ANNUAL REVIEWS

Annual Review of Fluid Mechanics

Dynamic Mode Decomposition and Its Variants

Peter J. Schmid

Department of Mathematics, Imperial College London, London, United Kingdom; email: pjschmid@imperial.ac.uk

Annu. Rev. Fluid Mech. 2022. 54:225–54

First published as a Review in Advance on October 5, 2021

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

https://doi.org/10.1146/annurev-fluid-030121-015835

Keywords

data decomposition, model reduction, quantitative flow analysis, Koopman analysis, dynamical systems, spectral analysis

Abstract

Dynamic mode decomposition (DMD) is a factorization and dimensionality reduction technique for data sequences. In its most common form, it processes high-dimensional sequential measurements, extracts coherent structures, isolates dynamic behavior, and reduces complex evolution processes to their dominant features and essential components. The decomposition is intimately related to Koopman analysis and, since its introduction, has spawned various extensions, generalizations, and improvements. It has been applied to numerical and experimental data sequences taken from simple to complex fluid systems and has also had an impact beyond fluid dynamics in, for example, video surveillance, epidemiology, neurobiology, and financial engineering. This review focuses on the practical aspects of DMD and its variants, as well as on its usage and characteristics as a quantitative tool for the analysis of complex fluid processes.

1. INTRODUCTION

The analysis of fluid systems often rests on the commonly accepted notion that the transport of conserved quantities or the evolution of observables is facilitated by only a small number of coherent structures or a few dynamic processes. These structures have their origin in instabilities or sensitivities that bring forth spatial patterns and specific scales that synchronize in time into persistent and observable motion. This compressed character of fluid flows motivates and gives rise to a concerted effort to extract these essential mechanisms from measurements, simulations, or data sequences in an attempt to reduce a system of many degrees of freedom to an analog of tractable size that still captures the essence of the overall process.

Statistical tools, such as variance analysis, conditional averaging (e.g., Alfredsson & Johansson 1984), and quadrant analysis (e.g., Wallace 2016), have dominated the field of fluid dynamics, particularly in experimental studies. Pattern extraction techniques based on linear algebra concepts, such as the Karhunen–Loève expansion, principal component analysis or proper orthogonal decomposition (POD) (Lumley 1971, Sirovich 1987, Berkooz et al. 1993, Holmes et al. 2012), and linear stochastic estimation (Adrian & Moin 1988), are part of the standard arsenal of tools for experimental and computational practitioners. Additionally, local linearization of dynamical systems (Perry & Chong 1987), Lagrangian coherent structures (Haller 2015), and, more recently, graph-theoretical methods (Gopalakrishnan Meena et al. 2018) have emerged in an effort to describe complex fluid behavior.

In this same endeavor, the past decade has seen a renaissance of an alternative viewpoint, originally introduced in the 1930s: the Koopman formalism of nonlinear dynamical systems (Koopman 1931). This approach provides a complementary formulation of evolution processes that adapts more readily to situations where data sequences of observables are available, but governing equations are difficult (or impractical) to formulate. This distinctly data-driven perspective has now established itself as an active research field to complement and support a more traditional model-based viewpoint. The advantages of analyzing fluid systems via the spectral properties of the Koopman operator have been expounded in previous articles (Mezić 2005, 2013); in this review, we use the same starting point, but concentrate on approximation algorithms to extract this spectral information from observed data sequences and emphasize various extensions and generalizations for a wider range of applications.

At the center of this extraction is a factorization and dimensionality reduction technique for sequential data streams, known as dynamic mode decomposition (DMD) (Schmid & Sesterhenn 2008, Rowley et al. 2009, Schmid 2010, Kutz et al. 2016a). It falls into the category of modal decompositions, where a high-dimensional spatiotemporal signal is broken apart into a triplet of purely spatial modes, scalar amplitudes, and purely temporal signals in a process that is akin to a separation of variables. The details of this decomposition allow us to describe the flow process by a hierarchy of simpler processes that, when linearly superimposed, recover the full data sequence. This decomposition facilitates the objective dissection of the flow into dynamic elements (spatial-mode temporal-signal pairs) that can then be studied in isolation. Superposition—and, hence, the linearity of this procedure—is critical. It is accomplished not, as commonly done, by a linearization about an equilibrium or quasi-equilibrium point, but rather by an embedding of the state variable into an infinite space of observables. DMD and its variants, described in this article, rely on the fact that a large but finite representation of this infinite-dimensional embedding is sufficient for an approximate description of the underlying dominant flow behavior.

This review provides a brief theoretical background on the Koopman concept, as it motivates the design of a data-driven algorithm for the extraction of coherent patterns from flow field observations. Further details on this have been given by Mezić (2005, 2013). The review then presents the standard DMD, the core algorithm, as a classical matrix decomposition/factorization method,

combined with a sparsity-promoting optimization to extract modal amplitudes. The next section gives an overview of the many improvements and extensions that have been suggested over the past years, addressing accuracy, robustness, and efficiency issues and widening the application field beyond flow analysis and evolution processes. Three representative examples, showcasing the flexibility of DMD as an analysis tool for simple to complex flow configurations, are given at the end of this review.

2. THEORETICAL BACKGROUND AND THE CORE ALGORITHM

2.1. The Koopman Approach to Dynamical Systems

The temporal evolution of a variable $\mathbf{x}$, which fully describes the state of a fluid system, is given by the nonlinear dynamical system

$$\mathbf{x}_{i+1} = \mathbf{F}(\mathbf{x}_i), \qquad 1.$$

where we have chosen the time-discrete formulation and where $\mathbf{F}$ represents the nonlinear governing equations, mapping the state $\mathbf{x}_i$ over a given time interval Δt onto $\mathbf{x}_{i+1}$.

The mapping $\mathbf{F}$ commonly stems from a discretization of partial differential equations that describe the key features of the underlying flow via conservation laws and physical models. Dynamical systems of this form have traditionally been analyzed by investigating their equilibrium points and the linear dynamics of small perturbations about them. This approach is central to hydrodynamic stability theory, where the repelling (unstable) or attracting (stable) behavior of equilibrium states (base flows) is investigated and tracked as the governing parameters are varied.

Rather than taking this approach, with its local linearization about equilibrium points, Koopman analysis accomplishes linearization by a different technique: It adopts a data-driven viewpoint by considering and tracing observables of the state-vector $\mathbf{x}$ as they evolve in time. We represent these observables of $\mathbf{x}$ as $\phi(\mathbf{x})$, a yet unspecified set of functions of the state-vector. The dynamics of the full nonlinear system is then projected onto the dynamics of these observables [see the review by Budišić et al. (2012)]. When introducing a mapping $\mathcal{K}$ that describes the evolution of observables over one discrete time unit Δt, we arrive at

$$\phi(\mathbf{x}_{i+1}) = \mathcal{K}\,\phi(\mathbf{x}_i) = \phi[\mathbf{F}(\mathbf{x}_i)]. \qquad 2.$$

From this expression it is easily shown that $\mathcal{K}$ is a linear operator, since we have $\mathcal{K}(c_1\phi_{(1)} + c_2\phi_{(2)}) = c_1\mathcal{K}\phi_{(1)} + c_2\mathcal{K}\phi_{(2)}$ for any two observables $\phi_{(1)}$ and $\phi_{(2)}$ and constants c_1 and c_2. The operator $\mathcal{K}$ is termed the Koopman operator. The above procedure can be thought of as a lifting or embedding of the state-vector dynamics onto an equivalent dynamics in the space of observables (see **Figure 1**).

We have transformed the nonlinear dynamics in the state variables into an equivalent linear dynamics in the observables. However, it is important to keep in mind that we also have transformed a

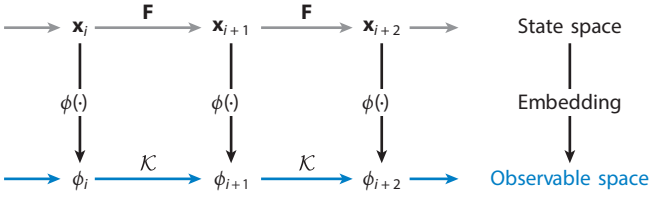

Figure 1

Projection of the state space dynamics given by a nonlinear mapping $\mathbf{F}$ onto the dynamics in an infinite-dimensional observable space given by the linear Koopman operator $\mathcal{K}$. We use the notation $\phi_i = \phi(\mathbf{x}_i)$, where $\mathbf{x}_i$ are state variables for the i-th snapshot and ϕ_i are the observables for the i-th snapshot.

finite-dimensional nonlinear system into an infinite-dimensional linear system. The linearization due to this embedding process is the key idea behind Koopman analysis.

The remaining question concerns the choice of observables that will render our nonlinear system linear. For dynamical systems $\mathbf{F}$ with polynomial nonlinearities in $\mathbf{x}$, it may appear sensible to also include polynomial terms in the suite of observables and define $\phi(z)$ as a composite entity including z and higher powers of z. This solution strategy is termed Carleman linearization (Carleman 1932). Despite some success in treating polynomial dynamical systems and control problems, Carleman linearization can often lead to divergences: the evolution of polynomial terms in the observable $\phi(z)$ produces still higher powers in z and requires their inclusion into our suite of observables, which in turn gives rise to even higher powers. This process clearly fails to arrive at a closed system.

To avoid outrunning the span of the chosen observables, we might consider, recalling the linear nature of $\mathcal{K}$, the invariant subspace of $\mathcal{K}$ for our choice of observables. By definition, the invariant subspace maps onto itself under the action of $\mathcal{K}$ and, thus, provides closure. We define

$$\mathcal{K}\Phi = \Phi\Lambda, \qquad 3.$$

with Φ the set of eigenfunctions of $\mathcal{K}$ and Λ containing the eigenvalues of $\mathcal{K}$ on its diagonal. We proceed by expressing our observables in this eigenfunction basis as $\phi = \Phi\,\xi$, and starting from an initial state-vector $\mathbf{x}_0$, we generate a temporal sequence of observables according to

$$\phi(\mathbf{x}_{0+k}) = \mathcal{K}^k\phi(\mathbf{x}_0) = \mathcal{K}^k\Phi\,\xi = \Phi\,\Lambda^k\xi. \qquad 4.$$

By their very nature, the eigenfunctions of the Koopman operator $\mathcal{K}$ form an invariant subspace, and repeated application of $\mathcal{K}$ will contain the dynamics in this subspace. Hence, we avoid the kind of divergences or closure problems mentioned above. This last expression then motivates a computational procedure that approximates the Koopman eigenfunctions—not from the operator itself, but rather from a sequence of observables the operator generates.

2.2. Koopman Eigenfunctions from Data: Dynamic Modes

With the spectral properties of the Koopman operator $\mathcal{K}$ identified as the key ingredients for our linearization process, we aim at devising a procedure to extract this information directly from a data sequence, thus avoiding the formation or approximation of $\mathcal{K}$. While in exceptional cases an analytical treatment of the Koopman eigenfunctions is available (see, e.g., Bagheri 2013), we generally have to resort to numerical approximations to compute the spectral properties of $\mathcal{K}$.

Using Equation 4, in **Figure 2** we construct a matrix whose n columns consist of snapshots of observables generated by a repeated application of the Koopman operator $\mathcal{K}$. The result can be further manipulated by expressing the powers of $\mathcal{K}$ by their spectral equivalents.

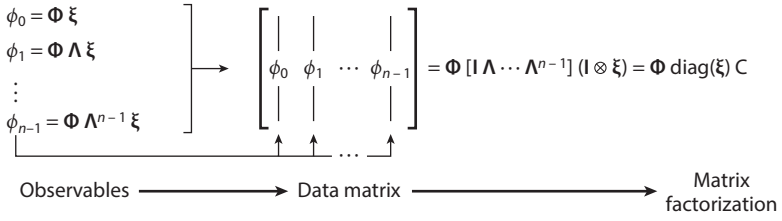

Figure 2

Transformation of a finite sequence of observables into a data matrix that is subsequently decomposed using a matrix factorization. The Kronecker product is denoted by ⊗, and I stands for the identity matrix.

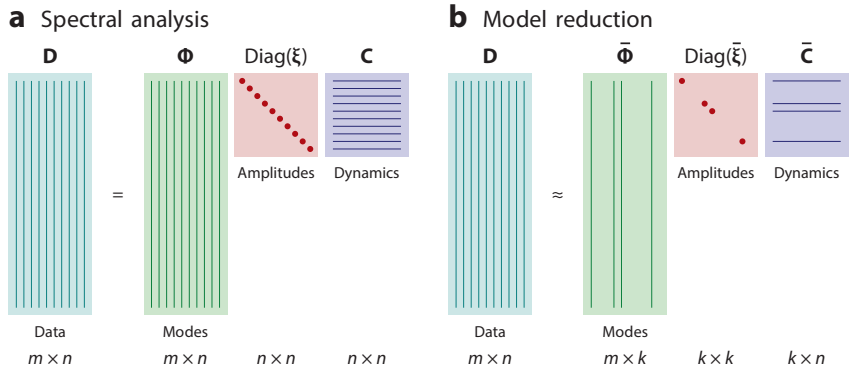

Figure 3

General factorization of a data matrix D into modes Φ, amplitudes diag(ξ), and dynamics C, applying (*a*) spectral analysis and (*b*) model reduction.

The expression in **Figure 2** introduces a sequence of observables and establishes a link between a data matrix and the spectral properties Φ and Λ of the Koopman operator $\mathcal{K}$. In other words, it suggests a data-driven method to compute the key components of Koopman analysis for dynamical systems. This formulation is reminiscent of a Krylov technique, where we express the dynamics of a linear system based on the repeated application of an operator to a starting vector; moreover, the final step establishes a classical matrix factorization problem. Denoting the snapshot matrix of observables by D, we aim at decomposing this data matrix into a triple product $\mathsf{D} = \Phi\,\mathrm{diag}(\xi)\,\mathsf{C}$ containing the spatial modes in Φ, the amplitudes in diag(ξ), and the temporal dynamics in C (see **Figure 3*a***). This type of decomposition is commonplace in many application areas, including fluid dynamics, when we seek to extract coherent patterns from a data sequence. POD and independent component analysis (ICA) (Hyvärinen 2013) are but two examples of this way of factoring a data stream into essential structures, their dynamics, and their representation within the original data sequence. The specifics of the decomposition are given by the constraints placed on the modal structures Φ (e.g., orthogonality in POD) or the temporal signals C (e.g., statistical independence in ICA) and vary by application field or processed data. The diagonality of the amplitudes diag(ξ) ensures a decoupling of the identified mode–dynamics pairs and allows for the study of each pair as a building block of a more complex process. Furthermore, the distribution of the amplitudes may indicate the prevalence of a few dominant processes and allow for the approximate representation of the full data sequence by a compressed hierarchy of essential subprocesses—a methodology termed model or dimensionality reduction (see **Figure 3*b***). These attributes of general matrix decompositions also carry over to our case.

In our case, the structure of the matrix C, containing the discrete temporal dynamics, uniquely identifies the partitioning of the data sequence into a triple factorization with a diagonal amplitude matrix. From the expression above, the matrix C takes on the following shape (see Rowley et al. 2009, Schmid 2010, Kutz et al. 2016a):

$$\mathsf{C} = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \cdots & \lambda_1^{n-1} \\ 1 & \lambda_2 & \lambda_2^2 & \cdots & \lambda_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \lambda_n & \lambda_n^2 & \cdots & \lambda_n^{n-1} \end{bmatrix}, \qquad 5.$$

known as a Vandermonde matrix. The entries λ_i can be identified as the Koopman eigenvalues represented by Λ in Equation 4. Following standard convention, the complex eigenvalues λ_i can be recast into an exponential form as $\lambda_i = \exp(\mu_i \Delta t)$ with Δt representing the (constant) time interval between the snapshots in our data matrix. The Vandermonde structure indicates that we are seeking a decomposition where each row of C constitutes a single (complex) frequency dynamics. In summary, then, to approximate the spectral properties Φ and Λ of the Koopman operator $\mathcal{K}$, we propose a factorization of our data matrix of observables D into a modal component Φ, a diagonal amplitude matrix diag(ξ), and a temporal Vandermonde matrix C.

While no direct linear algebra technique for this kind of factorization exists, we can exploit a relationship between Vandermonde and companion matrices in our design of a computational technique. Specifically, we recall that Vandermonde matrices can be used to diagonalize companion matrices (Rowley et al. 2009). Companion matrices are defined as square matrices comprising a unit subdiagonal and a nonzero last column:

$$\mathsf{S} = \begin{bmatrix} & & & a_1 \\ 1 & & & a_2 \\ & 1 & & a_3 \\ & & \ddots & \vdots \\ & & 1 & a_n \end{bmatrix}. \qquad 6.$$

We then state $\mathsf{S} = \mathsf{C}^{-1}\,\mathrm{diag}(\xi)\,\mathsf{C}$. Using this relation in our matrix factorization yields

$$\mathsf{D} = \Phi\,\mathrm{diag}(\xi)\,\mathsf{C} = \Phi\,\mathsf{C}\,\mathsf{S} = \mathsf{D}'\,\mathsf{S}, \qquad 7.$$

where we have introduced $\mathsf{D}' = \Phi\,\mathsf{C}$. The companion structure of S in this last expression allows for the following observation: The unit subdiagonal causes a simple shift in the order of the snapshots, whereas the coefficients in the last column represent the final snapshot in our data matrix D as a linear superposition of the snapshots in D'. A specific case based on seven snapshots is shown in **Figure 4** for demonstration.

The recasting of our matrix factorization from a Vandermonde-based to a companion-based formalism now suggests a data-based procedure for approximating the spectral properties of the Koopman operator. This algorithm will yield the DMD of the data matrix D. We commence with a data sequence of n snapshots of observables ϕ, sampled (for simplicity) at equispaced time intervals Δt. We break this sequence into two subsequences according to

$$\mathsf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_n] = \begin{cases} [\mathbf{d}_1, \ldots, \mathbf{d}_{n-1}] \equiv \mathsf{D}_1^{n-1}, \\ [\mathbf{d}_2, \ldots, \mathbf{d}_n] \quad \equiv \mathsf{D}_2^{n}. \end{cases} \qquad 8.$$

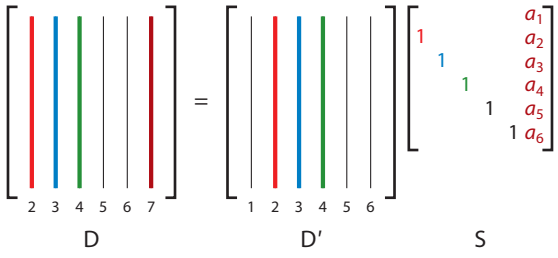

Figure 4

Mapping between two shifted data sequences using a companion matrix with, as an example, seven snapshots. The subdiagonal of the companion matrix represents the shifted snapshots; the last column of the companion matrix expresses the final snapshot as a linear combination of all previous snapshots.

Recalling Equation 7, we represent the last column of D_2^n (i.e., the snapshot $\mathbf{d}_n$) as a linear superposition of the snapshots in D_1^{n-1}. This yields a least-squares problem for the companion matrix S with the solution given by a reduced QR decomposition of D_1^{n-1}:

$$\mathsf{S} = \mathsf{R}^{-1}\mathsf{Q}^{\mathrm{H}}\mathsf{D}_2^n \qquad \text{with} \qquad \mathsf{Q}\,\mathsf{R} = \mathrm{qr}(\mathsf{D}_1^{n-1}), \tag{9.}$$

and $^{\mathrm{H}}$ the Hermitian (conjugate transpose) operation.

Returning to the Koopman problem, we can now restate the application of the Koopman operator on our data matrix as

$$\mathcal{K}\,\mathsf{D}_1^{n-1} = \mathsf{D}_2^n \approx \mathsf{D}_1^{n-1}\mathsf{S}, \tag{10.}$$

which expresses the action of the Koopman operator $\mathcal{K}$ on the data set D_1^{n-1}, advancing each snapshot over one time step Δt in the basis of the snapshots in D_1^{n-1} by the companion matrix S. We acknowledge at this point that in this review we use, in a slight abuse of notation, the same symbol $\mathcal{K}$ to denote the theoretical (infinite-dimensional) Koopman operator and its finite-dimensional representation.

This last expression is reminiscent of a partial decomposition, as employed in the Arnoldi iteration for computing eigenvalues of large-scale matrices via a Krylov subspace technique. Following the same idea, we gain approximate spectral information about the Koopman operator $\mathcal{K}$ from its lower-dimensional representation S. More specifically, with $[\mathsf{Y}, \Lambda] = \mathrm{eig}(\mathsf{S})$ or $\mathsf{S} = \mathsf{Y}\,\Lambda\,\mathsf{Y}^{-1}$ as the eigenvalue decomposition of S, we write

$$\mathcal{K}\,\underbrace{\mathsf{D}_1^{n-1}\mathsf{Y}}_{\Phi} = \underbrace{\mathsf{D}_1^{n-1}\mathsf{Y}}_{\Phi}\,\Lambda, \tag{11.}$$

which identifies the approximate eigenvalues and eigenfunctions of the Koopman operator $\mathcal{K}$ as Λ and $\Phi = \mathsf{D}_1^{n-1}\,\mathsf{Y}$, respectively.

The solution to the least-squares problem in Equation 9 implicitly assumes that the data matrix D_1^{n-1} has full rank, producing a nonsingular matrix R. For long data sequences or periodic flow phenomena, such as vortex shedding in the wake of a bluff body, the data sequence often contains redundancies and is hence rank deficient. In this case, we need to use a singular-value decomposition (SVD) in place of the QR decomposition to compute the low-dimensional representation of the Koopman operator. We factor the data sequence D_1^{n-1} by an SVD $\mathsf{D}_1^{n-1} = \mathsf{U}\Sigma\mathsf{V}^{\mathrm{H}}$ and then use

$$\mathcal{K}\,\mathsf{D}_1^{n-1} = \mathsf{D}_2^n = \mathcal{K}\,\mathsf{U}\Sigma\mathsf{V}^{\mathrm{H}} \tag{12.}$$

to form a modified expression $\tilde{\mathsf{S}}$ for the low-dimensional representation of the Koopman operator as the Rayleigh quotient matrix $\tilde{\mathsf{S}} = \mathsf{U}^{\mathrm{H}}\mathcal{K}\mathsf{U}$. We obtain an expression for $\tilde{\mathsf{S}}$ according to

$$\tilde{\mathsf{S}} = \mathsf{U}^{\mathrm{H}}\mathsf{D}_2^n\mathsf{V}\Sigma_\epsilon^{-1}, \tag{13.}$$

with

$$\Sigma_\epsilon^{-1} = \mathrm{diag}\left(\begin{cases} 1/\sigma_j & \text{for } \sigma_j \geq \epsilon \\ 0 & \text{for } \sigma_j < \epsilon \end{cases}\right) \qquad \text{and} \qquad \Sigma = \mathrm{diag}\{\sigma_1, \sigma_2, \ldots, \sigma_{n-1}\}. \tag{14.}$$

The last expression introduces a user-defined threshold parameter ϵ that has to be chosen to avoid rank-deficiency or near rank-deficiency in our data matrix D_1^{n-1}. In the above, we have expanded the modal structures in the orthogonal basis U. Together with the truncation of small singular values, this change of basis from the data snapshots D_1^{n-1} to their orthogonal representation

U adds robustness and numerical stability to the overall DMD algorithm. The modified procedure then requires the solution of the eigenvalue problem for $\tilde{\mathsf{S}}$ according to $[\mathsf{X}, \Lambda] = \mathrm{eig}(\tilde{\mathsf{S}})$, which in turn yields

$$\mathcal{K}\, \underbrace{\mathsf{U}\mathsf{X}}_{\Phi} = \underbrace{\mathsf{U}\mathsf{X}}_{\Phi}\, \Lambda \tag{15.}$$

and identifies the approximate eigenfunctions of $\mathcal{K}$ as $\Phi = \mathsf{U}\mathsf{X}$. Notably, the basis U consists of the POD modes of the data sequence D_1^{n-1}.

In summary, we have a simple algorithm for computing approximate Koopman eigenfunctions (dynamic modes) and Koopman eigenvalues directly from a sequence of snapshots of observables. Equations 8, 13, and 14, together with the eigenvalue problem $[\mathsf{X}, \Lambda] = \mathrm{eig}(\tilde{\mathsf{S}})$ and the representation $\Phi = \mathsf{U}\mathsf{X}$, constitute the key steps of the standard DMD algorithm.

While we have yet to address the choice of ϕ in our data matrix D, we first complete the matrix factorization $\mathsf{D} = \Phi\, \mathrm{diag}(\xi)\, \mathsf{C}$ by identifying the amplitudes ξ.

2.3. Amplitudes by Sparsity Promotion

With the modal structures Φ and their temporal dynamics C identified, we finally have to recover the amplitudes ξ from the data sequence. These amplitudes give weight to the identified mode–dynamics pairs (Φ_j, λ_j) and quantify their dominance in the observed data stream. Large amplitudes pinpoint essential flow processes, while structures with smaller amplitudes can be safely discarded when analyzing the data sequence.

The amplitudes ξ as a means to separate relevant from negligible fluid processes, however, can cause problems when the data matrix contains statistical outliers or is otherwise compromised by high-amplitude noise—a situation frequently encountered when processing experimental data. In particular, statistical outliers are reflected in the decomposition as structures with large amplitudes and large decay rates (modeling their time-local nature). For this reason, a naïve amplitude criterion for identifying physically pertinent flow structures is insufficient. Rather, an optimization approach is chosen that attempts a faithful reconstruction of the true data using a weighted superposition of the computed mode–dynamics pairs, but it does so with as few pairs as necessary (Jovanović et al. 2014). A balance has to be found between these two criteria: the reconstruction error and the cardinality (i.e., the number of nonzero entries) of the amplitude vector. Mathematically, the former criterion is expressed as the Frobenius norm $\|\cdot\|_F$ of the difference between the original and reconstructed data sequence, while the latter criterion is quantified by the L_1-norm of the amplitude vector ξ. We note that the L_1-norm is a regularized substitute for the true cardinality of ξ. We then obtain the following mixed-norm optimization problem:

$$\xi_{\mathrm{opt}} = \mathrm{argmin}_{\xi}\, \|\mathsf{D} - \Phi\, \mathrm{diag}(\xi)\, \mathsf{C}\|_F \;+\; \gamma \|\xi\|_1, \tag{16.}$$

with ξ_{opt} as the optimal amplitude vector and γ as a user-specified parameter directing emphasis toward the data recovery error or the sparsity of ξ. In the absence of sparsity constraints ($\gamma = 0$), we can recast the above expression into a quadratic programming problem

$$\xi_{\mathrm{opt}} = \mathrm{argmin}_{\xi}\, \left\{\xi^{\mathrm{H}} \mathsf{P}\, \xi - \mathbf{p}^{\mathrm{H}} \xi + s\right\}, \tag{17.}$$

with

$$\mathsf{P} = (\mathsf{X}^{\mathrm{H}}\mathsf{X}) \odot (\mathsf{C}\,\mathsf{C}^{\mathrm{H}})^{*}, \qquad \mathbf{p}^{*} = \mathrm{diag}(\mathsf{C}\,\mathsf{V}\,\Sigma\,\mathsf{X}), \qquad \text{and} \qquad s = \mathrm{trace}(\Sigma^2), \tag{18.}$$

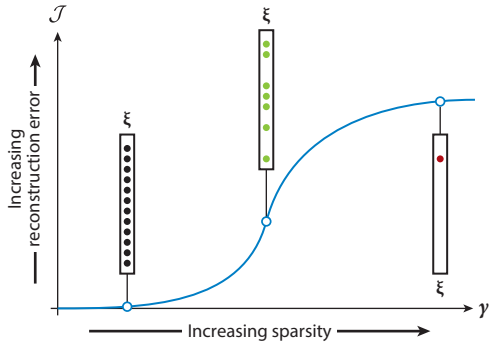

Figure 5

Reconstruction error as a function of the regularization parameter γ, producing amplitude vectors $\boldsymbol{\xi}$ of increasing sparsity.

where * denotes the complex conjugate, and $\odot$ stands for the elementwise (Hadamard) product. The optimal solution is given by $\boldsymbol{\xi}_{\text{opt}} = \mathsf{P}^{-1}\mathbf{p}$ and represents the amplitude distribution that minimizes the data recovery error, using all computed dynamic modes.

Bringing the sparsity term into play ($\gamma > 0$) yields a mixed-norm optimization problem that can be solved using the alternating directions of multiplier method (ADMM) (Boyd & Vandenberghe 2004), which iterates between a quadratic optimization problem (as in Equation 17) for a fixed sparsity and a pure L_1-regularization for a fixed recovery error. By varying the parameter γ, we trace out a reconstruction error, as sketched in **Figure 5**: For low values of γ we obtain a rather dense amplitude vector $\boldsymbol{\xi}$ and a low reconstruction error, while for large γ the amplitude vector is sparsely populated but the reconstruction error is substantial. As is common in regularized optimization problems, a value of γ has to be determined that still yields an acceptable reconstruction error, while eliminating a maximum of unnecessary or unphysical components from the amplitude vector.

Since there is no obvious relationship between the value of the regularization parameter γ and the number of retained components in $\boldsymbol{\xi}$, a computational sweep over γ is necessary to determine an acceptable value. However, since the ADMM problem contains matrices that scale with the number of snapshots in D, the computational burden of this sweep is tolerable (Jovanović et al. 2014). Once a sparsity pattern for $\boldsymbol{\xi}$ has been obtained, a polishing step, which reallocates residual components from the neglected to the retained modes, concludes the algorithm for determining the amplitudes and, thus, the full DMD.

3. EXTENDING THE CORE ALGORITHM

The procedural steps in the previous section are referred to below as the core algorithm. Over the decade since its introduction, many improvements, extensions, and generalizations have been devised, producing more accurate or robust methods and bringing Koopman ideas to many application areas beyond fluid flow analysis. While it is difficult to cover the many applications and extensions, we focus on some key developments that have had a significant impact on our way of analyzing flow data and enhancing our understanding of fluid processes.

We categorize these extensions into efforts to increase the accuracy of the decomposition, its robustness, and its efficient computation. In addition, we allude to expansions of the Koopman idea and DMD to other dynamical systems or data sets. Moreover, we mention complementary

operator-theoretic methods based on transfer operators, which are closely related to Koopman analysis.

3.1. Toward More Accuracy

A concerted effort has been directed toward the remaining question of the previous section: the choice of observables. This choice is crucial in arriving at a proper embedding of the nonlinear dynamical system and in the successful description of the resulting system by a linear operator (see **Figure 1**).

We recall that the Koopman operator $\mathcal{K}$ is infinite in the space of observables. The approximation of the Koopman operator by the DMD can be achieved in the limit as either the number n of snapshots or the number M of observables tends to infinity (see Klus et al. 2016, Korda & Mezić 2018). Realistically, both numbers are finite, and—while often the number of measurement points as observables is sufficiently large for a workable approximation of the infinite Koopman limit—the convergence toward the Koopman operator relies on a judicious choice of observables; incomplete or poor choices may yield spurious results (Tu et al. 2014b, Brunton et al. 2016). Within this context, the term "accuracy" in this section is to be understood in the sense of approaching the true infinite Koopman operator with our algorithm, that is, the accuracy of the approximation of an infinite operator by a finite representation.

3.1.1. The choice of observables and the extended DMD. In many applications in fluid dynamics, observables are chosen as the measurement points in experiments or as (all or some) state-vector components on the grid points of numerical simulations. In this case, we have $\phi(\mathbf{x}) = \mathbf{x}$. It is tacitly assumed that these observables capture the key dynamics and, thus, are sufficiently rich to ensure a proper embedding. While this assumption is often sensible, it may fail due to the presence of strong nonlinearities and other complexities in the underlying dynamical system. In this case, one may have to resort to more sophisticated observables consisting of a range of nonlinear functions of the measurements to effectively capture the complex manifold on which the dynamics unfolds (Tu et al. 2014b). The result of incorporating these modifications into the core algorithm is known as the extended DMD (EDMD) (Williams et al. 2015).

We start by defining two data sets, D_1 and D_2, consisting of a sequence of state-vector realizations. In a break with the earlier derivation, these realizations do not need to be sampled equidistantly in time; only their associated data points in the second set have to be displaced by a constant time interval Δt. We have $\mathsf{D}_1 = \{\mathbf{d}_{t_1}, \mathbf{d}_{t_2}, \ldots, \mathbf{d}_{t_n}\}$ and $\mathsf{D}_2 = \{\mathbf{d}_{t_1+\Delta t}, \mathbf{d}_{t_2+\Delta t}, \ldots, \mathbf{d}_{t_n+\Delta t}\}$. In addition, we introduce a set of M basis functions denoted by $\Psi(\mathbf{x}) = \{\psi_1(\mathbf{x}), \psi_2(\mathbf{x}), \ldots, \psi_M(\mathbf{x})\}^{\mathrm{T}}$. When applying this function to each member of our two data sets, we obtain $\Psi_1 = \{\Psi(\mathbf{d}_{t_1}), \Psi(\mathbf{d}_{t_2}), \ldots, \Psi(\mathbf{d}_{t_n})\}$ and $\Psi_2 = \{\Psi(\mathbf{d}_{t_1+\Delta t}), \Psi(\mathbf{d}_{t_2+\Delta t}), \ldots, \Psi(\mathbf{d}_{t_n+\Delta t})\}$. Expressing a general observable ϕ in the dictionary of functions Ψ according to $\phi = \Psi\mathbf{a}$ with coefficients $\mathbf{a}$, we proceed by minimizing the residual of mapping Ψ_1 onto Ψ_2 over the set of all n snapshots and compute an approximation of $\mathcal{K}$ in the space of our (nonlinear) observables by solving

$$\mathsf{S} = \mathrm{argmin}_{\mathsf{S}} \frac{1}{2} \sum_{i=1}^{n} \left| \left(\Psi(\mathbf{d}_{t_i+\Delta t}) - \Psi(\mathbf{d}_{t_i})\mathsf{S} \right) \mathbf{a} \right|^2 . \tag{19.}$$

We obtain $\mathsf{S} = \mathsf{G}^{+}\mathsf{A}$, with

$$\mathsf{G} = \frac{1}{n} \sum_{i=1}^{n} \Psi(\mathbf{d}_{t_i}) \Psi(\mathbf{d}_{t_i})^{\mathrm{H}} \qquad \text{and} \qquad \mathsf{A} = \frac{1}{n} \sum_{i=1}^{n} \Psi(\mathbf{d}_{t_i}) \Psi(\mathbf{d}_{t_i+\Delta t})^{\mathrm{H}}, \tag{20.}$$

and where $^+$ denotes the Moore–Penrose pseudo-inverse. With S established, we can determine the spectral properties of $\mathcal{K}$ following $[\mathsf{X}, \Lambda] = \text{eig}(\mathsf{S})$ and $\Phi = \Psi\mathsf{X}$.

The EDMD can be thought of as a higher-order Taylor series expansion near equilibrium points, whereas the standard DMD only captures the linear term and is thus restricted to motion near fixed points. The choice of a dictionary of observables that effectively represents the Koopman operator is an open question and the topic of ongoing research. Various approximations, such as Hermite polynomials, radial basis functions, or discontinuous spectral elements, have been suggested (Li et al. 2017), and cross-validation techniques are necessary to avoid overfitting by the introduced observables. The increased computational burden resulting from a larger observable space can be mitigated by a kernel trick that implicitly generates a suite of observables (see Williams et al. 2014, Pan & Duraisamy 2020a).

3.1.2. Learning the embedding. While the EDMD provides a computational framework for accounting for a set of (nonlinear) observables, it does not answer the question of how to choose them. An attractive approach for tackling the latter issue is a merging of Koopman-based data analysis and modern optimization methodology. Efforts in this direction have been further fueled by the recent rise of machine learning techniques.

Introducing a vector function $\Psi(\mathbf{x})$ as a dictionary of nonlinear observables, we can formulate a minimization problem for the residual of the mapping expressed as $\sum_i \|\Psi(\mathbf{x}_{i+1}) - \mathcal{K}\Psi(\mathbf{x}_i)\|^2$. In this expression, both the dictionary Ψ and the operator $\mathcal{K}$ are the control variables for rendering the residual a minimum (Li et al. 2017). We have

$$\mathcal{K}, \Psi = \text{argmin}_{\mathcal{K},\Psi} \sum_{i=1}^{n} \|\Psi(\mathbf{x}_{i+1}) - \mathcal{K}\Psi(\mathbf{x}_i)\|^2 + \text{reg}(\mathcal{K}, \Psi), \qquad 21.$$

where reg$(\cdot)$ stands for a suitable regularizer (e.g., a Tikhonov term). To be practical, one must parameterize the dictionary by, for example, a three-layer feed-forward neural network of the form $\Psi(\mathbf{x}) = \Psi(\mathbf{x}; \theta) = \mathsf{W}_{\text{out}}\mathbf{h}_3 + \mathbf{b}_{\text{out}}$ and $\mathbf{h}_{k+1} = \tanh(\mathsf{W}_k\mathbf{h}_k + \mathbf{b}_k)$ for $k = 0, 1, 2$ (see Li et al. 2017). The parameters θ in this expression are the network weights W and the biases $\mathbf{b}$ (both for the internal and output layers). The ensuing optimization in $(\mathcal{K}, \theta)$ can be accomplished using an alternating approach: iterating between (*a*) keeping $\mathcal{K}$ fixed, while optimizing with respect to θ, and (*b*) optimizing with respect to $\mathcal{K}$, while keeping θ fixed. Standard algorithms, such as pseudo-inverses and stochastic gradient descent, are applied at each part of the iteration.

An alternative attempt to dictionary learning is the formulation of the embedding by a specific neural network: an autoencoder. In this formulation, we identify a nonlinear mapping to a latent state via multiple layers of neural nets (the encoder), which are followed by a second group of nets (the decoder) that reconstitutes the latent variables back to physical space (see **Figure 6** for a sketch of the layout). Networks of this type are often used in nonlinear model reduction or the compression of data and images. Crucial in our application is the enforcement of linearity in the mapping of the identified latent variables over one time step—the condition for the linear Koopman operator acting on the embedded state-vector. During the training phase of the autoencoder, the reconstruction residual $\mathbf{x}_{i+1} - \phi^{-1}[\mathcal{K}\phi(\mathbf{x}_i)]$, a linearity constraint, and a multiple–time step residual are used to update the weights in the encoder, decoder, and linear mapping (**Figure 6**). The converged design identifies the embedding ϕ and the Koopman operator in the latent variables $\phi(\mathbf{x})$. This type of machine learning technique for finding effective observables for Koopman analysis has been applied to systems with discrete spectra (Otto & Rowley 2019) and with discrete and continuous spectra (Lusch et al. 2018), as well as to uncertain and stabilized systems (Pan & Duraisamy 2020b).

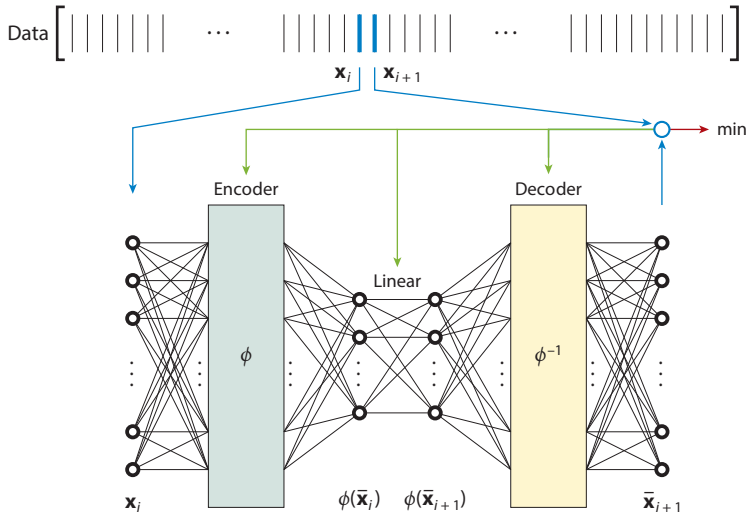

Figure 6

Autoencoder approach to learning the Koopman embedding ϕ and the Koopman operator from training data. The encoder and decoder parts consist of multiple layers of neural nets; the innermost network enforces a linear mapping via the activation function.

3.2. Toward More Robustness

This set of extensions and improvements aims at making the dynamic mode decomposition more insensitive to noise in the data, to outliers, or to a wide range of disparate scales. In addition, certain symmetries in the data, which may be poorly represented by the standard decomposition, call for a preprocessing step or an algorithmic modification.

3.2.1. Sensitivity analysis. Noise is an integral component of experimental data, and it is important to assess the propagation of noise and uncertainty in the input data through the algorithmic steps of the decomposition, as well as to quantify how noise manifests itself in the computed spectral properties. Duke et al. (2012) explored the sensitivity of output variables to noisy data; in particular, the signal-to-noise ratio in the incoming data, the record length of the data sequence, and the resolution of characteristic wavelengths present in the data have been identified as critical parameters that influence the accuracy of spectral information. Several suggestions for the improvement of the algorithm and for effective data preprocessing have been made.

Building on weak-noise theory for noisy limit-cycles, Bagheri (2014) demonstrated a damping effect on the DMD spectrum with a quadratic dependence on the frequency and a linear dependence on the amplitude. This effect resulted in an alignment of the computed eigenvalues along parabolic arcs in the complex spectral plane.

3.2.2. Data preprocessing. Preprocessing the data sets before applying a decomposition is a convenient and simple way of (*a*) minimizing the influence of noise or (*b*) rebalancing data sets with vastly different amplitudes. While manipulating the columns of the data matrix does not influence $\mathcal{K}$, it may well improve the condition number of the data matrix and thus yield a more well-conditioned algorithm. Drmač et al. (2017) suggested a general row-and-column scaling, combined with refined Rayleigh–Ritz techniques, to obtain precise and robust spectral properties of the Koopman operator, even from noisy data. Caution has to be exercised, however, in

the presence of substantial noise, as certain noise components could be amplified via the scaling procedure.

A different approach with the same goal is based on the L+S decomposition, a technique for factoring a (noisy) data sequence into a low-rank component L and sparse component S (Candès et al. 2011). This method is commonly applied in video surveillance when breaking video footage into persistent (low-rank) structures and intermittent (sparse) events. Mathematically, we have

$$\mathsf{L}, \mathsf{S} = \text{argmin}_{\mathsf{L},\mathsf{S}} \|\mathsf{L}\|_* + \mu\|\mathsf{S}\|_1, \qquad \text{such that} \quad \mathsf{D} = \mathsf{L} + \mathsf{S}, \tag{22.}$$

where $\|\cdot\|_*$ denotes the nuclear norm, a differentiable proxy for rank; $\|\cdot\|_1$ represents the L_1-norm, which is sparsity promoting (as already exploited for the amplitude calculations); and μ is a user-supplied parameter weighing one effect against the other. The above mixed-norm optimization problem can readily be solved using iterative techniques, such as ADMM, the iteratively reweighted least-squares algorithm, or the augmented Lagrange multiplier method (as in Scherl et al. 2020). Eliminating the noise by discarding the identified S component leaves a low-rank replacement L of the original data sequence D. This denoising procedure (i.e., replacing D by L) adds robustness to the subsequent DMD analysis and yields spectral information with increased fidelity.

3.2.3. Translational structures and rotated reference frames. Translational structures, as they often occur in convectively dominated flows, are difficult to represent as a modal superposition of single-frequency dynamics. Convected coherent structures often appear as multiple modes displaying a sequence of higher-harmonic frequencies and spatial wavenumbers, but nearly constant local phase/group speeds. The difficulty in representing this type of structure can be alleviated by using a preprocessing step that aligns the analysis along characteristic coordinates in the data set (Sesterhenn & Shahirpour 2019). By maximizing the drop in the dominant singular values of D via a rotation of the reference frame from a physical to a characteristic formulation, one can make notable improvements, with a more compact representation of the modal structures.

3.2.4. Adjusting the bases. In the core algorithm, the Koopman operator is expressed in the basis of proper orthogonal modes of the data sequence D_1^{n-1}. This choice will result in a robust and efficient algorithm, despite the fact that this basis may not be optimal. An optimal representation of the Koopman operator can be formulated using the expression $\mathcal{K} = \mathsf{L}\mathsf{M}\mathsf{L}^{\mathrm{H}}$, with L denoting a low-rank basis and M a reduced description of $\mathcal{K}$. The associated optimization problem can then be stated as

$$\mathsf{L}, \mathsf{M} = \text{argmin}_{\mathsf{L},\mathsf{M}} \|\mathsf{D}_2^n - \mathsf{L}\mathsf{M}\mathsf{L}^{\mathrm{H}}\mathsf{D}_1^{n-1}\|, \qquad \text{with} \quad \mathsf{L}^{\mathrm{H}}\mathsf{L} = \mathsf{I}. \tag{23.}$$

In other words, rather than computing the Koopman operator in a given basis (U), we simultaneously determine the optimal basis (L) and the Koopman operator (M) expressed in this basis (Wynn et al. 2013). The spectral information is then given by $[\mathsf{X}, \Lambda] = \text{eig}(\mathsf{M})$ and $\Phi = \mathsf{L}\mathsf{X}$. While this optimal mode decomposition shows improved robustness to noisy data, it requires a more complex (nonconvex) algorithm for solving the above optimization problem on a Grassmannian manifold given by $\mathsf{L}^{\mathrm{H}}\mathsf{L} = \mathsf{I}$; moreover, the rank of the Koopman representation, $r = \text{rank}(\mathsf{L}\mathsf{M}\mathsf{L}^{\mathrm{H}})$, has to be user supplied.

Another attempt at modifying the basis for the DMD has been undertaken by Noack et al. (2016), who introduced the recursive DMD. It commences with a standard DMD run, after which the member of the dynamic mode basis is selected that maximally reduces the residual norm of the current reconstruction error. This selection procedure is then repeated in the space orthogonal

to the already identified modes, and terminates when the desired number of modes has been determined. By design, the recursively computed modes form an orthogonal basis, while preserving the monochromatic (single-frequency) feature of DMD modes.

3.2.5. Exact DMD. The projection of $\mathcal{K}$ onto the basis U of the data sequence D_1^{n-1} leads to a more robust DMD algorithm. The approximate eigenfunctions $\Phi = \mathsf{U}\mathsf{X}$ are eigenfunctions of the projected operator $\mathsf{U}^{\mathrm{H}}\mathcal{K}\mathsf{U}$, but not necessarily of $\mathcal{K}$ (Tu et al. 2014b). The eigenfunctions of $\mathcal{K}$ should be in the column space of D_2^n rather than D_1^{n-1}. In practice, the column spaces of D_2^n and D_1^{n-1} are nearly identical owing to the fact that the snapshots $[\mathbf{d}_2, \ldots, \mathbf{d}_{n-1}]$ appear in both data sets. Nonetheless, for exact eigenfunctions of $\mathcal{K}$ we have to use the identities $\mathcal{K}\Phi = \mathsf{D}_2^n\mathsf{V}\mathsf{\Sigma}_\epsilon^{-1}\mathsf{U}^{\mathrm{H}}\Phi = \Phi\Lambda$ and $\mathsf{U}^{\mathrm{H}}\Phi\Lambda = \bar{\mathsf{S}}\mathsf{U}^{\mathrm{H}}\Phi$, from which we recover the exact modes via $\Phi = \mathsf{D}_2^n\mathsf{V}\mathsf{\Sigma}_\epsilon^{-1}\mathsf{X}$. The projected and exact formulations are equivalent when the two data sequences D_1^{n-1} and D_2^n span the same column space. The algorithm with the exact recovery of the eigenfunctions of $\mathcal{K}$ is termed exact DMD (Tu et al. 2014b).

3.2.6. Debiased or total least-squares DMD and forward–backward DMD. In the formulation of the core DMD algorithm, a least-squares problem has been solved when matching two shifted data sequences, D_1^{n-1} and D_2^n. In this process, the shifted data set D_2^n has been taken as noise-free, while D_1^{n-1} has been assumed to contain noise. Since the bulk of the snapshots can be found in both sequences, it appears more rational to accept noise-contaminated snapshots in both data sequences, debias the projection of one set onto the other, and formulate a total least-squares rather than a standard least-squares problem (Hemati et al. 2017). In this manner, the residual of the matching process is spread among both data sequences. We form a stacked data sequence $\bar{\mathsf{D}} = [\mathsf{D}_1^{n-1}; \mathsf{D}_2^n]$ and compute the SVD according to $\bar{\mathsf{D}} = \mathsf{U}\mathsf{\Sigma}\mathsf{V}^{\mathrm{H}}$. The columns of V are then used to construct a projector $\mathbb{V} = \mathsf{V}\mathsf{V}^{\mathrm{H}}$. The debiased data sequences are then given as $\mathsf{D}_1^{n-1}\mathbb{V}$ and $\mathsf{D}_2^n\mathbb{V}$, which are then fed as the two data sequences into the core algorithm. This modification improves the recovery of spectral information, particularly in the presence of noise in the measurement snapshots.

In another attempt at robustifying the DMD, Dawson et al. (2016) suggested a two-step process where the core algorithm is applied twice: first to extract the usual mapping from D_1^{n-1} to D_2^n, and second to determine a (inverse) mapping from D_2^n to D_1^{n-1}. These two mappings, denoted respectively by $\bar{\mathsf{S}}_{\mathrm{f}}$ (forward) and $\bar{\mathsf{S}}_{\mathrm{b}}$ (backward), are then combined into $\bar{\mathsf{S}}_{\mathrm{fb}} = \sqrt{\bar{\mathsf{S}}_{\mathrm{f}}\bar{\mathsf{S}}_{\mathrm{b}}}$ and used in place of $\bar{\mathsf{S}}$ in the core algorithm. The main idea behind this forward–backward DMD is the approximate cancellation of noise in the data sequence, as the noise component is assumed to be independent of the respective evolution direction. Care must be taken when computing the square root of the matrix product $\bar{\mathsf{S}}_{\mathrm{f}}\bar{\mathsf{S}}_{\mathrm{b}}$.

3.3. Toward More Efficiency

When processing large data, challenges arise on many levels, and our decomposition algorithms have to be adjusted to accommodate large matrices. In general, we are only interested in the few modal structures that describe the bulk of the dynamics. For this reason, low-rank and sparsity concepts can be and should be exploited.

3.3.1. Parallelization. The key algorithm underlying all DMD computations is the QR decomposition, which is central to the SVD of tall-and-skinny matrices. For highly resolved, 3D data fields, the resulting snapshot sequences can take on dimensions that require a parallel approach to computing the SVD. This can be accomplished by incorporating the parallel QR decomposition

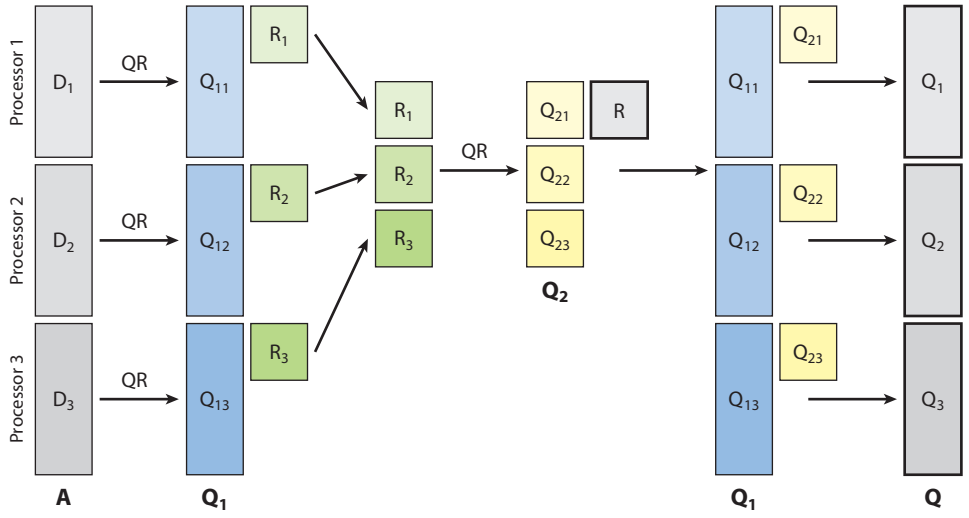

Figure 7

Parallel QR algorithm for a tall-and-skinny matrix $\mathsf{D} = \mathsf{QR}$, illustrated for three processors. The original matrix is partitioned, and each partition is QR-decomposed. The triangular parts are gathered and again QR-decomposed, resulting in the final triangular part. The final orthogonal part consists of the orthogonal part of the partitions and the orthogonal part of the gathered triangular part.

of Demmel et al. (2012) into our core algorithm. It partitions the data matrix, performs a QR decomposition of each partition, gathers the R part of each partition, and performs another QR decomposition. This final QR decomposition then produces the ultimate R component, while the Q part is used to assemble the Q parts of each partition into the final Q component. The procedural steps of this algorithm are illustrated in **Figure 7** for the case of three processors.

An SVD of the final R component then produces the SVD of the entire data matrix. This parallelization strategy has been incorporated into the core DMD algorithm and has been applied to very large data sets (Sayadi & Schmid 2016) from direct numerical simulations of transitional boundary layers and thermal convection. It has shown excellent scaling on a massively parallel computing architecture.

3.3.2. Randomized algorithms. The underlying assumption of coherence in the data stream often leads to matrices with a substantial gap in their singular values. This near rank-deficiency can be exploited effectively by randomized methods (Halko et al. 2011). This type of algorithm allows for the computation of many characteristics, such as the range or singular vectors, from the action of the low-rank matrix and its transpose on a suite of random vectors. These random matrix-vector-products are also termed sketches.

Randomized algorithms provide an alternative viewpoint to Krylov subspace methods where we express the action of a large matrix A in a space generated by repeated application of A on a single starting vector $\mathbf{b}$. In the randomized approach, a single application of A on multiple random vectors produces the basis for the extraction of spectral information:

$$\underset{\text{Krylov techniques}}{\text{span}\{\mathbf{b}, \mathsf{A}\mathbf{b}, \mathsf{A}^2\mathbf{b}, \ldots, \mathsf{A}^{n-1}\mathbf{b}\}} \quad \longleftrightarrow \quad \underset{\text{Randomized approach}}{\text{span}\{\mathsf{A}\omega_1, \mathsf{A}\omega_2, \ldots, \mathsf{A}\omega_n\}}.$$

Algorithms based on the projection onto these spans are highly efficient, accurate (in a stochastic sense), and particularly suited to data sequences and matrices with a reduced range—a situation

we often encounter in data stemming from coherent fluid motion. The incorporation of randomized algorithms can significantly increase the range of applicability of decomposition techniques in general, and of DMD in particular. Demonstrations of the efficiency of this randomized approach are given by Erichson et al. (2019).

3.3.3. Streaming and incremental algorithms. Especially in experimental settings, we encounter situations where data become available in a streaming manner—even in real time as the experiment progresses. In this case, we are particularly interested in incremental versions of the key algorithms underlying our chosen decomposition. Given a data set $\mathsf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n]$ and the accompanying SVD U, Σ, and V, we are concerned with the updated decomposition as a snapshot $\mathbf{d}_{n+1}$ is added to the data set. Rather than computing a new SVD from scratch, we seek to adjust the current components U, Σ, and V to account for the additional data point $\mathbf{d}_{n+1}$. This is accomplished by formulating the addition of $\mathbf{d}_{n+1}$ as a rank-one modification of an augmented data set (Brand 2002). Within this formalism, only small adjustments have to be made to incorporate the new snapshot into the updated decomposition. A version of DMD based on this incremental methodology has been proposed by Hemati et al. (2014) and successfully applied to streaming experimental data. An additional advantage of this technology is also the markedly reduced memory footprint, which allows for the treatment of data that otherwise could not reside fully in memory. Schmid (2021) has provided details of updating (adding new snapshots) and downdating (removing old snapshots) modifications for the SVD, and thus for DMD, that cover the addition, removal, and shifting of data points in a finite snapshot window with a known decomposition. Incremental algorithms are a key technology for the realization of an online, real-time DMD analysis.

3.3.4. Multiresolution DMD. The occurrence of widely separated timescales in the data sequence poses algorithmic challenges to any data decomposition and, hence, to DMD. For these situations, an extension to the core algorithm can be employed consisting of a recursive computation that progressively removes low-frequency (slow-motion) features from the data sequence and processes the filtered remainder. At each step the remaining data sequence is split into two subsequences on which the same low-frequency removal process is continued. In this manner, a time–frequency analysis is performed that is capable of and effective at detecting elements in the data sequence that are localized in time and widely separated in scale. The removal-splitting step is continued until the entire data sequence is processed. Regime-switching in time–frequency space can be detected, and an adjustment of the sampling rate at each level can further improve computational efficiency. The resulting multiresolution DMD (Kutz et al. 2016b) incorporates concepts from wavelet-based multiscale analysis and filter banks and thus exhibits superior resolution power of flow features that are localized both in the primal (e.g., time) and dual (e.g., frequency) independent variable.

3.3.5. Compressed sensing and sub-Nyquist sampled data. Compressed sensing (Candès et al. 2006) has had a tremendous impact on many application areas, from general signal processing to digital photography, and from medical image recovery to network tomography. It relies on the expression of a signal in a basis in which it appears sparsely. When combined with a sparsity-promoting optimization method (similar to the method applied in the amplitude computation above), compressed sensing allows for the recovery of signals from highly incomplete frequency information. In this manner, the common Nyquist–Shannon criterion can be surpassed, and signals can be accurately recovered, even when sampled at sub-Nyquist rates. This technology can also be incorporated into the identification of coherent structures by the DMD (Tu et al. 2014a, Brunton et al. 2015). The application to particle image velocimetry (PIV) data for flow past a

cylinder, sampled at sub-Nyquist rates, has been reported by Tu et al. (2014a). The correct frequencies and associated modal structures could be recovered successfully, which demonstrates the potential of combining compressed-sensing techniques with data decomposition methods to optimally leverage sensors and produce accurate information beyond their Nyquist limits.

3.4. Conceptual Extensions

The core algorithm has also been carried beyond its application to quantitative flow analysis. It has been incorporated as a subcomponent into many, more complex algorithms and has been adapted to data sequences stemming from systems other than evolution processes. In addition, important advances in the underlying theory of transfer operators have produced—and continue to produce—new tools for the analysis of dynamical systems.

3.4.1. DMD with control. An important extension of the DMD is concerned with the treatment of externally driven data sequences, which extend the analysis of intrinsic flow features to the description of the response behavior to external forcing (Brunton et al. 2016, Proctor et al. 2016). The governing equation for the discrete-time data sequence is given as

$$\mathsf{D}_2^n = \mathcal{K}\mathsf{D}_1^{n-1} + \mathsf{B}\mathsf{F}_1^{n-1}, \tag{24.}$$

with F as the forcing sequence $\mathsf{F}_1^{n-1} = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_{n-1}]$ and B as the control matrix, which describes how the forcing interacts with the system governed by $\mathcal{K}$. Two cases have to be distinguished: (*a*) the case where the details of the driving, encapsulated in B, are known, and (*b*) the case where B is unknown and has to be identified, together with $\mathcal{K}$, from the data.

In the first case, we can employ a simple lifting of the forcing to reformulate the problem as $\mathsf{D}_2^n - \mathsf{B}\mathsf{F}_1^{n-1} = \mathcal{K}\mathsf{D}_1^{n-1}$. We then proceed as before with the compensated output data sequence $\mathsf{D}_2^n - \mathsf{B}\mathsf{F}_1^{n-1}$ and the input data sequence D_1^{n-1} and apply the core algorithm to arrive at

$$\tilde{\mathsf{S}} = \mathsf{U}^{\mathrm{H}}\left(\mathsf{D}_2^n - \mathsf{B}\mathsf{F}_1^{n-1}\right)\mathsf{V}\boldsymbol{\Sigma}_\epsilon^{-1}, \qquad \text{with} \qquad \mathsf{D}_1^{n-1} = \mathsf{U}\boldsymbol{\Sigma}\mathsf{V}^{\mathrm{H}}, \tag{25.}$$

as the projected approximation of the Koopman operator $\mathcal{K}$ for the driven problem. The Koopman eigenfunctions $\boldsymbol{\Phi}$ follow from there in the usual manner, that is, $\boldsymbol{\Phi} = \mathsf{U}\mathsf{X}$ with $[\mathsf{X}, \boldsymbol{\Lambda}] = \mathrm{eig}(\tilde{\mathsf{S}})$, or $\left(\mathsf{D}_2^n - \mathsf{B}\mathsf{F}_1^{n-1}\right)\mathsf{V}\boldsymbol{\Sigma}_\epsilon^{-1}\mathsf{X}$ for the exact recovery.

For the second case (with unknown B), we augment the evolution operator according to

$$\mathsf{D}_2^n = [\mathcal{K}\ \mathsf{B}]\begin{bmatrix}\mathsf{D}_1^{n-1}\\ \mathsf{F}_1^{n-1}\end{bmatrix} \tag{26.}$$

and treat the composite operator $[\mathcal{K}\ \mathsf{B}]$ as the unknown to be determined from the data sequences on both sides of the equation. This formulation is quite common in subspace system identification (Van Overschee & De Moor 2012). We proceed as before using an SVD of the composite data sequence on the right, that is,

$$\begin{bmatrix}\mathsf{U}_\mathsf{D}\\ \mathsf{U}_\mathsf{F}\end{bmatrix}\boldsymbol{\Sigma}\mathsf{V}^{\mathrm{H}} = \mathrm{svd}\left(\begin{bmatrix}\mathsf{D}_1^{n-1}\\ \mathsf{F}_1^{n-1}\end{bmatrix}\right), \tag{27.}$$

which gives us an input space representation. From this decomposition, we approximate the two unknown operators as

$$[\mathcal{K},\ \mathsf{B}] = [\mathsf{D}_2^n\mathsf{V}\boldsymbol{\Sigma}_\epsilon^{-1}\mathsf{U}_\mathsf{D}^{\mathrm{H}},\ \ \mathsf{D}_2^n\mathsf{V}\boldsymbol{\Sigma}_\epsilon^{-1}\mathsf{U}_\mathsf{F}^{\mathrm{H}}]. \tag{28.}$$

In a departure from earlier analysis, it is necessary to express the dynamics in output space (i.e., in a basis that spans the output sequence D_2^n). To this end, we have to employ a final transformation

of the above identified operators $[\mathcal{K},\ \mathsf{B}]$ onto the basis $\hat{\mathsf{U}}$ given by $\hat{\mathsf{U}}\hat{\Sigma}\hat{\mathsf{V}}^{\mathrm{H}} = \mathrm{svd}(\mathsf{D}_2^n)$. We finally have $\hat{\mathsf{S}} = \hat{\mathsf{U}}^{\mathrm{H}}\mathcal{K}\hat{\mathsf{U}}$ and $\hat{\mathsf{B}} = \hat{\mathsf{U}}^{\mathrm{H}}\mathsf{B}$ as the two operators properly transformed to the output space. The spectral information follows as $\Phi = \hat{\mathsf{U}}\mathsf{X}$, or, for the exact version, $\Phi = \mathsf{D}_2^n\mathsf{V}\Sigma_\epsilon^{-1}\mathsf{U}_{\mathsf{D}}^{\mathrm{H}}\hat{\mathsf{U}}\mathsf{X}$ with $[\mathsf{X}, \Lambda] = \mathrm{eig}(\hat{\mathsf{S}})$.

This framework applies to dynamical systems with inputs: It allows for the analysis of driven processes and paves the way for a wide range of control-theoretical applications to processes that are only observable through measurements and data.

3.4.2. Time-delay embedding, Hankel DMD, and higher-order DMD. DMD is as attractive in deducing coherent flow features from data sequences (spectral analysis) as it is in representing the principal dynamics reflected in the data (system identification). The transfer from an analysis tool to a prediction tool is greatly aided by embedding the data sequence using delay coordinates. This delay embedding is a common method for expressing ergodic attractors of nonlinear dynamical systems (Takens 1981); for example, it has been applied in signal analysis and forecasting using singular spectral analysis (Golyandina et al. 2001).

Given a data sequence D consisting of n samples, we construct a Hankel matrix D_{H} of the form

$$\mathsf{D} = [\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n] \qquad \longrightarrow \qquad \mathsf{D}_{\mathsf{H}} = \begin{bmatrix} \mathbf{d}_1 & \mathbf{d}_2 & \mathbf{d}_3 & \cdots & \mathbf{d}_{n-3} \\ \mathbf{d}_2 & \mathbf{d}_3 & \mathbf{d}_4 & \cdots & \mathbf{d}_{n-2} \\ \mathbf{d}_3 & \mathbf{d}_4 & \mathbf{d}_5 & \cdots & \mathbf{d}_{n-1} \\ \mathbf{d}_4 & \mathbf{d}_5 & \mathbf{d}_6 & \cdots & \mathbf{d}_n \end{bmatrix}, \tag{29.}$$

where we have applied an embedding dimension of three. The stacked data matrix D_{H} is then used in place of D and processed by the core algorithm. Arbabi & Mezić (2017) have shown the convergence of this Hankelized approach to the eigenfunctions of the Koopman operator; they also illustrated remarkable improvements in the prediction of simple and complex fluid systems. Further examples and theoretical results on delay embedding and the Hankel viewpoint of Koopman analysis have been given by Brunton et al. (2017), Kamb et al. (2020), and Pan & Duraisamy (2020a); they convincingly demonstrated that linear time-delayed models are an effective and efficient tool to capture nonlinear and chaotic dynamics.

Delay embedding is also an important technique when the temporal/spectral complexity of a dynamical system exceeds the spatial complexity, for example, in systems that are characterized by a broadband spectrum or are spatially undersampled. In this case, we arrive at a short-and-wide, rather than a tall-and-skinny, data matrix D, and the core algorithm fails at extracting all relevant spectral features. A higher-order DMD has been introduced by LeClainche & Vega (2017) to address this issue. It relies on a delay embedding across multiple dimensions, with

$$\mathbf{d}_{i+d} = \mathsf{R}_1\mathbf{d}_i + \mathsf{R}_2\mathbf{d}_{i+1} + \cdots + \mathsf{R}_d\mathbf{d}_{i+d-1}, \qquad i = 1, \ldots, n-d, \tag{30.}$$

for a d-dimensional delay embedding. The resulting mapping is given by $\mathsf{D}_{\mathsf{H},2}^{\ n} = \mathcal{K}_{\mathsf{H}}\mathsf{D}_{\mathsf{H},1}^{\ n-1}$, with

$$\mathcal{K}_{\mathsf{H}} = \begin{bmatrix} 0 & \mathsf{I} & & & \\ & 0 & \mathsf{I} & & \\ & & \ddots & \ddots & \\ & & & 0 & \mathsf{I} \\ \mathsf{R}_1 & \mathsf{R}_2 & \mathsf{R}_3 & \cdots & \mathsf{R}_d \end{bmatrix}. \tag{31.}$$

With the augmented (Hankelized) data matrices and the definition of the higher-order Koopman operator $\mathcal{K}_{\mathsf{H}}$, we can apply the core algorithm and extract spectral information from temporally

broadband or spatially sparse data sequences. The higher-order extension adds more robustness and flexibility to the standard algorithm and enables the analysis of systems for which temporal resolution is substituted for spatial resolution.

3.4.3. The adjoint Koopman operator. While the linear Koopman operator $\mathcal{K}$ describes the mapping of observables over one time step and builds the theoretical foundation of the data-driven DMD, much can be learned from the linear operator adjoint to $\mathcal{K}$, termed the Frobenius–Perron operator $\mathcal{P}$ (Lasota & Mackey 1994). Over the past decade, it has been used successfully in the description of dynamical systems based on models or observational data, with examples from molecular dynamics, meteorology, oceanography, and engineering systems. The body of literature and recent accomplishments are too expansive to review here; only a few key tools for the analysis of fluid systems are given below. The reader is urged to refer to the work of Froyland & Padberg (2009), Junge & Koltai (2009), and Klus et al. (2018, 2020) for more details.

The Frobenius–Perron operator $\mathcal{P}$ describes the evolution of densities. Given a probability density $\rho(\mathbf{x})$ we define a mapping of ρ over a discrete time step as

$$\rho_{i+1}(\mathbf{x}_{i+1}) = \mathcal{P}\rho_i(\mathbf{x}_{i+1}) = \int \delta[\mathbf{x}_{i+1} - \mathbf{F}(\mathbf{x}_i)]\rho_i(\mathbf{x}_i)\, \mathrm{d}\mathbf{x}_i. \qquad 32.$$

The kernel expression $\delta[\mathbf{x}_{i+1} - \mathbf{F}(\mathbf{x}_i)]$ realizes the conditional probability density of reaching the state $\mathbf{x}_{i+1}$ from $\mathbf{x}_i$, over one time step, following the dynamics $\mathbf{F}$. In other words, while the Koopman operator propagates observables forward in time from $\phi(\mathbf{x}_i)$ to $\phi(\mathbf{x}_{i+1})$, the Frobenius–Perron operator evolves the probability density ρ by accounting for all densities that are linked via the one-step dynamics encoded in $\mathbf{F}$.

From a computational perspective, the Frobenius–Perron operator is most conveniently discretized using Ulam's method (Ulam 1960, Junge & Koltai 2009, Bollt & Santitissadeekorn 2013). To this end, the data sequence is described by tracing it in a sufficiently high-dimensional phase space and by assigning each data snapshot $\mathbf{d}_i$ to one of a finite number of disjoint hypercubes $\mathbb{B}_k$ covering the part of phase space that is occupied by the data sequence trajectories. A discrete approximation of $\mathcal{P}$ is then given by the matrix

$$\mathsf{P}_{ij} = \frac{\#_k\left\{(\mathbf{d}_k \in \mathbb{B}_i) \text{ and } (\mathbf{d}_{k-1} \in \mathbb{B}_j)\right\}}{\#_k\left\{(\mathbf{d}_k \in \mathsf{B}_i)\right\}}, \qquad 33.$$

where $\#_k\{\cdot\}$ denotes a simple counting function over the realizations k and $(\cdot)$ represents a Boolean expression. In words, we count the number of occurrences where snapshots from hypercube $\mathbb{B}_i$ were, one time step ago, in hypercube $\mathbb{B}_j$ and divide by the total number of snapshots in $\mathbb{B}_i$. This provides a probabilistic link between regions of phase space under the governing dynamics $\mathbf{F}$.

Under this formulation, the matrix P is row stochastic, represents a finite Markov chain, and encodes the dynamics $\mathbf{F}$. Coherence in time, or temporal persistence, is expressed by eigenvectors of P corresponding to eigenvalues close to one in magnitude, as they describe nearly invariant structures (see, e.g., Froyland & Padberg 2009, Junge & Koltai 2009, Froyland et al. 2013). Transitory structures, in contrast, would have eigenvalues less than one in magnitude. Tools from dynamical systems, linear algebra, graph theory, and Markov chains can be brought to bear on this computational framework, and much information can be gained from an analysis of the Frobenius–Perron operator $\mathcal{P}$. This information will enhance and complement the results from a Koopman analysis based on $\mathcal{K}$. In quantitative flow analysis, the Frobenius–Perron approach has been exploited in a data-driven cluster-based analysis of shear flow instabilities (Kaiser et al. 2014) and the description of rare events in models of turbulent flows (Schmid et al. 2018).

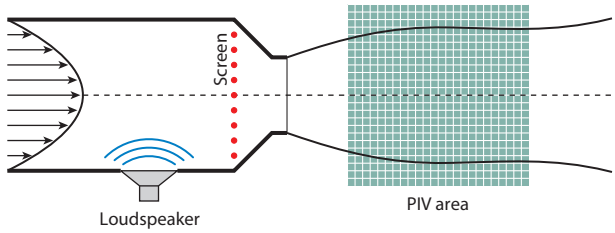

Figure 8

Dynamic mode decomposition analysis of an axisymmetric jet subject to harmonic forcing. Time-resolved particle image velocimetry (PIV) data are processed for a square domain downstream of the nozzle. Two cases are considered in the main text: without acoustic forcing, displaying the intrinsic stability behavior of the jet, and with acoustic forcing, probing the response behavior of the jet to harmonic excitation.

4. APPLICATIONS

DMD and its many variants and extensions have been applied to many configurations and areas—far too many to summarize here. Three examples will be given to highlight typical features, to venture beyond standard flow analysis, and to encourage exploration of complex flows with this effective tool.

4.1. Temporal Analysis

The first example is temporal analysis, arguably the most common application of DMD. Observables are obtained from measurement points in experiments or from state-vector components on grid points in simulations and gathered into a data sequence with time as the evolution direction. If these observables are sufficiently rich in capturing the intrinsic dynamics, the Koopman operator can be approximated by the DMD. The data sets can range in complexity from a small collection of hotwire signals to image sequences from a high-speed camera, and from Schlieren movies to time-resolved tomographic PIV measurements. Even combinations of data sets of varying complexities are possible.

For our example, we consider a circular jet that can be acoustically driven by a loudspeaker in a plenum [see **Figure 8** for a sketch (Schmid et al. 2011)]. The structures arising in the jet downstream from the nozzle are measured by time-resolved PIV. The Reynolds number of the flow based on the centerline velocity and jet exit diameter is $Re = 14{,}767$. We acquire $n = 101$ snapshots with a spatial resolution of 63×63 for the two measured velocity components. Two data sets will be considered: the unforced case with no acoustic excitation by the loudspeaker, and a forced case probing the response of the jet to acoustic forcing. Both cases have been processed by the sparsity-promoting DMD algorithm. For the forced case, a scalar signal from the loudspeaker could optionally be included in the state-vector and the resulting data matrix. This extension would uncover a phase relationship between the forcing loudspeaker and the forced jet response—and follow a simplified version of the approach outlined above for DMD with control.

In the unforced case, a DMD analysis reveals the intrinsic frequency response at Strouhal numbers of $St = 0.28$ and $St = 0.46$, which fall within the typical range of preferred modes in axisymmetric jets. The associated structures are displayed in **Figure 9*a*,*b***. We observe a primary instability at $St = 0.46$ that undergoes a vortex-merging process further downstream, inducing a pronounced response at $St = 0.28$.

In the forced case, the jet responds quite differently: We observe a lock-on to the forcing frequency in the jet, giving rise to a distinct structure with significant amplitude in the data sequence

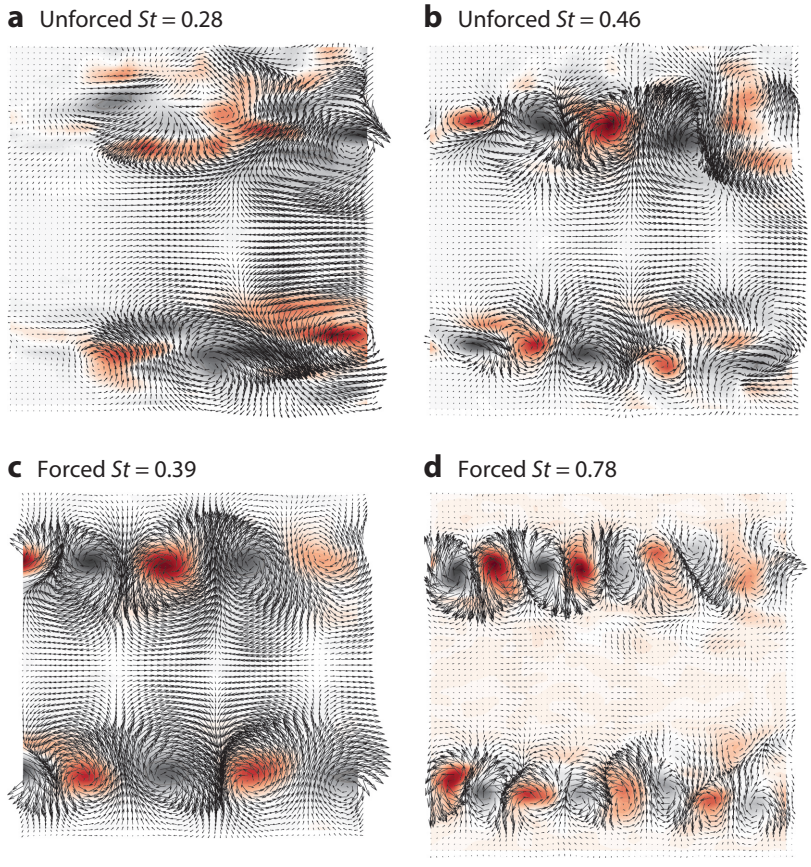

Figure 9

Dominant dynamic modes of an axisymmetric jet (*a*,*b*) without and (*c*,*d*) under harmonic forcing for different Strouhal numbers *St*.

(see **Figure 9*c***); a Strouhal number of $St = 0.39$ is detected, with a pronounced second peak at a higher-harmonic frequency of $St = 0.78$ (see **Figure 9*d***). The higher harmonic shows the typical doubling of the streamwise wavenumber, suggesting a superposition of the two forced modes in **Figure 9*c*,*d*** in a phase-locked linear combination. This flow behavior is strongly suggestive of a convective global instability in the axisymmetric jet.

4.2. Composite Analysis

DMD is a data-based method and thus not bound to governing equations or other mathematical constraints. Instead, we can uninhibitedly construct and explore data sets as to their patterns and correlations. This is demonstrated with this example.

We consider a compressible axisymmetric jet exiting a circular nozzle, with a Mach number of $Ma = 0.9$ (see **Figure 10**). We are particularly interested in the coherent structures preferred by the jet, regarding both the hydrodynamic component and the associated acoustic radiation. To this end, we construct composite snapshots from the numerical simulations, consisting of a first component that represents hydrodynamic effects (the azimuthal vorticity), and a second component that acts as a proxy for acoustic activity (the dilatation). Processing this synchronized data set

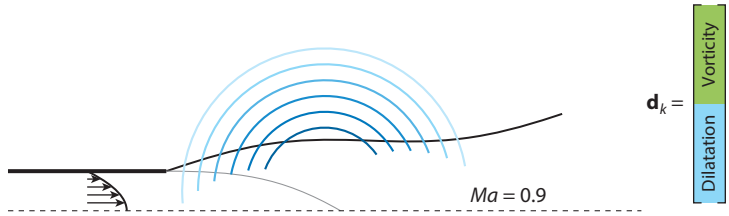

Figure 10

Composite analysis of a compressible axisymmetric jet emanating from a circular pipe with a Mach number of $Ma = 0.9$. The data sequence consists of the azimuthal vorticity component and the dilatation, capturing respectively hydrodynamic and aeroacoustic effects.

by the core algorithm reveals a hierarchy of modal patterns; three of these modes with significant amplitudes are shown in **Figure 11**. We observe modes with only hydrodynamic effects, but a negligible acoustic footprint (**Figure 11*a*,*b***). This stands in contrast to modes that exhibit a substantial acoustic component, with similar hydrodynamic features (**Figure 11*c–f***). In this manner, we can distinguish between quiet and loud modal structures and relate features of the hydrodynamics to sound pressure patterns.

This flexibility in data analysis, owing to the equation-free approach of DMD, can be exploited in many situations and can become an attractive feature in the analysis of multiphysics phenomena or inherent feedback loops. For example, Sayadi et al. (2014) investigated a link between the skin friction signature on the wall and the near-wall structures responsible for it. A composite DMD analysis of a transitional boundary layer uncovered a surprisingly low-dimensional connection between streak-like fluid elements and the induced skin friction pattern. Similar analyses can be performed, for example, for fluid–structure interactions, where structural deformations and flow patterns are linked in a causal relationship through composite data processing. Combustion and multiphase data would be amenable to a similar analysis.

Composite DMD analysis is closely linked to EDMD, which uses a set of observables to capture the essence of a dynamical system. While the latter approach (EDMD) is motivated by a completeness argument for the embedding of a nonlinear system, the former approach (composite DMD) aims at a similar complete description of the dynamics using physical intuition.

4.3. Streamline-Based Analysis

The final example takes the DMD's flexibility of data processing due to its detachment from governing equations one step further. It abandons the common assumption of a temporal evolution direction and, hence, a stacking of columns in D corresponding to snapshots at different instants in time. Rather, we use a spatial coordinate to describe the evolution of structures. Moreover, this evolution will be along a curved trajectory in space.

As an example, we consider a jet in cross-flow, exiting perpendicularly into a boundary layer with a jet-to-freestream ratio of $R = 3$. The Reynolds number based on the jet diameter is $Re = 500$. The flow is characterized by a breakdown of vortical structures developing along a curved counter-rotating vortex pair (CVP). A typical snapshot is presented in **Figure 12*a***. For the DMD analysis, we determine a base-flow streamline emanating from the center of the transverse jet. We then define normal planes at equispaced distances along the streamline's arclength; a total of 110 planes have been chosen (see **Figure 12*b*** for a sketch).

Projecting the 3D velocities onto these normal planes and stacking them as columns into the data matrix D, we can perform a streamline-based DMD analysis of the spatial evolution of fluid structures along the counter-rotating vortex sheet. The six most dominant modes are displayed

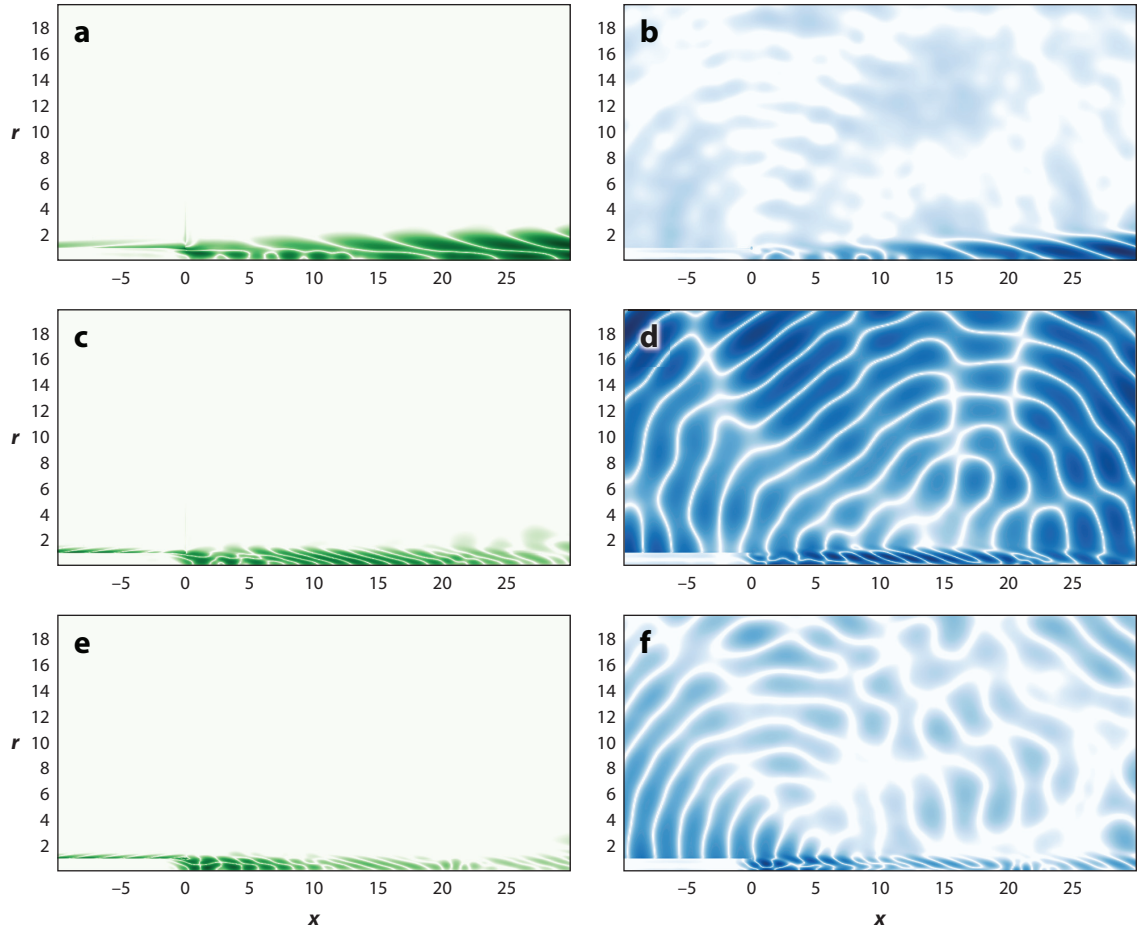

Figure 11

Three dominant composite modes, showing the vortical (*left column*; *a*,*c*,*e*) and acoustic (*right column*; *b*,*d*,*f*) components of the dynamic modes.

in **Figure 13**, showing a vortical breakdown of the CVP, which increasingly concentrates on the flanks of the base-flow vortex sheet for higher DMD modes, in accordance with numerical and experimental observations. It should be stressed that these modes will evolve spatially, and the DMD eigenvalues represent complex wavenumbers (rather than frequencies) along the curved streamline.

This example demonstrates the adaptability of the analysis tool to complex flow situations, rooted in the equation-free formulation of the Koopman framework. An equation-based equivalent analysis would be too unwieldy, if not infeasible. By contrast, a geometric rearrangement of the 3D data is all it took to perform a data-driven analysis of the essential flow patterns.

4.4. Other Application Areas

Koopman analysis and the DMD have seen a wide expansion across many application areas. They are employed in the detection of patterns, the extraction of reduced-order models, and the prediction of time series based on previous observations. Applications in epidemiology (spreading of

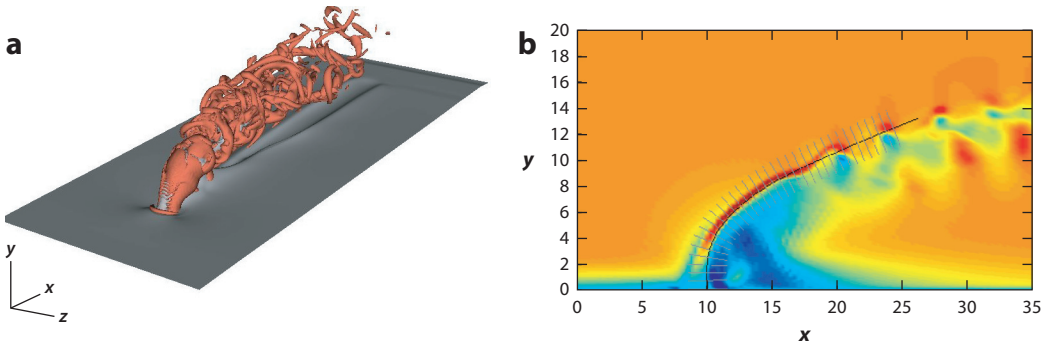

Figure 12

Streamline-based dynamic mode decomposition analysis of a jet in cross-flow. (*a*) Representative snapshot, visualized by isocontours of the Q-criterion. (*b*) Layout of the spatial slices normal to the base-flow streamline. The 3D velocity fields are projected into in-plane and plane-normal components and incorporated into the data matrix.

flu, measles, polio, etc.), in medical imaging and signal analysis [low-resolution MRI (magnetic resonance imaging), neurodegenerative patterns in Parkinson's disease, mass cytometry, epileptic EEG (electroencephalogram) classification, decoding of electrocorticographic signals, etc.], in climate science and oceanography (sea ice patterns and prediction, El Niño–Southern Oscillation analysis and prediction, flow patterns through the Strait of Gibraltar, etc.), in robotics (external perturbation estimation, etc.), and in financial engineering (trading strategies, cycle analysis of markets, etc.) account for a cross section of disciplines where a quantitative description of salient dynamic features by DMD has had a distinct impact.

5. CONCLUSIONS

The past 15 years have seen a remarkable renaissance of an established idea of treating nonlinear dynamical systems by linear techniques: Koopman analysis. Combined with computational methods, as described in this review, this alternative to classical system theory provides new perspectives and many opportunities for the analysis, control, and prediction of fluid flows. It is based on a linearization step that is not accomplished by expansion about equilibrium or quasi-equilibrium states, but rather by an embedding of the nonlinear dynamics in an infinite observable space. With proper embedding, the dynamics in this observable space appears linear, governed by the Koopman operator. The eigenfunctions of this operator, mapping the observables over a specified time interval, play a crucial role and represent modal, monochromatic coherent structures. An approximation of these modes can be determined efficiently by the DMD, directly from data sequences and without appeal to an underlying model or governing equations.

This review outlined the theoretical background, the core algorithm, and extensions/generalizations that ensure the accuracy, robustness, and efficiency of the computational algorithms that determine the proper embedding, the linear observable dynamics, and the desired spectral properties. Variants that allow for the analysis of driven systems, sparse measurements, or sub-Nyquist sampling have also been described. The advantage of a data-driven approach—in an era where data are available at an unprecedented scale, both in quality and quantity—can hardly be overestimated.

While there certainly remain challenges of a theoretical, algorithmic, and practical nature, it appears safe to foresee a continued impact of these techniques on application areas where model-based analysis is difficult, or even prohibitive, and where observations suggest the presence

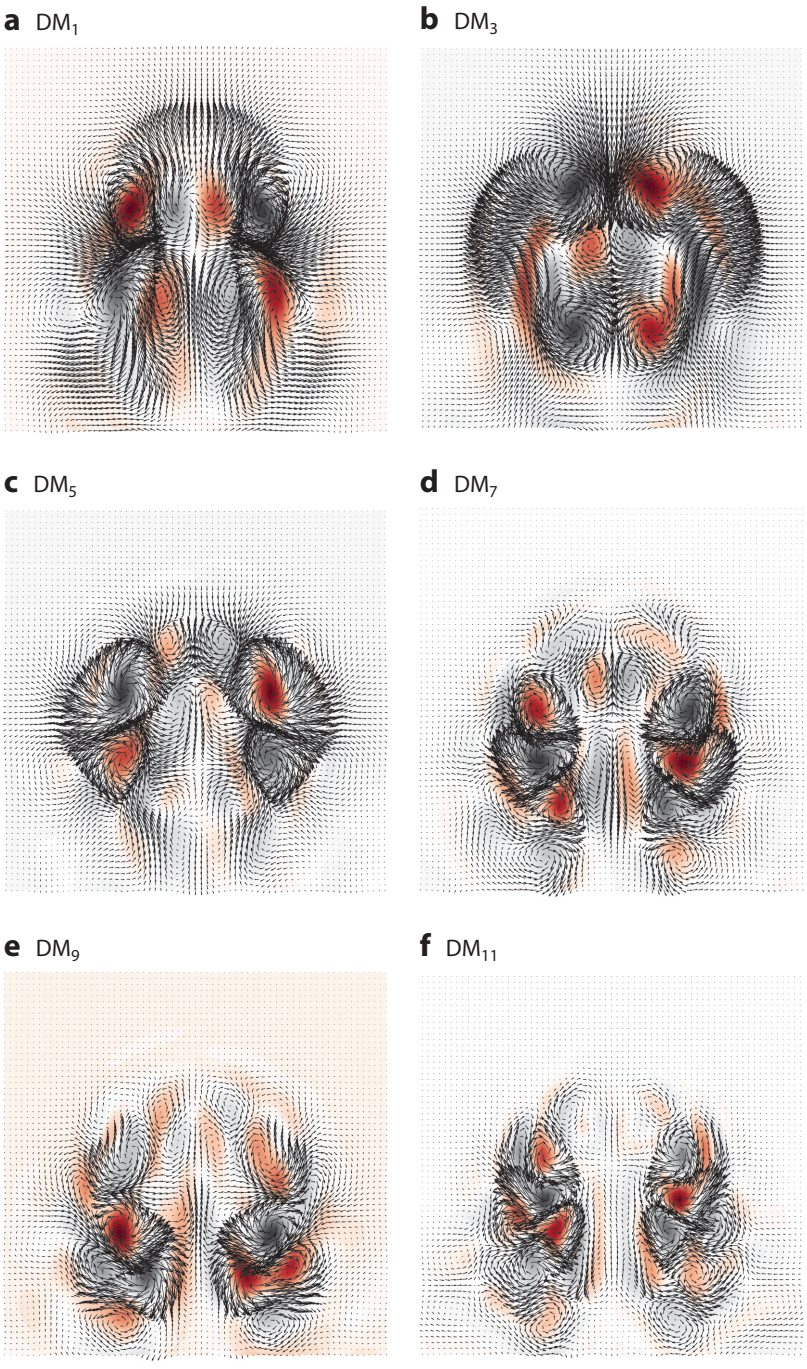

Figure 13

Dominant dynamic modes (DMs) for a jet in cross-flow. A spatial analysis along the base-flow streamline has been performed, and the modes are visualized by velocity vectors (*black*) and vorticity contours (*red-gray*) in a plane normal to the base-flow streamline emanating from the jet exit.

of emerging spatiotemporal patterns and subprocesses. New developments of this data-centric methodology and new results by applying it to real-life problems are greatly anticipated.

SUMMARY POINTS

1. Koopman theory provides an alternative and attractive perspective to analyzing nonlinear dynamical systems within a linear framework. Dynamic mode decomposition (DMD) provides approximate spectral information about the Koopman operator.
2. Flow analysis by DMD is inherently data driven and equation-free, a key feature that adds flexibility and applicability to a wide range of data sets.
3. The core algorithm is straightforward, relying on standard linear algebra concepts and modern optimization techniques.
4. Many extensions and generalizations are available, addressing issues of convergence toward the Koopman limit, robustness to noise and uncertainty in the data, and efficiency of the computational steps. Advances in peripheral research fields (such as optimization, control theory, system identification, harmonic analysis, statistical learning, and L_1-regression) can be readily incorporated into DMD.
5. Data can be processed in creative ways: Composite observables, spatially evolving data sets, curved evolution directions, and nonuniformly or sub-Nyquist sampled data allow for the quantitative analysis of many complex fluid (and nonfluid) systems.

FUTURE ISSUES

1. Machine learning, a very active field of research, will continue to influence the Koopman and DMD framework, even beyond the use of autoencoders. New developments in representing and learning embeddings will have a direct impact on Koopman and DMD analysis, and constraints based on known physical properties (symmetries, conservation laws, etc.) will play a major role in this effort.
2. Delay embedding will be a key methodology for predicting complex dynamical behavior beyond the horizon of the training data set. This technology, paired with, for example, optimal prediction based on a Mori–Zwanzig representation (Chorin et al. 2000), could have important consequences for the forecasting precision for many nonlinear evolution processes.
3. Sparsity concepts will play an increasing role in aiding the decomposition toward finding a minimal representation of the overall data sequence in terms of principal subprocesses.
4. DMD and its variants may find their way into many other algorithms. The convergence behavior of iterative algorithms (such as, for example, stochastic gradient descent) can be interpreted as a discrete dynamical system, whose dynamic modes could give valuable information about the slow and fast dynamics of residual fields.
5. Many more application fields, beyond the ones mentioned here, may benefit from the sort of quantitative analysis expounded in this article. This will be particularly valuable for application fields where first-principle governing equations are difficult or impossible to come by.

6. Many generalizations and improvements have been developed as single features. Synthesizing and combining these specific extensions into a comprehensive framework would provide a platform for the data-centric analysis of a great many dynamical systems.

DISCLOSURE STATEMENT

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

ACKNOWLEDGMENTS

I am grateful for many discussions and exchanges with key scientists in the field of Koopman analysis, decomposition techniques, and flow analysis over the years. They all have contributed to a richer, more complete understanding of this exciting technique and its potential. Partial support from the US AFOSR (Air Force Office of Scientific Research)/EOARD (European Office of Aerospace Research and Development) grant FA9550-18-1-0127 (program managers: Drs. D. Smith, I. Leyva, and S. Popkin) and the EU's Marie Skłodowska-Curie Innovative Training Programme grant 675008 is gratefully acknowledged.

LITERATURE CITED

Adrian RJ, Moin P. 1988. Stochastic estimation of organized turbulent structure: homogeneous shear flow. *J. Fluid Mech.* 190:531–59

Alfredsson PH, Johansson AV. 1984. On the detection of turbulence-generating events. *J. Fluid Mech.* 139:325–345

Arbabi H, Mezić I. 2017. Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator. *SIAM J. Appl. Dyn. Sys.* 16:2096–126

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

Bagheri S. 2014. Effects of weak noise on oscillating flows: linking quality factor, Floquet modes, and Koopman spectrum. *Phys. Fluids* 26:094104

Berkooz G, Holmes P, Lumley JL. 1993. The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Rev. Fluid Mech.* 25:539–75

Bollt EM, Santitissadeekorn N. 2013. *Applied and Computational Measurable Dynamics.* Philadelphia: SIAM

Boyd S, Vandenberghe L. 2004. *Convex Optimization*. Cambridge, UK: Cambridge Univ. Press

Brand M. 2002. Incremental singular value decomposition of uncertain data with missing values. In *Proceedings of the 7th European Conference on Computer Vision*, ed. A Heyden, G Sparr, M Nielsen, P Johansen, pp. 707–20. Berlin: Springer-Verlag

Brunton SL, Brunton BW, Proctor JL, Kaiser E, Kutz JN. 2017. Chaos as an intermittently forced linear system. *Nat. Commun.* 8:19

Brunton SL, Brunton BW, Proctor JL, Kutz JN. 2016. Koopman observable sub-spaces and finite linear representations of nonlinear dynamical systems for control. *PLOS ONE* 11:e0150171

Brunton SL, Proctor JL, Tu JH, Kutz JN. 2015. Compressed sensing and dynamic mode decomposition. *J. Comput. Dyn.* 2:165–91

Budišić M, Mohr R, Mezić I. 2012. Applied Koopmanism. *Chaos* 22:047510

Candès EJ, Li X, Ma Y, Wright J. 2011. Robust principal component analysis? *J. ACM* 58:11

Candès EJ, Romberg J, Tao T. 2006. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* 52:489–509

Carleman T. 1932. Application de la théorie des équations intégrales linéaires aux systèmes d'équations différentielles nonlinéaires. *Acta Math.* 59:63–87

Chorin AJ, Hald OH, Kupferman R. 2000. Optimal prediction and the Mori–Zwanzig representation of irreversible processes. *PNAS* 97:2968–73

Dawson S, 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

Demmel J, Grigori L, Hoemmen M, Langou J. 2012. Communication-optimal parallel and sequential QR and LU factorizations. *SIAM J. Sci. Comput.* 34:206–39

Drmač Z, Mezić I, Mohr R. 2017. *Data driven model decompositions: analysis and enhancements*. Tech. Rep. 201708, AIMDyn Inc., Santa Barbara, CA

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

Erichson NB, Mathelin L, Kutz JN, Brunton SL. 2019. Randomized dynamic mode decomposition. *SIAM J. Appl. Dyn. Sys.* 18:1867–91

Froyland G, Junge O, Koltai P. 2013. Estimating long-term behavior of flows without trajectory integration: the infinitesimal generator approach. *SIAM J. Numer. Anal.* 51:223–47

Froyland G, Padberg K. 2009. Almost-invariant sets and invariant manifolds—connecting probabilistic and geometric descriptions of coherent structures in flow. *Physica D* 238:1507–23

Golyandina N, Nekrutin V, Zhigljavsky A. 2001. *Analysis of Time Series Structure: SSA and Related Techniques*. Boca Raton, FL: Chapman & Hall/CRC

Gopalakrishnan Meena M, Nair AG, Taira K. 2018. Network community-based model reduction for vortical flows. *Phys. Rev. E* 97:063103

Halko N, Martinsson PG, Tropp JA. 2011. Finding structure in randomness: probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* 53:217–88

Haller G. 2015. Lagrangian coherent structures. *Annu. Rev. Fluid Mech.* 47:137–62

Hemati MS, Rowley CW, Deem EA, Cattafesta LN. 2017. De-biasing the dynamic mode decomposition for applied Koopman spectral analysis of noisy datasets. *Theor. Comput. Fluid Dyn.* 31:349–68

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

Holmes P, Lumley J, Berkooz G, Rowley CW. 2012. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge, UK: Cambridge Univ. Press

Hyvärinen A. 2013. Independent component analysis: recent advances. *Philos. Trans. R. Soc. A* 371:20110534

Jovanović MR, Schmid PJ, Nichols JW. 2014. Sparsity-promoting dynamic mode decomposition. *Phys. Fluids* 26:024103

Junge O, Koltai P. 2009. Discretization of the Frobenius–Perron operator using a sparse Haar tensor basis: the sparse Ulam method. *SIAM J. Num. Anal.* 47:3464–85

Kaiser E, Noack B, Cordier L, Spohn A, Segond M, et al. 2014. Cluster-based reduced-order modelling of a mixing layer. *J. Fluid Mech.* 754:365–414

Kamb M, Kaiser E, Brunton SL, Kutz JN. 2020. Time-delay observables for Koopman: theory and applications. *SIAM J. Appl. Dyn. Sys.* 19:886–917

Klus S, Koltai P, Schütte C. 2016. On the numerical approximation of the Perron-Frobenius and Koopman operator. *J. Comput. Dyn.* 3:51–79

Klus S, Nüske F, Koltai P, Wu H, Kevrekidis IG, Schütte C, Noé F. 2018. Data-driven model reduction and transfer operator approximation. *J. Nonlinear Sci.* 28:985–1010

Klus S, Nüske F, Peitz S, Niemann JH, Clementi C, Schütte C. 2020. Data-driven approximation of the Koopman generator: model reduction, system identification and control. *Physica D* 406:132416

Koopman BO. 1931. Hamiltonian systems and transformations in Hilbert space. *PNAS* 17:315–18

Korda M, Mezić I. 2018. On convergence of extended dynamic mode decomposition to the Koopman operator. *J. Nonlinear Sci.* 28:687–710

Kutz JN, Brunton SL, Brunton BW, Proctor JL. 2016a. *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*. Philadelphia: SIAM

Kutz JN, Fu X, Brunton SL. 2016b. Multiresolution dynamic mode decomposition. *SIAM J. Appl. Dyn. Sys.* 15:713–35

Lasota A, Mackey CM. 1994. *Chaos, Fractals and Noise: Stochastic Aspects of Dynamics*. New York: Springer-Verlag

LeClainche S, Vega JM. 2017. Higher order dynamic mode decomposition. *SIAM J. Appl. Dyn. Sys.* 16:882–925
Li Q, Dietrich F, Bollt EM, Kevrekidis IG. 2017. Extended dynamic mode decomposition with dictionary learning: a data-driven adaptive spectral decomposition of the Koopman operator. *Chaos* 27:103111
Lumley JL. 1971. *Stochastic Tools in Turbulence*. New York: Academic
Lusch B, Kutz JN, Brunton SL. 2018. Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.* 9:4950
Mezić I. 2005. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dyn.* 41:309–25
Mezić I. 2013. Analysis of fluid flows via spectral properties of the Koopman operator. *Annu. Rev. Fluid Mech.* 45:357–78
Noack BR, Stankiewicz W, Morzyński M, Schmid PJ. 2016. Recursive dynamic mode decomposition of transient and post-transient wake flows. *J. Fluid Mech.* 809:843–72
Otto SE, Rowley CW. 2019. Linearly recurrent autoencoder networks for learning dynamics. *SIAM J. Appl. Dyn. Sys.* 18:558–93
Pan S, Duraisamy K. 2020a. On the structure of time-delay embedding in linear models of non-linear dynamical systems. *Chaos* 30:073135
Pan S, Duraisamy K. 2020b. Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability. *SIAM J. Appl. Dyn. Sys.* 19:480–509
Perry AE, Chong MS. 1987. A description of eddying motions and flow patterns using critical-point concepts. *Annu. Rev. Fluid Mech.* 19:125–55
Proctor JL, Brunton SL, Kutz JN. 2016. Dynamic mode decomposition with control. *SIAM J. Appl. Dyn. Sys.* 15:142–61
Rowley CW, Mezić I, Bagheri S, Schlatter P, Henningson DS. 2009. Spectral analysis of nonlinear flows. *J. Fluid Mech.* 641:115–27
Sayadi T, Schmid PJ. 2016. Parallel data-driven decomposition algorithm for large-scale data sets: with application to transitional boundary layers. *Theor. Comput. Fluid Dyn.* 30:415–28
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
Scherl I, Strom B, Shang JK, Williams O, Polagye BL, Brunton SL. 2020. Robust principal component analysis for modal decomposition of corrupt fluid flows. *Phys. Rev. Fluids* 5:054401
Schmid PJ. 2010. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* 656:5–28
Schmid PJ. 2021. Data-driven and operator-based tools for the analysis of turbulent flows. In *Advanced Approaches in Turbulence*, ed. P Durbin, pp. 243–306. Amsterdam: Elsevier
Schmid PJ, Li L, Juniper MP, Pust O. 2011. Applications of the dynamic mode decomposition. *Theor. Comput. Fluid Dyn.* 25:249–59
Schmid PJ, Schmidt OT, Towne A, Hack MJP. 2018. Analysis and prediction of rare events in turbulent flows. In *Proceedings of the Summer Program 2018*, pp. 139–48. Stanford, CA: Cent. Turbul. Res.
Schmid PJ, Sesterhenn JL. 2008. *Dynamic mode decomposition of numerical and experimental data*. Paper presented at 61st Annual Meeting of the APS Division of Fluid Dynamics, Nov. 25, San Antonio, TX
Sesterhenn JL, Shahirpour A. 2019. A characteristic dynamic mode decomposition. *Theor. Comput. Fluid Dyn.* 33:281–305
Sirovich L. 1987. Turbulence and the dynamics of coherent structures. *Q. J. Appl. Math.* 45:561–90
Takens F. 1981. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence: Proceedings of a Symposium Held at the University of Warwick 1979/80*, ed. D Rand, L-S Young, pp. 366–81. Berlin: Springer-Verlag
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, Luchtenberg DM, Brunton SL, Kutz JN. 2014b. On dynamic mode decomposition: theory and applications. *J. Comput. Dyn.* 1:391–421
Ulam SM. 1960. *A Collection of Mathematical Problems*. New York: InterScience
Van Overschee P, De Moor BL. 2012. *Subspace Identification for Linear Systems: Theory—Implementation—Applications*. Boston: Kluwer Academic

Wallace JM. 2016. Quadrant analysis in turbulence research: history and evolution. *Annu. Rev. Fluid Mech.* 48:131–58

Williams MO, Kevrekidis IG, Rowley CW. 2015. 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 approach to data-driven Koopman spectral analysis. *J. Comput. Dyn.* 2:247–65

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