW, Cattafesta LN III, Ukeiley LS. 2013. Integration of non-time-resolved PIV and time-resolved velocity point sensors for dynamic estimation of velocity fields. *Exp. Fluids* 54:1–20

Tu JH, Rowley CW, Kutz JN, Shang JK. 2014a. Spectral analysis of fluid flows using sub-Nyquist-rate PIV data. *Exp. Fluids* 55:1805

Tu JH, Rowley CW, Luchtenburg DM, Brunton SL, Kutz JN. 2014b. On dynamic mode decomposition: theory and applications. *J. Comput. Dyn.* 1:391–421

Verhaegen M, Dewilde P. 1992. Subspace model identification part 1. The output-error state-space model identification class of algorithms. *Int. J. Control* 56:1187–210

Viberg M. 1995. Subspace-based methods for the identification of linear time-invariant systems. *Automatica* 31:1835–51

von Neumann J. 1947. The mathematician. In *The Works of the Mind*, ed. RB Heywood, Vol. I, pp. 180–96. Chicago: Univ. Chicago Press

Willcox K. 2007. Model reduction for large-scale applications in computational fluid dynamics. In *Real-Time PDE-Constrained Optimization*, ed. LT Biegler, O Ghattas, M Heinkenschloss, D Keyes, B van Bloemen Waanders, pp. 217–32. Philadelphia: SIAM

Willcox K, Peraire J. 2002. Balanced model reduction via the proper orthogonal decomposition. *AIAA J.* 40:2323–30

Williams DR, An X, Iliev S, King R, Reißner F. 2015a. Dynamic hysteresis control of lift on a pitching wing. *Exp. Fluids* 56:112

Williams MO, Kevrekidis IG, Rowley CW. 2015b. A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. *J. Nonlinear Sci.* 25:1307–46

Williams MO, Rowley CW, Kevrekidis IG. 2014. A kernel-based approach to data-driven Koopman spectral analysis. arXiv:1411.2260 [math.DS]

Wynn A, Pearson DS, Ganapathisubramani B, Goulart PJ. 2013. Optimal mode decomposition for unsteady flows. *J. Fluid Mech.* 733:473–503

Zhou K, Salomon G, Wu E. 1999. Balanced realization and model reduction for unstable systems. *Int. J. Robust Nonlinear Control* 9:183–98

Annual Review of
Fluid Mechanics

Volume 49, 2017

Contents

Indexes

Errata

An online log of corrections to *Annual Review of Fluid Mechanics* articles may be found at http://www.annualreviews.org/errata/fluid