SIAM J. Applied Dynamical Systems
Vol. 15, No. 1, pp. 142–161

© 2016 SIAM. Published by SIAM under the terms
of the Creative Commons 4.0 license

Dynamic Mode Decomposition with Control*

Joshua L. Proctor†, Steven L. Brunton‡, and J. Nathan Kutz§

Abstract. We develop a new method which extends dynamic mode decomposition (DMD) to incorporate the effect of control to extract low-order models from high-dimensional, complex systems. DMD finds spatial-temporal coherent modes, connects local-linear analysis to nonlinear operator theory, and provides an equation-free architecture which is compatible with compressive sensing. In actuated systems, DMD is incapable of producing an input-output model; moreover, the dynamics and the modes will be corrupted by external forcing. Our new method, dynamic mode decomposition with control (DMDc), capitalizes on all of the advantages of DMD and provides the additional innovation of being able to disambiguate between the underlying dynamics and the effects of actuation, resulting in accurate input-output models. The method is data-driven in that it does not require knowledge of the underlying governing equations—only snapshots in time of observables and actuation data from historical, experimental, or black-box simulations. We demonstrate the method on high-dimensional dynamical systems, including a model with relevance to the analysis of infectious disease data with mass vaccination (actuation).

Key words. model reduction, dynamic mode decomposition, data-driven, equation-free, input-output models

AMS subject classifications. 65P99, 37M99, 37M10, 37N10, 37N35

DOI. 10.1137/15M1013857

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

1. Introduction.

We introduce the method of dynamic mode decomposition with control (DMDc) to analyze observational data arising from complex, high-dimensional systems that exhibit dynamics and require control. By utilizing both measurements of the system and the applied external control, the underlying, unforced dynamics can be extracted and specified in an *equation-free* manner; i.e., the underlying equations of motion do not have to be known. In addition, a description of how the control inputs affect the system are also discovered and characterized. With a quantitative understanding of the input-output characteristics, a reduced-order model can be generated for both prediction and design of controllers for high-dimensional, complex systems.

Controlling high-dimensional systems remains an extremely challenging task as many control strategies do not scale well with the dimension of the system. In particular, controllers developed on a full system may be computationally prohibitive to implement, introducing unacceptably large latencies [2]. Moreover, many control laws are determined by solving a large Riccati equation ($\mathcal{H}_2$) or through an iterative procedure ($\mathcal{H}_\infty$), constituting an enormous

*Received by the editors March 25, 2015; accepted for publication (in revised form) December 1, 2015; published electronically January 26, 2016.
http://www.siam.org/journals/siads/15-1/M101385.html

†Corresponding author. Institute for Disease Modeling, Bellevue, WA 98004 (joproctor@intven.com).
‡Mechanical Engineering and Applied Mathematics, University of Washington, Seattle, WA 98195 (sbrunton@uw.edu).
§Applied Mathematics, University of Washington, Seattle, WA 98195 (kutz@uw.edu). This author acknowledges support from the U.S. Air Force Office of Scientific Research (FA9550-09-0174).

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

upfront cost. Thus, practical engineering control strategies for dealing with high-dimensional observational data revolve around dimensionality-reduction techniques. Such methods, often based on the singular value decomposition (SVD) of the data, allow one to construct low-dimensional subspaces where computationally tractable controllers can be designed and implemented [35, 24, 20, 40, 39, 58, 18]. Balanced truncation is a classic method developed to specifically take advantage of underlying low-dimensional observable and controllable subspaces to create a balanced, reduced-order model [35]. Generalizations of this scheme which combine balanced truncation with the SVD on empirical data, such as the balanced proper orthogonal decomposition, have already been shown to overcome some of the computational difficulties associated with the high dimension of complex systems, but still require a pernicious linear adjoint calculation [57, 40, 21]. Further innovations around system identification methods, such as the Eigensystem Realization Algorithm (ERA) and the Observer Kalman Filter Identification (OKID), were developed to aid in the discovery of input-output models for systems with control [24, 25, 13]. The dimensions of the measurements, though, were assumed to be low and the system linear. Recent work has demonstrated that ERA can produce exactly the same balanced input-output models as those produced by Balanced Proper Orthogonal Decomposition (BPOD) [31]. Other system identification methods called subspace identification methods are also focused on constructing input-output models from measurement data, but these techniques bypass the identification of Markov parameters [38].

DMDc has a number of advantages for high-dimensional, complex systems. First, it is based upon the dynamic mode decomposition (DMD) algorithm, which is a data-driven, *equation-free* architecture that reconstructs the underlying dynamics of the system from snapshot measurements alone [46, 45, 41, 43, 9, 52, 51]. Substantial success has been achieved in the application of DMD to fields such as fluid dynamics which have been historically difficult to analyze and construct controllers due to the enormous number of spatial states required for simulation [16, 44, 47, 3, 53, 51]. Second, DMD has acquired popularity as a method for systems with nonlinear dynamics, due to a strong connection between DMD and Koopman operator theory [29, 32, 41, 7, 33]. Finally, DMD can be modified to take advantage of sparse, or limited, measurements of the complex system [6, 53, 22]. Sparse measurements have recently been leveraged in a variety of complex systems—some for control [4, 42, 30, 12]. Such a scenario arises in many physical, biological, and engineering systems due to limited numbers of sensors. Such advantages, in combination with the control architecture advocated here, warrant serious consideration of the DMDc as an equation-free control strategy in complex systems.

As a motivating example, DMDc can be applied to the field of computational epidemiology focusing on the eradication of diseases. The advent of new monitoring tools and a substantial focus on the quantitative assessment of resource allocation are beginning to generate large sets of data describing the spread of infectious disease. A substantial literature exists focused on mathematically modeling the spread of infectious disease and the effect of external control (e.g., vaccinations for Polio and bed nets for Malaria) [1]. A common challenge in computational epidemiology is deciding how to model the spread of disease leading to an enormous number of phenomenological models [28]. Equation-free techniques such as DMD and DMDc provide a complementary modeling tool for analyzing the spatial-temporal spread of infectious disease. Focusing on *only* the historical data containing state information (i.e., number

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

of infections in a spatial location in a given time) and whether control interventions have been applied (i.e., number of vaccinations in a spatial location in a given time), DMDc discovers the dynamical properties of the complex systems.

The outline of the paper is as follows: Section 2 describes the background on the method DMD. Section 3 describes the new method dynamic mode decomposition with control (DMDc). Section 4 presents a number of numerical examples including an artificial application based on an epidemiological problem. Section 5 discusses a number of similarities to and differences from system identification methods.

2. Background: Dynamic mode decomposition. Dynamic mode decomposition (DMD) is a powerful data-driven method for analyzing complex systems. Using measurement data from numerical simulations or laboratory experiments, DMD attempts to extract important dynamic characteristics such as unstable growth modes, resonance, and spectral properties. This section provides the mathematical background of DMD [46, 45, 41, 43, 52].

2.1. Dynamical systems and data. At a fundamental level, DMD analyzes the relationship between pairs of measurements from a dynamical system. The measurements, $\mathbf{x_k}$ and $\mathbf{x_{k+1}}$, where k indicates the temporal iteration from a discrete dynamical system, are assumed to be approximately related by a linear operator:

$$\mathbf{x}_{k+1} \approx \mathbf{A}\mathbf{x}_k, \tag{2.1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{A} \in \mathbb{R}^{n\times n}$. This approximation is assumed to hold for all pairs of measurements. The subsequent description and discussion of DMD will be centered around finding a best-fit solution of the operator $\mathbf{A}$ for all pairs of measurements. It is important to note that the relationship in (2.1) does not need to hold exactly. Previous work has theoretically justified using this approximating operator on data generated by nonlinear dynamical systems [52]. Also, DMD has been primarily applied to data collected from high-dimensional nonlinear systems [16, 44, 47, 3, 53, 51].

The process under observation is often continuous (whether from a numerical model or experiment), and measurements $\mathbf{x}(t)$ can be collected at regular time intervals Δt denoted by $\mathbf{x}_k = \mathbf{x}(k\Delta t)$. Each measurement in time $\mathbf{x}_k$ will be referred to as snapshots within this manuscript [48, 49, 50]. Note that previous applications of DMD to numerical simulations of fluid dynamic problems often assume full-state access [41]. The more general case is described here where measurements of a system, whether historical, numerical, or experimental, are utilized to construct the data matrix. A control theoretic perspective would frame (2.1) with a measurement function. A linear measurement function $\mathbf{x} = \mathbf{C}\tilde{x}$ would suggest there is an underlying system involving the variable $\tilde{\mathbf{x}}$. In section 5, we demonstrate how this measurement function can take different forms. For example in DMD, the matrix $\mathbf{C}$ acts to reduce the dimension of the system, whereas for the Eigensystem Realization Algorithm (ERA) the matrix acts to increase the dimension of the system. A comparison of DMD to system identification methods with this perspective is included in section 5. We denote the sequence of snapshots collected by the following description:

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_{m-1} \\ | & | & & | \end{bmatrix},$$

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

$$\mathbf{X}' = \begin{bmatrix} | & | & & | \\ \mathbf{x}_2 & \mathbf{x}_3 & \dots & \mathbf{x}_m \\ | & | & & | \end{bmatrix}, \tag{2.2}$$

where m is the total number of snapshots and $\mathbf{X}'$ is the time-shifted snapshot matrix of $\mathbf{X}$, i.e., $\mathbf{X}' = \mathbf{A}\mathbf{X}$. For DMD, data is often collected at regular time intervals Δt. The number of snapshots required for DMD varies with the application, but is intimately related to the linearity properties of the Koopman operator. The solution will converge by decreasing the recording interval $\Delta t \to 0$, thus indicating the number of snapshots required (for an illuminating numerical example see [43]). New directions for DMD have focused on novel paradigms for collecting data in time [52] and across the state of the system [6]. Each utilizes the concepts of sparsity and compressed sensing techniques [5].

The relationship between pairs of measurement in (2.1) and the combined data snapshots (2.2) can be described more compactly in the following matrix form:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X}. \tag{2.3}$$

Solving for an approximation of the process matrix $\mathbf{A}$ for the measurement matrix pair $\mathbf{X}$ and $\mathbf{X}'$ is the primary objective of DMD.

2.2. Dynamic mode decomposition. The following section describes how to find the dynamic modes and eigenvalues of the underlying system $\mathbf{A}$ described in (2.3). The DMD of the measurement matrix pair $\mathbf{X}$ and $\mathbf{X}'$ is the eigendecomposition of the matrix $\mathbf{A}$. The operator $\mathbf{A}$ is defined by the following:

$$\mathbf{A} = \mathbf{X}'\mathbf{X}^{\dagger}, \tag{2.4}$$

where † is the Moore–Penrose pseudoinverse [52]. A least-squares solution $\mathbf{A}$ to the underdetermined problem $\mathbf{X}' = \mathbf{A}\mathbf{X}$ can be found by minimizing the Frobenius norm of $\|\mathbf{X}' - \mathbf{A}\mathbf{X}\|_F$. When the problem is overdetermined, $\mathbf{A}$ can be solved as a minimum-norm solution [52]. As mentioned earlier, the relationship does not need to hold exactly. A computationally efficient and accurate method for finding the pseudoinverse is via the SVD. The SVD of $\mathbf{X}$ results in the well-known decomposition

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* = \begin{bmatrix} \tilde{\mathbf{U}} & \tilde{\mathbf{U}}_{\text{rem}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{\Sigma}} & 0 \\ 0 & \mathbf{\Sigma}_{\text{rem}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}^* \\ \tilde{\mathbf{V}}^*_{\text{rem}} \end{bmatrix} \tag{2.5}$$

$$\approx \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*, \tag{2.6}$$

where $\mathbf{U} \in \mathbb{R}^{n\times n}$, $\mathbf{\Sigma} \in \mathbb{R}^{n\times m-1}$, $\tilde{\mathbf{V}}^* \in \mathbb{R}^{m-1\times m-1}$, $\tilde{\mathbf{U}} \in \mathbb{R}^{n\times r}$, $\tilde{\mathbf{\Sigma}} \in \mathbb{R}^{r\times r}$, $\tilde{\mathbf{V}}^* \in \mathbb{R}^{r\times m-1}$, $_{\text{rem}}$ indicates the remaining $m-1-r$ singular values, and * denotes the complex conjugate transpose. Equation (2.6) demonstrates how to reduce the dimension of the data matrix $\mathbf{X}$ by appropriately choosing a truncation value r of the singular values, thus eliminating the remainder (rem) terms and allowing for the psuedoinverse to be accomplished since $\tilde{\mathbf{\Sigma}}$ is square. Choosing the appropriate truncation value r has a rich scientific history; notably, the Eckart–Young theorem provides a rigorous and popular method for choosing r [11, 34, 15]. In

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

addition, there are recent theoretical developments attempting to identify the correct r when $\mathbf{X}$ may have additive noise [10, 14].

Using the SVD of the snapshot matrix $\mathbf{X}$ in (2.6), the following approximation of the matrix $\mathbf{A}$ can be computed:

$$\mathbf{A} \approx \bar{\mathbf{A}} = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}^*, \tag{2.7}$$

where $\bar{\mathbf{A}}$ is an approximation of the operator $\mathbf{A}$ from (2.6). A dynamic model of the process can be constructed given by the following:

$$\mathbf{x}_{k+1} = \bar{\mathbf{A}}\mathbf{x}_k, \tag{2.8}$$

where $\mathbf{x}$ and $\bar{\mathbf{A}}$ have the same dimension as the matrices described earlier in (2.1). An eigenvalue analysis of the matrix $\bar{\mathbf{A}}$ would produce the dynamic modes and eigenvalues of the system. The computation, though, can be prohibitively expensive if $n \gg 1$.

If $r \ll n$, a more compact and computationally efficient model can be found by projecting $\mathbf{x}_k$ onto a linear subspace of dimension r. This basis transformation takes the form $\mathbf{P}\mathbf{x} = \tilde{\mathbf{x}}$. As previously shown by DMD, a convenient transformation has already been computed via the SVD of $\mathbf{X}$, given by $\mathbf{P} = \tilde{\mathbf{U}}^*$. The reduced-order model can be derived as follows:

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{U}}^*\bar{\mathbf{A}}\tilde{\mathbf{U}}\tilde{\mathbf{x}}_k \tag{2.9}$$

$$= \tilde{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{x}}_k \tag{2.10}$$

$$= \tilde{\mathbf{A}}\tilde{\mathbf{x}}_k. \tag{2.11}$$

The reduced-order model is given by the following:

$$\tilde{\mathbf{A}} = \tilde{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}. \tag{2.12}$$

The eigendecomposition of $\tilde{\mathbf{A}}$ defined by $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\Lambda$ yields eigenvalues and eigenvectors that can be investigated for fundamental properties of the underlying system such as growth modes and resonance frequencies. In addition, the computation is efficient since $\tilde{\mathbf{A}} \in \mathbb{R}^{r\times r}$ and $r \ll n$.

Remark. Computing the eigendecomposition of $\tilde{\mathbf{A}}$ versus $\bar{\mathbf{A}}$ can be a computationally crucial step for efficiency. For example, the domain discretization of a fluids or epidemiological problem can have an arbitrarily large set of dimensions n. The direct solution of the $n \times n$ eigenvalue problem might not be feasible; thus solving the $r \times r$ is substantially more attractive. The observation is reminiscent of the method of snapshots by Sirovich [48, 49, 50].

For DMD, the eigenvalues of $\tilde{\mathbf{A}}$ and $\bar{\mathbf{A}}$ are equivalent [43], and the eigenvectors are related via a linear transformation. The eigenvectors of $\bar{\mathbf{A}}$ are called dynamic modes [43, 52]. Note that there is a difference between computing the dynamic modes with the Exact DMD method from Tu et al. [52] and Schmid [43]. Here we describe the Exact DMD method giving the following relationship between the eigenvectors of $\tilde{\mathbf{A}}$ and the dynamic modes ϕ of $\bar{\mathbf{A}}$:

$$\phi = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\mathbf{w}. \tag{2.13}$$

If $\lambda \neq 0$, then this is the DMD mode for λ. If the eigenvalue is 0, then the dynamic mode is computed using $\phi = \tilde{\mathbf{U}}\mathbf{w}$. The Exact DMD algorithm has a number of advantages over the original procedure; for a detailed discussion, see [52].

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

3. Dynamic mode decomposition with control. This section presents the mathematical description of dynamic mode decomposition with control (DMDc). Understanding the dynamic characteristics of complex systems that have both internal dynamics and applied external control is fundamental to controller design and sensor placement. The DMDc method helps discover the underlying dynamics without the confounding effect of external control. In addition, the method also quantifies the effect of control inputs on the measurements of the system. Figure 1 illustrates the data collection, the algorithm, and applications of DMDc.

The underlying dynamical system and measured data matrices are redefined to include systems with control inputs in section 3.1. The subsequent section, section 3.2, describes how to solve for the dynamic modes if the effect of the inputs on the system is already well known or well estimated. The last section, section 3.3, shows how to solve for both the dynamic modes and the input matrix.

3.1. Dynamical system with control. The new method modifies the basic assumption of DMD. Now, a trio of measurements are assumed to be connected. The goal of DMDc is to analyze the relationship between a future system measurement $\mathbf{x}_{k+1}$ with the current measurement $\mathbf{x}_k$ *and* the current control $\mathbf{u}_k$. For each trio of measurement data, a pair of linear operators provides the following relationship:

$$\mathbf{x}_{k+1} \approx \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{3.1}$$

where $\mathbf{x}_j \in \mathbb{R}^n$, $\mathbf{u}_j \in \mathbb{R}^l$, $\mathbf{A} \in \mathbb{R}^{n\times n}$, and $\mathbf{B} \in \mathbb{R}^{n\times l}$. The operators $\mathbf{A}$ and $\mathbf{B}$ are best-fit solutions for all trios of data. Similar to DMD, the relationship in (3.1) does not need to hold exactly. Data matrices can be constructed with temporal snapshots of the system measurements and control input over time. The measurement snapshot matrices, $\mathbf{X}$ and $\mathbf{X}'$, are collected in the same manner as in (2.2). We denote a new sequence of control input snapshots collected by the following description:

$$\mathbf{\Upsilon} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_{m-1} \\ | & | & & | \end{bmatrix}. \tag{3.2}$$

Equation (3.1) can be rewritten in matrix form to include the new data matrices:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{\Upsilon}. \tag{3.3}$$

Utilizing the three data matrices, DMDc is focused on finding best-fit approximations to the mappings $\mathbf{A}$ and $\mathbf{B}$. As noted in section 2.1, a natural extension to (3.1) would be to consider a measurement equation. Here, a linear measurement equation $\mathbf{x} = \mathbf{C}\tilde{\mathbf{x}} + \mathbf{D}\mathbf{u}$ suggests there is an underlying system with $\tilde{\mathbf{x}}$. This perspective is discussed in more detail in section 5, where we elaborate on how the measurement equation fits into the definition of DMD and system identification methods such as ERA. In the following two sections, we describe how to find the dynamic modes of $\mathbf{A}$ given the inclusion of control snapshots. The first section outlines the analysis and algorithm if the matrix $\mathbf{B}$ is known or well estimated. If unknown, the second section describes how to discover both $\mathbf{A}$ and $\mathbf{B}$ from the observation matrices.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

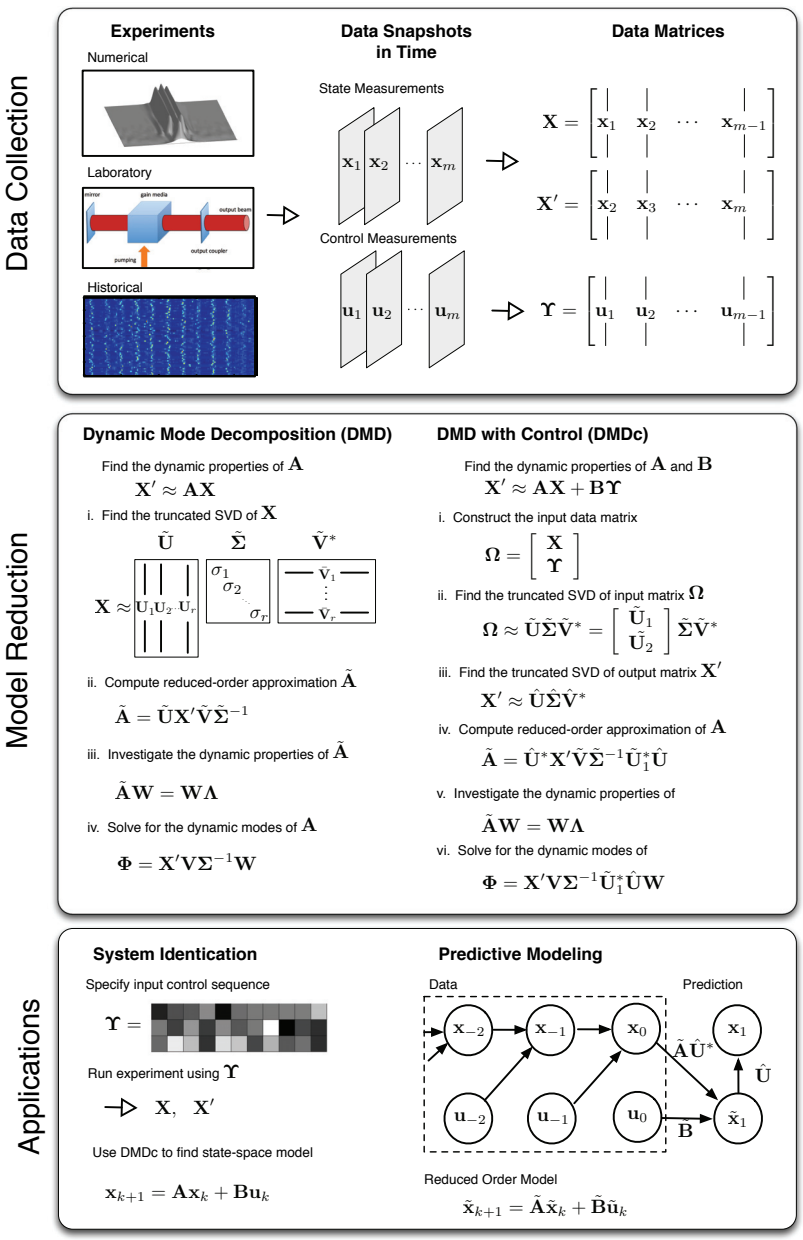

Figure 1. *The illustration outlines the three major components of applying DMDc. The top panel describes the collection of data from numerical, laboratory, or historical data and the curation of the data into matrices for the methods. Note that the figure in the historical plot is the data representing prevaccination Measles cases in the UK normalized similarly to that found in [27]. The middle panel outlines the procedure for DMD and DMDc for comparison. The bottom panel illustrates two practical applications of DMDc.*

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

3.2. The map B is known. This section describes how to find the dynamic modes and eigenvalues of the underlying system $\mathbf{A}$ when the matrix $\mathbf{B}$ is known. The assumption that $\mathbf{B}$ is known or well estimated is an idealistic view of most complex systems, but it helps provide one of the major motivations for this work. Finding the underlying dynamics $\mathbf{A}$ in a complex system where control has been applied is essential for designing controllers and placement of sensors. If external control has been applied to the system, standard DMD would produce incorrect dynamic information. The more general case where $\mathbf{B}$ is unknown will be described in the following section.

Equation (3.3) can be rearranged by pairing the time-shifted measurement snapshot matrix with the control snapshot matrix and the known matrix $\mathbf{B}$:

$$\mathbf{X}' - \mathbf{B}\boldsymbol{\Upsilon} \approx \mathbf{A}\mathbf{X}. \tag{3.4}$$

The mapping $\mathbf{A}$ can be solved for similarly to (2.4). Again, the truncated SVD of $\mathbf{X}$ gives the matrix factorization $\tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*$. Thus, the approximation of $\mathbf{A}$ is given by the following description:

$$\mathbf{A} \approx \bar{\mathbf{A}} = (\mathbf{X}' - \mathbf{B}\boldsymbol{\Upsilon})\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}^*. \tag{3.5}$$

Note that if the control snapshots are $\mathbf{u}_j = \mathbf{0}\ \forall\ j\ \in [1, m-1]$, then the derivation is equivalent to DMD. A dynamic model of both the computed process and the given input matrix can be constructed described by the following:

$$\mathbf{x}_{k+1} = \bar{\mathbf{A}}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{3.6}$$

where $\mathbf{x}$, $\bar{\mathbf{A}}$, and $\mathbf{B}$ are the same dimensions of the matrices described earlier in (2.1). If $r \ll n$, though, a more compact and computationally efficient model can be found using the same basis transformation $\mathbf{P}\mathbf{x} = \tilde{\mathbf{x}}$ as described earlier for DMD. Again, a convenient transformation has already been computed via the SVD of $\mathbf{X}$, given by $\mathbf{P} = \tilde{\mathbf{U}}^*$. The reduced-order model can be derived as follows:

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{U}}^*\bar{\mathbf{A}}\tilde{\mathbf{U}}\tilde{\mathbf{x}}_k + \tilde{\mathbf{U}}^*\mathbf{B}\mathbf{u}_k \tag{3.7}$$

$$= \tilde{\mathbf{U}}^*(\mathbf{X}' - \mathbf{B}\boldsymbol{\Upsilon})\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{x}}_k + \tilde{\mathbf{U}}^*\mathbf{B}\mathbf{u}_k \tag{3.8}$$

$$= \tilde{\mathbf{A}}\tilde{\mathbf{x}}_k + \tilde{\mathbf{B}}\mathbf{u}_k. \tag{3.9}$$

The reduced-order approximation of $\mathbf{A}$ is given by the following:

$$\tilde{\mathbf{A}} = \tilde{\mathbf{U}}^*(\mathbf{X}' - \mathbf{B}\boldsymbol{\Upsilon})\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}. \tag{3.10}$$

The eigendecomposition of $\tilde{\mathbf{A}}$ defined by $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\boldsymbol{\Lambda}$ yields eigenvectors that can be used to find the dynamic modes. Similar to Exact DMD, the dynamic modes can be found with the following description:

$$\phi = (\mathbf{X}' - \mathbf{B}\boldsymbol{\Upsilon})\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\mathbf{w}. \tag{3.11}$$

If $\lambda \neq 0$, then this is the DMD mode for λ. If the eigenvalue is 0, then the dynamic mode is computed using $\phi = \tilde{\mathbf{U}}\mathbf{w}$.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

3.3. The map B is unknown. The assumption that $\mathbf{B}$ is known indicates a significant amount of knowledge about how control inputs affect the system. This section relaxes that assumption and notably demonstrates that approximations of the matrices $\mathbf{A}$ and $\mathbf{B}$ can both be found from system measurements and control snapshots. To the experimentalist or analyst, this is by far more interesting since only the snapshots of the control and output measurements are required to find the properties of the underlying process $\mathbf{A}$ and how that process is affected by input control given by the operator $\mathbf{B}$.

The approximate relationship between the data matrices $\mathbf{X}$, $\mathbf{\Upsilon}$, and $\mathbf{X}'$ in (3.3) can be rewritten in the following form:

$$\mathbf{X}' \approx \mathbf{G\Omega}, \tag{3.12}$$

where $\mathbf{G} = [\mathbf{A} \;\; \mathbf{B}]$ and $\mathbf{\Omega} = \left[\begin{smallmatrix}\mathbf{X}\\ \mathbf{\Upsilon}\end{smallmatrix}\right]$. Thus, similar to the definition of DMD, DMDc of the measurement trio $\mathbf{X}$, $\mathbf{\Upsilon}$, and $\mathbf{X}'$ is the eigendecomposition of the operator $\mathbf{A}$ defined by the following:

$$\mathbf{G} = \mathbf{X}'\mathbf{\Omega}^{\dagger}, \tag{3.13}$$

$$[\mathbf{A} \;\; \mathbf{B}] = \mathbf{X}' \begin{bmatrix} \mathbf{X} \\ \mathbf{\Upsilon} \end{bmatrix}^{\dagger}, \tag{3.14}$$

where $\mathbf{\Omega}$ contains both the measurement and control snapshot information. Here, we again seek a best-fit solution of the operator $\mathbf{G}$ which now contains the process dynamics $\mathbf{A}$ and input matrix $\mathbf{B}$. Further, a least-squares solution $\mathbf{G}$ to the underdetermined problem $\mathbf{X}' = \mathbf{G\Omega}$ can be found by minimizing the Frobenius norm of $\|\mathbf{X}' - \mathbf{G\Omega}\|_F$. Note that, similar to DMD, the equality does not need to hold exactly. To solve, we again utilize an SVD, now performed on the augmented data matrix giving $\mathbf{\Omega} = \mathbf{U\Sigma V}^* \approx \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*$. The truncation value of the SVD for $\mathbf{\Omega}$ will be defined as p. Note that the truncation value of $\mathbf{\Omega}$ should be larger than that of $\mathbf{X}$. The following computation provides an approximation of $\mathbf{G}$:

$$\mathbf{G} \approx \bar{\mathbf{G}} = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}^*, \tag{3.15}$$

where $\mathbf{G} \in \mathbb{R}^{n\times(n+l)}$. We can now find approximations of the matrices $\mathbf{A}$ and $\mathbf{B}$ by breaking the linear operator $\tilde{\mathbf{U}}$ into two separate components given by the following:

$$[\mathbf{A}, \;\; \mathbf{B}] \approx [\bar{\mathbf{A}}, \;\; \bar{\mathbf{B}}] \tag{3.16}$$

$$\approx [\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*, \;\; \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_2^*], \tag{3.17}$$

where $\tilde{\mathbf{U}}_1 \in \mathbb{R}^{n\times p}$, $\tilde{\mathbf{U}}_2 \in \mathbb{R}^{l\times p}$, and $\tilde{\mathbf{U}}^* = [\tilde{\mathbf{U}}_1^* \;\; \tilde{\mathbf{U}}_2^*]$. Similar to (3.17), a dynamic model using the matrices $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$, but for a large dimensional system where $n \gg 1$, this is computationally prohibitive. Here, we again seek a reduced-order model of rank $r \ll n$ where a transformation is required such that $\mathbf{x} = \mathbf{P}\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}} \in \mathbb{R}^r$.

Unlike DMD, the truncated left singular vectors $\tilde{\mathbf{U}}$ cannot be used to define the subspace on which the state evolves. For (3.17), the truncated left singular vectors of $\mathbf{\Omega}$ define the *input* space. To find a linear transformation $\mathbf{P}$ for the measurement $\mathbf{x}$, we utilize a reduced-order subspace of the *output* subspace. This fundamental observation allows for DMDc to discover a reduced-order representation of the dynamics $\mathbf{A}$ and input matrix $\mathbf{B}$.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

To find the reduced-order subspace of the output space, a second SVD is required. The data matrix of the *output* space $\mathbf{X}'$ can be approximated by the familiar SVD $\hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^*$, where the truncation value is r and $\hat{\mathbf{U}} \in \mathbb{R}^{n\times r}$, $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{r\times r}$, and $\hat{\mathbf{V}}^* \in \mathbb{R}^{r\times m-1}$. Note that the two SVDs will likely have different truncation values of the input and output matrices p and r and $p > r$. Using the transformation $\mathbf{x} = \hat{\mathbf{U}}\tilde{\mathbf{x}}$, the following reduced-order approximations of $\mathbf{A}$ and $\mathbf{B}$ can be computed:

$$\tilde{\mathbf{A}} = \hat{\mathbf{U}}^*\bar{\mathbf{A}}\hat{\mathbf{U}} = \hat{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*\hat{\mathbf{U}}, \tag{3.18}$$

$$\tilde{\mathbf{B}} = \hat{\mathbf{U}}^*\bar{\mathbf{B}} \quad = \hat{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_2^*, \tag{3.19}$$

where $\tilde{\mathbf{A}} \in \mathbb{R}^{r\times r}$ and $\tilde{\mathbf{B}} \in \mathbb{R}^{r\times l}$. We can then form the reduced-order equation as (3.9) given by the following:

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{A}}\tilde{\mathbf{x}}_k + \tilde{\mathbf{B}}\mathbf{u}_k. \tag{3.20}$$

Similar to DMD, the dynamic modes of $\mathbf{A}$ can be found by first solving the eigenvalue decomposition $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}$. The transformation from eigenvectors to dynamic modes of $\mathbf{A}$ is slightly modified and is given by the following:

$$\phi = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*\hat{\mathbf{U}}\mathbf{w}, \tag{3.21}$$

where the relationship between ϕ and $\mathbf{w}$ is similar to Exact DMD. Note that the second SVD on the state is utilized to fulfill the requirement of exactness for constructing the dynamic modes as in [52].

3.4. The algorithm. This section outlines the algorithm.

1. *Collect and construct the snapshot matrices.* Collect the system measurement and control snapshots, and form the matrices $\mathbf{X}$, $\mathbf{X}'$, and $\mathbf{\Upsilon}$ as described in (2.2) and (3.2). Stack the data matrices $\mathbf{X}$ and $\mathbf{\Upsilon}$ to construct the matrix $\mathbf{\Omega}$.
2. *Compute the SVD of the input space* $\mathbf{\Omega}$. Compute the SVD of $\mathbf{\Omega}$ as described in (2.6), thereby obtaining the decomposition $\mathbf{\Omega} \approx \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*$ with truncation value p.
3. *Compute the SVD of the output space* $\mathbf{X}'$. Compute the SVD of $\mathbf{X}'$ as described in (2.6), thereby obtaining the decomposition $\mathbf{X}' \approx \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^*$ with truncation value r.
4. *Compute the approximation of the operators* $\mathbf{G} = [\mathbf{A} \;\; \mathbf{B}]$. Compute the following:

$$\tilde{\mathbf{A}} = \hat{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*\hat{\mathbf{U}}, \tag{3.22}$$

$$\tilde{\mathbf{B}} = \hat{\mathbf{U}}^*\mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_2^*. \tag{3.23}$$

5. *Perform the eigenvalue decomposition of* $\tilde{\mathbf{A}}$. Perform the eigenvalue decomposition given by the following:

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}. \tag{3.24}$$

6. *Compute the dynamic modes of the operator* $\mathbf{A}$.

$$\mathbf{\Phi} = \mathbf{X}'\tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1}\tilde{\mathbf{U}}_1^*\hat{\mathbf{U}}\mathbf{W}. \tag{3.25}$$

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

4. Applications. This section describes a number of numerical examples for the application of this method. The examples increase in complexity as the section progresses. The emphasis for each of these examples is the benefit of including control snapshot information to the analysis.

4.1. Example 1: Unstable linear system with proportional controller. DMDc can help discover the underlying dynamics of a system through measurements of both the state and external inputs. Here, we demonstrate the idea on a simple two-dimensional unstable linear system with a stabilizing controller. Despite the simplicity of the mathematical problem, the example is illustrative for the general concept of DMDc. Consider the following dynamical system:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{k+1} = \begin{bmatrix} 1.5 & 0 \\ 0 & 0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \tag{4.1}$$

where $u_k = K[x_1]_k$ and $K = -1$. The proportional controller clearly stabilizes the system by moving the unstable eigenvalue within the unit circle. If we have access to the input data and the $\mathbf{B}$ matrix as described in section 3.3, we can collect state and control snapshots to perform the DMDc computation. For an initial condition $[4 \quad 7]^T$, the following are the data matrices constructed from computing the first five temporal snapshots of (4.1):

$$\mathbf{X} = \begin{bmatrix} 4 & 2 & 1 & 0.5 \\ 7 & 0.7 & 0.07 & 0.007 \end{bmatrix}, \tag{4.2}$$

$$\mathbf{X}' = \begin{bmatrix} 2 & 1 & 0.5 & 0.25 \\ 0.7 & 0.07 & 0.007 & 0.0007 \end{bmatrix}, \tag{4.3}$$

$$\boldsymbol{\Upsilon} = \begin{bmatrix} -4 & -2 & -1 & -0.5 \end{bmatrix}. \tag{4.4}$$

Following the description in section 3.2, we compute the SVD of $\mathbf{X}$. Here, we use the MATLAB economy-sized SVD algorithm to give the following matrix factorization of $\mathbf{X}$:

$$\tilde{\mathbf{U}} = \begin{bmatrix} -0.5239 & -0.8462 \\ -0.8462 & 0.5329 \end{bmatrix}, \tag{4.5}$$

$$\tilde{\boldsymbol{\Sigma}} = \begin{bmatrix} 8.2495 & 0 \\ 0 & 1.6402 \end{bmatrix}, \tag{4.6}$$

$$\tilde{\mathbf{V}} = \begin{bmatrix} -0.9764 & 0.2105 \\ -0.2010 & -0.8044 \\ -0.0718 & -0.4932 \\ -0.0330 & -0.2557 \end{bmatrix}. \tag{4.7}$$

Now, we can compute (3.5) using the data matrices in (4.4), the SVD matrices in (4.7), and the matrix $\mathbf{B}$ in (4.1), giving the following approximation to $\mathbf{A}$:

$$\bar{\mathbf{A}} = \begin{bmatrix} 1.5 & 0 \\ 0 & 0.1 \end{bmatrix}, \tag{4.8}$$

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

where we recover the unstable linear dynamics from data of the state and control snapshots. This example demonstrates the utility of DMDc with recovering unstable dynamics from a system that would otherwise appear to be stable.

Note that this example assumes that the matrix $\mathbf{B}$ is known. This can be prohibitively restrictive when investigating systems with state-dependent feedback. With a control signal that is dependent on the state, it is impossible to reconstruct $\mathbf{B}$ using the method described in section 3.3. The matrix $\mathbf{B}$ can be recovered, though, if a small random disturbance is added to the controller $u_k = K[x_1]_k + \delta_k$, where each δ is drawn from a Gaussian distribution with zero mean. This breaks the symmetry of the state-dependent feedback, allowing both $\mathbf{A}$ and $\mathbf{B}$ to be investigated independently. Further, this provides the numericist or experimentalist a procedure for investigating high-dimensional systems with state feedback.

4.2. Example 2: Large-scale, stable linear systems. In this section, we investigate stable linear systems where the number of measurements is significantly greater than the dimensionality of the underlying system. The previous example demonstrated the utility of the method on a low-dimensional unstable model. Here, the method is applied to large-scale dynamical systems that have an underlying low-dimensional attractor.

To construct these large-scale systems, a low-dimensional stable model is generated and subsequently embedded into a higher-dimensional subspace. There are three steps for generating the model and data matrices to compare the output of DMDc and the generated model:

1. *Generate a low-dimensional stable state-space model,* $\mathbf{A}$ *and* $\mathbf{B}$*.* Generate discrete random state-space systems using the MATLAB command Discrete Random State Space Method. These stable-discrete state-space models can be used as numerical experiments for DMDc. Here, we have chosen a five-dimensional model, two input variables, and 100 measurement variables. The output is a state-space model $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$, and $\mathbf{C}$.
2. *Generate random input data* $\boldsymbol{\Upsilon}$*.* Using the MATLAB randn command, generate a matrix of random inputs, $\boldsymbol{\Upsilon} \in \mathbb{R}^{2\times m-1}$.
3. *Use the model and input vector to generate the data matrices* $\mathbf{X}$ *and* $\mathbf{X}'$*.* Using the model and the input matrix, generate output data for the snapshot matrix.

Using the data matrices $\mathbf{X}$, $\mathbf{X}'$, and $\boldsymbol{\Upsilon}$, the DMDc computation can be performed to find an approximation of $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{B}}$. To compare the generated model and the model produced by DMDc, we assign $\tilde{\mathbf{C}} = \hat{\mathbf{U}}$. The assignment allows for the comparison of state-space models.

The singular values of the frequency response, a multi-input multi-output (MIMO) generalization of a Bode plot, are used to compare the two models. The MATLAB command sigma will generate the frequency response for both systems. Figure 2 illustrates one such comparison arising from a single numerical realization from the ensemble. Note that there is no distinction between the generated model (in red) and the model from DMDc (in blue) for both control inputs (both lines).

4.3. Example 3: A sparse linear system in the Fourier domain. The final example for DMDc is a large-scale dynamical system on a spatial grid. The system consists of high-dimensional full-state measurements, although the dynamics are governed by a low-dimensional dynamical system in the Fourier domain. The motivation for this example

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

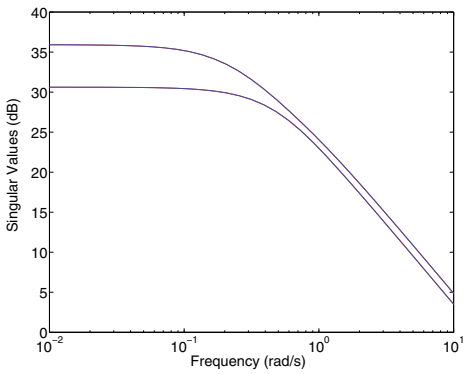

Figure 2. *The singular values of the frequency response for a large-scale stable linear system. The blue line is from the model from DMDc, and the red is from the real model. Note that an equivalent frequency response cannot be constructed from DMD alone since it does not consider input-output systems.*

comes from epidemiology and infectious disease spread, where the measurements can be high-dimensional in both space and time. For example, consider the number of possible states of a dynamical system to represent flu infections across the world over a decade, including both spatial discretization and disease heterogeneity factors. In this example, the underlying attractor could be quite low-dimensional. To complicate this picture, actuation in the form of a spatial delivery of vaccinations is also occurring each year, which can directly affect the dynamics of an infectious disease.

Here, we construct a sparse dynamical system in a two-dimensional Fourier domain as an abstraction of the problem described above. Only five modes are allowed to be nonzero. The dynamical system on these spatial modes is constructed in the following way: for each mode, a temporal oscillation frequency is chosen randomly, and a small, stable damping rate is similarly chosen. The boundary conditions are periodic, thus restricting the dynamics to a torus. This system was previously constructed in [6] to demonstrate compressive DMD. Here, though, the example is extended to allow for actuation in the spatial domain. The spatial actuation is then Fourier transformed in order to compute the effect on the underlying dynamical system. The spatial grid used is 128×128.

Similar to the previous examples, the underlying dynamics of the system can be discovered solely from state and control snapshots in the spatial domain using DMDc. The top left plot of Figure 3 shows the evolution of one such unforced system in space. The right plot shows the effect of actuation on the same system. The actuation is a localized negative control input applied in the spatial domain, shown in the lower left plot. The eigenvalue plot shows that DMDc discovers the underlying eigenvalues more accurately than DMD. In addition, the zero-valued Fourier modes can be contaminated with Gaussian noise without a qualitative change in the behavior of DMDc.

5. Connections to system identification methods. In comparing DMDc with the system identification methods described in this section, we first discuss a number of important general differences. First, DMD and DMDc are modal decomposition methods focused on discovering

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

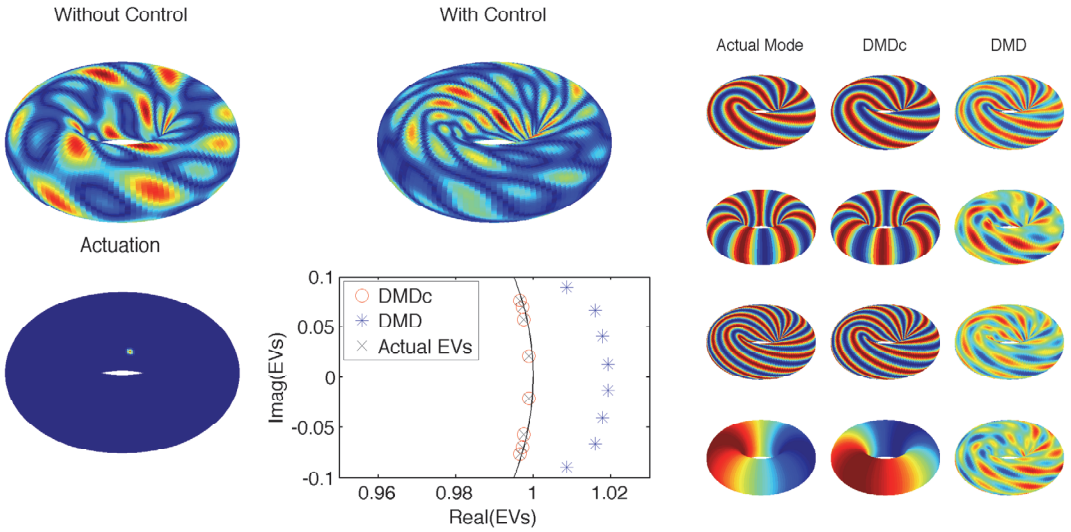

Figure 3. *The top left panel illustrates one realization of Example 3 without actuation over time. The right panel illustrates the same dynamical system but with actuation. The bottom left panel illustrates the actuation applied in the spatial domain. The bottom middle panel shows a comparison between the actual eigenvalues and the eigenvalues found from DMD and DMDc. On the right the first four dynamic modes of DMD and DMDc are compared to the actual underlying spatial modes.*

coherent spatial-temporal modes, *dynamic modes*, from high-dimensional data [43]. Second, DMD has been connected to Koopman spectral analysis providing theoretical justification for characterizing nonlinear systems [29, 41, 33]. The system identification methods are focused on a slightly different task, namely, producing linear models for control from input-output data [55, 26]. However, there are a number of algorithmic similarities between DMDc and system identification methods if we assume access to full-state measurements, as we will discuss in this section.

A substantial portion of this section is spent exploring the connection to two system identification methods in particular: the Eigensystem Realization Algorithm (ERA) and Observer Kalman Filter Identification (OKID). We discuss these particular methods in more detail in this section due to the historical connection between data-driven methods like ERA [24, 23] and modal decomposition methods such as Balanced Proper Orthogonal Decomposition (BPOD) [40], which produce balanced input-output models [35]. ERA was recently demonstrated to produce exactly the same balanced models as BPOD [31], but without the need for information about the adjoint system. Further, strong connections have been established between ERA and DMD [52]. ERA constructs an input-output model using impulse response data, and OKID is typically used in conjunction when convenient impulse response data is unavailable [25, 37, 36, 23]. In this section, we show the connections between DMDc and ERA/OKID. Importantly, we also include a discussion on the connections of DMDc to subspace identification methods such as N4SID, MOESP, and CVA [54, 55, 26, 38].

ERA and OKID were developed to derive a state-space model for control in aerospace applications involving flexible structures [24, 25]. System identification methods such as

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

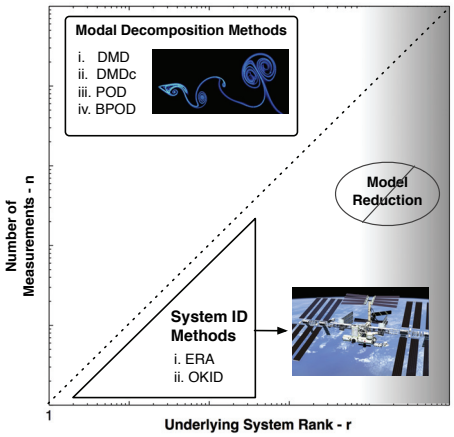

Figure 4. *An illustration depicting the different regimes, with respect to the rank of the system and the number of measurements, of the modal decomposition methods and the system identification methods.*

ERA/OKID were developed for input-output systems which typically have a higher rank/dimensionality than the number of observables $r > n$ [19, 24]. In contrast, modal decomposition methods such as DMD, DMDc, Proper Orthogonal Decomposition (POD), and BPOD are typically applied to complex systems where the number of measurements are significantly larger than the rank of the underlying attractor $n \gg r$, e.g., fluid dynamics problems. Figure 4 illustrates the regime of applications where each of these methods is typically applied. In addition, DMD and POD have been previously established as analysis methods for *nonlinear* complex systems [29, 41, 33].

Previous work by Tu et al. [52] has established a number of connections between DMD and ERA. The similarities and differences between DMDc and ERA listed in this section, though, are more readily compared since both algorithms assume input-output systems. The following list briefly explains how DMDc and ERA differ for the construction of a typical input-output model:

- *The data matrix construction:* The data from the ERA procedure is fundamentally impulse-response data, whereas DMDc can have arbitrary input histories. The input histories are fundamental to the DMDc procedure. Despite this difference, DMDc's and ERA's constructions of the measurement data matrices are similar. Both the matrix $\mathbf{X}$ of DMDc and the Hankel matrix $\mathbf{H}$ of ERA assume snapshots at regular intervals. The data matrix $\mathbf{H}$ is also vertically stacked with time shifted versions of the snapshot. DMDc does not require shift-stacking the matrix since there is little risk of column-rank deficiency due to the typically large number of observables. The two data matrices are equivalent on the condition that $\mathbf{H}$ is not vertically stacked with snapshots [52].
- *The* $\mathbf{A}$ *matrix:* It was previously shown that if the data matrices described in the first bullet are the same, the matrices $\mathbf{A}$ produced by DMD and ERA are equivalent up to a similarity transformation [52]. The $\mathbf{A}$ matrix constructed by DMDc is different

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

from those produced by DMD and ERA since the *input* space and thus the subspace $\tilde{\mathbf{U}}_1^*$ contains added information from the control snapshots.

- *The* $\mathbf{B}$ *matrix:* To compute the matrix $\mathbf{B}$ using ERA, only the first data snapshot after the impulse is utilized, which translates to the discrete dynamical system relationship $\mathbf{x_1} = \mathbf{CBu_0}$, where $\mathbf{u_0}$ is an impulse and $\mathbf{C}$ is the standard linear map for the observable equation. Note that ERA also requires a projection of the single data snapshot onto the left singular vectors of the data matrix $\mathbf{H}$ and the same similarity transform $\mathbf{\Sigma}^{1/2}$ described for the matrix $\mathbf{A}$. The ERA formulation can be contrasted with DMDc through the illustration of the difference in the data matrix construction for DMDc given by the following:

$$\mathbf{\Omega} = \begin{bmatrix} | & | & | & & | \\ \mathbf{0} & \mathbf{x_1} & \mathbf{x_2} & \dots & \mathbf{x_{m_c}} \\ | & | & | & & | \\ \mathbf{u_0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{bmatrix}. \tag{5.1}$$

  The computation to find the matrix $\mathbf{B}$ is quite different for DMDc. Arbitrary control histories can be included in $\mathbf{\Omega}$ to compute $\mathbf{B}$, whereas ERA is primarily impulse response focused. Further, finding the matrix $\mathbf{B}$ with ERA will not be as robust to noise compared with using DMDc and a longer input history.
- *The* $\mathbf{C}$ *matrix:* For DMDc, ERA, and DMD, a linear transformation matrix maps the model state to the measurements. Each of these methods utilizes the left singular vectors of its data matrices for the mapping. There is an important distinction between the role of the left singular vectors for DMDc and ERA. The mapping for DMDc projects a high-dimensional set of observables onto a lower-dimensional subspace. In ERA, the left singular vectors often lift the dimension of the observables; see Figure 4 for an illustration of the rank of the model versus the dimension of the observables.

The observer/Kalman filter identification method allows minimal realization algorithms such as ERA to be generalized from impulse response data to data that is driven by rich input signals [25, 37, 36, 23]. The calculation of the above matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ is typically considered more robust when combining OKID with ERA. An often cited computational challenge confronting ERA is the analysis of lightly damped systems. The magnitude of data (number of snapshots) may be prohibitively large for lightly damped systems, and factoring the Hankel matrix using the SVD is computationally prohibitive. A major similarity between OKID and DMDc is the construction of the data matrix; OKID constructs an augmented data matrix that also stacks the control with the measurements. Similar to DMD and ERA, in the limit of only evaluating the first row of the augmented Hankel matrix, the data matrices between DMDc and OKID are equivalent.

OKID has also been grouped into a larger set of system identification methods called subspace identification methods [38]. There are a number of connections between DMDc and other popular subspace identification methods such as N4SID, MOESP, and CVA [54, 55, 26, 38]. Algorithmically, these methods involve a regression, model reduction, and parameter estimation step similar to DMDc, as described in [38]. In contrast to DMDc, a pre-estimation step is often utilized that defines an estimated variable that helps define an input-output model. The numerical procedure of N4SID, MOESP, and CVA is similar to DMDc in the

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

construction of the regression problem between the input and output data, instead of forming the Hankel matrix as in ERA/OKID [56]. There are, though, interesting differences similar to those found for ERA/OKID, namely in the choice of the projection scaling (see the earlier discussion concerning the similarity transformation required to compare ERA and DMD) and in using the orthogonal complement of the control data matrix Υ for computing the solution [38]. Recent work has combined elements of system identification methods with POD to extract balanced input-output models from data snapshots for describing the flow of a nonlinear system [18]. We believe adapting DMDc using current subspace identification methods as a generalization of the method could be quite impactful.

6. Discussion. Complex, high-dimensional data has become ubiquitous in traditional scientific and engineering applications as well as in modern data-rich fields such as internet traffic, distribution systems, and transportation networks. Machine-learning and statistical methods have been successfully applied to characterize many of these so-called *big-data* problems. Similarly, scientific and engineering fields, exemplified by control theoretic community, have focused on the development of quantitative and automatic dimensionality reduction methods to both characterize and *control* complex systems. In order to construct effective controllers, the underlying system needs to be well understood. Accurately describing the underlying system is a challenge when the system is complex, high-dimensional, and without well-characterized governing equations.

Dynamic mode decomposition (DMD) is a data-driven, equation-free method that helps meet a number of these modern-day challenges. The method has strong connection to nonlinear operator theory and discovers spatial-temporal coherent modes from data. DMD, though, does not produce accurate reduced-order models from complex systems with exogenous forcing. Dynamic mode decomposition with control (DMDc) inherits the advantages of DMD but also provides accurate input-output models for complex systems with actuation. The method can be applied to data from a variety of sources including historical, experimental, and black-box simulations. In addition, DMDc can be naturally extended to consider input-output systems with internally evolving states and a measurement equation. This perspective is more natural in the control theoretic community, in contrast to the DMD literature, which has tended to focus on constructing a relationship between measurements and model reduction.

Methods such as DMDc will play an increasing role in the analysis of large-scale datasets from complex systems. DMD already has a significant number of applications in the fluid dynamics community [44, 47, 16, 3, 53, 51] and is expanding to a variety of other applications like background subtraction in video processing [17]. We believe DMDc is poised to similarly excel as a tool for a diverse set of engineering and applied science applications where control of the complex system is important. Further, the DMDc method is well suited to couple with innovative sparsity-promoting sampling and control strategies [8, 42, 4, 12]. This connection has already been demonstrated for DMD both in time and space [6, 22, 53]. DMDc is therefore positioned to have a dramatic effect on the analysis and control of large-scale complex systems.

Acknowledgments. The authors would like to thank Bill and Melinda Gates for their active support of the Institute for Disease Modeling and their sponsorship through the Global Good Fund. Productive discussions about dynamic mode decomposition with Bing Brunton,

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

Clancy Rowley, and Jonathan Tu are likewise greatly appreciated.

REFERENCES

[1] R. Anderson, R. May, and B. Anderson, *Infectious Diseases of Humans: Dynamics and Control*, Oxford University Press, Oxford, UK, 1992.

[2] K. J. Aström and P. R. Kumar, *Control: A perspective*, Automatica J. IFAC, 50 (2014), pp. 3–43.

[3] S. Bagheri, *Koopman-mode decomposition of the cylinder wake*, J. Fluid Mech., 726 (2013), pp. 596–623.

[4] Z. Bai, T. Wimalajeewa, Z. Berger, G. Wang, M. Glauser, and P. K. Varshney, *Physics based compressive sensing approach applied to airfoil data collection and analysis*, AIAA Paper 2013-0772, 51st Aerospace Sciences Meeting, 2013.

[5] R. G. Baraniuk, *Compressive sensing*, IEEE Signal Process. Mag., 24 (2007), pp. 118–120.

[6] S. L. Brunton, J. L. Proctor, and J. N. Kutz, *Compressive Sampling and Dynamic Mode Decomposition*, preprint, arXiv:1312.5186[math.DS], 2013.

[7] M. Budišić, R. M. Mohr, and I. Mezić, *Applied Koopmanism*, Chaos, 22 (2012), 047510.

[8] E. J. Candès, *Compressive sensing*, in Proceedings of the International Congress of Mathematics, 2006.

[9] K. K. Chen, J. H. Tu, and C. W. Rowley, *Variants of dynamic mode decomposition: Boundary condition, Koopman, and Fourier analyses*, J. Nonlinear Sci., 22 (2012), pp. 887–915.

[10] D. L. Donoho and M. Gavish, *Minimax risk of matrix denoising by singular value thresholding*, Ann. Statist., 42 (2014), pp. 2413–2440.

[11] C. Eckart and G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika, 1 (1936), pp. 211–218.

[12] M. Fardad, F. Lin, and M. R. Jovanović, *Design of optimal sparse interconnection graphs for synchronization of oscillator networks*, IEEE Trans. Automat. Control, 29 (2014), pp. 2281–2291.

[13] M. Forgione, X. Bombois, and P. M. J. Van den Hof, *Data-driven model improvement for model-based control*, Automatica J. IFAC, 52 (2015), pp. 118–124.

[14] M. Gavish and D. L. Donoho, *The Optimal Hard Threshold for Singular Values Is* $4/\sqrt{3}$, preprint, arxiv:1305.5870[stat.ME], 2013.

[15] G. H. Golub and C. Reinsch, *Singular value decomposition and least squares solutions*, Numer. Math., 14 (1970), pp. 403–420.

[16] M. Grilli, P. J. Schmid, S. Hickel, and N. A. Adams, *Analysis of unsteady behaviour in shockwave turbulent boundary layer interaction*, J. Fluid Mech., 700 (2012), pp. 16–28.

[17] J. Grosek and J. N. Kutz, *Dynamic Mode Decomposition for Real-Time Background/Foreground Separation in Video*, preprint, arxiv:1404.7529[cs.CV], 2014.

[18] J. Guzmán I, D. Sipp, and P. J. Schmid, *A dynamic observer to capture and control perturbation energy in noise amplifiers*, J. Fluid Mech., 758 (2014), pp. 728–753.

[19] B. L. Ho and R. E. Kalman, *Effective construction of linear state-variable models from input/output data*, in Proceedings of the 3rd Annual Allerton Conference on Circuit and System Theory, University of Illinois, Urbana, IL, 1965, pp. 449–459.

[20] P. J. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, 2nd ed., Cambridge Monogr. Mech., Cambridge University Press, Cambridge, UK, 2012.

[21] M. Ilak and C. W. Rowley, *Modeling of transitional channel flow using balanced proper orthogonal decomposition*, Phys. Fluids, 20 (2008), 034103.

[22] M. R. Jovanović, P. J. Schmid, and J. W. Nichols, *Sparsity-promoting dynamic mode decomposition*, Phys. Fluids, 26 (2014), 024103.

[23] J. N. Juang, *Applied System Identification*, Prentice Hall PTR, Upper Saddle River, NJ, 1994.

[24] J. N. Juang and R. S. Pappa, *An eigensystem realization algorithm for modal parameter identification and model reduction*, J. Guidance Control Dynam., 8 (1985), pp. 620–627.

[25] J. N. Juang, M. Phan, L. G. Horta, and R. W. Longman, *Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments*, Technical Memorandum 104069, NASA, 1991.

[26] T. Katayama, *Subspace Methods for System Identification*, Springer-Verlag, London, 2005.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

[27] M. J. Keeling and B. T. Grenfell, *Disease extinction and community size: Modeling the persistence of measles*, Science, 275 (1997), pp. 65–67.
[28] M. Keeling and P. Rohani, *Modeling Infectious Diseases in Humans and Animals*, Princeton University Press, Princeton, NJ, 2008.
[29] B. O. Koopman, *Hamiltonian systems and transformation in Hilbert space*, Proc. Natl. Acad. Sci. USA, 17 (1931), pp. 315–318.
[30] F. Lin, M. Fardad, and M. R. Jovanović, *Design of optimal sparse feedback gains via the alternating direction method of multipliers*, IEEE Trans. Automat. Control, 58 (2013), pp. 2426–2431.
[31] Z. Ma, S. Ahuja, and C. W. Rowley, *Reduced order models for control of fluids using the eigensystem realization algorithm*, Theoret. Comput. Fluid Dynam., 25 (2011), pp. 233–247.
[32] I. Mezić, *Spectral properties of dynamical systems, model reduction and decompositions*, J. Nonlinear Dyn., 41 (2005), pp. 309–325.
[33] I. Mezić, *Analysis of fluid flows via spectral properties of the Koopman operator*, in Annual Review of Fluid Mechanics, Annu. Rev. Fluid Mech. 45, Annual Reviews, Palo Alto, CA, 2013, pp. 357–378.
[34] L. Mirsky, *Symmetric gauge functions and unitarily invariant norms*, Quart. J. Math. Oxford Ser. (2), 11 (1960), pp. 50–59.
[35] B. C. Moore, *Principal component analysis in linear systems: Controllability, observability, and model reduction*, IEEE Trans. Automat. Control, 26 (1981), pp. 17–32.
[36] M. Phan, L. G. Horta, J. N. Juang, and R. W. Longman, *Linear system identification via an asymptotically stable observer*, J. Optim. Theory Appl., 79 (1993), pp. 59–86.
[37] M. Phan, J. N. Juang, and R. W. Longman, *Identification of linear-multivariable systems by identification of observers with assigned real eigenvalues*, J. Astronautical Sci., 40 (1992), pp. 261–279.
[38] S. J. Qin, *An overview of subspace identification*, Computers and Chemical Engineering, 30 (2006), pp. 1502–1513.
[39] P. Rapisarda and H. L. Trentelman, *Identification and data-driven model reduction of state-space representations of lossless and dissipative systems from noise-free data*, Automatica J. IFAC, 47 (2011), pp. 1721–1728.
[40] C. W. Rowley, *Model reduction for fluids using balanced proper orthogonal decomposition*, Internat. J. Bifur. Chaos Appl. Sci. Engrg., 15 (2005), pp. 997–1013.
[41] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, *Spectral analysis of nonlinear flows*, J. Fluid Mech., 641 (2009), pp. 115–127.
[42] H. Schaeffer, R. Caflisch, C. D. Hauck, and S. Osher, *Sparse dynamics for partial differential equations*, Proc. Natl. Acad. Sci. USA, 110 (2013), pp. 6634–6639.
[43] P. J. Schmid, *Dynamic mode decomposition of numerical and experimental data*, J. Fluid Mech., 656 (2010), pp. 5–28.
[44] P. J. Schmid, *Application of the dynamic mode decomposition to experimental data*, Experiments in Fluids, 50 (2011), pp. 1123–1130.
[45] P. J. Schmid, K. E. Meyer, and O. Pust, *Dynamic mode decomposition and proper orthogonal decomposition of flow in a lid-driven cylindrical cavity*, in Proceedings of the 8th International Symposium on Particle Image Velocimetry, 2009, PIV09-0186.
[46] P. J. Schmid and J. L. Sesterhenn, *Dynamic mode decomposition of numerical and experimental data*, in Bull. Amer. Phys. Soc. 61st Annual Meeting of the APS Division of Fluid Dynamics (San Antonio, TX), Vol. 53, Amer. Phys. Soc., College Park, MD, 2008, p. 208.
[47] P. J. Schmid, D. Violato, and F. Scarano, *Decomposition of time-resolved tomographic PIV*, Experiments in Fluids, 52 (2012), pp. 1567–1579.
[48] L. Sirovich, *Turbulence and the dynamics of coherent structures.* I. *Coherent structures*, Quart. Appl. Math., 45 (1987), pp. 561–571.
[49] L. Sirovich, *Turbulence and the dynamics of coherent structures.* II. *Symmetries and transformations*, Quart. Appl. Math., 45 (1987), pp. 573–582.
[50] L. Sirovich, *Turbulence and the dynamics of coherent structures.* III. *Dynamics and scaling*, Quart. Appl. Math., 45 (1987), pp. 583–590.
[51] G. Tissot, L. Cordier, N. Benard, and B. R. Noack, *Model reduction using dynamic mode decomposition*, Comptes Rendus Mécanique, 342 (2014), pp. 410–416.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license

[52] J. H. Tu, D. M. Luchtenburg, C. W. Rowley, S. L. Brunton, and J. N. Kutz, *On dynamic mode decomposition: Theory and applications*, J. Comput. Dynam., 1 (2014), pp. 391–421.
[53] J. H. Tu, C. W. Rowley, J. N. Kutz, and J. K. Shang, *Toward compressed DMD: Spectral analysis of fluid flows using sub-Nyquist-rate PIV data*, Experiments in Fluids, to appear; preprint, arXiv:1401.7047[physics.flu-dyn], 2014.
[54] P. Van Overschee and B. De Moor, *N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems*, Automatica J. IFAC, 30 (1994), pp. 75–93.
[55] P. Van Overschee and B. De Moor, *Subspace Identification for Linear Systems: Theory - Implementation - Applications*, Springer, New York, 1996.
[56] M. Viberg, *Subspace-based methods for the identification of linear time-invariant systems*, Automatica J. IFAC, 31 (1995), pp. 1835–1851.
[57] K. Willcox and J. Peraire, *Balanced model reduction via the proper orthogonal decomposition*, AIAA J., 40 (2002), pp. 2323–2330.
[58] R. C. Winck and W. J. Book, *Dimension reduction in a feedback loop using the SVD: Results on controllability and stability*, Automatica J. IFAC, 49 (2013), pp. 3084–3089.

Downloaded 11/19/26 to 46.12.210.136 . Redistribution subject to CCBY license

© 2016 SIAM. Published by SIAM under the terms of the Creative Commons 4.0 license