

Article

# Randomized Projection Learning Method for Dynamic Mode Decomposition

Sudam Surasinghe <sup>1,\*</sup>,†,‡  and Erik M. Bollt <sup>2,‡</sup><sup>1</sup> Department of Mathematics, Clarkson University, Potsdam, NY 13699, USA<sup>2</sup> Electrical and Computer Engineering and C<sup>3</sup>S<sup>2</sup> the Clarkson Center for Complex Systems Science, Clarkson University, Potsdam, NY 13699, USA; ebollt@clarkson.edu

\* Correspondence: surasinc@clarkson.edu

† Current address: Clarkson University, Potsdam, NY 13699, USA.

‡ These authors contributed equally to this work.

**Abstract:** A data-driven analysis method known as dynamic mode decomposition (DMD) approximates the linear Koopman operator on a projected space. In the spirit of Johnson–Lindenstrauss lemma, we will use a random projection to estimate the DMD modes in a reduced dimensional space. In practical applications, snapshots are in a high-dimensional observable space and the DMD operator matrix is massive. Hence, computing DMD with the full spectrum is expensive, so our main computational goal is to estimate the eigenvalue and eigenvectors of the DMD operator in a projected domain. We generalize the current algorithm to estimate a projected DMD operator. We focus on a powerful and simple random projection algorithm that will reduce the computational and storage costs. While, clearly, a random projection simplifies the algorithmic complexity of a detailed optimal projection, as we will show, the results can generally be excellent, nonetheless, and the quality could be understood through a well-developed theory of random projections. We will demonstrate that modes could be calculated for a low cost by the projected data with sufficient dimension.

**Keywords:** Koopman operator; dynamic mode decomposition (DMD); Johnson–Lindenstrauss lemma; random projection; data-driven method



**Citation:** Surasinghe, S.; Bollt, E. Randomized Projection Learning Method for Dynamic Mode Decomposition. *Mathematics* **2021**, *9*, 2803. <https://doi.org/10.3390/math9212803>

Academic Editor: Alexandre Mauroy, Yoshihiko Susuki and Igor Mezic

Received: 28 September 2021

Accepted: 29 October 2021

Published: 4 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Modeling real-world phenomena in physical sciences to engineering, videography, and economics is limited due to computational costs. Real-world systems require dynamic nonlinear modeling. Dynamic mode decomposition (DMD) [1–3] is an emerging tool in this area that could use the data directly rather than intermediary differential equations. However, the data in real-world applications is enormous. The DMD algorithm's reliance on a singular value decomposition (SVD) appears to be a limiting factor due to storing and calculating the SVD of such a matrix. One can notice that SVD calculations of snapshot matrices in big projects would require the use of supercomputers or days of computation. To allow processing on small-scale computers and in a shorter time frame, we propose developing an algorithm based on a randomized projection [4], which is used to reduce the dimension of observable space in DMD, which we call rDMD. In order to utilize and carefully analyzed the rDMD, we will use the Johnson–Lindenstrauss lemma. It is clear that a random projection is simple as compared to a detailed optimal projection method, but our analysis and examples demonstrate, nonetheless, the quality and efficiency.

Strong theoretical support from Johnson–Lindenstrauss (JL) lemma [5] makes the random projection method reliable and has extensive utilization in the field of data science. According to the JL lemma, if data points lie in a sufficiently high-dimensional space, then those data points may be projected into a sufficiently low-dimensional space while approximately preserving the distance of the data points. Furthermore, the projection can be done by a random matrix, which makes algorithms based on the JL lemma both

past and simple. Hence, this tool is more powerful and adopted heavily in data science. JL lemma-based random projection and SVD-based projection can be used to project  $N$  dimensional data into a lower dimension  $L \ll N$ . Data matrix  $X_{N \times M}$  can be projected (by random projection) into a lower dimension ( $L$ ) subspace as  $X_L := RX$ , where  $R$  is a random matrix with unit length. Hence, the random projection is very simple because it relies only on matrix multiplication. Moreover, computational complexity is  $\mathcal{O}(MLN)$ , while SVD has computational complexity  $\mathcal{O}(NM^2)$  when  $M < N$  [6]. We will use the random projection to project high-dimensional snapshot matrices into a manageable low-dimensional space. In a theoretical perspective, the dimensions of the input and output spaces in the Koopman operator can be reduced by the random projection method; thus, reducing the storage and computational cost of the DMD algorithm.

Both DMD [2] and rDMD are grounded in theory through the application of the Koopman operator. They are numerical methods used to estimate the Koopman operator that identifies spatial and temporal patterns from a dynamical system. Theoretical support of the Koopman operator theory makes these algorithms strong. A major objective of the DMD method is to isolate and interpret the spatial features of a dynamical system through the associated eigenvalues of the DMD operator (see Section 4.3). Our new randomized DMD (rDMD) algorithm targets to address issues that arise in SVD-based existing DMD methods by reducing the dimensionality of the matrix just by using matrix multiplication. rDMD can achieve very accurate results with low-dimensional data embedded in high-dimensional observable space.

This paper will summarize the Koopman operator theory, existing DMD algorithms, and random projection theory in Section 2. Then we will discuss our proposed randomized DMD algorithm in Section 3, and finally, in Section 4, we will provide examples that support our approach. Major notations and abbreviations used in this article are listed in Table 1.

**Table 1.** Major notations and abbreviations used in the article.

Notation and Abbreviations	Meaning
$\mathcal{K}$	Koopman operator
$M$	Number of snapshots
$N$	Number of observable
$L$	Dimension of the projected observable space
$\mathbb{K}$	DMD on original space
$\mathbb{K}$	DMD on projected space
$X$	Snapshot matrix: time stape 0 to $M$
$Y$	Snapshot matrix: time stape 0 to $M$
$X_L, Y_L$	Projected snapshot matrices
$P$	Projection matrix of dimension $L \times N$
$R$	Random projection matrix
$(\lambda, \phi)$	Eigenvalue and eigenvector of $\mathbb{K}$
$(\lambda_L, \phi_L)$	Eigenvalue and eigenvector of $\mathbb{K}$
DMD	Dynamic Mode Decomposition
SVD	Extended Dynamic Mode Decomposition
rDMD	randomized Dynamic Mode Decomposition
RP	Random Projection
JL	Johnson–Lindenstrauss

**Remark 1** (Problem Statement). *In this article, we investigate the estimation of Koopman modes by the computationally efficient method. Standard existing methods calculate the SVD of the snapshot matrix. However, direct SVD calculations of such matrices can be quite large and computationally intensive. Here, we propose a simple method based on the JL lemma to perform calculations simply on random subsamples, doing so in an analytically rigorous and controlled way. In addition, we will generalize the current SVD-based methods as a DMD in a projected space. This allows extension of choices that can be used in DMD algorithms. We will show the quality of our*

theoretical results through the well-developed theory of random projections in high-dimensional geometry. In addition, experimental and real-world examples validate excellent results with the theory underlying our method.

### 2. Dynamic Mode Decomposition and Background Theory

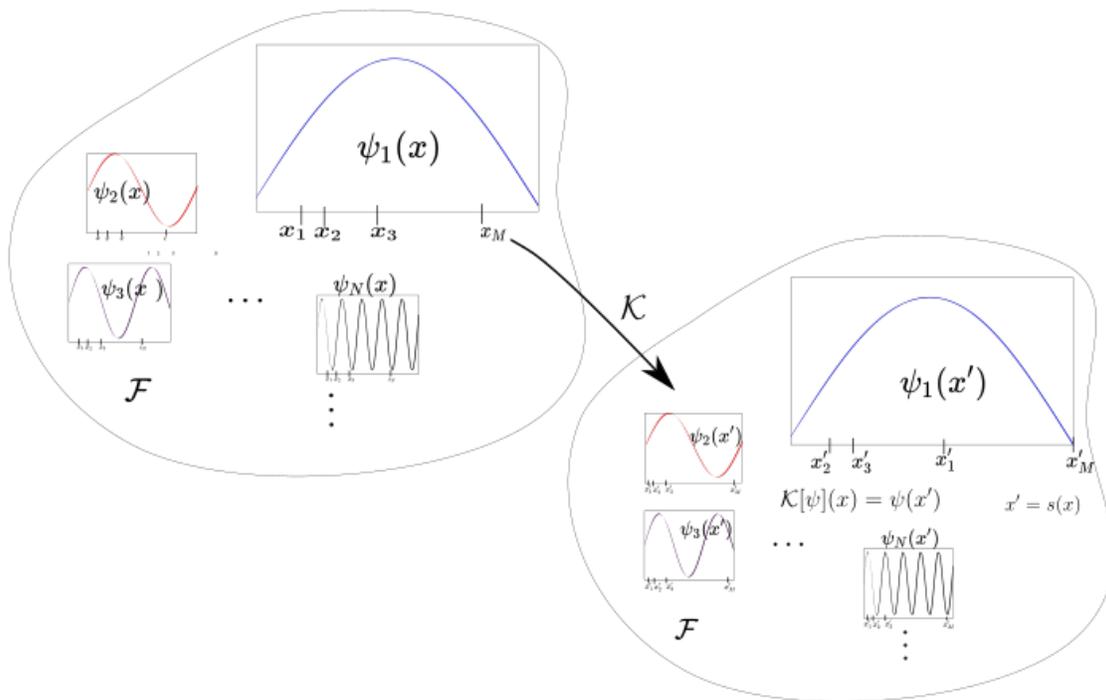
The focus of this paper is to approximate the eigenpairs of the Koopman operator based on the randomized dynamic mode decomposition method. We will first review the underlining theory about the Koopman operator.

#### 2.1. Koopman Operator

Consider a discrete-time dynamical system

$$x_{n+1} = S(x_n), \tag{1}$$

where  $S : \mathcal{M} \rightarrow \mathcal{M}$  and  $\mathcal{M}$  is a finite dimensional manifold. (If we have a differential equation or continuous time dynamical system, the flow map can be considered.) The variable  $x$  is often recognized as a state variable and  $\mathcal{M}$  as phase space. The associated Koopman operator is described as the evaluation of observable functions (Figure 1)  $\psi : \mathcal{M} \rightarrow \mathbb{R}$  in function space  $\mathcal{F}$ . Instead of analyzing the individual trajectories in phase space, the Koopman operator operates on the observations [2,3,7,8].



**Figure 1.** This figure shows the behavior of the Koopman operator  $\mathcal{K}$  in observable space  $\mathcal{F}$  associated with a dynamical system  $S$ . The Koopman operator evaluates the observable  $\psi$  at downstream or future  $x' = x_{n+1} = S(x_n)$ .

**Definition 1** (Koopman operator [7]). *The Koopman operator  $\mathcal{K}$  for a map  $S$  is defined as the following composition,*

$$\begin{aligned} \mathcal{K} : \mathcal{F} &\rightarrow \mathcal{F} \\ \psi &\mapsto \mathcal{K}[\psi] = \psi \circ S \end{aligned} \tag{2}$$

on the function space  $\mathcal{F}$ .

It is straightforward, to prove [7],

$$\mathcal{K}[a\psi_1 + b\psi_2] = a(\psi_1 \circ S) + b(\psi_2 \circ S) = a\mathcal{K}[\psi_1] + b\mathcal{K}[\psi_2] \tag{3}$$

for  $\psi_1, \psi_2 \in \mathcal{F}$  and  $a, b \in \mathbb{C}$  and, therefore, the Koopman operator is linear on  $\mathcal{F}$ . This is an interesting and important property of the operator because the associated map  $S$  most probably will be non-linear. Even though the operator is associated with a map that evolves in a finite dimensional space,  $\mathcal{F}$  the function space in which the operator acts on could possibly be an infinite dimensional. This is the trade-off between costs for the linearity [8].

Spectral analysis of the Koopman operator can be used to decompose the dynamics, which is the key success in the DMD. Assuming the spectrum of the Koopman operator  $\mathcal{K}$  is given by

$$\mathcal{K}\psi_i(x) = \lambda_i\psi_i(x) \quad i = 1, 2, 3, \dots \tag{4}$$

then vector-valued observables  $\mathbf{g} : \mathcal{M} \rightarrow \mathbb{R}^N$  (or  $\mathbb{C}^N$ ) can be represented by

$$\mathbf{g}(\mathbf{x}) = \sum_{i=1}^{\infty} \psi_i(\mathbf{x})\boldsymbol{\phi}_i, \tag{5}$$

where  $\boldsymbol{\phi}_i \in \mathbb{R}^N$  (or  $\mathbb{C}^N$ ) are the vector coefficients of the expansion and called ‘‘Koopman modes’’ (here we assumed that components of  $\mathbf{g}$  lie within the span of the eigenfunctions of  $\mathcal{K}$ ). Note that the observable value at time  $n + 1$  is given by

$$\mathbf{g}(\mathbf{x}_{n+1}) = \sum_{i=1}^{\infty} \lambda_i^n \psi_i(\mathbf{x}_0)\boldsymbol{\phi}_i. \tag{6}$$

This decomposition can be used to separate the spacial and time components of the dynamical system and can be used to isolate the specific dynamics.

### 2.2. Dynamic Mode Decomposition

The dynamic mode decomposition is a data-driven method to estimate the Koopman modes from numerical or experimental data [2]. Suppose dynamics are governed by Equation (1) for any state  $\mathbf{x}$  and vector valued measurements are given by observable  $\mathbf{g}(\mathbf{x}) \in \mathbb{R}^N$ . For a given set of data  $X = [\mathbf{g}(\mathbf{x}_0) \mathbf{g}(\mathbf{x}_1) \dots \mathbf{g}(\mathbf{x}_{M-1})]$ ,  $Y = [\mathbf{y}_0 \mathbf{y}_1 \dots \mathbf{y}_{M-1}]$  where  $\mathbf{y}_i = \mathbf{g}(\mathbf{s}(\mathbf{x}_i))$ , the Koopman modes and eigenvalues of the Koopman operator can be estimated through solving the least-squares problem

$$\mathbb{K} = \arg \min_K \|KX - Y\|_F^2 = \arg \min_K \sum_{i=0}^{M-1} \|K\mathbf{g}(\mathbf{x}_i) - \mathbf{y}_i\|_2^2 \tag{7}$$

and  $\mathbb{K} = YX^\dagger$  (here  $X^\dagger$  is the pseudo-inverse of  $X$ ) is defined as the ‘‘Exact DMD’’ operator [3]. The eigenvalue ( $\hat{\lambda}$ ) of  $\mathbb{K}$  is an approximation of an eigenvalue ( $\lambda$ ) of  $\mathcal{K}$ ; the corresponding right eigenvector ( $\hat{\phi}$ ) is called the DMD mode and approximates the Koopman mode ( $\phi$ ). Then the observable value  $\mathbf{g}(\mathbf{x}(t))$  at time  $t$  can be modeled as

$$\mathbf{g}(\mathbf{x}(t)) = \sum_{i=1}^r \psi_i(\mathbf{x}_0)\hat{\boldsymbol{\phi}}_i\hat{\lambda}_i^t \tag{8}$$

where  $r$  is the number of selected DMD modes and demonstrates the finite dimensional approximation for vector-valued observable  $\mathbf{g}$  under the Koopman operator. Based on this decomposition, data matrices can be expressed as

$$\begin{aligned} X_{N \times M} &= \Phi_{N \times r} T_{r \times M} \\ Y_{N \times M} &= \Phi_{N \times r} \Lambda_{r \times r} T_{r \times M} \end{aligned} \tag{9}$$

where  $\Phi = [\psi_1(x_0)\phi_1 \ \psi_2(x_0)\phi_2 \ \dots \ \psi_r(x_0)\phi_r]$ ,  $T$  is a Vandermonde matrix with  $T_{ij} = \lambda_i^{j-1}$  for  $i = 1, 2, \dots, r, j = 1, 2, \dots, M$  and  $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_r\}$ . Note that with the above decomposition  $\mathbb{K} = YX^T = \Phi\Lambda TT^T\Phi^T$ . We will suppose  $\mathbb{K}$  has distinct eigenvalues  $\lambda_i$ , columns of  $X$  are linearly independent and  $r \leq M$ . In practical applications, we are expected to fully understand the data set by relatively few ( $r \ll M$ ) modes. This can be considered one of the dimension reduction steps of the algorithm. Additionally, the dimension of columns of the data matrix need to be reduced.

In practice, the columns of data matrix  $X$  (and  $Y$ ) are constructed by the snapshot matrices of spatial observable data. More often, those snapshots lie in a high-dimensional space  $\mathbb{R}^N$  ( $N \gg 1$  and roughly  $\mathcal{O}(10^{15})$  to  $\mathcal{O}(10^{10})$ ), but the number of snapshots or time steps ( $M$ ) are small and often it is  $\mathcal{O}(10^3)$  to  $\mathcal{O}(10^1)$  [9]. Hence, computing the spectrum of matrix  $\mathbb{K}$  by direct SVD is computationally intensive, even though most of the eigenvalues will be zero. Method-of-snapshot, parallel version of SVD, or randomized SVD can be used to attack this difficulty [10,11]. In this project, we use a more simple randomized method by generalizing the DMD algorithm. We can project our data matrices  $X, Y$  into a low-dimensional space  $\mathbb{R}^L$  with  $r \leq L \leq M \ll N$ ; therefore, we need to estimate the spectrum of  $\mathbb{K}$  based on the computation on the projected space. Our proposed rDMD method is focused on this dimension reduction step.

In the next section (Section 3), we will discuss more details about the calculation. Note that the current methods are based on the singular value decomposition of the data matrix  $X$  to construct a projection, and our proposed algorithm is based on the random projection method to project data into a low-dimensional space.

### 2.3. Random Projection

The random projection method is based on the Johnson–Lindenstrauss lemma, which is applied by many data analysis methods. In this article, we use a random matrix  $R$  generated by a Gaussian distribution, such that each element  $r_{ij} \stackrel{iid}{\sim} \mathcal{N}(0, 1)$  with a normalized column is of unit length.

**Theorem 1** (Johnson–Lindenstrauss lemma [5]). *For any  $0 < \epsilon < 1$  and any integer  $M > 1$ , let  $L$  be a positive integer, such that  $L \geq L_0$  with  $L_0 = \frac{C \ln M}{\epsilon^2}$ , where  $C$  is a suitable constant ( $C \approx 8$  in practice,  $C = 2$  is good enough). Then for any set  $X$  of  $M$  data points in  $\mathbb{R}^N$ , there exists a map  $f : \mathbb{R}^N \rightarrow \mathbb{R}^L$  such that for all  $x_1, x_2 \in X$ ,*

$$(1 - \epsilon) \|x_1 - x_2\|^2 \leq \|f(x_1) - f(x_2)\|^2 \leq (1 + \epsilon) \|x_1 - x_2\|^2.$$

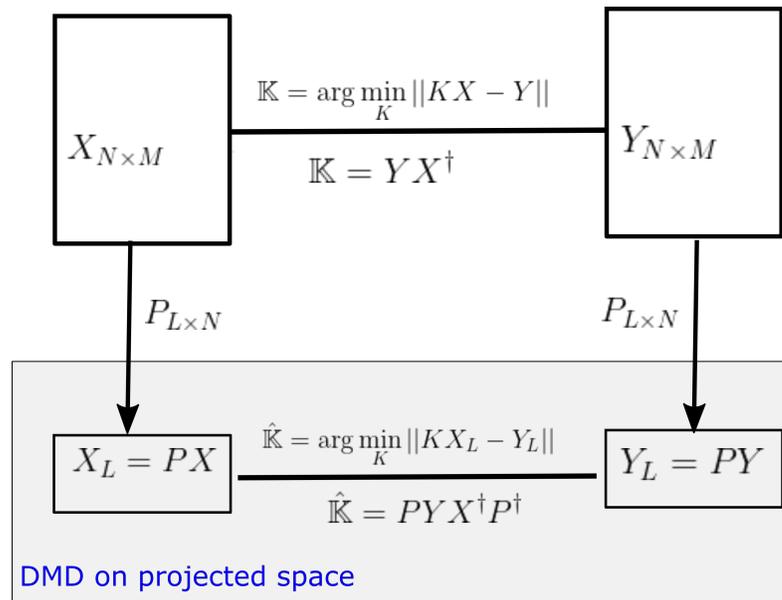
**Theorem 2** (Random Projection [4]). *For any  $0 < \epsilon, \delta < \frac{1}{2}$  and positive integer  $N$ , there exists a random matrix  $R$  of size  $L \times N$  such that for  $L \geq L_0$  with  $L_0 = \frac{C \ln(1/\delta)}{\epsilon^2}$ . and for any unit-length vector  $x \in \mathbb{R}^N$*

$$Pr\{|||Rx||^2 - 1| > \epsilon\} \leq \delta$$

or

$$Pr\{|||Rx||^2 - 1| > \epsilon\} \leq e^{-CL\epsilon^2}$$

A low rank approximation for both  $X, Y$  can be found using the random projection method. Notice that both these matrices have  $M$  points from  $N$  dimensional observable space and, therefore, we can use random projection matrix  $R$  of size  $L \times N$  with the  $L \geq \frac{C \ln N}{\epsilon^2}$ , which provides  $\epsilon$ -isometry to  $\mathbb{R}^N$ . (See Figure 2 for details).



**Figure 2.** DMD operator on the projected space. This figure shows the relationship between the DMD operator on original space and DMD on projected space. The operator  $\hat{\mathbb{K}}$  on the projected space is defined in Equation (10) and can be calculated by Equation (10).

### 3. Randomized Dynamic Mode Decomposition

In this section, we generalize currently used DMD algorithms and then discuss our proposed randomized DMD algorithm.

#### 3.1. DMD on Projected Space

As mentioned in Section 2.2, computational and storage costs of DMD can be reduced by projecting data into a low-dimensional observable space. Let  $P \in \mathbb{R}^{L \times N}$  be any rank  $L$  projection matrix, then dimension of data matrices  $X, Y \in \mathbb{R}^{N \times M}$  can be reduced to  $L \times M$  by the projection  $X_L = PX, Y_L = PY$ . The DMD operator on the projected space (see Figure 2) is given by,

$$\hat{\mathbb{K}} = \arg \min_{K \in \mathbb{R}^{L \times L}} \|KX_L - Y_L\|_F^2 = \arg \min_{K \in \mathbb{R}^{L \times L}} \|KPX - PY\|_F^2 \tag{10}$$

and  $\hat{\mathbb{K}} = PY(PX)^\dagger$ . Therefore

$$\hat{\mathbb{K}} = PY(PX)^\dagger = PYX^\dagger P^\dagger = P\mathbb{K}P^\dagger \tag{11}$$

where  $\mathbb{K} = YX^\dagger$  is the DMD operator on the original space.

**Proposition 1.** Some eigenpairs  $(\lambda, \phi)$  of  $\mathbb{K}$  can be obtain by  $(\lambda_L, \phi_L)$  of projected DMD  $\hat{\mathbb{K}}$  with  $\lambda = \lambda_L$  and  $\phi = P^\dagger \phi_L$ .

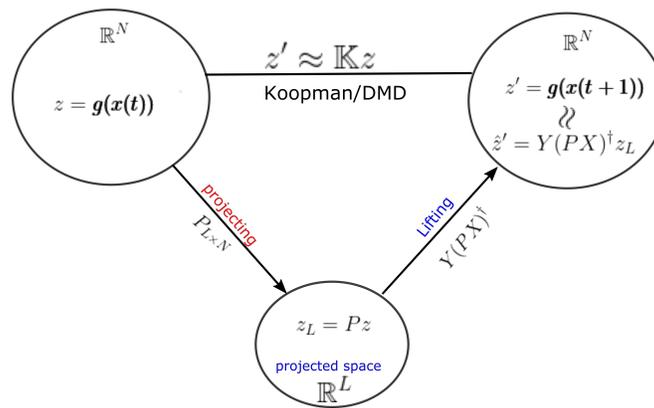
**Proof.** Let  $(\lambda_L, \phi_L)$  be an eigenpair of  $\hat{\mathbb{K}}$ . Then  $\hat{\mathbb{K}}\phi_L = \lambda_L \phi_L$  and by Equation (11),  $P\mathbb{K}P^\dagger \phi_L = \lambda_L \phi_L$ . Now let  $P^\dagger \phi_L = \phi$ , then  $\phi_L = P\phi$  because  $PP^\dagger = I$ . Hence,  $P\mathbb{K}P^\dagger \phi_L = P\mathbb{K}\phi = \lambda_L P\phi$  and  $P(\mathbb{K}\phi - \lambda_L \phi) = 0$ . Since  $\mathbb{K}\phi - \lambda_L \phi = 0$  is a solution to the above equation,  $\lambda_L$  is an eigenvalue and the corresponding eigenvector is  $\phi = P^\dagger \phi_L$  of  $\mathbb{K}$ .  $\square$

In other words, we can lift up the dimension of eigenvectors in a projected space by  $P^\dagger$  to obtain an eigenvector in the original data space. However to avoid the direct calculation of the pseudo-inverse of the projection matrix, we can calculate the eigenvector in the output space  $Y_L$  of the DMD operator and lift up the vector into the original output

space  $Y$ . We can easily show that  $\hat{\phi} = Y(PX)^\dagger \phi_L$  is an eigenvector of  $\mathbb{K}$  for corresponding non-zero eigenvalues.

$$\begin{aligned} \mathbb{K}\hat{\phi} &= \mathbb{K}Y(PX)^\dagger \phi_L = \mathbb{K}YX^\dagger P^\dagger \phi_L \\ &= \mathbb{K}\mathbb{K}P^\dagger \phi_L \\ &= \mathbb{K}\lambda P^\dagger \phi_L \\ &= \lambda YX^\dagger P^\dagger \phi_L \\ &= \lambda Y(PX)^\dagger \phi_L \\ &= \lambda \hat{\phi} \end{aligned} \tag{12}$$

Moreover, notice,  $P\hat{\phi} = PY(PX)^\dagger \phi_L = \hat{\mathbb{K}}\phi_L$  and, therefore,  $\hat{\phi}$  estimates the eigenvector on the output space  $Y$ . A detailed view of this lifting operator is shown in Figure 3. It provides the relationship of the lifting operator with the DMD operator acting on any general observable vector  $z = \mathbf{g}(\mathbf{x}(t)) \in \mathbb{R}^N$ .



**Figure 3.** The figure shows the projecting operator  $P$  and DMD related lifting operator  $Y(PX)^\dagger = \mathbb{K}P^\dagger$ , which should be used in DMD algorithms. Instead of using  $P^\dagger$  as the lifting operator,  $Y(PX)^\dagger$  can be used for efficient calculations. Moreover, notice that  $\hat{z}'_L := Pz' = \hat{\mathbb{K}}z_L$ .

Next, the focus moves to the spatial–temporal decomposition of the projected data matrices by spectrum of the DMD operator. Note that the observable value  $\mathbf{g}(\mathbf{x}(t))$  at time  $t$  can be modeled as  $\mathbf{g}(\mathbf{x}(t)) = \sum_{i=1}^r \psi_i(\mathbf{x}_0)P^\dagger(\phi_L)_i \hat{\lambda}_i^t$  and similar to the Equation (9), data can be decomposed as

$$\begin{aligned} X_{N \times M} &= P_{N \times L}^\dagger \tilde{\Phi}_{L \times r} T_{r \times M} \\ Y_{N \times M} &= P_{N \times L}^\dagger \tilde{\Phi}_{L \times r} \Lambda_{r \times r} T_{r \times M}. \end{aligned} \tag{13}$$

This decomposition leads to  $\mathbb{K} = YX^\dagger = P^\dagger \tilde{\Phi} \Lambda T T^\dagger \tilde{\Phi}^\dagger P$  and if  $r \leq L$  all the non-zero eigenvalues and corresponding eigenvectors of  $\mathbb{K}$  can be constructed by the projected DMD operator. Further, Equation (13) can be used to isolate the spatial profile of interesting dynamical components, such as attractors, periodic behaviors, etc.

Based on the choice of the projection matrix, we have alternative ways to estimate the spectrum of the DMD operator.

**Remark 2** (Projection by SVD). A commonly used projection matrix is based on SVD of the input matrix  $X = U\Sigma V^*$  and the projection matrix is chosen to be  $P = U^*$ , here,  $*$  represents the conjugate transpose of a matrix. Using Equation (11) and SVD of  $X$ , the operator on the projected space can be formulated as  $\hat{\mathbb{K}} = U^*YV\Sigma^{-1}$ .

**Remark 3** (Standard DMD and Exact DMD). Let eigenpair of an SVD-based  $\hat{\mathbb{K}} = U^*YV\Sigma^{-1}$  be given by  $(\lambda, \phi_L)$ . In a standard DMD (Reference Schmid’s paper and Tu’s paper) use the eigenvector  $P^\dagger\phi_L = U\phi_L$  to estimate eigenvectors of  $\mathbb{K}$ . On the other hand, in the exact DMD (reference Tu’s paper) this eigenvector is estimated by  $Y(PX)^\dagger\phi_L = YV\Sigma^{-1}\phi_L$ .

**Remark 4.** One can also use QR decomposition-based projection methods. QR decomposition of the input–output snapshot data matrix  $[X Y]$  is used in some existing methods [3], and so our philosophy of random projection methods used here will greatly improve efficiency, but, nonetheless, with quality that is controlled in an analytically rigorous way.

In this paper, we propose a simple random projection-based method to estimate the spectrum of the DMD operator.

### 3.2. Randomized Methods

Our suggested randomized dynamic mode decomposition (rDMD) is based on the random projection applied to the theory of DMD on a projected space. We can reduce the dimension of the data matrix  $X, Y$  in DMD by using a random projection matrix  $R_{L \times N}$ . In other words, we construct a projection matrix  $P$  discussed in Section 3.1 as a random matrix  $R$  whose columns have unit lengths and entries that are selected independently and identically from a probability distribution. Therefore, the rDMD matrix on the projected space is given by  $\hat{\mathbb{K}} = RY(RX)^\dagger$ , and if an eigenpair of  $\hat{\mathbb{K}}$  is given by  $(\lambda, \phi_L)$ , then the eigenpair of  $\mathbb{K}$  is given  $(\lambda, Y(RX)^\dagger\phi_L)$ . Algorithm 1 represents the major steps needed to estimate the eigenvalues and corresponding eigenvectors of the DMD operator with the random projection method. In addition, Figure 4 shows the details of the input–output variables of the algorithm, spatiotemporal decomposition of the data, and how to use the eigendecomposition of the Koopman operator to isolate and interpret the spatial features of a dynamical system.

The calculation of the projection matrix of a standard or exact DMD algorithm based on the SVD of the snapshot matrix  $X$  is needed to store a full high-resolution data matrix, which leads to memory issues. Our proposed rDMD algorithm can avoid these storage issues, because low-dimensional matrices  $X_L, Y_L$  obtained by matrix multiplications only need to store one row and one column of each matrix at a time. Additionally, this algorithm reduces the computational cost, since we only need to calculate the pseudo-inverse of comparatively lower dimensional matrix. The choice of the distribution of  $R$  can further reduce the computational cost [12].

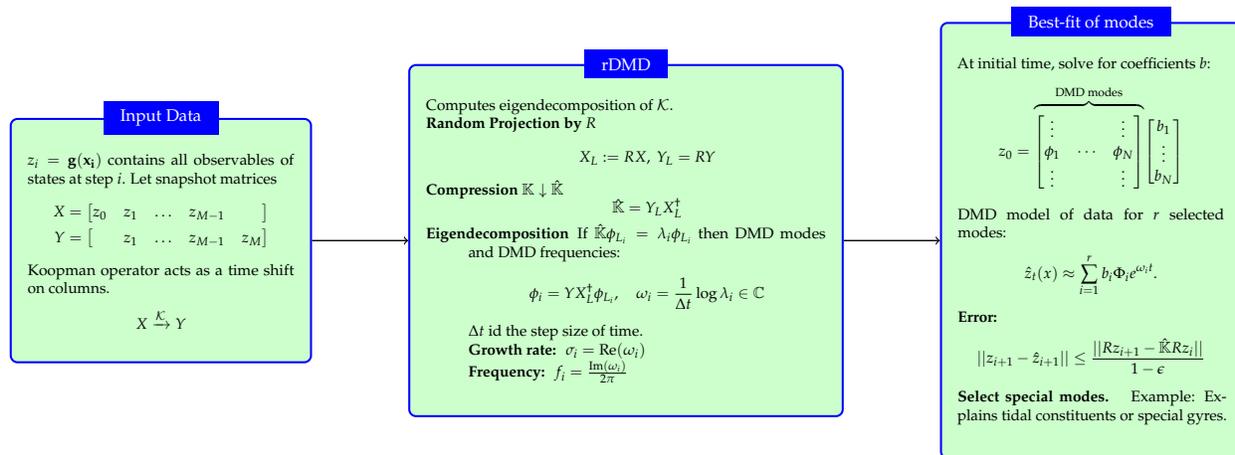
---

**Algorithm 1** Randomized DMD (rDMD). Figure 4 shows the details of the input and output variable.

---

**Data:**  $X, Y \in \mathbb{R}^{N \times M}$   
**Input:**  $\epsilon$   
 $L_0 = \frac{C \ln M}{\epsilon^2}$ ;  
 Choose  $L$  such that  $L \geq L_0$ ;  
 Construct a random matrix  $R = \frac{1}{\sqrt{L}}(r_{ij}) \in \mathbb{R}^{L \times N}$  such that  $r_{ij} \sim \mathcal{N}(0, 1)$ ;  
 Calculate  $X_L := RX, Y_L := RY$  ;  
 Calculate  $\hat{\mathbb{K}} = Y_L X_L^\dagger$ ;  
 $[\Lambda \ \Phi_L] = \text{eigs}(\hat{\mathbb{K}})$ ;  
**Result:**  $\text{diag}(\Lambda), Y X_L^\dagger \Phi_L$

---



**Figure 4.** The figure summarizes the rDMD algorithm, its input data, and the output variables. This also explains how to use the eigendecomposition of the Koopman operator to isolate and interpret the spatial features of a dynamical system. This figure is new from the previous manuscript.

One time step forecasting error for any given snapshot by using the rDMD algorithm can be bounded by using the JL theory.

**Theorem 3 (Error Bound).** Let  $z = \mathbf{g}(x(t))$ ,  $z' = \mathbf{g}(x(t+1)) \in \mathbb{R}^N$ . Error bound of estimating  $z'$  by using the rDMD as  $\hat{z}' = YX^\dagger R^\dagger Rz$  is given by

$$E(z'; L) := \|z' - \hat{z}'\| \leq \frac{\|Rz' - \hat{\mathbb{K}}Rz\|}{1 - \epsilon} := UB \tag{14}$$

with at least the probability of  $\mathcal{O}(1/M^2)$  for any  $0 < \epsilon < 1$  with  $L > \frac{C \log(M)}{\epsilon^2}$ .

**Proof.** Since  $\hat{\mathbb{K}} = RYX^\dagger R^\dagger$ , the rDMD acts on the projected vector, which can be rearranged as  $\hat{\mathbb{K}}Rz = R\hat{z}'$ . Therefore,

$$\|Rz' - \hat{\mathbb{K}}Rz\| = \|Rz' - R\hat{z}'\|.$$

Now we can apply the JL theory to attain the desired error bound.

$$(1 - \epsilon)\|z' - \hat{z}'\| \leq \|Rz' - R\hat{z}'\| = \|Rz' - \hat{\mathbb{K}}Rz\|.$$

Hence  $\|z' - \hat{z}'\| \leq \frac{\|Rz' - \hat{\mathbb{K}}Rz\|}{1 - \epsilon}$ .  $\square$

### 4. Results and Discussion

In this section, we demonstrate the theory of rDMD with a few examples. The first two examples consider the computation for known dynamics and demonstrate the error analysis. The final example demonstrates application in the field of oceanography and isolates the interesting features by rDMD, comparing the resulting modes with the exact DMD results.

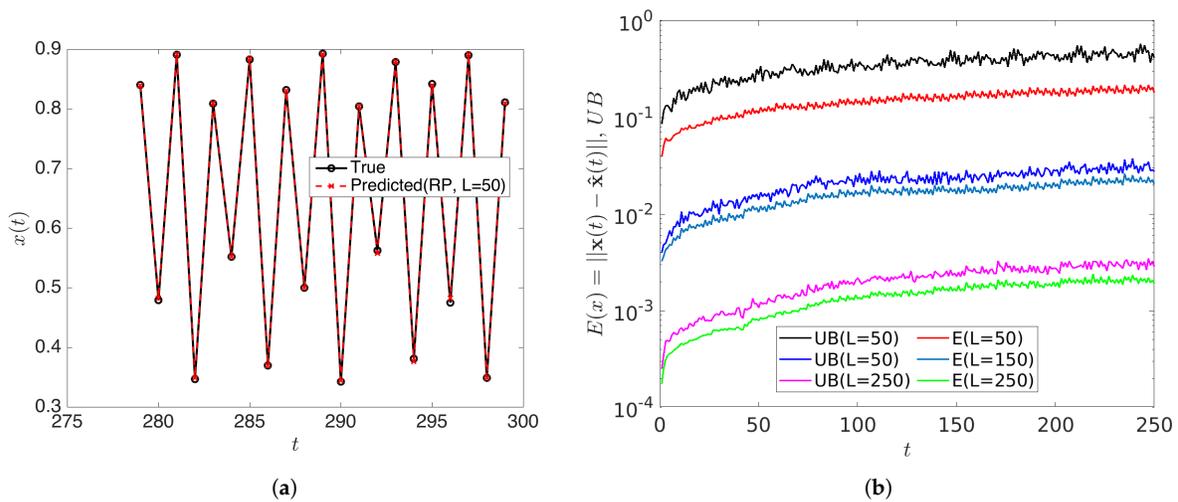
#### 4.1. Logistic Map

We first consider a dataset of 300 snapshots from a logistic map,

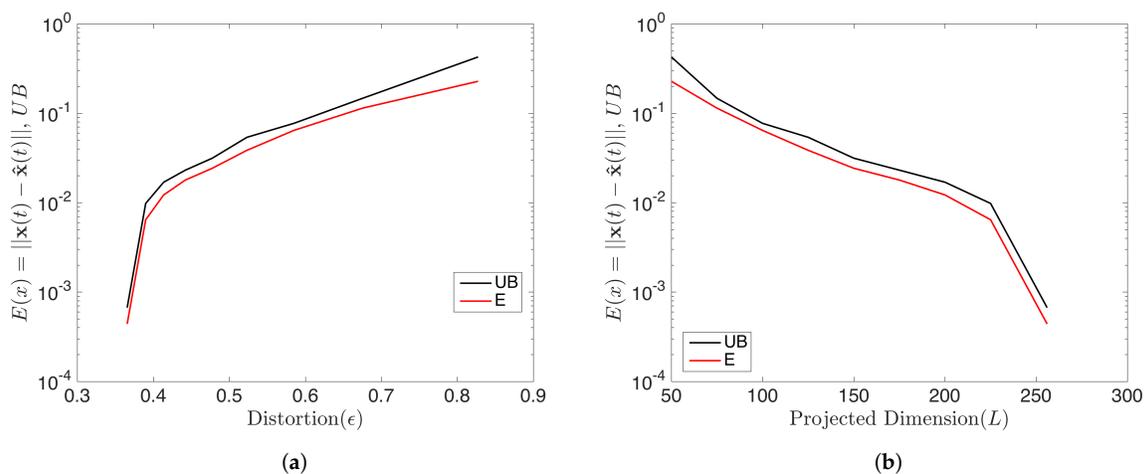
$$x_{n+1} = ax_n(1 - x_n)$$

with  $a = 3.56994$ . In this case, all initial conditions will converge to a period-256 orbit. Therefore the rank of the snapshot matrix with relatively high samples should be 256.

We forecasted the data by using the rDMD method and then analyzed the error of the prediction and compared it with the theoretical upper bound. With  $N = 5000$  initial conditions and  $M = 300$  samples, the dimension  $L$  of the projecting space can be chosen as  $L \geq \frac{C \ln(300)}{\epsilon^2} \approx \frac{34.22}{\epsilon^2}$  when  $C = 6$ . rDMD with projection into a 50 dimensional space can accurately forecast the time series data. (Figure 5 shows the original vs. predicted data for one trajectory.) Furthermore, Figure 6 demonstrates the bound of the error of the forecast explained in Equation (14) and how the error relates to the distortion parameter  $\epsilon$  (Figure 6a) and the dimension of the projected space (Figure 6b). Since the rank of the snapshot matrix is 256, any  $L \geq 256$  will perform very accurately. This example validates the error bound we discussed in Equation (14) and the error of the prediction depends on the error exhibited by the projected DMD operator and the distortion parameter ( $\epsilon$  or the projected dimension) from the JL theory.



**Figure 5.** (a) Shows the predicted data using the rDMD algorithm with projected dimension  $L = 50$  compared to the original data from logistic map  $x_{n+1} = 3.56994x_n(1 - x_n)$  for initial condition  $x(0) = 0.4967$ . Further, (b) shows the estimated error and theoretical upper bounds (Equation (14)) for some projected dimension  $L$ , and this example validates the theoretical bound.



**Figure 6.** This shows the prediction error of the logistic map  $x_{n+1} = 3.56994x_n(1 - x_n)$  by rDMD and its theoretical upper bounds(Equation (14)). (a) Represents the error with respect to the distortion  $\epsilon$  and (b) shows the error with the dimension of the projected space that will guarantee the bound for this example.

4.2. Toy Example: Demonstrates the Variable Separation and Isolating Dynamics

To demonstrate the variable separation and to isolate the spatial structures based on the time dynamics, we consider a toy example (motivated by [13]),

$$z(x, t) = \sum_{j=1}^{20} j \operatorname{sech}(0.1x + j) e^{i\gamma_j t} = \sum_{j=1}^{20} \Phi_j(x) T_j(t) \tag{15}$$

where  $\gamma_j$ 's are constants, and let  $\Phi_j(x) = j \operatorname{sech}(0.1x + j)$  and  $T_j(t) = e^{i\gamma_j t}$ .

Comparing this Equation (15) with decomposition Equation (8), the rDMD algorithm is expected to isolate 20 periodic modes by rDMD algorithm. The data set (snapshot matrix) for this problem is constructed by  $N = 20,000$  spatial grid points and  $M = 5001$  temporal grid points with  $\gamma_j = j$  (see Figure 7). As discussed in the previous section, if  $L \geq 20$  then those expected modes can be isolated and there exist eigenvalues  $\lambda_j$  of an rDMD operator, such that

$$\omega_j = \ln(\lambda_j) = ji$$

for  $j = 1, 2, \dots, 20$  (see Figure 8). Furthermore, we expect corresponding rDMD modes equal to spatial variables of the model, such that

$$b_j(x_0)\phi_j(x) = j \operatorname{sech}(0.1x + j) = \Phi_j(x).$$

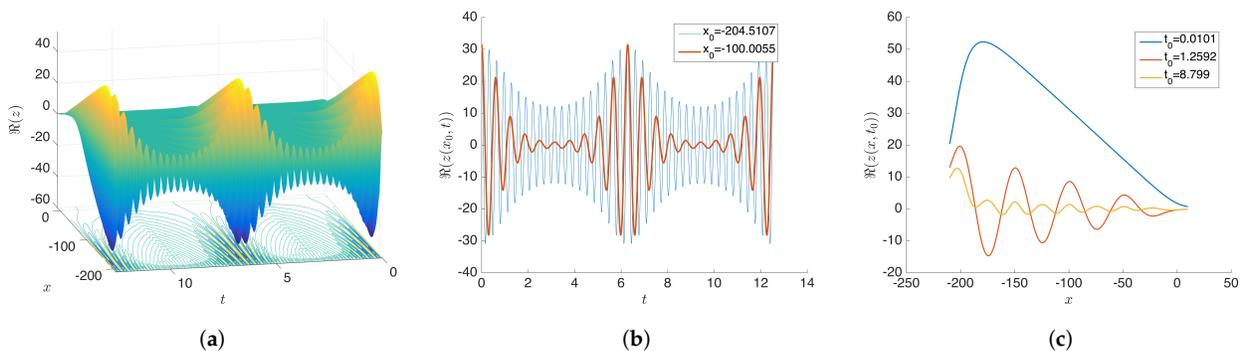


Figure 7. Original dataset constructed by Equation (15). (a) shows the  $\Re(z(x, t))$  plot for all  $x, t$  values at  $20,000 \times 5000$  grid points. (b) represents the time series plot for two initial conditions. (c) provides the snapshots of few different time points. Our goal is separate and isolate spatial variables  $\Phi_j(x) = j \operatorname{sech}(0.1x + j)$  and  $T_j(t) = e^{ijt}$  from the given data constructed by  $z(x, t)$ .

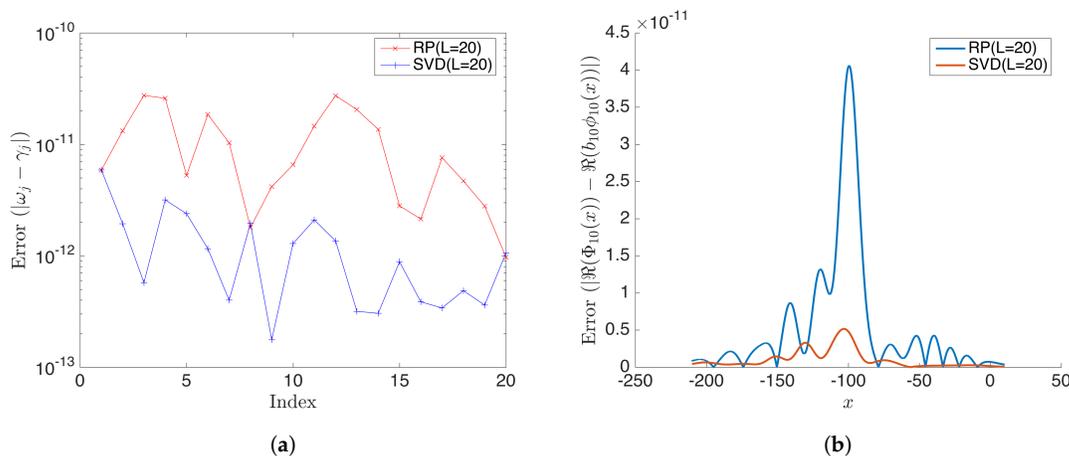


Figure 8. (a) Shows the absolute error for estimated eigenvalues from rDMD and the exact DMD when the dimension of the projected space  $L=20$ . (b) Shows the absolute error for the estimated 10<sup>th</sup> DMD mode by rDMD and exact DMD methods. In this case, both methods have very accurate results and error is less than  $10^{-10}$ .

As expected, we noticed that the calculated modes have negligible error when the dimension of projected space  $L \geq r = 20$ . Figure 8 shows the absolute error of eigenvalues and DMD modes. All modes behave similarly; here, we present mode 10 for demonstration purposes by the SVD-based exact DMD method and the random projection based rDMD method. Notice that errors of both methods are less than  $10^{-10}$  when  $L \geq r = 20$ .

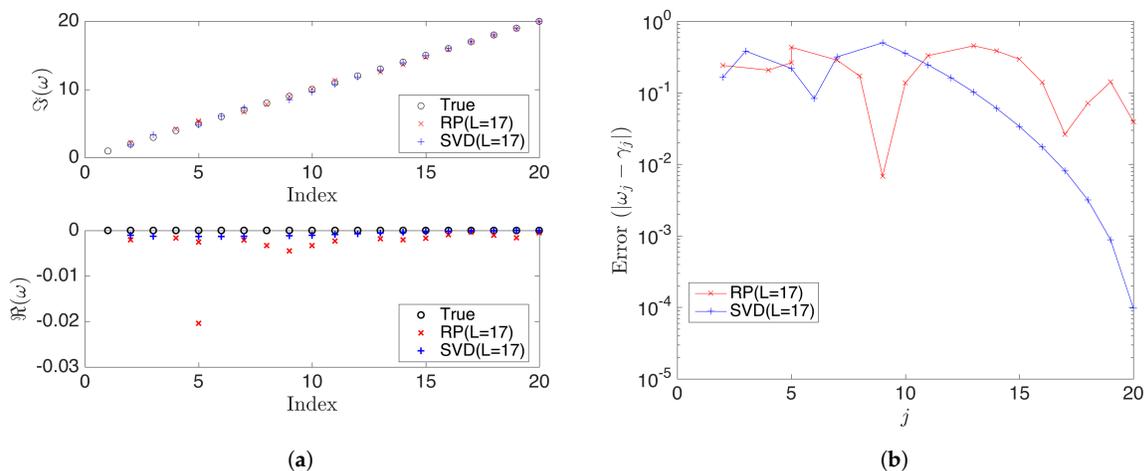
Further, we examine the case when the projected dimension  $L = 17 < r = 20$  and compare the results of rDMD with the exact DMD. We notice that both methods demonstrate similar errors and rDMD is almost as good as the SVD projection-based exact DMD (see Figures 9 and 10). When the number of actual modes( $r$ ) is larger than the dimension of the projected space ( $L$ ), the projected DMD operator only estimates the  $L$  number of modes, leading to both truncation errors and errors for eigenpair estimation based on the projected DMD operator. The  $L < r$  case can be modeled as,

$$z(x, t) = \sum_{j=1}^L \hat{b}_j \hat{\phi}_j(x) e^{\hat{\omega}_j i t} + E_{L+1} \tag{16}$$

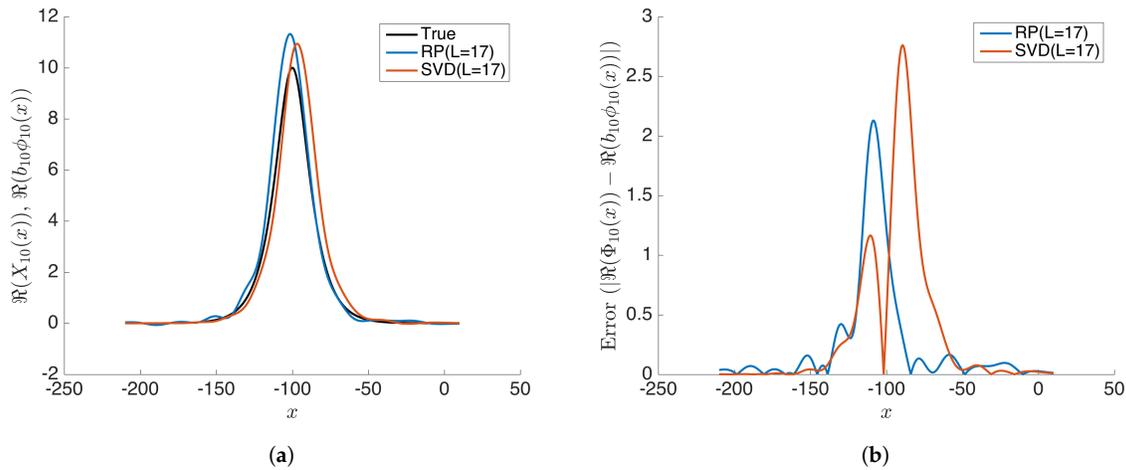
where  $E_{L+1} = \sum_{L=j+1}^m b_j \phi_j(x) e^{\omega_j i t}$  is the truncated error that also affects the estimation process of eigenpairs. Therefore, if  $L < r$ , then there exists an error in eigenvalues and eigenvectors calculated by any method based on the projected DMD. However, this example demonstrates that rDMD can provide the results as good as the SVD projection-based method with very low computational costs (See Table 2).

**Table 2.** Computational costs for the SVD-based exact DMD and random projection-based rDMD method for the data simulated by Equation (15). Computational costs of SVD for the high-dimensional snapshot matrix is relatively larger than random projection.

Method	Projected by	Computational Time (s)
Exact DMD	SVD	521.09
rDMD	Random Projection	2.35



**Figure 9.** (a) Compares the eigenvalues  $\omega_j = \ln \lambda_j$  calculated from rDMD (random projection(RP) with  $L = 17$ ) and exact DMD (SVD projection with  $L = 17$ ) methods with the expected true values  $\gamma_j = j i$ . Here,  $L = 17 < 20$  is the dimension of the projected space. (b) Shows the absolute error for the estimated eigenvalues from rDMD and exact DMD.



**Figure 10.** (a) Compares the modes  $b_{10}\phi_{10}(x)$  calculated from rDMD (random projection(RP) with  $L = 17$ ) and exact DMD (SVD projection with  $L = 17$ ) methods with the expected true values  $\Phi_{10}(x) = 10 \operatorname{sech}(0.1x + 10)$ . (b) Shows the absolute error for estimated values from rDMD and exact DMD.

### 4.3. Gulf of Mexico

In this example, we consider the data from Hybrid Coordinate Ocean Model (HYCOM) [14], which simulates the ocean data around the Gulf of Mexico. We used hourly surface velocity component  $(u, v)$  with  $1/25^0$  spatial resolution ( $N = 541 \times 347$  grid points) data for 10 days (240 h and  $M = 239$ ). Understanding the dynamics from the oceanographic data is an interesting application of DMD because those dynamics can be decomposed by tidal constituents. Hence, we are expected to isolate the dynamics associated with the tidal period; in other words, the final DMD mode selection is based on the period  $P_i = 2\pi / \operatorname{Im}(\ln(\lambda_i))$  of the modes(see Table 3). We constructed the snapshot matrix

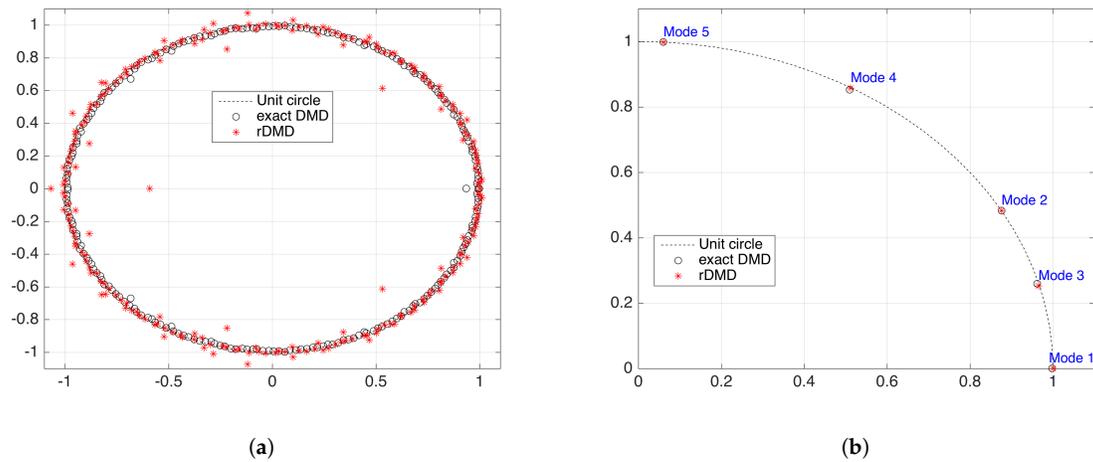
$$X = \begin{bmatrix} u \\ v \end{bmatrix} \tag{17}$$

by stacking the snapshots of velocity components  $(u, v)$  in each column to perform the DMD analysis.

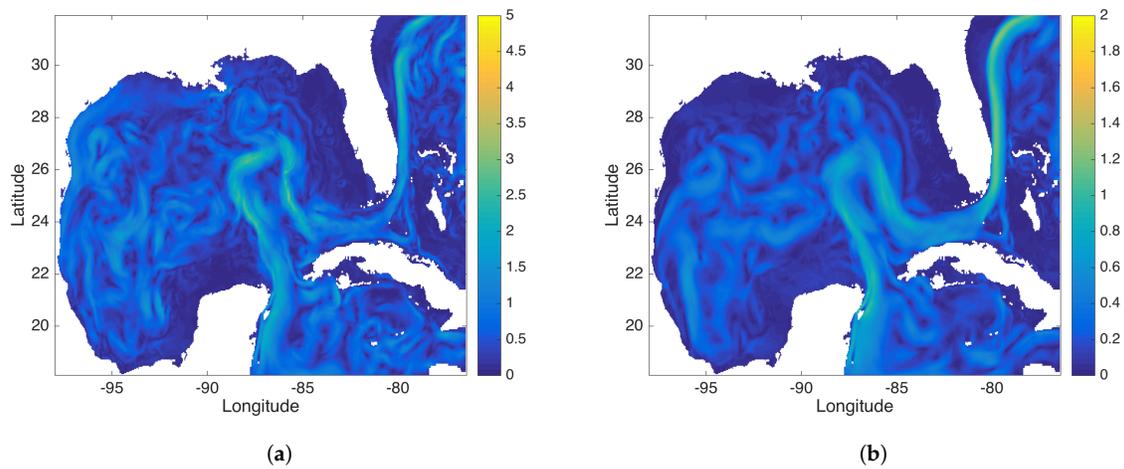
Figure 11 shows that most of the eigenvalues calculated from the SVD-based exact DMD and random projection-based rDMD are in agreement. Furthermore, eigenvalues that isolated the specific dynamics are almost equal. Additionally, Figures 12–14 show the spacial profile of those modes from the exact DMD and rDMD methods. Moreover, each mode clearly isolate the interesting oceanographic features (see Table 3) and both methods provide almost the same spacial structures(see Figures 12–14) as expected.

**Table 3.** DMD modes for the Gulf of Mexico data set. Modes are selected based on the association to the tidal periods.

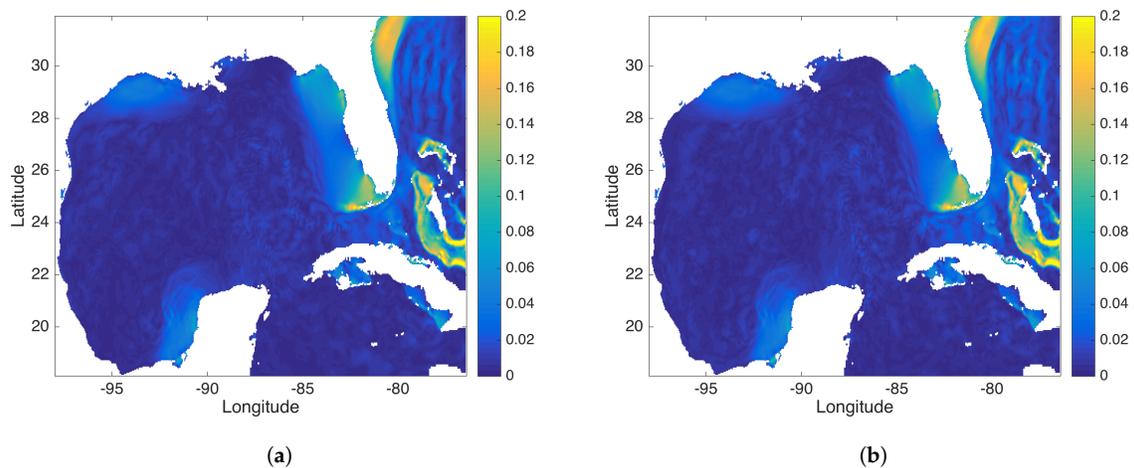
Mode	Period (h)		Associated Feature
	DMD	rDMD	
1	$\infty$	$\infty$	Gulf stream around the GOM (see Figure 12)
2	12.47	12.47	Semi-diurnal tidal constituents (see Figure 13)
3	23.85	24.56	Diurnal tidal constituents (see Figure 14)
4	6.08	6.07	Second harmonic to semi-diurnal tidal constituents (see Figure 14)
5	4.16	4.17	Third harmonic to semi-diurnal tidal constituents (see Figure 14)



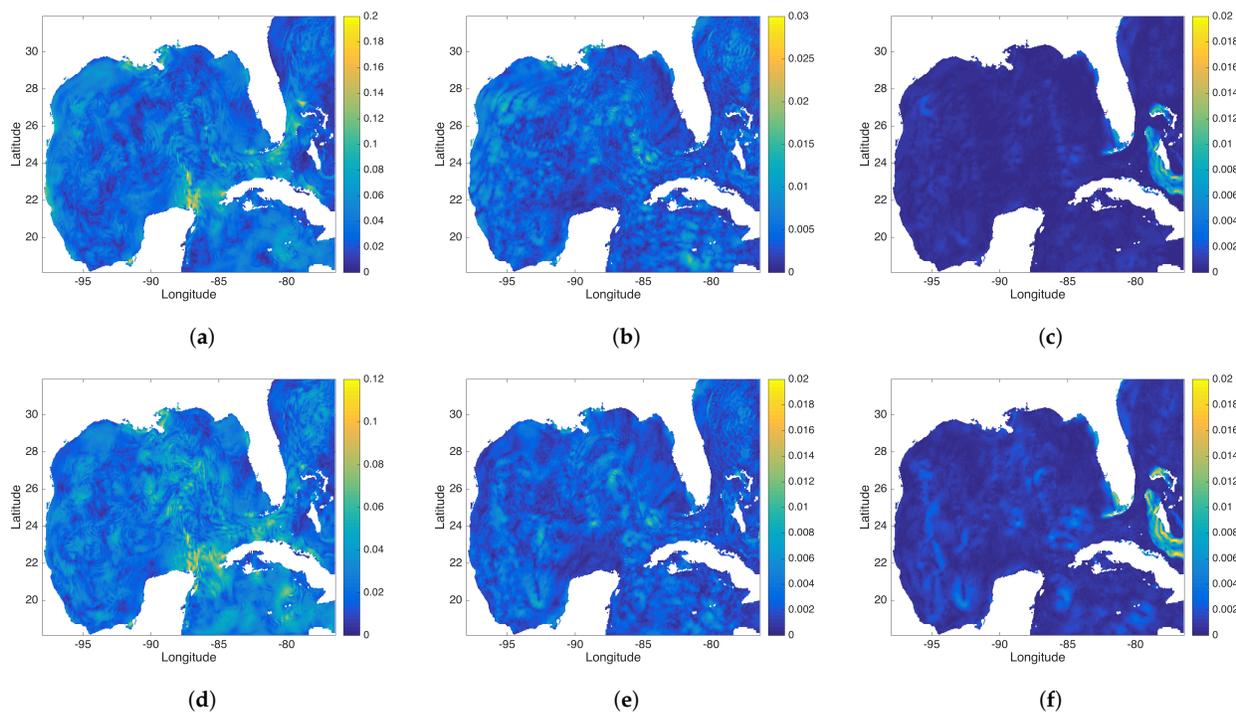
**Figure 11.** Eigenvalues  $\lambda_i$  calculated from the exact DMD and rDMD methods. (a) Full spectrum of the two methods with projected space dimension  $L = 239$  and (b) shows the first five modes. The mode selection is based on the comparison of the tidal periods with period of the DMD modes.



**Figure 12.** This figure compares the (a) DMD and (b) rDMD background mode identified by data from the Gulf of Mexico (GOM). This background mode captures the ocean current passing through the GOM.



**Figure 13.** This figure compares the (a) DMD and (b) rDMD mode associated with the M2 tidal frequency. This mode capture the “red tides”.



**Figure 14.** (a)–(c) represent the exact DMD modes 3, 4, and 5 and (d)–(e) show the rDMD modes 3, 4, and 5. Mode 3 is a diurnal mode with period 23.85 h for the exact DMD case and 24.56 h for the rDMD case. Modes 4 and 5 are associated with the second and third harmonic of semi-diurnal tidal constituents, respectively.

Notice that the dimension of the snapshot matrix is  $375,454 \times 239$  and the SVD calculation of this matrix is more costly for both computation and storage. On the other hand, random projection only performs by the matrix multiplication, which could be done at a relatively low cost. Hence, we achieve almost the same results by using the random projection method, at relatively lower computational and storage costs.

## 5. Conclusions

We demonstrated that our rDMD can achieve very accurate results with low-dimensional data embedded in a high-dimensional observable space. Recent analytic technology from the concepts of high-dimensional geometry of data, and concentration of measure, have shown—surprising, if not initially intuitively—that even random projection methods can be quite powerful and capable. Here, in the setting of DMD methods approximating and projecting the action of a Koopman operator, we show that randomized projection can be developed and analyzed rigorously by the Johnson–Lindenstrauss theorem formalism, showing a powerful and simple approach. We provided a theoretical framework and experimental results to address those issues raised from SVD-based methods by introducing our new rDMD algorithm. The theoretical framework is based on generalizing the SVD-based concept as a projection of high-dimensional data into a low-dimensional space. We proved that eigenpairs of DMD in the original space can be estimated by using any rank  $L$  projection matrix  $P$ . Being able to estimate eigenpairs allowed us to use the powerful and simple Johnson–Lindenstrauss lemma and the random projection method, allowing us to project data with matrix multiplication. Therefore, our proposed random projection-based DMD (rDMD) can estimate eigenpairs of the DMD operator with low storage and computational costs. Further, the error of the estimation can be controlled by choosing the dimension of the projected space; we demonstrated this error bound through the “logistic map” example.

DMD promises the separation of the spatial and time variables from data. Hence, we experimentally demonstrated how well the rDMD algorithm performed this task by

a toy example. Notice that the number of those isolated modes ( $m$ ) are relatively (i.e., to spatial and temporal resolution) low in practical applications. If  $m \ll M$ , then the rank of the data matrix is much lower, and those eigenvalues and vectors of interest can be estimated accurately by projecting data into the much lower dimensional space  $L \geq m$ . The SVD projection-based exact DMD method still needs to calculate the SVD of a high-dimensional (roughly  $10^{10} \times 10^3$ ) data matrix, while rDMD only requires multiplying the data matrix by a much lower-dimensional projection matrix. Furthermore, we noticed that both exact and random DMD methods experience similar errors. However random projection is much faster and needs less space for the calculations. We also demonstrate that practical applications provide similar results by using oceanographic data from the Gulf of Mexico.

Since the size of the DMD matrix is enormous in those applications (this could be roughly  $10^{10} \times 10^{10}$ ), the eigenpairs must be estimated by projecting data into the low-dimensional space. Estimating eigenvalues and eigenvectors of a DMD operator using a high-dimensional snapshot data matrix (in applications, this could be  $10^{10} \times 10^3$ ) with existing SVD-based methods is expensive. The computational efficiency of the rDMD led to a new path of the current Koopman analysis. It allows using more observable variables in the data matrix without need for much extra computational power. Hence, state variables and more non-linear terms can be used in an analysis, with low costs, to improve the Koopman modes. The JL theory can be adopted further into the field of numerical methods of the Koopman theory. As a next step, we can use the random projection concept in the extended DMD and kernel DMD methods.

**Author Contributions:** Conceptualization, S.S. and E.M.B.; data curation, S.S. and E.M.B.; formal analysis, S.S. and E.M.B.; funding acquisition, E.M.B.; investigation, S.S. and E.M.B.; methodology, S.S. and E.M.B.; project administration, E.M.B.; resources, S.S. and E.M.B.; software, S.S. and E.M.B.; supervision, E.M.B.; validation, S.S. and E.M.B.; visualization, S.S. and E.M.B.; writing – original draft, S.S. and E.M.B.; writing – review and editing, S.S. and E.M.B.

**Funding:** E.B. gratefully acknowledges funding from the Army Research Office W911NF16-1-0081 (Dr. Samuel Stanton) as well as from DARPA.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

DMD	dynamic mode decomposition
rDMD	randomized dynamic mode decomposition
SVD	singular value decomposition
RP	random projection
JL	Johnson–Lindenstrauss
MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	directory of open access journals

## References

- Schmid, P.J. Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **2010**, *656*, 5–28.
- Rowley, C.W.; Mezić, I.; Bagheri, S.; Schlatter, P.; Henningson, D.S. Spectral analysis of nonlinear flows. *J. Fluid Mech.* **2009**, *641*, 115–127.
- Tu, J.H.; Rowley, C.W.; Luchtenburg, D.M.; Brunton, S.L.; Kutz, J.N. On dynamic mode decomposition: Theory and applications. *J. Comput. Dyn.* **2014**, *1*, 391–421.
- Dasgupta, S. Experiments with Random Projection. In Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI '00), Stanford, CA, USA, 30 June–3 July 2000; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2000; pp. 143–151.

5. Johnson, W.B.; Lindenstrauss, J.; Schechtman, G. Extensions of Lipschitz maps into Banach spaces. *Isr. J. Math.* **1986**, *54*, 129–138.
6. Bingham, E.; Mannila, H. Random Projection in Dimensionality Reduction: Applications to Image and Text Data. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01)*; Association for Computing Machinery: New York, NY, USA, 2001; pp. 245–250.
7. Koopman, B.O. Hamiltonian Systems and Transformation in Hilbert Space. *Proc. Natl. Acad. Sci. USA* **1931**, *17*, 315–318.
8. Bollt, E.M. Geometric considerations of a good dictionary for Koopman analysis of dynamical systems: Cardinality, “primary eigenfunction,” and efficient representation. *Commun. Nonlinear Sci. Numer. Simul.* **2021**, *100*, 105833.
9. Chen, K.K.; Tu, J.H.; Rowley, C.W. Variants of Dynamic Mode Decomposition: Boundary Condition, Koopman, and Fourier Analyses. *J. Nonlinear Sci.* **2012**, *22*, 887–915.
10. McQuarrie, S.A.; Huang, C.; Willcox, K.E. Data-driven reduced-order models via regularised Operator Inference for a single-injector combustion process. *J. R. Soc. N. Z.* **2021**, *51*, 194–211.
11. Pan, S.; Arnold-Medabalimi, N.; Duraisamy, K. Sparsity-promoting algorithms for the discovery of informative Koopman-invariant subspaces. *J. Fluid Mech.* **2021**, *917*, <https://doi.org/10.1017/jfm.2021.271>.
12. Achlioptas, D. Database-friendly random projections: Johnson–Lindenstrauss with binary coins. *J. Comput. Syst. Sci.* **2003**, *66*, 671–687. Special Issue on PODS 2001.
13. Kutz, N.J.; Brunton, S.L.; Brunton, B.W.; Proctor, J.L. *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2016.
14. HYCOM + NCODA Gulf of Mexico 1/25° Analysis. 2021. Available online: <http://xxx.lanl.gov/abs/https://www.hycom.org/data/gomuOpt04/expt-90pt1m000> (accessed on 3 November 2021).