Physics of Fluids

RESEARCH ARTICLE | JUNE 10 2022

Dynamic mode decomposition with core sketch FREE

Shady E. Ahmed ; Pedram H. Dabaghian; Omer San ; Diana A. Bistrian ; Ionel M. Navon

Check for updates

Physics of Fluids 34, 066603 (2022)
https://doi.org/10.1063/5.0095163
CHORUS

CrossMark
View Online
Export Citation

Articles You May Be Interested In

Double layer secure sketch

AIP Conference Proceedings (September 2012)

Fashion sketch design by interactive genetic algorithms

AIP Conference Proceedings (November 2012)

Preliminary research on prospective teacher's ability in understanding the physics problems by sketching

AIP Conference Proceedings (November 2019)

31 August 2023 11:05:41

AIP Publishing

Physics of Fluids ARTICLE scitation.org/journal/phf

Dynamic mode decomposition with core sketch

Cite as: Phys. Fluids **34**, 066603 (2022); doi: 10.1063/5.0095163
Submitted: 8 April 2022 · Accepted: 31 May 2022 ·
Published Online: 10 June 2022

View Online Export Citation CrossMark

Shady E. Ahmed,[1] Pedram H. Dabaghian,[1] Omer San,[1,a)] Diana A. Bistrian,[2] and Ionel M. Navon[3]

AFFILIATIONS

[1]School of Mechanical & Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078, USA
[2]Department of Electrical Engineering and Industrial Informatics, Politehnica University of Timisoara, Hunedoara, Romania
[3]Department of Scientific Computing, Florida State University, Tallahassee, Florida 32306, USA

[a)]Author to whom correspondence should be addressed: osan@okstate.edu

ABSTRACT

With the increase in collected data volumes, either from experimental measurements or high fidelity simulations, there is an ever-growing need to develop computationally efficient tools to process, analyze, and interpret these datasets. Modal analysis techniques have gained great interest due to their ability to identify patterns in the data and extract valuable information about the system being considered. Dynamic mode decomposition (DMD) relies on elements of the Koopman approximation theory to compute a set of modes, each associated with a fixed oscillation frequency and a decay/growth rate. Extracting these details from large datasets can be computationally expensive due to the need to implement singular value decomposition of the input data matrix. Sketching algorithms have become popular in numerical linear algebra where statistical theoretic approaches are utilized to reduce the cost of major operations. A sketch of a matrix is another matrix, which is significantly smaller, but still sufficiently approximates the original system. We put forth an efficient DMD framework, SketchyDMD, based on a core sketching algorithm that captures information about the range and corange (their mutual relationship) of input data. The proposed sketching-based framework can accelerate various portions of the DMD routines, compared to classical methods that operate directly on the raw input data. We conduct numerical experiments using the spherical shallow water equations as a prototypical model in the context of geophysical flows. We show that the proposed SketchyDMD is superior to existing randomized DMD methods that are based on capturing only the range of the input data.

Published under an exclusive license by AIP Publishing. https://doi.org/10.1063/5.0095163

31 August 2023 11:05:41

I. INTRODUCTION

The volumes of datasets that are being available at our disposal are expanding exponentially. Advances in experimental setups and sensing technologies yield denser field measurements than ever before. In addition, sensors are becoming cheaper, more accurate, and more accessible to the point that crowd-sourcing can continuously provide feeds of data at multiple scales. Moreover, modern computing facilities and numerical algorithms have made large-to-extreme scale computations feasible. Recent exascale computing efforts resolve a wide range of spatial and temporal scales and lead to enormous amounts of datasets. On the one hand, the scientific community is fortunate with such luxury of big data to enhance our understanding of the underlying physical phenomena. On the other hand, processing these datasets and extracting meaningful information from them are major challenges. The questions of "what to look at" and "what to look for" are not often easy to answer. Therefore, there is an utmost need for computationally efficient tools that facilitate the analysis of the collected data.

It is generally accepted that most high-dimensional data lie considerably lower dimensional patterns that constitute the solution manifold.[1] Modal analysis techniques have been developed to identify the spatiotemporal structures, which dominate the evolution of the system.[2–4] This introduces the concept of reduced order models (ROMs) where the entire system can be well represented by relatively compact surrogates, opening ways to analyze the prominent dynamics.[5–7] Proper orthogonal decomposition (POD), also known as principal component analysis (PCA), has been a major driving force for ROM developments in the last few decades. More recently, dynamic mode decomposition (DMD) has gained special attention with roots from the Koopman theory. A rigorous description of the Koopman operator and its properties can be found in Refs. 8–14. The Koopman approximation relies on a transformation that lifts the dynamics from state space where dynamics might be nonlinear to the observable space where the dynamics are linear but infinite-dimensional (see Fig. 1). It is worth noting that extensive research efforts have been dedicated to define the state-to-observable mappings so as to better approximate the Koopman operator and optimize its linear embeddings. Such transformation enables the implementation of dynamical system theoretical analyses, which augments our understanding of the system and our ability to predict its behavior.

Published under an exclusive license by AIP Publishing

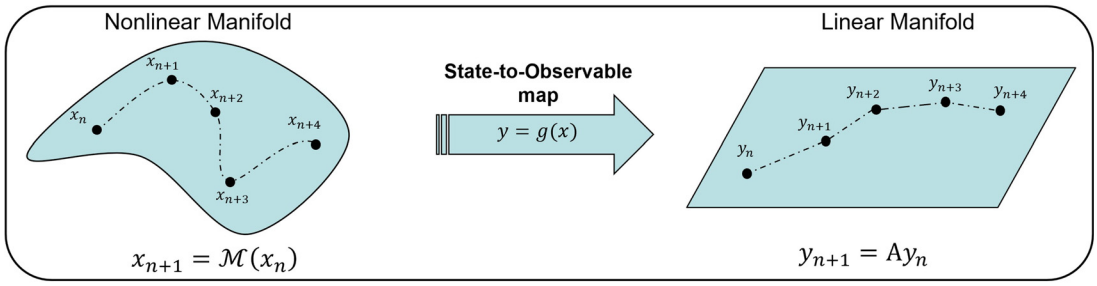

FIG. 1. The Koopman viewpoint: while the original system with state x evolves according to $\mathcal{M}$ on a nonlinear manifold (left), the selection of appropriate observables $y = g(x)$ provides a substitute system where the dynamics can be approximated using a linear flow map (right).

DMD can be viewed as a data-driven approximation of the Koopman operator spectrum, and it can be applied to both experimental and numerical data with success. In DMD, each mode is associated with a unique growth/decay rate and oscillation frequency. Therefore, DMD not only reveals information about the system's spatial characteristics but also delivers knowledge about the system's temporal evolution.[15,16] DMD was introduced to the fluid dynamics community more than a decade ago[17,18] and has been fostered in many other disciplines as well.[19–21] For instance, DMD has been used to predict the behavior of physiological and ecological systems[22] and applied in epidemiology studies,[23] including the modeling of COVID-19.[24,25] Also, DMD has found its place in the world of images[26–28] and medical diagnostics.[29] In addition, research has been conducted on the application of DMD in combination with other existing methods such as data assimilation.[30]

A. Computational aspects

Standard off-the-shelf DMD implementation can be fragile for large scale systems. The algorithm proposed by Rowley *et al.*[31] involves the computation of a companion matrix that helps to construct, in a least squares sense, the final data vector as a linear combination of all previous data vectors. However, Schmid[17] showed that this version may be ill-conditioned in practice, and an alternative algorithm based on the singular value decomposition (SVD) of the snapshot data matrix is recommended. With increasing popularity, various strategies have been developed to overcome many of the early DMD shortcomings.[16,32] For example, Williams *et al.*[33] introduced an extended dynamic mode decomposition approach to better approximate the Koopaman operator tuples (i.e., eigenvalues, eigenfunctions, and modes). We also refer readers to Refs. 34–39 for the integration of machine learning approaches in extracting the Koopman operator. In addition, many studies employed pre-processing and post-processing techniques to improve the DMD performance using ensemble-averaging methods,[40] mean subtraction,[41] de-biasing algorithms,[42] sparsity inducing approaches,[43] online update techniques,[44,45] improved least squares methods,[46,47] and the Hankel DMD algorithm.[10] Furthermore, efforts have been made to make the DMD algorithm robust against noisy data[17,48] and incomplete measurements Katrutsa *et al.*[49]

The majority of DMD developments rely on the SVD-based DMD implementation, which becomes computationally prohibitive for high dimensional systems. Maryada and Norris[50] proposed a parallel DMD framework with reduced communications using a combination of horizontal-slicing and QR factorization to compress the original data matrix. Alternatively, sketching methods can be used to alleviate the computational burden of classical numerical linear algebra algorithms. These methods rely upon informed projections to construct a smaller (memory-wise) matrix, called the "sketch" while preserving important properties of the original data.[51,52] The sketch may represent any combination of the row space, the column space, or the space generated by the intersection of rows and columns (core space). Sketching algorithms are increasingly being embedded in computational pipelines to ensure their scalability. Moreover, sketching-based tools are streaming-compatible where individual data snapshots can be accessed only once, which is an advantage for processing large datasets. There have been several successful studies that aimed to utilize random projections to derive low-rank randomized DMD.[53–57] These developments have focused mainly on capturing the range (column-space) of the original data matrix.

In this study, we build a new sketching-based DMD framework, denoted as SketchyDMD, based on capturing both the range and co-range of the input data. The original algorithm for defining the elementary sketch matrices was proposed by Tropp *et al.*[58] The definition of a core sketch, which relates the column- and row-spaces, is what makes this algorithm particularly interesting. It provides fresh information about the system's singular values, eventually leading to improved approximation of the DMD spectrum. Moreover, the SketchyDMD algorithm can be implemented using a single pass over the input data matrix. This makes the proposed framework more favorable in applications with strict memory limitations and/or streaming datasets. In contrast, previous rDMD methods require at least two distinct passes over the input data to generate the low-rank approximation. Moreover, SketchyDMD includes free parameters that can be tuned based on the given memory constraints.

The rest of the paper is structured as follows. In Sec. II, we briefly describe the spherical shallow water equation (SWE) system that we adopt for numerical experiments. SWEs are a relatively simple, but very representative model for geophysical flows. The SWEs formulation is often considered a first step in developing general circulation models in large scales. We then detail the DMD formulations and the proposed sketching approach in Secs. III and IV, respectively. Results are provided with corresponding discussions in Sec. V, while concluding remarks are drawn in Sec. VI.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

II. SPHERICAL SHALLOW WATER EQS. SYSTEM

The shallow water equations (SWEs) describe the flow field of a free fluid surface where the horizontal length scales are significantly larger than the vertical length scale. This assumption is valid for many geophysical systems, including the oceanic and atmospheric flows.[59,60] Mathematically, SWEs are obtained by the integrated average of the Navier–Stokes equations (NSEs) along the vertical direction and using the hydrostatic approximation that relates the pressure to the depth of the fluid column. In addition, the Coriolis term is included to account for the forces due to Earth's rotation. For the atmospheric flow on Earth, the SWEs in spherical coordinates can be written as follows:[61]

$$\frac{\partial h}{\partial t}+\frac{1}{\rho\cos\theta}\frac{\partial hu_\phi}{\partial\phi}+\frac{1}{\rho}\frac{\partial hu_\theta}{\partial\theta}=\frac{hu_\theta}{\zeta}\tan\theta, \tag{1}$$

$$\frac{\partial hu_\phi}{\partial t}+\frac{1}{\rho\cos\theta}\frac{\partial}{\partial\phi}\left(hu_\phi^2+\frac{1}{2}gh^2\right)+\frac{1}{\rho}\frac{\partial(hu_\phi u_\theta)}{\partial\theta}=Fhu_\theta-\frac{gh}{\zeta\cos\theta}\frac{\partial H}{\partial\phi}+\frac{hu_\phi u_\theta}{\zeta}\tan\theta, \tag{2}$$

$$\frac{\partial hu_\theta}{\partial t}+\frac{1}{\rho\cos\theta}\frac{\partial(hu_\phi u_\theta)}{\partial\phi}+\frac{1}{\rho}\frac{\partial}{\partial\theta}\left(hu_\theta^2+\frac{1}{2}gh^2\right)=-Fhu_\phi-\frac{gh}{\zeta}\frac{\partial H}{\partial\theta}+\frac{hu_\phi^2}{\zeta}\tan\theta, \tag{3}$$

where $\zeta=\rho+H$ with ρ being the Earth's radius ($\rho=6.4\times10^6$ m) and H being the height of the bottom topography. The depth of the water surface is denoted by h, and the gravitational acceleration is g ($g=9.8$ m/s^2). The velocity components in the longitudinal and latitudinal directions are defined by u_θ and u_ϕ, respectively, where θ represents the latitudes (North/South) and ϕ denotes the longitudes (East/West). $F=2f\sin\theta$ is the Coriolis parameter with $f=\frac{2\pi}{24\times3600}$ rad/s representing Earth's rotation rate. We use the Lax–Wendroff method for solving the SWEs in order to generate the snapshots utilized in our DMD analysis. Further details of the problem setting and the numerical approaches can be found in Ref. 61.

III. DYNAMIC MODE DECOMPOSITION

Dynamic mode decomposition (DMD) is a completely data-driven reduced order modeling technique that does not require prior knowledge of the system. Let $\mathbf{x}\in\mathbb{R}^n$ (where $n\gg1$) be the state of a system that evolves in time through some specific dynamics as $\frac{d\mathbf{x}}{dt}=f(\mathbf{x},t)$, where $f(\cdot)$ can be a nonlinear function. The objective of DMD is to identify the leading DMD eigenvalues and the corresponding eigenvectors of the best fit operator $\mathbf{A}$, which would evolve the system linearly in time as

$$\frac{d\mathbf{x}}{dt}\approx\mathbf{A}\mathbf{x}. \tag{4}$$

A. DMD notion

By estimating the operator $\mathbf{A}$, spatially coherent DMD modes can be computed, each of which is associated with a growth or decay rate and an oscillation frequency that collectively define its time dynamics. Dimensionality reduction is obtained by choosing a few significant modes while rejecting the others. In problems of fluid dynamics, the system dimension n can be very large so as to account for the relevant scales in the flow field, and thus, frameworks such as DMD are of high interest. In practice, one would have a set of realizations of the system's state $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_m]\in\mathbb{R}^{n\times m}$ collected at various time instants $t_k=[t_1,t_2,\ldots,t_m]\in\mathbb{R}^m$, where n is the number of degrees of freedom and m is the number of discrete time steps at which data are available (i.e., number of collected snapshots). These data can be either actual field data from sensor measurements or even synthetic data generated from high fidelity numerical simulations. In the present study, data are obtained from the solution of the SWEs, which constitute the full order model (FOM) of the system of interest. For two-dimensional (2D) field data, each matrix of dimension $N_\phi\times N_\theta$, corresponding to a particular time t_k, is rearranged in a column vector $\mathbf{x}^{(k)}\in\mathbb{R}^{N_\phi\cdot N_\theta}$. Thus, the full data matrix $\mathbf{x}\in\mathbb{R}^{n\times m}$ is formed as

$$\mathbf{X}=\begin{bmatrix}| & | & & |\\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m\\ | & | & & |\end{bmatrix}. \tag{5}$$

The primary objective of DMD is to extract the eigenvectors and eigenvalues of the matrix $\hat{\mathbf{A}}$, such that

$$\mathbf{x}_{k+1}\approx\hat{\mathbf{A}}\mathbf{x}_k, \tag{6}$$

where Eq. (6) is the time discretized form of Eq. (4). The full dataset is split into two matrices $\mathbf{X}_1\in\mathbb{R}^{n\times(m-1)}$ and $\mathbf{X}_2\in\mathbb{R}^{n\times(m-1)}$ as defined as follows:

$$\mathbf{X}_1=\begin{bmatrix}| & | & & |\\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{m-1}\\ | & | & & |\end{bmatrix},\quad \mathbf{X}_2=\begin{bmatrix}| & | & & |\\ \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_m\\ | & | & & |\end{bmatrix}. \tag{7}$$

Thus, Eq. (6) can be rewritten in matrix format as follows:

$$\mathbf{X}_2\approx\hat{\mathbf{A}}\mathbf{X}_1, \tag{8}$$

and the operator $\hat{\mathbf{A}}$ can be computed through a least squares minimization as follows:

$$\hat{\mathbf{A}}=\underset{\hat{\mathbf{A}}}{\operatorname{argmin}}\,||\mathbf{X}_2-\hat{\mathbf{A}}\mathbf{X}_1||_F, \tag{9}$$

where $||\cdot||_F$ is the Frobenius norm. Solving for $\hat{\mathbf{A}}\in\mathbb{R}^{n\times n}$ would require high computational power when n is very large (as is the case in practical fluid dynamics problems). So, the motivation here is to seek ways to replace it with lower rank approximations. This will be described in detail in Sec. III B.

Once the DMD modes are computed, we can form a matrix $\mathbf{\Psi}=\{\boldsymbol{\psi}_i\}_{i=1}^m\in\mathbb{C}^{n\times m}$, where each column $\boldsymbol{\psi}_i$ represents a DMD mode. The next step is to truncate the DMD mode matrix $\mathbf{\Psi}$ to $\mathbf{\Psi}_\mathbf{r}=\{\boldsymbol{\psi}_i\}_{i=1}^r\in\mathbb{C}^{n\times r}$, where r is the number of retained modes. The information of the time dynamics in each DMD basis is inferred from the corresponding eigenvalues $\mathbf{\Lambda}=\text{diag}\{\lambda_i\}_{i=1}^r\in\mathbb{C}^{r\times r}$. Here, λ_i denotes the discrete-time eigenvalues, while the continuous-time eigenvalues can be evaluated as follows:

$$\alpha_i=\frac{\ln\lambda_i}{\Delta T}, \tag{10}$$

where ΔT is the time interval between two consecutive snapshots and $\boldsymbol{\alpha}=\{\alpha_i\}_{i=1}^r\in\mathbb{C}^r$ constitutes the vector of the continuous-time eigenvalues. Next, we reconstruct the high dimensional dynamics from the lower order subspace as

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

$$\mathbf{x}_k^{ROM} = \sum_{i=1}^{r} \boldsymbol{\psi}_i \lambda_i^{k-1} b_i = \boldsymbol{\Psi}_\mathbf{r} \boldsymbol{\Lambda}^{k-1} \mathbf{b}, \tag{11}$$

where $\mathbf{b} = \{b_i\}_{i=1}^{r} \in \mathbb{C}^r$ is the vector of initial amplitudes of the DMD modes, given as

$$\mathbf{b} = \boldsymbol{\Psi}_\mathbf{r}^{\dagger} \mathbf{x}_1, \tag{12}$$

where $\boldsymbol{\Psi}_r^{\dagger}$ is the Moore–Penrose pseudoinverse of $\boldsymbol{\Psi}_\mathbf{r}$. Finally, for field reconstruction, Eq. (11) can be written in terms of the continuous-time eigenvalues by using the relation in Eq. (10) as follows:

$$\mathbf{x}_k^{ROM} = \sum_{i=1}^{r} \boldsymbol{\psi}_i e^{\alpha_i t_k} b_i = \boldsymbol{\Psi}_\mathbf{r} \mathrm{diag}[e^{\boldsymbol{\alpha} t_k}] \mathbf{b}. \tag{13}$$

In Eq. (13), the real part of the continuous-time eigenvalues is responsible for the growth or decay rate of the mode. The mode grows for a positive real part and conversely decays over time for a negative real part. On the contrary, the imaginary part determines the oscillating frequency of the corresponding mode.

B. Deterministic DMD

In the standard form of the DMD, the linear operator $\hat{\mathbf{A}}$ is projected onto a lower R-dimensional subspace to reduce the computational cost of solving the optimization problem in (9). Projection basis functions can be obtained from the singular value decomposition (SVD) of the matrix $\mathbf{X}_1$ as $\mathbf{X}_1 = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^*$ (where $\mathbf{V}^*$ denotes the complex conjugate transpose of $\mathbf{V}$). Compact SVD can be computed such that $\mathbf{X}_1 \approx \mathbf{U}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}\mathbf{V}_\mathbf{R}^*$, where $\mathbf{U}_\mathbf{R} \in \mathbb{R}^{R\times R}$ and $\mathbf{V}_\mathbf{R} \in \mathbb{R}^{R\times R}$ denote the matrix of the first R columns of $\mathbf{U}$ and $\mathbf{V}$, respectively, while $\boldsymbol{\Sigma}_\mathbf{R} \in \mathbb{R}^{R\times R}$ is the top-left $R \times R$ sub-block of $\boldsymbol{\Sigma}$, with R being the rank of $\boldsymbol{\Sigma}$. The projection of $\hat{\mathbf{A}}$ onto the R-dimensional space is taken as

$$\tilde{\mathbf{A}} = \mathbf{U}_\mathbf{R}^* \hat{\mathbf{A}} \mathbf{U}_\mathbf{R}. \tag{14}$$

Thus, the optimization problem in Eq. (9) can be reformulated as follows:

$$\tilde{\mathbf{A}} = \underset{\tilde{\mathbf{A}}}{\mathrm{argmin}} \, ||\mathbf{X}_2 - \mathbf{U}_\mathbf{R}\tilde{\mathbf{A}}\boldsymbol{\Sigma}_\mathbf{R}\mathbf{V}_\mathbf{R}^*||_F, \tag{15}$$

which gives $\tilde{\mathbf{A}} = \mathbf{U}_\mathbf{R}^*\mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1} \in \mathbb{R}^{R\times R}$. The eigenvalues of $\tilde{\mathbf{A}}$ represent the Ritz values (approximate eigenvalues) of $\hat{\mathbf{A}}$. These eigenvalues (also called DMD eigenvalues) are computed as $\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\boldsymbol{\Lambda}$. It should be noted that the eigen decomposition of $\tilde{\mathbf{A}}$ results in complex eigenvalues and eigenvectors. Here, the columns of $\mathbf{W} \in \mathbb{C}^{R\times R}$ are the eigenvectors, while $\boldsymbol{\Lambda} = \mathrm{diag}(\{\lambda_i\}_{i=1}^{R}) \in \mathbb{C}^{R\times R}$ is a diagonal matrix containing the corresponding eigenvalues. The DMD modes can be evaluated as follows:

$$\boldsymbol{\Psi} = \mathbf{U}_\mathbf{R}\mathbf{W}, \tag{16}$$

while the so-called "exact" DMD modes can be written as[62,63]

$$\boldsymbol{\Psi} = \mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}\mathbf{W}. \tag{17}$$

A suitable reduced order approximation $\boldsymbol{\Psi}_\mathbf{r}$ can be obtained by retaining r columns of $\boldsymbol{\Psi}$. From here, the continuous-time eigenvalues can be easily calculated from Eq. (10), and the estimated higher order dynamics from the lower order approximation can be found from either Eq. (11) or Eq. (13). Algorithm 1 shows the standard SVD-based DMD implementation.

ALGORITHM 1. Deterministic dynamic mode decomposition.

1: The matrix $\mathbf{X}$ is split into two matrices $\mathbf{X}_1 = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\}$ and $\mathbf{X}_2 = \{\mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_m\}$.

2: Perform SVD on $\mathbf{X}_1$

$$\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^* = \mathrm{svd}(\mathbf{X}_1)$$

3: Rank truncation [to reduce noise]

$$\mathbf{U}_\mathbf{R} = \mathbf{U}(:, 1:R)$$
$$\mathbf{V}_\mathbf{R} = \mathbf{V}(:, 1:R)$$
$$\boldsymbol{\Sigma}_\mathbf{R} = \boldsymbol{\Sigma}(1:R, 1:R)$$

4: Low-rank dynamics

$$\tilde{\mathbf{A}} = \mathbf{U}_\mathbf{R}^*\mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}$$

5: Eigenvalue decomposition

$$[\mathbf{W}, \boldsymbol{\Lambda}] = \mathrm{eig}(\tilde{\mathbf{A}})$$

6: Compute DMD modes and spectrum

$$\boldsymbol{\Psi} = \mathbf{U}_\mathbf{R}\mathbf{W} \quad \text{or} \quad \boldsymbol{\Psi} = \mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}\mathbf{W}$$
$$\lambda_i = \{\mathrm{diag}(\boldsymbol{\Lambda})\}$$
$$\alpha_i = \ln(\lambda_i)/\Delta T$$

IV. SKETCHY DYNAMIC MODE DECOMPOSITION

As we discussed before, one of the main bottlenecks in the computational pipeline of the stable DMD implementation is the size of the given data matrix. In particular, for high-dimensional systems, the size of the input data matrix $\mathbf{X}$ can be very large and, thus, loading $\mathbf{X}_1$ to perform SVD becomes computationally expensive. Streaming algorithms can be adopted to mitigate this burden by passing the snapshots one-by-one and performing SVD on increments of the data.[44,64] Another approach, which we consider in the present study, is to utilize sketching as a tool to reduce the size of the processed data matrix. In particular, we seek a low-order embedding of the original data matrix represented by a smaller matrix, called the sketch that captures the main information of the original one. The success of the sketching method relies on the assumption that big data matrices are often low-rank,[1] with an exponential decay of the underlying singular values. Indeed, this is the main reason why model order reduction has witnessed substantial success in many applications.

Sketching-based algorithms exploit informed projections to transform the original data matrix to a more compact one. The expensive computations are thus performed onto the latter, after which post-processing takes place to map (and maybe correct) the outputs to the original space. Randomized projections have been especially effective for this purpose. Figure 2 depicts the use of randomized projection for dimensionality reduction by extracting a summary of the data column space. Randomized projections can efficiently and accurately extract the spatiotemporal coherent

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

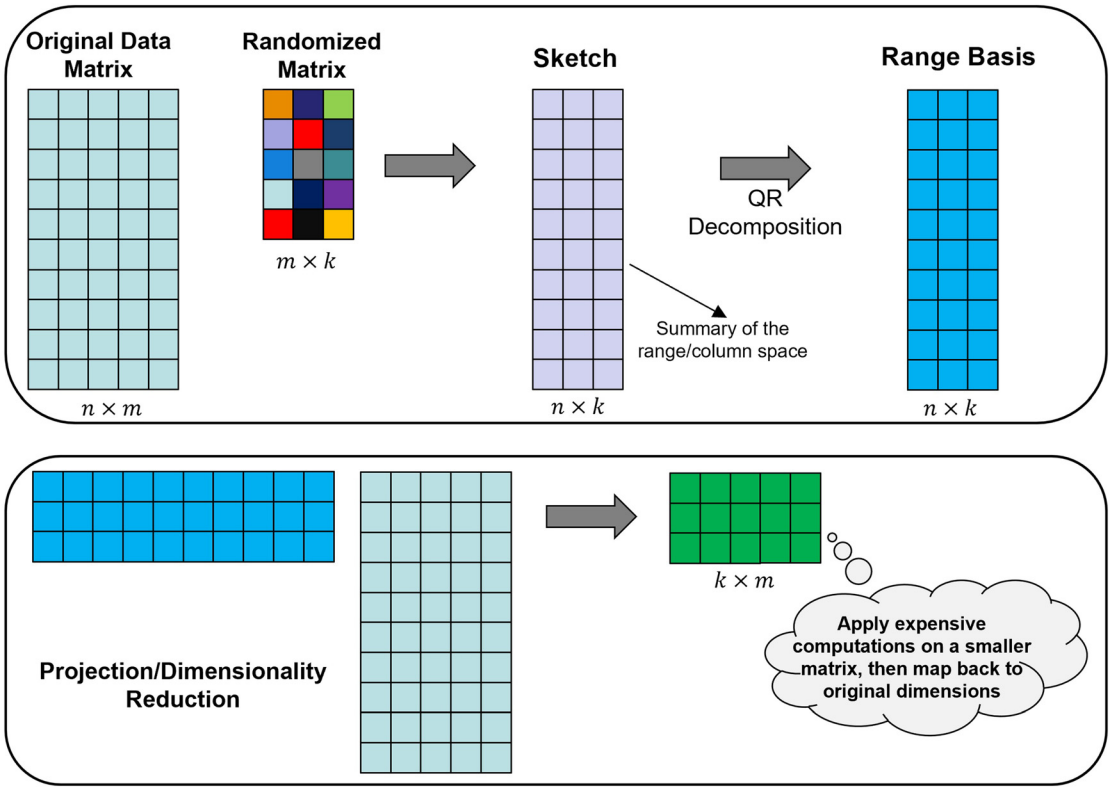

FIG. 2. Demonstration for dimensionality reduction using a range sketch with randomized projections.

structures from high-dimensional data, with high probability.[65] We consider three variants of sketching-based DMD, to reduce the computational costs of different portions of the deterministic DMD algorithm.

A. Sketching the range of $\mathbf{X}_1$

Bistrian and Navon[54] introduced a randomized DMD (rDMD) framework that aims at mitigating the cost incurred by computing the SVD of the data matrix $\mathbf{X}_1 \in \mathbb{R}^{n\times(m-1)}$. In particular, a near-optimal basis with a target rank k is defined using random projections, such that it captures the range of $\mathbf{X}_1$ and provides a smaller sketch matrix $\mathbf{B}_1 \in \mathbb{R}^{k\times m-1}$. In rDMD, an orthonormal basis $\mathbf{Q}$ is estimated from the full order dense data matrix $\mathbf{X}_1 = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(\mathbf{m-1})}\} \in \mathbb{R}^{n\times(m-1)}$, such that $\mathbf{X}_1 \approx \mathbf{Q}\mathbf{Q}^*\mathbf{X}_1$. In order to do this, a random sampling of the input data matrix $\mathbf{X}_1$ is first performed as follows:

$$\mathbf{Y}_1 = \mathbf{X}_1\mathbf{\Omega}_1, \tag{18}$$

where $\mathbf{\Omega}_1 \in \mathbb{R}^{(m-1)\times k}$ is a random matrix drawn from Gaussian distribution, which incorporates the randomized concept, while $\mathbf{Y}_1 \in \mathbb{R}^{n\times k}$ is a summary of the action of $\mathbf{X}_1$. We highlight that the computational cost of this matrix–matrix multiplication can be reduced by exploiting structured randomized matrices. Here, $k = r + s$ represents the target rank, where s is defined as the oversampling factor that helps to obtain an improved basis. Bistrian and Navon[54] suggested using an oversampling factor of $s = r$, which we adopt in the present study.

Next, a QR decomposition of $\mathbf{Y}_1$ is performed as $\mathbf{Y}_1 = \mathbf{Q}_1\mathbf{R}_1$ with $\mathbf{Q}_1 \in \mathbb{R}^{n\times k}$ and $\mathbf{R}_1 \in \mathbb{R}^{k\times k}$. Indeed, we only need $\mathbf{Q}_1$ for our computations and can safely discard $\mathbf{R}_1$. The full data $\mathbf{X}_1$ are then projected onto the basis $\mathbf{Q}_1$ to obtain a lower dimension matrix $\mathbf{B}_1 \in \mathbb{R}^{k\times(m-1)}$ as follows:

$$\mathbf{B}_1 = \mathbf{Q}_1^*\mathbf{X}_1, \tag{19}$$

where $\mathbf{Q}^*$ is the conjugate transpose of $\mathbf{Q}$. Thus, SVD can be performed on $\mathbf{B}_1$ instead of $\mathbf{X}_1$ as $\mathbf{B}_1 = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*$. The left and right singular vectors with the corresponding singular values of $\mathbf{X}_1$ can be recovered as follows:

$$\mathbf{U} = \mathbf{Q}_1\tilde{\mathbf{U}}, \tag{20}$$

$$\mathbf{\Sigma} = \tilde{\mathbf{\Sigma}}, \tag{21}$$

$$\mathbf{V} = \tilde{\mathbf{V}}. \tag{22}$$

The algorithmic steps for the randomized DMD based on sketching the range of $\mathbf{X}_1$ are summarized in Algorithm 2.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

ALGORITHM 2. Randomized DMD by sketching the range of $\mathbf{X}_1$.

1: The matrix $\mathbf{X}$ is split into two matrices $\mathbf{X}_1 = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\}$ and $\mathbf{X}_2 = \{\mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_m\}$.
2: Draw a random matrix $\boldsymbol{\Omega}_1 \in \mathbb{R}^{(m-1)\times k}$ from Gaussian distribution and perform the randomized projection of $\mathbf{X}_1$

$$\mathbf{Y}_1 = \mathbf{X}_1\boldsymbol{\Omega}_1$$

3: Perform QR decomposition as $\mathbf{Y}_1 = \mathbf{Q}_1\mathbf{R}_1$ to obtain a near-optimal basis $\mathbf{Q}_1$ for $\mathbf{X}_1$ and discard $\mathbf{R}_1$.
4: A sketch $\mathbf{B}_1$ of $\mathbf{X}_1$ is obtained as

$$\mathbf{B}_1 = \mathbf{Q}_1^*\mathbf{X}_1$$

5: Perform SVD on $\mathbf{B}_1$

$$\tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^* = \mathrm{svd}(\mathbf{B}_1)$$

6: Recover SVD of $\mathbf{X}_1$

$$\mathbf{U} = \mathbf{Q}_1\tilde{\mathbf{U}}$$
$$\boldsymbol{\Sigma} = \tilde{\boldsymbol{\Sigma}}$$
$$\mathbf{V} = \tilde{\mathbf{V}}$$

7: Rank truncation [to reduce noise]

$$\mathbf{U}_\mathbf{R} = \mathbf{U}(:, 1:R)$$
$$\mathbf{V}_\mathbf{R} = \mathbf{V}(:, 1:R)$$
$$\boldsymbol{\Sigma}_\mathbf{R} = \boldsymbol{\Sigma}(1:R, 1:R)$$

8: Low-rank dynamics

$$\tilde{\mathbf{A}} = \mathbf{U}_\mathbf{R}^*\mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}$$

9: Eigenvalue decomposition

$$[\mathbf{W}, \boldsymbol{\Lambda}] = \mathrm{eig}(\tilde{\mathbf{A}})$$

10: Compute DMD modes and spectrum

$$\boldsymbol{\Psi} = \mathbf{U}_\mathbf{R}\mathbf{W} \quad \text{or} \quad \boldsymbol{\Psi} = \mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}\mathbf{W}$$
$$\lambda_i = \{\mathrm{diag}(\boldsymbol{\Lambda})\}$$
$$\alpha_i = \ln(\lambda_i)/\Delta T$$

B. Sketching the range of X

In Sec. IV A, the deterministic SVD of $\mathbf{X}_1$ is bypassed by a randomized projection of $\mathbf{X}_1$ to obtain a sketch that captures its range and efficiently applying SVD onto this sketch. After the SVD of $\mathbf{X}_1$ is recovered, the remaining steps are performed in a high-dimensional space. Alternatively, Erichson *et al.*[56] developed a randomized DMD algorithm that is based on sketching the range of $\mathbf{X}$ and performing the whole DMD procedure in a low-order space, while the DMD of the original system is recovered at the very end. In this algorithm, a random projection is defined to capture the range of $\mathbf{X}$ as follows:

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\Omega}, \tag{23}$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{m\times k}$ is a random matrix and $\mathbf{Y} \in \mathbb{R}^{n\times k}$ is the summary of the action of $\mathbf{X}$. Next, a QR decomposition of $\mathbf{Y}$ is performed as $\mathbf{Y} = \mathbf{QR}$ with $\mathbf{Q} \in \mathbb{R}^{n\times k}$ and $\mathbf{R} \in \mathbb{R}^{k\times k}$, where $\mathbf{Q}$ is considered a near-optimal orthonormal basis such that $\mathbf{X} \approx \mathbf{QQ}^*\mathbf{X}$. The full data matrix $\mathbf{X}$ is projected onto $\mathbf{Q}$ to obtain a lower dimension matrix $\mathbf{B} \in \mathbb{R}^{k\times m}$ as follows:

$$\mathbf{B} = \mathbf{Q}^*\mathbf{X}. \tag{24}$$

$\mathbf{B}$ can be then split into two matrices $\mathbf{B}_1 \in \mathbb{R}^{k\times(m-1)}$ and $\mathbf{B}_2 \in \mathbb{R}^{k\times(m-1)}$ by selecting the first and last $(m-1)$ columns of $\mathbf{B}$, respectively.

Therefore, the DMD algorithm can be implemented using the low-order sketches to yield the DMD modes $\tilde{\boldsymbol{\Psi}}$ of the low-order matrix $\mathbf{B}$. The sought DMD modes are recovered at the end as follows:

$$\boldsymbol{\Psi} = \mathbf{Q}\tilde{\boldsymbol{\Psi}}. \tag{25}$$

The algorithmic steps for the randomized DMD based on sketching the range of $\mathbf{X}$ are summarized in Algorithm 3.

ALGORITHM 3. Randomized DMD by sketching the range of $\mathbf{X}$.

1: Draw a random matrix $\boldsymbol{\Omega} \in \mathbb{R}^{m\times k}$ from Gaussian distribution and perform the randomized projection of $\mathbf{X}$

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\Omega}$$

2: Perform QR decomposition as $\mathbf{Y} = \mathbf{QR}$ to obtain a near-optimal basis $\mathbf{Q}$ for $\mathbf{X}$ and discard $\mathbf{R}$.
3: A sketch $\mathbf{B}$ of $\mathbf{X}$ is obtained as

$$\mathbf{B} = \mathbf{Q}^*\mathbf{X}$$

4: The matrix $\mathbf{B}$ is split into two matrices $\mathbf{B}_1 \in \mathbb{R}^{k\times(m-1)}$ and $\mathbf{B}_2 \in \mathbb{R}^{k\times(m-1)}$.
5: Perform SVD on $\mathbf{B}_1$

$$\tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^* = \mathrm{svd}(\mathbf{B}_1)$$

6: Rank truncation [to reduce noise]

$$\tilde{\mathbf{U}}_\mathbf{R} = \tilde{\mathbf{U}}(:, 1:R)$$
$$\tilde{\mathbf{V}}_\mathbf{R} = \tilde{\mathbf{V}}(:, 1:R)$$
$$\tilde{\boldsymbol{\Sigma}}_\mathbf{R} = \tilde{\boldsymbol{\Sigma}}(1:R, 1:R)$$

7: Low-rank dynamics

$$\tilde{\mathbf{A}} = \tilde{\mathbf{U}}_\mathbf{R}^*\mathbf{B}_2\tilde{\mathbf{V}}_\mathbf{R}\tilde{\boldsymbol{\Sigma}}_\mathbf{R}^{-1}$$

8: Eigenvalue decomposition

$$[\mathbf{W}, \boldsymbol{\Lambda}] = \mathrm{eig}(\tilde{\mathbf{A}})$$

9: Compute DMD modes for $\mathbf{B}$

$$\tilde{\boldsymbol{\Psi}} = \tilde{\mathbf{U}}_R\mathbf{W} \quad \text{or} \quad \tilde{\boldsymbol{\Psi}} = \mathbf{B}_2\tilde{\mathbf{V}}_\mathbf{R}\tilde{\boldsymbol{\Sigma}}_\mathbf{R}^{-1}\mathbf{W}$$

10: Recover DMD modes of $\mathbf{X}$

$$\boldsymbol{\Psi} = \mathbf{Q}\tilde{\boldsymbol{\Psi}}$$

11: Compute DMD spectrum

$$\lambda_i = \{\mathrm{diag}(\boldsymbol{\Lambda})\}$$
$$\alpha_i = \ln(\lambda_i)/\Delta T$$

C. Sketching the range and corange of $\mathbf{X}_1$

In this study, we further develop an efficient DMD implementation based on a sketching approach that captures the range and corange of the data matrix $\mathbf{X}_1 \in \mathbb{R}^{n\times(m-1)}$, resulting in an even smaller

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

sketch than the range sketches.[58,66] We first introduce the sketching operators parametrized by a "range" parameter k and a "core" parameter p that satisfy $r \leq k \leq p \leq \min(n, m-1)$, where the parameter k determines the maximum rank of an approximation. We independently draw and fix four randomized linear reduction maps (often called test matrices) as follows:

$$\begin{aligned} \boldsymbol{\Omega}_1 &\in \mathbb{R}^{(m-1)\times k} \quad \text{and} \quad \boldsymbol{\Gamma}_1 \in \mathbb{R}^{k\times n} \\ \boldsymbol{\Theta}_1 &\in \mathbb{R}^{(m-1)\times p} \quad \text{and} \quad \boldsymbol{\Phi}_1 \in \mathbb{R}^{p\times n}. \end{aligned}$$

Then, we define three matrices comprising our sketch as follows:

$$\mathbf{F}_1 = \mathbf{X}_1\boldsymbol{\Omega}_1 \in \mathbb{R}^{n\times k}, \tag{26}$$

$$\mathbf{G}_1 = \boldsymbol{\Gamma}_1\mathbf{X}_1 \in \mathbb{R}^{k\times(m-1)}, \tag{27}$$

$$\mathbf{H}_1 = \boldsymbol{\Phi}_1\mathbf{X}_1\boldsymbol{\Theta}_1 \in \mathbb{R}^{p\times p}, \tag{28}$$

where $\mathbf{F}_1$ and $\mathbf{G}_1$ capture the range and corange of $\mathbf{X}_1$, respectively, while $\mathbf{H}_1$ is called the core sketch and describes how $\mathbf{X}_1$ acts between the spaces captured by sketches $\mathbf{F}_1$ and $\mathbf{G}_1$.[67] Near-optimal bases for the range and corange of $\mathbf{X}_1$ are computed by the thin QR factorization of $\mathbf{F}_1$ and $\mathbf{G}_1$, respectively, as follows:

$$\mathbf{F}_1 = \mathbf{Q}_1\mathbf{R}_1, \tag{29}$$

$$\mathbf{G}_1^* = \mathbf{P}_1\mathbf{T}_1, \tag{30}$$

where we can again discard the triangular matrices $\mathbf{T}_1$ and $\mathbf{R}_1$. The third sketch $\mathbf{H}_1$ is used to compute the core approximation $\mathbf{C}_1 \in \mathbb{R}^{k\times k}$ of $\mathbf{X}_1$ as follows:

$$\mathbf{C}_1 = (\boldsymbol{\Phi}_1\mathbf{Q}_1)^\dagger\mathbf{H}_1(\mathbf{P}_1^*\boldsymbol{\Theta}_1)^\dagger, \tag{31}$$

which can be implemented efficiency by solving a family of least squares problems. We note that the core sketch is a key element of the proposed SketchyDMD framework as it is responsible for an improved approximation of the system's singular values. This, in turn, results in an enhanced estimate of the DMD spectrum. The original data matrix is now related to the core sketch by the following relation:

$$\mathbf{X}_1 = \mathbf{Q}_1\mathbf{C}_1\mathbf{P}_1^*. \tag{32}$$

Note that $\mathbf{C}_1 \in \mathbb{R}^{k\times k}$ is a square matrix with small size k, compared to $\mathbf{B}_1$ and $\mathbf{B}$ in Secs. IV A and IV B. Thus, we use the core sketch to compute the SVD of $\mathbf{C}_1 = \tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^*$ efficiently, and the singular values and vectors of $\mathbf{X}_1$ can be recovered as follows:

$$\mathbf{U} = \mathbf{Q}_1\tilde{\mathbf{U}}, \tag{33}$$

$$\boldsymbol{\Sigma} = \tilde{\boldsymbol{\Sigma}}, \tag{34}$$

$$\mathbf{V} = \mathbf{P}_1\tilde{\mathbf{V}}. \tag{35}$$

The algorithmic steps for the randomized DMD based on sketching the range of $\mathbf{X}_1$ are summarized in Algorithm 4. We highlight that the randomized sampling in Step 2 of Algorithm 4 can be implemented simultaneously with a single pass over the data matrix $\mathbf{X}_1$. In contrast, Algorithms 2 and 3 require access to the original data matrix at least twice, i.e., first to extract the range basis (top panel in Fig. 2) and second to obtain the low-rank approximation (bottom panel in Fig. 2). This makes SketchyDMD more favorable for applications with memory limitations where a single-view algorithm is a must. Nonetheless, Algorithm 4 requires a more aggressive over-sampling than Algorithms 2 and 3. If a second view is permissible, a better core approximation of $\mathbf{X}_1$ can be defined as $\mathbf{C}_1 = \mathbf{Q}_1^*\mathbf{X}_1\mathbf{P}_1$.

ALGORITHM 4. SketchyDMD: Randomized DMD by sketching the range and corange of $\mathbf{X}_1$.

1: The matrix $\mathbf{X}$ is split into two matrices $\mathbf{X}_1 = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\}$ and $\mathbf{X}_2 = \{\mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_m\}$.

2: Draw random matrices $\boldsymbol{\Omega}_1 \in \mathbb{R}^{(m-1)\times k}$, $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{k\times n}$, $\boldsymbol{\Theta}_1 \in \mathbb{R}^{(m-1)\times s}$, and $\boldsymbol{\Phi}_1 \in \mathbb{R}^{s\times n}$ independently from Gaussian distribution and perform the randomized projection of $\mathbf{X}_1$

$$\begin{aligned} \mathbf{F}_1 &= \mathbf{X}_1\boldsymbol{\Omega}_1 \in \mathbb{R}^{n\times k}, \\ \mathbf{G}_1 &= \boldsymbol{\Gamma}_1\mathbf{X}_1 \in \mathbb{R}^{k\times(m-1)}, \\ \mathbf{H}_1 &= \boldsymbol{\Phi}_1\mathbf{X}_1\boldsymbol{\Theta}_1 \in \mathbb{R}^{s\times s}, \end{aligned}$$

3: Perform QR decomposition of $\mathbf{F}_1$ and $\mathbf{G}_1$ as $\mathbf{F}_1 = \mathbf{Q}_1\mathbf{R}_1$ and $\mathbf{G}_1^* = \mathbf{P}_1\mathbf{T}_1$ to obtain a near-optimal basis $\mathbf{Q}_1$ and $\mathbf{P}_1$ for the range and corange of $\mathbf{X}_1$, respectively. Discard $\mathbf{T}_1$ and $\mathbf{R}_1$.

4: A core sketch $\mathbf{C}_1 \in \mathbb{R}^{k\times k}$ of $\mathbf{X}_1$ is obtained as follows:

$$\mathbf{C}_1 = (\boldsymbol{\Phi}_1\mathbf{Q}_1)^\dagger\mathbf{H}_1(\mathbf{P}_1^*\boldsymbol{\Theta}_1)^\dagger$$

5: Perform SVD on $\mathbf{C}_1$

$$\tilde{\mathbf{U}}\tilde{\boldsymbol{\Sigma}}\tilde{\mathbf{V}}^* = \text{svd}(\mathbf{C}_1)$$

6: Recover SVD of $\mathbf{X}_1$

$$\begin{aligned} \mathbf{U} &= \mathbf{Q}_1\tilde{\mathbf{U}} \\ \boldsymbol{\Sigma} &= \tilde{\boldsymbol{\Sigma}} \\ \mathbf{V} &= \mathbf{P}_1\tilde{\mathbf{V}} \end{aligned}$$

7: Rank truncation [to reduce noise]

$$\begin{aligned} \mathbf{U}_\mathbf{R} &= \mathbf{U}(:, 1:R) \\ \mathbf{V}_\mathbf{R} &= \mathbf{V}(:, 1:R) \\ \boldsymbol{\Sigma}_\mathbf{R} &= \boldsymbol{\Sigma}(1:R, 1:R) \end{aligned}$$

8: Low-rank dynamics

$$\tilde{\mathbf{A}} = \mathbf{U}_\mathbf{R}^*\mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}$$

9: Eigenvalue decomposition

$$[\mathbf{W}, \boldsymbol{\Lambda}] = \text{eig}(\tilde{\mathbf{A}})$$

10: Compute DMD modes and spectrum

$$\begin{aligned} \boldsymbol{\Psi} = \mathbf{U}_\mathbf{R}\mathbf{W} \quad \text{or} \quad \boldsymbol{\Psi} &= \mathbf{X}_2\mathbf{V}_\mathbf{R}\boldsymbol{\Sigma}_\mathbf{R}^{-1}\mathbf{W} \\ \lambda_i &= \{\text{diag}(\boldsymbol{\Lambda})\} \\ \alpha_i &= \ln(\lambda_i)/\Delta T \end{aligned}$$

V. RESULTS AND DISCUSSION

We define the problem setup of testing the presented DMD methodologies using numerical experiments of the SWEs in Sec. V A. Then, we highlight the effect of the modal selection criterion on the DMD performance in Sec. V B. The results of applying the SketchyDMD framework are presented in Sec. V C along with a comparison with existing methodologies in terms of error and central processing unit (CPU) time.

A. Problem setup

For the numerical experiments using the spherical SWEs system, we consider a domain given by longitudes $\Phi = [0, 360^\circ]$ and latitudes $\Theta = [-79.5^\circ, 79.5^\circ]$ (to avoid pole singularity in the polar region).

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

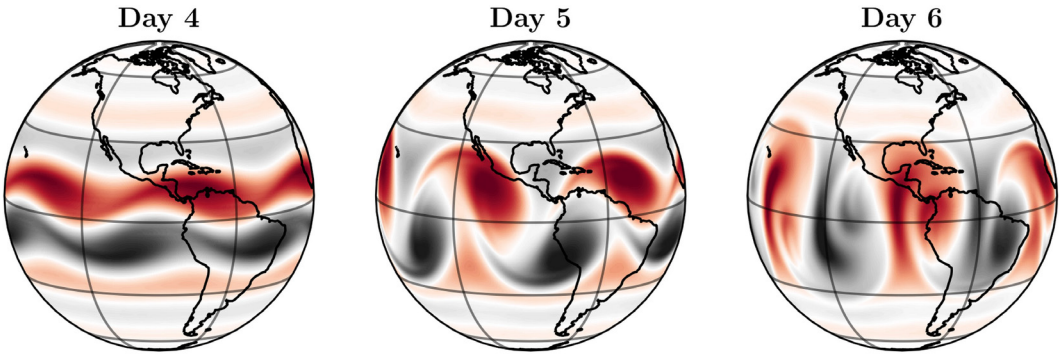

FIG. 3. Evolution of the vorticity over the Earth's surface as predicted by the SWEs at the end of three consecutive days.

We use a spatial resolution of one degree in both directions (i.e., $\Delta\theta = \Delta\phi = 1^\circ$, $N_\theta = 160$, and $N_\phi = 360$). The bottom surface is taken to be flat, i.e., $H(\phi, \theta) = 0 \, \forall \, \phi \in \Phi$ and $\theta \in \Theta$. We define the following initial height h_0:

$$h_0(\cdot, \theta) = 10\,000 - 60\cos(4\pi\theta)e^{-\theta^2}; \;\; \forall \phi \in \Phi. \tag{36}$$

Then, we inject random perturbation to trigger shear layer instability as follows:

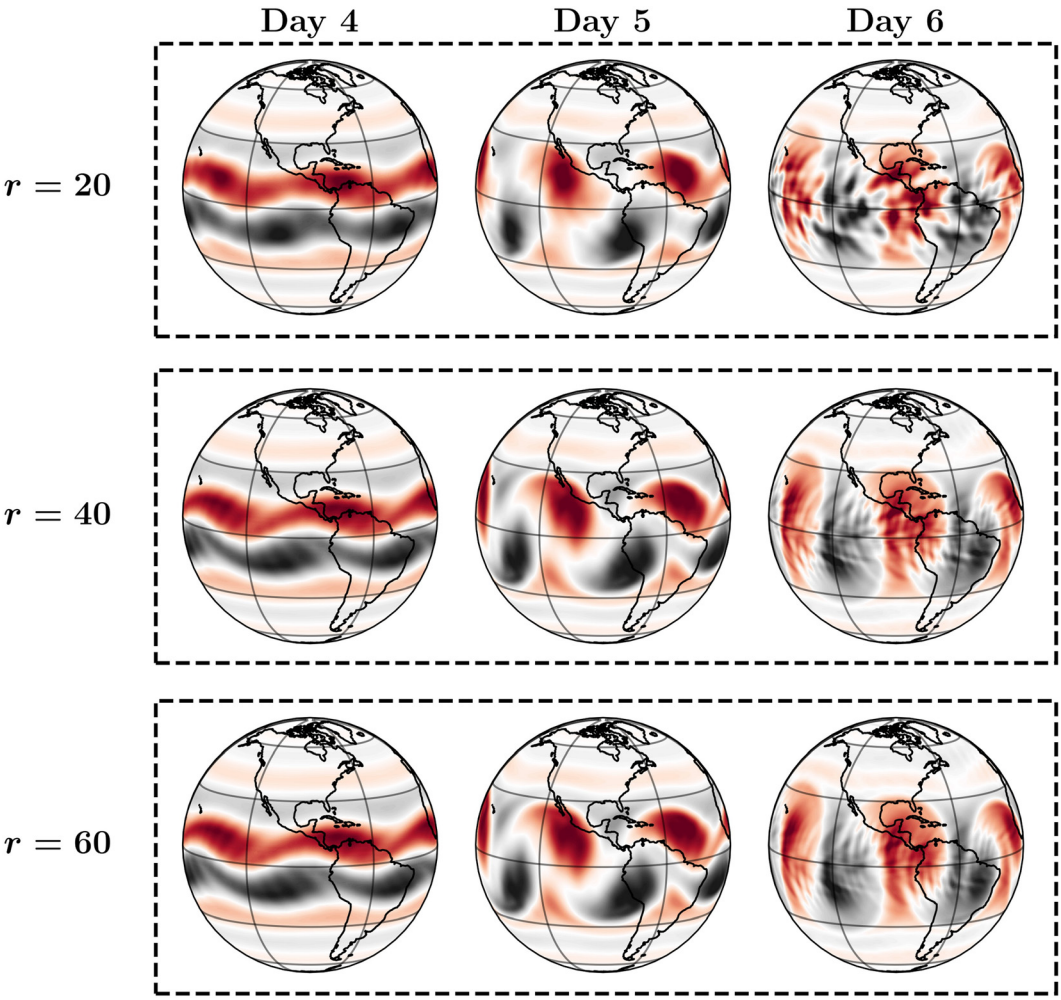

FIG. 4. Deterministic DMD predictions with early truncation using $r = 20$ (top), $r = 40$ (middle), and $r = 60$ (bottom) modes.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

$$h(\cdot,\theta) = h_0(\cdot,\theta) + \kappa\frac{\Delta\theta(N_\theta - 1)}{\pi}|F| \times 10^4 \cos\theta, \tag{37}$$

where κ is a uniformly distributed random number between 0 and 1. Initial velocities fields are computed using the geostrophic wind conditions. We use periodic boundary conditions in the longitudinal (ϕ) direction and slip boundary conditions in the latitudinal (θ) direction.

We carry out the numerical simulations for a time period of 6 days (144 h) and collect snapshots of the vorticity field ($\boldsymbol{\omega} = \nabla \times \mathbf{u}$) each 15 min (referred to as the sampling frequency), resulting in a total of 96 snapshots per day. Moreover, we remove the first 3 days from the collected datasets (corresponding to initial transition period) and consider only snapshots from the end of the third day to the end of the sixth day for the DMD analysis. The evolution of the vorticity fields within the considered time frame is depicted in Fig. 3, where the Cartopy package[68] is utilized for data processing and visualization.

B. Modal selection

In DMD, each mode is associated with a unique oscillation frequency and a growth/decay rate. However, unlike proper orthogonal decomposition where the modes are ordered based on their contribution to the total variance in the data, the DMD modes are not sorted by default. A common approach to overcome this limitation is to take advantage of the singular value decomposition of the data matrix $\mathbf{X}_1$ (see step 2 in Algorithm 1). Since the resulting singular values are sorted in descending order, we can retain the largest r singular values with the corresponding r columns of $\mathbf{U}$ and $\mathbf{V}$. In other words, the r-rank approximation of $\mathbf{X}_1$ (i.e., $R = r$ in Sec. III B) is utilized to compute a square $r \times r$ $\tilde{\mathbf{A}}$ matrix. Therefore, only r DMD modes are obtained from Algorithm 1 eliminating the need for further sorting or selection mechanisms. We call this methodology the "early truncation."

Figure 4 displays the predicted vorticity fields from deploying the deterministic DMD with early truncation using $r = 20$, $r = 40$, and $r = 60$ modes. We notice that the DMD with early truncation can yield inaccurate results, especially with small values of r due to the underestimation of the data matrix $\mathbf{X}_1$. Ahmed *et al.*[69] reported similar observations where early truncation produced a low-quality reconstruction and prediction for a SWE system on a Cartesian grid.

Alternative sorting and selection criteria have been developed to improve the DMD prediction. For example, the DMD modes could be sorted based on their initial amplitudes $\mathbf{b}$ or their growth rate. Using either of these features can overlook modes with large initial amplitudes but rapidly decaying and/or modes with small initial amplitudes but rapidly growing. Instead, we utilize a sorting scheme that is based on the combination of the DMD initial amplitudes and growth/decay rates.[61] In particular, the contribution of each mode over the entire sampling window is estimated from the projection of the FOM solution onto the DMD modes as follows:

$$I_i = |b_i|\frac{e^{\sigma_i T} - 1}{\sigma_i T}, \tag{38}$$

where T is the total sampling time (length of time window). The DMD modes are assigned a value I_i and sorted in decreasing order of I_i. In other words, the DMD model is produced by finding the solution to the following constrained minimization problem:

$$\begin{aligned} &\text{Find truncation rank} && R := \text{argmin}||\omega_{FOM} - \omega_{DMD}|| \\ &\text{subject to :} && I_1 > I_2 > \cdots > I_R. \end{aligned} \tag{39}$$

The first r modes are selected based on their importance index I_i. A comparison between DMD predictions with early truncation and DMD with the sorting criteria in Eq. (38) is demonstrated in Fig. 5.

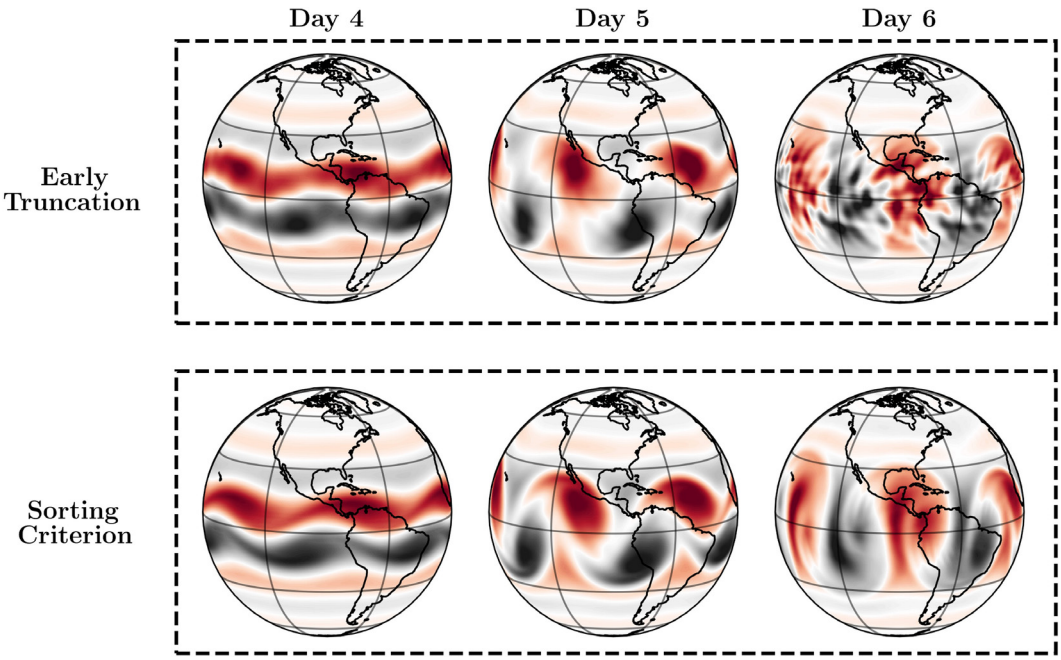

FIG. 5. Deterministic DMD predictions with early truncation (top) and sorting criterion (bottom) using $r = 20$ modes.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

We observe that, even with $r = 20$, embedding a good DMD sorting criterion results in significant improvements in the DMD predictions. Therefore, we adopt this sorting criterion in our numerical investigations of the proposed SketchyDMD framework. It was also shown in Ref. 55 that a similar criterion is very effective along with the imposition of a limit for the relative error of reconstruction.

It should be noted that the selection of modal truncation and selection have been the topic of many DMD studies. For example, Pan *et al.*[70] focused on mode selection by applying a multi-task learning strategy as an extension of sparsity promoting DMD. In the meantime, research efforts have been devoted to discover truncation methods with updatable parameters to maintain important properties of a system, while the order is reduced. Wilson[71] extended the DMD with control algorithm by implementing adaptive parameters to mitigate the truncation errors in the dynamical models.

C. SketchyDMD

Figure 6 displays the vorticity field for the SWEs system at three consecutive days using the proposed SketchyDMD algorithm with $r = 20$. We observe that the quality of SketchyDMD is comparable to the deterministic approach, which operates on the large input data matrix. We also notice that the performance of the SketchyDMD with core sketch is quite sensitive to the values of k and s. In particular, Martinsson and Tropp[72] recommended that more aggressive oversampling is required for a good core sketch approximation and suggested values of the free parameters as $k = 4r$ and $s = 2k$. Power iterations are also common for improving sketching-based algorithms. Moreover, Tropp *et al.*[58] provide some guidelines on setting these hyperparameters given the memory constraints with error bounds guarantees.

As highlighted in Sec. IV C, the SketchyDMD is based on a core sketch approximation, which relates the range and corange of the input data matrix. This relationship provides additional information about the corresponding singular values, which eventually improves the approximation of the DMD spectrum. The approximated DMD spectrum is depicted in Fig. 7 where the computed DMD eigenvalues (λ_i) are plotted in the complex plane for different sketching algorithms. In Fig. 7, we can see that SketchyDMD yields eigenvalues that are closer to the deterministic DMD than the eigenvalues predicted by existing randomized DMD approaches. In addition, ℓ_2 error for SketchyDMD predictions at different times is shown in Fig. 8, compared to the deterministic DMD approach as well as existing randomized methods based on $\mathbf{X}_1$ range (see Sec. IV A) and $\mathbf{X}$ range (see Sec. IV B). Due to the stochastic nature of sketching-based algorithms, we repeat the sampling process with 10 different calls of the pseudo-random number generator. Results in Fig. 8 correspond to the sample mean banded with 95% confidence interval.

Finally, an empirical comparison in terms of CPU time is illustrated in Fig. 9 for different numbers of target DMD modes. We observe that sketching-based DMD algorithms are considerably more efficient than the deterministic implementation. The CPU time gain would be even more pronounced for larger datasets (i.e., higher resolutions and longer time intervals). In the meantime, we emphasize that the SketchyDMD procedure in Algorithm 4 is based on a single pass over the input data matrix $\mathbf{X}_1$, while Algorithms 2 and 3 require at least two passes over the given matrix. Thus, Algorithm 4 is potentially favorable in applications with strict memory limits and streaming datasets. Nonetheless, this single-view SketchyDMD approach requires larger over-sampling factors than the multiple-view versions. As a result, the CPU time of SketchyDMD can be slightly larger than the CPU time for previous randomized DMD frameworks.

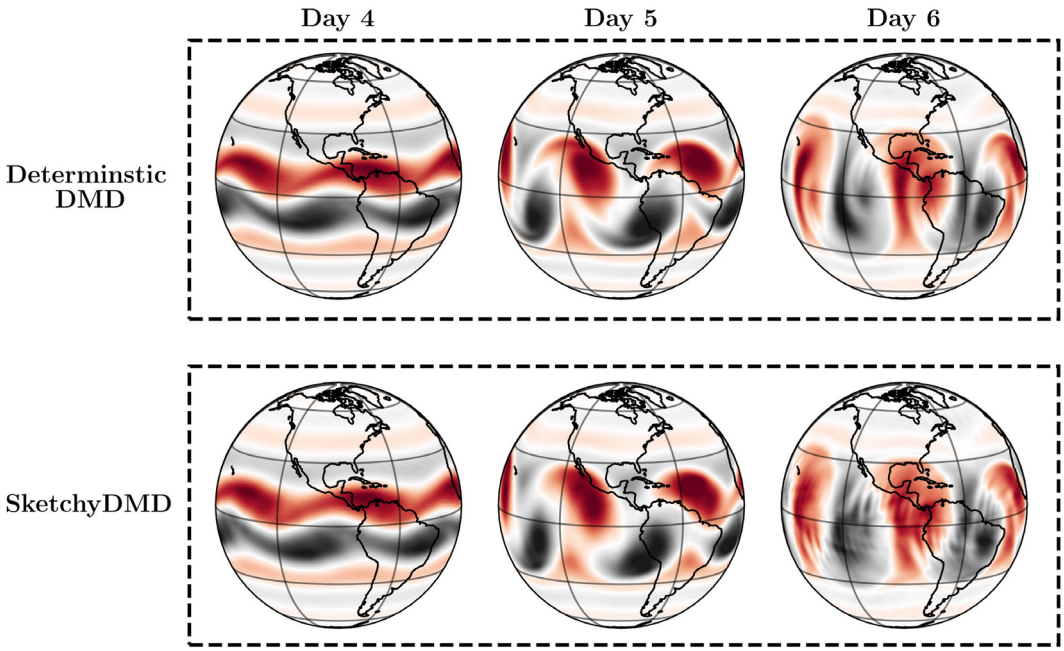

FIG. 6. DMD predictions with sketching the range and corange of $\mathbf{X}_1$ using $r = 20$ modes and sorting criterion.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

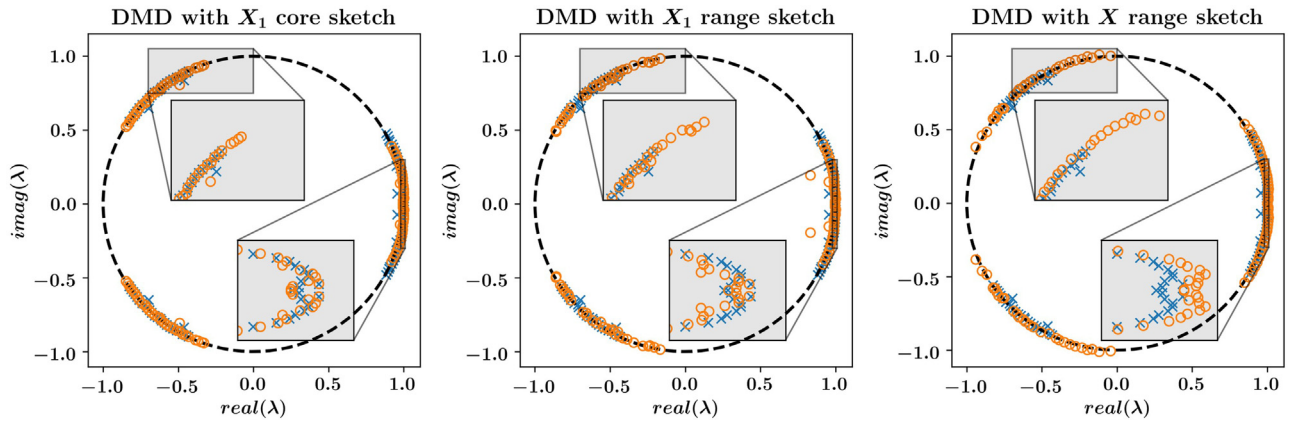

FIG. 7. The DMD eigenvalues for sketching-based DMD (circles) compared to the deterministic DMD spectrum (× symbols). SketchyDMD (with core sketch) provides more accurate DMD spectrum than randomized DMD with only range sketches.

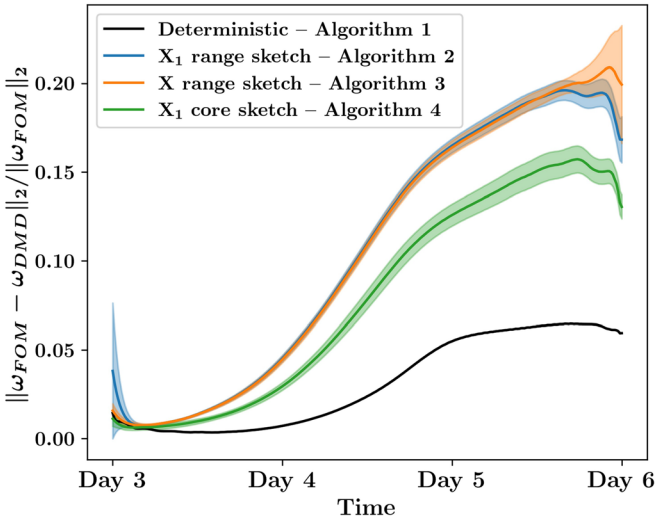

FIG. 8. Normalized ℓ_2 errors in the predicted vorticity fields of the spherical SWEs system using the DMD methodologies listed in Secs. III and IV with respect to the FOM solution. Shaded areas represent 95% confidence intervals using a sample of 10 different calls of the pseudo-random number generator.

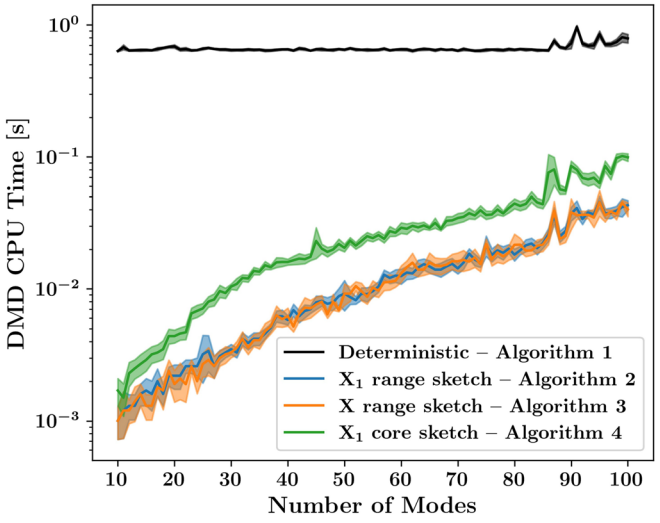

FIG. 9. CPU time for implementing the DMD methodologies listed in Secs. III and IV at different numbers of modes. We note that the CPU time for SketchyDMD is slightly larger than previous randomized DMD methods due to the need for higher over-sampling to compute the core approximation with a single memory pass. Shaded areas represent 95% confidence intervals using 10 different runs to mitigate the effects of irregularity in hardware performance and loading.

VI. CONCLUSIONS

We provide a new sketching-based algorithm to enable the efficient dynamic mode decomposition (DMD) implementation for large datasets. In particular, the SketchyDMD approach is based on random projections to capture the range and corange of the input data matrix. By computing smaller sketches than the original data matrix, expensive portions of the DMD procedure can be bypassed and implemented in a smaller space. In addition, a core sketch is defined to relate the range and corange and enrich the approximated DMD spectrum. The presented SketchyDMD requires a single pass over the input data, which makes it more favorable in applications with limited memory since existing methodologies view elements of the large matrix at least twice. Moreover, the core sketch in SketchyDMD is a square matrix with maximum rank of k, which can be tuned for specific memory constraints. We also highlight the importance of defining problem-specific sorting criteria to select the most dominant DMD modes. We demonstrate the capability of SketchyDMD through numerical experiments using the spherical shallow water equations system. SketchyDMD is shown to be superior to existing randomized DMD for approximating the DMD spectrum. Nonetheless, the performance is quite sensitive to the selection of the over-sampling parameters k and s. In future studies, we will explore improved projection algorithms and power iterations to improve robustness of SketchyDMD.

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Award No. DE-SC0019290. O.S. gratefully acknowledges their support. Direct numerical simulations for this project were carried out using resources of the Oklahoma State University High-Performance Computing Center.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Shady E. Ahmed: Data curation (equal); Formal analysis (equal); Software (lead); Writing – original draft (lead); Writing – review and editing (equal). **Pedram H. Dabaghian:** Investigation (supporting); Software (supporting). **Omer San:** Conceptualization (lead); Data curation (equal); Formal analysis (equal); Funding acquisition (lead); Investigation (lead); Methodology (lead); Project administration (lead); Supervision (lead); Writing – original draft (equal); Writing – review and editing (equal). **Diana A. Bistrian:** Conceptualization (equal); Methodology (equal); Writing – review and editing (equal). **Ionel M. Navon:** Conceptualization (equal); Investigation (equal); Writing – review and editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available within the article. The GitHub repository can also be found at https://github.com/Shady-Ahmed/SketchyDMD.

REFERENCES

[1]M. Udell and A. Townsend, "Why are big data matrices approximately low rank?," SIAM J. Math. Data Sci. **1**, 144–160 (2019).
[2]K. Taira, S. L. Brunton, S. T. Dawson, C. W. Rowley, T. Colonius, B. J. McKeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley, "Modal analysis of fluid flows: An overview," AIAA J. **55**, 4013–4041 (2017).
[3]F. Guéniat, L. Mathelin, and L. R. Pastur, "A dynamic mode decomposition approach for large and arbitrarily sampled systems," Phys. Fluids **27**, 025113 (2015).
[4]V. Puzyrev, M. Ghommem, and S. Meka, "PyROM: A computational framework for reduced order modeling," J. Comput. Sci. **30**, 157–173 (2019).
[5]A. Quarteroni, A. Manzoni, and F. Negri, *Reduced Basis Methods for Partial Differential Equations: An Introduction* (Springer, New York, 2015), Vol. 92.
[6]C. W. Rowley and S. T. Dawson, "Model reduction for flow analysis and control," Annu. Rev. Fluid Mech. **49**, 387–417 (2017).
[7]S. E. Ahmed, S. Pawar, O. San, A. Rasheed, T. Iliescu, and B. R. Noack, "On closures for reduced order models: A spectrum of first-principle to machine-learned avenues," Phys. Fluids **33**, 091301 (2021).
[8]I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," Nonlinear Dyn. **41**, 309–325 (2005).
[9]I. Mezić, "Analysis of fluid flows via spectral properties of the Koopman operator," Annu. Rev. Fluid Mech. **45**, 357–378 (2013).
[10]H. Arbabi and I. Mezic, "Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator," SIAM J. Appl. Dyn. Syst. **16**, 2096–2126 (2017).
[11]J. N. Kutz, J. L. Proctor, and S. L. Brunton, "Applied Koopman theory for partial differential equations and data-driven modeling of spatio-temporal systems," Complexity **2018**, 6010634.
[12]N. Črnjarić-Žic, S. Maćešić, and I. Mezić, "Koopman operator spectrum for random dynamical systems," J. Nonlinear Sci. **30**, 2007–2056 (2020).
[13]M. Korda, M. Putinar, and I. Mezić, "Data-driven spectral analysis of the Koopman operator," Appl. Comput. Harmonic Anal. **48**, 599–629 (2020).
[14]A. Mauroy, Y. Susuki, and I. Mezić, *Koopman Operator in Systems and Control* (Springer, Berlin, 2020).
[15]A. K. Alekseev, D. A. Bistrian, A. E. Bondarev, and I. M. Navon, "On linear and nonlinear aspects of dynamic mode decomposition," Int. J. Numer. Methods Fluids **82**, 348–371 (2016).
[16]M. Rot, M. Horvat, and G. Kosec, "Dynamic mode decomposition as an analysis tool for time-dependent partial differential equations," arXiv:2203.04728 (2022).
[17]P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," J. Fluid Mech. **656**, 5–28 (2010).
[18]C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," J. Fluid Mech. **641**, 115–127 (2009).
[19]N. B. Erichson, S. L. Brunton, and J. N. Kutz, "Compressed dynamic mode decomposition for background modeling," J. Real-Time Image Process. **16**, 1479–1492 (2019).
[20]H. Natsume, H. Tanaka, S. Kajita, and N. Ohno, "Application of dynamic mode decomposition to rotating structures in detached linear plasmas," Phys. Plasmas **27**, 042301 (2020).
[21]S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz, "Modern Koopman theory for dynamical systems," arXiv:2102.12086 (2021).
[22]G. Donovan and C. Brand, "Spatial early warning signals for tipping points using dynamic mode decomposition," Physica A **596**, 127152 (2022).
[23]D. A. Bistrian, G. Dimitriu, and I. M. Navon, "Processing epidemiological data using dynamic mode decomposition method," AIP Conf. Proc. **2164**, 080002 (2019).
[24]D. A. Bistrian, G. Dimitriu, and I. M. Navon, "Modeling dynamic patterns from COVID-19 data using randomized dynamic mode decomposition in predictive mode and ARIMA," AIP Conf. Proc. **2302**, 080002 (2020).
[25]A. Viguerie, G. F. Barros, M. Grave, A. Reali, and A. L. Coutinho, "Coupled and uncoupled dynamic mode decomposition in multi-compartmental systems with applications to epidemiological and additive manufacturing problems," Comput. Methods Appl. Mech. Eng. **391**, 114600 (2022).
[26]T. Sundby, J. M. Graham, A. Rasheed, M. Tabib, and O. San, "Geometric change detection in digital twins," Digital **1**, 111–129 (2021).
[27]Q. Lu and V. M. Zavala, "Image-based model predictive control via dynamic mode decomposition," J. Process Control **104**, 146–157 (2021).
[28]I. U. Haq, T. Iwata, and Y. Kawahara, "Dynamic mode decomposition via convolutional autoencoders for dynamics modeling in videos," Comput. Vision Image Understanding **216**, 103355 (2022).
[29]N. Groun, M. Villalba-Orero, E. Lara-Pezzi, E. Valero, J. Garicano-Mena, and S. L. Clainche, "Higher order dynamic mode decomposition: From fluid dynamics to heart disease analysis," arXiv:2201.03030 (2022).
[30]A. Albarakati, M. Budišić, R. Crocker, J. Glass-Klaiber, S. Iams, J. Maclean, N. Marshall, C. Roberts, and E. S. Van Vleck, "Model and data reduction for data assimilation: Particle filters employing projected forecasts and data with application to a shallow water model," Comput. Math. Appl. (2021) (published online).
[31]C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Reduced-order models for flow control: Balanced models and Koopman

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing

modes," in *Seventh IUTAM Symposium on Laminar-Turbulent Transition* (Springer, 2010), pp. 43–50.
[32]C. Y. Li, Z. Chen, T. K. Tse, A. U. Weerasuriya, X. Zhang, Y. Fu, and X. Lin, "A parametric and feasibility study for data sampling of the dynamic mode decomposition: Spectral insights and further explorations," Phys. Fluids **34**, 035102 (2022).
[33]M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," J. Nonlinear Sci. **25**, 1307–1346 (2015).
[34]S. E. Otto and C. W. Rowley, "Linearly recurrent autoencoder networks for learning dynamics," SIAM J. Appl. Dyn. Syst. **18**, 558–593 (2019).
[35]S. Pan and K. Duraisamy, "Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability," SIAM J. Appl. Dyn. Syst. **19**, 480–509 (2020).
[36]T. Iwata and Y. Kawahara, "Neural dynamic mode decomposition for end-to-end modeling of nonlinear dynamics," arXiv:2012.06191 (2020).
[37]S. C. Puligilla and B. Jayaraman, "Assessment of end-to-end and sequential data-driven learning for non-intrusive modeling of fluid flows," Adv. Comput. Math. **46**, 55 (2020).
[38]O. Azencot, N. B. Erichson, V. Lin, and M. Mahoney, "Forecasting sequential data using consistent Koopman autoencoders," in *International Conference on Machine Learning* (PMLR, 2020), pp. 475–485.
[39]H. Eivazi, L. Guastoni, P. Schlatter, H. Azizpour, and R. Vinuesa, "Recurrent neural networks and Koopman-based frameworks for temporal predictions in a low-order model of turbulence," Int. J. Heat Fluid Flow **90**, 108816 (2021).
[40]S. Sarmast, R. Dadfar, R. F. Mikkelsen, P. Schlatter, S. Ivanell, J. N. Sørensen, and D. S. Henningson, "Mutual inductance instability of the tip vortices behind a wind turbine," J. Fluid Mech. **755**, 705–731 (2014).
[41]S. M. Hirsh, K. D. Harris, J. N. Kutz, and B. W. Brunton, "Centering data improves the dynamic mode decomposition," SIAM J. Appl. Dyn. Syst. **19**, 1920–1955 (2020).
[42]M. S. Hemati, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "De-biasing the dynamic mode decomposition for applied Koopman spectral analysis of noisy datasets," Theor. Comput. Fluid Dyn. **31**, 349–368 (2017).
[43]M. R. Jovanović, P. J. Schmid, and J. W. Nichols, "Sparsity-promoting dynamic mode decomposition," Phys. Fluids **26**, 024103 (2014).
[44]M. S. Hemati, M. O. Williams, and C. W. Rowley, "Dynamic mode decomposition for large and streaming datasets," Phys. Fluids **26**, 111701 (2014).
[45]H. Zhang, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "Online dynamic mode decomposition for time-varying systems," SIAM J. Appl. Dyn. Syst. **18**, 1586–1609 (2019).
[46]S. L. Brunton, J. H. Tu, I. Bright, and J. N. Kutz, "Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems," SIAM J. Appl. Dyn. Syst. **13**, 1716–1732 (2014).
[47]J. N. Kutz, X. Fu, and S. L. Brunton, "Multiresolution dynamic mode decomposition," SIAM J. Appl. Dyn. Syst. **15**, 713–735 (2016).
[48]I. Scherl, B. Strom, J. K. Shang, O. Williams, B. L. Polagye, and S. L. Brunton, "Robust principal component analysis for modal decomposition of corrupt fluid flows," Phys. Rev. Fluids **5**, 054401 (2020).
[49]A. Katrutsa, S. Utyuzhnikov, and I. Oseledets, "Extension of dynamic mode decomposition for dynamic systems with incomplete information based on t-model of optimal prediction," arXiv:2202.11432 (2022).
[50]K. Maryada and S. Norris, "Reduced-communication parallel dynamic mode decomposition," J. Comput. Sci. **61**, 101599 (2022).
[51]M. W. Mahoney, "Randomized algorithms for matrices and data," arXiv:1104.5557 (2011).
[52]D. P. Woodruff, "Sketching as a tool for numerical linear algebra," arXiv:1411.4357 (2014).
[53]N. B. Erichson and C. Donovan, "Randomized low-rank dynamic mode decomposition for motion detection," Comput. Vision Image Understanding **146**, 40–50 (2016).
[54]D. A. Bistrian and I. M. Navon, "Randomized dynamic mode decomposition for nonintrusive reduced order modelling," Int. J. Numer. Methods Eng. **112**, 3–25 (2017).
[55]D. A. Bistrian and I. M. Navon, "Efficiency of andomized dynamic mode decomposition for reduced order modelling," Int. J. Comput. Fluid Dyn. **32**, 88–103 (2018).
[56]N. B. Erichson, L. Mathelin, J. N. Kutz, and S. L. Brunton, "Randomized dynamic mode decomposition," SIAM J. Appl. Dyn. Syst. **18**, 1867–1891 (2019).
[57]A. Alla and J. N. Kutz, "Randomized model order reduction," Adv. Comput. Math. **45**, 1251–1271 (2019).
[58]J. A. Tropp, A. Yurtsever, M. Udell, and V. Cevher, "Streaming low-rank matrix approximation with an application to scientific simulation," SIAM J. Sci. Comput. **41**, A2430–A2463 (2019).
[59]D. L. Williamson, J. B. Drake, J. J. Hack, R. Jakob, and P. N. Swarztrauber, "A standard test set for numerical approximations to the shallow water equations in spherical geometry," J. Comput. Phys. **102**, 211–224 (1992).
[60]J. Thuburn and Y. Li, "Numerical simulations of Rossby–Haurwitz waves," Tellus A **52**, 181–189 (2000).
[61]S. E. Ahmed, O. San, D. A. Bistrian, and I. M. Navon, "Sketching methods for dynamic mode decomposition in spherical shallow water equations," AIAA 2022-2325, 2022.
[62]J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems* (SIAM, 2016).
[63]J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," J. Comput. Dyn. **1**, 391 (2014).
[64]S. D. Pendergrass, J. N. Kutz, and S. L. Brunton, "Streaming GPU singular value and dynamic mode decompositions," arXiv:1612.07875 (2016).
[65]S. Surasinghe and E. M. Bollt, "Randomized projection learning method for dynamic mode decomposition," Mathematics **9**, 2803 (2021).
[66]A. Yurtsever, M. Udell, J. Tropp, and V. Cevher, "Sketchy decisions: Convex low-rank matrix optimization with optimal storage," in *Artificial Intelligence and Statistics* (PMLR, 2017), pp. 1188–1196.
[67]D. Rajaram, C. Perron, T. G. Puranik, and D. N. Mavris, "Randomized algorithms for non-intrusive parametric reduced order modeling," AIAA J. **58**, 5389–5407 (2020).
[68]Met Office, *Cartopy: A Cartographic Python Library With a Matplotlib Interface* (Met Office, Exeter, Devon, 2015).
[69]S. E. Ahmed, O. San, D. A. Bistrian, and I. M. Navon, "Sampling and resolution characteristics in reduced order models of shallow water equations: Intrusive vs nonintrusive," Int. J. Numer. Methods Fluids **92**, 992–1036 (2020).
[70]S. Pan, N. Arnold-Medabalimi, and K. Duraisamy, "Sparsity-promoting algorithms for the discovery of informative Koopman-invariant subspaces," J. Fluid Mech. **917**, A18 (2021).
[71]D. Wilson, "Data-driven identification of dynamical models using adaptive parameter sets," Chaos **32**, 023118 (2022).
[72]P.-G. Martinsson and J. A. Tropp, "Randomized numerical linear algebra: Foundations and algorithms," Acta Numer. **29**, 403–572 (2020).

31 August 2023 11:05:41

Published under an exclusive license by AIP Publishing