
ON PROPER ORTHOGONAL DECOMPOSITION AND DYNAMIC MODE DECOMPOSITION FOR FLUID-DYNAMICS

A PREPRINT

Zeyad M. Manaa

Aerospace Engineering Department

King Fahd University of Petroleum and Minerals
Dhahran, KSA

Mohamed R. Elbalshy

Aerospace Engineering Department

King Fahd University of Petroleum and Minerals
Dhahran, KSA

ABSTRACT

This project focuses on applying Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD) to analyze complex flow field (e.g. flow past a cylinder). The goal of this project is to explore the potential of these techniques in analyzing large and complex datasets resulting from complex dynamical systems, especially in fluid mechanics using model reduction techniques. Additionally, the project aims to establish a connection between mathematical concepts, including Singular Value Decomposition (SVD), Fourier Series and Fourier Transform, Partial Differential Equations (PDEs) and their practical applications in the aerospace field.

Keywords Proper Orthogonal Decomposition · Dynamic Mode Decomposition · Singular Value Decomposition · Aerodynamics · Coherent Structures

1 Introduction

The next great era may produce a method of understanding the qualitative content of equations ... today we cannot ... today we can not see that the water flow equations contain such things as turbulence ...

Feynman - 1977

The advancements on the experimental work combined with the direct numerical simulations (DNS) have led to an abundance of data that are hard to interpret. Simple aerodynamic configurations and other likely physical phenomena exhibit complex behavior with a myriad of temporal and spatial structures. Fortunately, most of these complex aerodynamic behaviours are governed with rigid mathematical formulation that describes the flow behavior: the *Navier-Stokes* equations. But, the complexity and ambiguity of the *Navier-Stokes* equations do not allow us to further explore those phenomena analytically in depth. On the other hand, a growing body of literature is interested in decomposing those data into an interpretable structures. Some methods discussed in the literature are the well known POD, Galerkin projection, and more recent techniques developed for linear systems, such as balanced truncation and DMD - just to name a few. Also, variants of those techniques are developed as an extension for them to account for more complex scenarios. Readers are referred to [1–3] for a comprehensive reviews of the recent advances.

We will focus herein in the mathematical establishment for both POD [4] and DMD [5]. Then, a case study will be developed in order to link between the mathematical abstraction and applications.

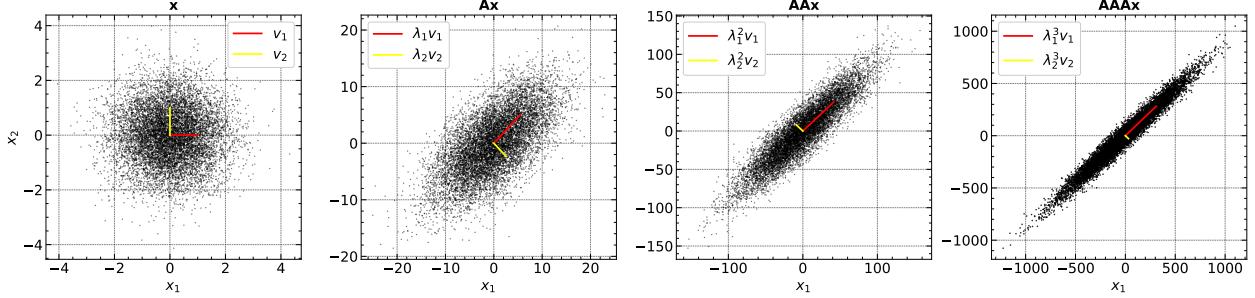


Figure 1: Collection of random entries x_1 and x_2 denoted as \mathbf{x} stretched in the direction of dominant eigenvector \mathbf{v}_1 with iterative operation A^κ , which has eigenvalues of $\lambda_1 = 7.4$ and $\lambda_2 = -3.4$.

2 Methodology

2.1 Eigen and Singular Value Decomposition

2.1.1 Eigenvalue Decomposition

The eigenvalues and eigenvectors of a matrix give information about the direction in which vectors can expand or shrink. A matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ has an eigenvector $\mathbf{v} \in \mathbb{C}^n$ and an eigenvalue $\lambda \in \mathbb{C}$ if they satisfy,

$$\mathbf{Av} = \lambda\mathbf{v} \quad (1)$$

The set of all eigenvalues of \mathbf{A} is called the spectrum of \mathbf{A} . The eigenvectors are unique up to a complex scalar (i.e. if \mathbf{v} is an eigenvector, $\beta\mathbf{v}$ is also an eigenvector; where $\beta \in \mathbb{C}$).

Although the preceding expression in Eq. 1 is simple, it has significant consequences as it tells information about the effect of premultiplying a vector \mathbf{v} by a linear transformation \mathbf{A} . It tells that the effect of this transformation is captured by only multiplying the same vector by scalar λ , which is the eigenvalue associated with the direction. This can be depicted in Fig. 1 by showing an iterative premultiplication by a matrix \mathbf{A} which stretches the \mathbf{x} vector in the direction of dominant eigenvector.

If the matrix \mathbf{A} has n linearly independent eigenvectors \mathbf{v}_i with corresponding eigenvalues λ_i with $i = 1, 2, \dots, n$, we can write,

$$\mathbf{Av} = \mathbf{V}\Lambda \quad (2)$$

where $\mathbf{V} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \dots \ \mathbf{v}_n] \in \mathbb{C}^{n \times n}$ and $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n) \in \mathbb{C}^{n \times n}$. Postmultiplying Eq. 2 by \mathbf{V}^{-1} yields,

$$\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^{-1} \quad (3)$$

This is called *eigenvalue decomposition*, and for this to hold, \mathbf{A} should have a full rank eigenvector matrix¹

2.1.2 Singular Value Decomposition

The singular value decomposition (SVD) is a matrix factorization that has many uses, especially for dimensionality reduction, image compression, low-rank approximation – just to name a few. The issue of high dimensionality is a prevalent obstacle in the handling of data obtained from intricate systems [8]. These systems entail voluminous collections of measured data, encompassing various forms of data such as audio, image, or video data. The data may also arise from physical systems, such as neural recordings from a brain or fluid velocity measurements from a simulation or experiment. Many complex systems display predominant patterns in the data that may be distinguished by a low-dimensional attractor or manifold [9, 10].

SVD reveals how a rectangular matrix stretches and rotates a vector. For instance, by premultiplying unit vectors $\phi_i \in \mathbb{R}^m$ with a rectangular matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ as shown in Fig. 2, an ellipse (ellipsoid) with semiaxes represented by the unit vectors $p\sigma_i$ and magnitudes σ_i is obtained. Singular values capture the stretching amount imposed by matrix \mathbf{X} along the axes of the ellipse.

¹In such case \mathbf{A} called non-defective or diagonalizable, otherwise, the canonical Jordan form can be used to get the eigenvalue decomposition as $\mathbf{A} = \mathbf{V}\mathbf{J}\mathbf{V}^{-1}$; where \mathbf{J} is the canonical Jordan form as found in [6, 7].

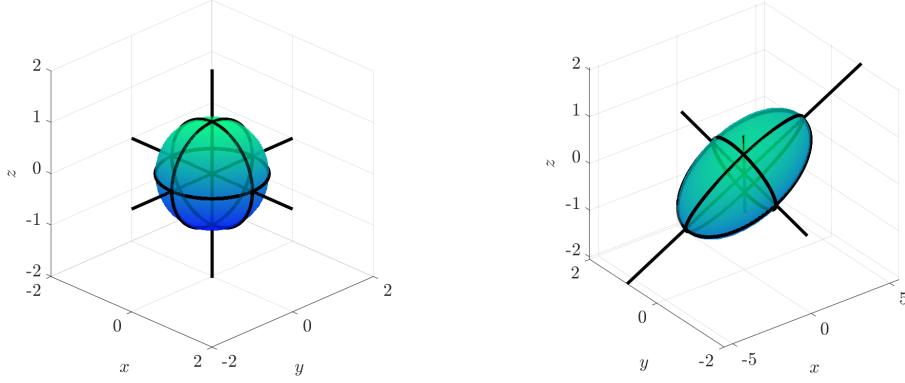


Figure 2: SVD transforms a sphere of radius one, represented by the right singular vectors ϕ_i , into an ellipsoid with semiaxes determined by the left singular vectors ψ_i and size denoted by the singular values σ_i . This transformation is depicted graphically, and the example considered here is $\mathbf{Y} \in \mathbb{R}^{3 \times 3}$.

Description Starting from the generalization of eigenvalue problem for complex rectangular data matrix $\mathbf{Y} \in \mathbb{C}^{n \times m}$, $\phi_i \in \mathbb{C}^m$, and $\psi_i \in \mathbb{C}^n$ ends up with,

$$\mathbf{Y}\phi_i = \sigma_i\psi_i \quad (4)$$

This can be formulated in a matrix format as follows,

$$\mathbf{Y}\Phi = \Psi\Sigma \quad (5)$$

where $\Psi \in \mathbb{C}^{n \times n}$ and $\Phi \in \mathbb{C}^{m \times m}$ are both unitary matrices² and $\Sigma \in \mathbb{R}^{n \times m}$ is a diagonal matrix with $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$ along its diagonal.

Postmultiplying Eq. 5 by $\Phi^{-1} = \Phi^\top$ yields,

$$\mathbf{Y} = \Psi\Sigma\Phi^\top \quad (6)$$

which is known as the singular value decomposition [8, 11].

The matrix σ can be written as,

$$\Sigma = \begin{bmatrix} \Sigma^r & 0 \\ 0 & 0 \end{bmatrix} \quad (7)$$

where $\Sigma^r \in \mathbb{C}^{r \times r}$, and $\Sigma^r = \text{diag}(\sigma_1 \dots \sigma_r)$ sorted in a descending order as discussed above. From this, Eq. 6 can be rewritten in form of,

$$\mathbf{Y} = \tilde{\Psi}\tilde{\Sigma}\tilde{\Phi}^\top \quad (8)$$

where $\tilde{\Psi} \in \mathbb{C}^{n \times r}$, $\tilde{\Phi} \in \mathbb{C}^{m \times r}$, and $\tilde{\Sigma} = \Sigma^r \in \mathbb{C}^{r \times r}$. This might be referred to as the truncated SVD.

Note on the relationship between the SVD and Eigenvalue Decomposition The eigenvalue and singular value decompositions have a close relationship. The left and right singular vectors of \mathbf{Y} are the orthonormal eigenvectors of $\mathbf{Y}\mathbf{Y}^\top$ and $\mathbf{Y}^\top\mathbf{Y}$, respectively. Additionally, the nonzero singular values of \mathbf{A} correspond to the square roots of the nonzero eigenvalues of $\mathbf{Y}^\top\mathbf{Y}$ and $\mathbf{Y}^\top\mathbf{Y}$. Thus, instead of performing the computationally expensive SVD, the eigenvalue decomposition can be used on $\mathbf{Y}\mathbf{Y}^\top$ or $\mathbf{Y}^\top\mathbf{Y}$ to obtain the singular vectors and singular values of \mathbf{A} . Since flow field data typically yields a high-dimensional rectangular matrix, the smaller of the square matrices of $\mathbf{Y}^\top\mathbf{Y}$ and $\mathbf{Y}\mathbf{Y}^\top$ is usually chosen to perform the decomposition in a computationally efficient manner. This property is utilized in some of the decomposition methods discussed below.

Then $\{\psi_i\}_{i=1}^j$ and $\{\phi_i\}_{i=1}^j$ are the eigenvectors of $\mathbf{Y}\mathbf{Y}^\top$ and $\mathbf{Y}^\top\mathbf{Y}$ respectively with eigenvalues of $\lambda_i = \sigma_i^2 > 0$, $\forall 1 < i < r$.

²In the realm of linear algebra, the condition for unitary matrices is that they are invertible and satisfy the property $\mathbf{U}^* = \mathbf{U}^{-1}$ and $\mathbf{V}^* = \mathbf{V}^{-1}$, where the symbol * denotes the conjugate transpose operation, it will be used with \top interchangeably in the scope of this paper.

2.2 Proper Orthogonal Decomposition

This section provides an introduction to the proper orthogonal decomposition (POD) and its application in \mathbb{R}^m . The POD is utilized to approximate high-dimensional data sets such as solutions obtained from a direct numerical simulation, while preserving essential information. The method is also employed in various fields such as stochastics, data analysis, statistics, meteorology, and geophysics under different names such as Karhunen-Loëve decomposition [12], Principal Component Analysis [13], Hotelling transform [14], and empirical orthogonal functions [15]. Interested readers may refer to [9, 16–18] for more comprehensive introductions to the POD.

2.2.1 Description

POD is a mathematical technique that is also used to extract modes by optimizing the mean square of the field variable. It was first introduced by Lumley [4] as a way to extract coherent structures from turbulent flow fields. This method decomposes a set of data into a minimal number of basis functions or modes to capture as much energy as possible, and it is used in various fields. The roots of POD can be traced back to the matrix diagonalization technique of the mid-19th century, which is related to SVD discussed in Sec. 2.1.2.

In the application of the POD to get the optimal minimal basis vectors or functions in the Hilbert space we assume to have a vector quantity $\mathbf{f}(\mathbf{x}, t)$ which can describe a vector field quantity in our case (i.e. velocity field) or even a scalar quantity (i.e. pressure). Here, \mathbf{x} is considered a spatial vector. Assume that $\mathbf{f}(\mathbf{x}, t)$ has a temporal mean denoted $\bar{\mathbf{f}}(\mathbf{x})$. We can decompose the vector field as

$$\mathbf{f}(\mathbf{x}, t) - \bar{\mathbf{f}}(\mathbf{x}) = \sum_i a_i \psi_i(\mathbf{x}, t) \quad (9)$$

where $\psi_i(\mathbf{x}, t)$ and a_i represent the basis vectors (i.e. modes) and expansion coefficient respectively. One may note that the expansion takes the form of the generalized Fourier series³ for some set of basis functions $\psi_i(\mathbf{x}, t)$. In the framework of POD, we seek the optimal set of basis functions or vectors for given quantity.

However, Eq. 9 is considered a classical way to express the POD. Contemporary uses of modal decompositions have aimed to separate space and time, and therefore require only spatial modes⁴. In this regard, the previous equation, Eq. 9 can be expressed as

$$\mathbf{f}(\mathbf{x}, t) - \bar{\mathbf{f}}(\mathbf{x}) = \sum_i a_i(t) \psi_i(\mathbf{x}) \quad (10)$$

2.2.2 Algorithm

There are three main approaches to discuss the POD algorithm⁵; the *spatial (classical) POD* method, the *the snapshot POD* method, and the *SVD-POD* method. Here we will give a brief description of these three methods.

The spatial POD method Recalling that the objective of POD analysis is to find the optimal basis with which a vector field can be expanded with maximum energy and least number of modes. So, we seek the vectors $\psi_i(\mathbf{x})$ in Eq. 10 that can satisfy the preceding conditions while expanding the vector field $\mathbf{f}(\mathbf{x})$.

One way to solve this problem, as discussed by Eckart et. al [19], is to get the eigenvectors ψ_i and the eigenvalues λ_i from solving the eigen value problem with the covariance matrix of the vector $\mathbf{y}(t)$ denoted as \mathbf{R}

$$\mathbf{R}\psi_i = \lambda_i \psi_i, \quad \psi_i \in \mathbb{R}^n, \quad \lambda_1 \geq \dots \geq \lambda_n \geq 0 \quad (11)$$

where \mathbf{R} can be mathematically expressed as

$$\mathbf{R} = \sum_{j=1}^m \mathbf{y}_j \mathbf{y}_j^\top = \mathbf{Y} \mathbf{Y}^\top \in \mathbb{R}^{n \times n} \quad (12)$$

where the matrix \mathbf{Y} represents the data matrix with m snapshots stacked into

$$\mathbf{Y} = [\mathbf{y}_1 \ \mathbf{y}_2 \ \dots \ \mathbf{y}_m] \in \mathbb{R}^{n \times m} \quad (13)$$

³The proof for this conclusion is found in appendix A.1

⁴Here, a separation of variable has taken place which may not be suitable for all problems. So, choosing the formulation should depend on the problem itself as discussed by [9].

⁵We will consider that the data matrix are real-numbered. In other words, the discussion will be limited to \mathbb{R} .

The eigenvectors ψ_i retrieved from Eq. 11 are the POD modes which by nature are orthonormal. That means that they satisfy

$$\langle \psi_k, \psi_l \rangle = \delta_{kl} = \begin{cases} 1, & k = l \\ 0, & k \neq l \end{cases} \quad j, k = 1, \dots, n \quad (14)$$

Further, the eigenvalues λ_i tells information about how much energy is contained on each eigenmode ψ_i in L_2 sense.

The snapshot POD method This is an efficient method when the spatial size of the data matrix is very large. In most practical cases in aerodynamics and fluid mechanics the number of degrees of freedom in the spatial discretization of the PDE, denoted by m , is significantly larger than the number of time steps n . Consequently, the correlation matrix becomes very large, making the *spatial POD* method for getting the eigenfunctions computationally expensive and even impossible. As a result, in most applications, the method of snapshots, which was introduced by Sirovich [20–22], is used. To implement this method, we need to solve the $n \times n$ eigenvalue problem.

$$\mathbf{Y}^\top \mathbf{Y} \phi_i = \lambda_i \phi_i, \quad \forall 1 \leq i \leq \kappa \in \mathbb{N} \quad (15)$$

whose eigenvectors are the right singular vectors $\{\phi_i\}_{i=1}^\kappa$ and the eigenvalues are $\lambda_i = \sigma_i^2$ for $1 \leq i \leq \kappa$. From this the POD basis $\{\psi_i\}_{i=1}^\kappa$ can be calculated through the relation

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} \mathbf{Y} \phi_i \quad \forall 1 \leq i \leq \kappa \in \mathbb{N} \quad (16)$$

SVD-POD The singular value decomposition (SVD) can be utilized to factorize a rectangular matrix into its left and right singular vectors. Mathematically, the data matrix \mathbf{Y} can be directly decomposed using SVD as found in Eq. 8. Further, as discussed in section 2.1.2, SVD can be directly performed on \mathbf{Y} to determine the POD modes Ψ .

Although the method of snapshots is often the preferred approach for dealing with large data sets, it is worth noting that the SVD-based technique for determining POD modes is known for its robustness against roundoff errors [23].

Finally, the eigenvalues λ_i can be used to choose how many modes are needed to represent the energy/fluctuations in the flow field. Let κ be the number of modes needed to represent the energy of the system, so a proper choice for this parameter may be formulated as follows⁶,

$$\mathcal{E}(\kappa) := \frac{\sum_{i=1}^{\kappa} \lambda_i}{\sum_{i=1}^r \lambda_i} \geq \delta_{\mathcal{E}} \quad (17)$$

where $\mathcal{E}(\kappa)$ is called energy ratio, and the threshold $\delta_{\mathcal{E}} \in [0, 1]$. In real life scenarios we choose $\delta_{\mathcal{E}} \approx 1$ (e.g. $\delta_{\mathcal{E}} = 0.99$).

Below, a summary of the POD approach that might be used in the context of aerodynamic analysis is presented in Algorithm 1.

2.3 Dynamic Mode Decomposition

This section presents an introduction to dynamic mode decomposition (DMD) and its applications in many fields such as fluid dynamics, neuroscience and epidemiology. The DMD method is a matrix decomposition technique that does not require equations and instead relies on data analysis. This decomposition method is different from POD and balanced POD because it provides growth rates and frequencies for each mode, which are determined by the eigenvalues' magnitude and phase. If the data is generated by a linear dynamical operator, the method can recover the leading eigenvalues and eigenvectors. If the data is periodic, the decomposition is similar to a discrete Fourier transform (DFT) in time.

2.3.1 Description

It may appear unlikely that a nonlinear system can be described using eigenvalues and mode superposition, which require a linear operator. However, Dynamic Mode Decomposition (DMD) has demonstrated as in [24] that it is possible by utilizing a spectral analysis of the Koopman operator, a linear and infinite-dimensional operator that captures the evolution of observables for any system, even nonlinear ones.

The DMD method is a combination of techniques that reduce spatial dimensions, like the POD, with Fourier transforms in time [25, 26]. This allows for spatial modes to be associated with a specific temporal frequency and growth/decay

⁶The summation here is operated upon the eigenvalues λ_i themselves, not the singular values σ_i .

Algorithm 1 POD basis in \mathbb{R}^m

Input: a data matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ with snapshots $\{\mathbf{y}\}_{i=1}^n \subset \mathbb{R}^m$ and a threshold $\delta_{\mathcal{E}} \in [0, 1]$
Output: POD basis $\{\psi\}_{i=1}^{\kappa} \subset \mathbb{R}^m$ and eigenvalues $\{\lambda\}_{i=1}^{\kappa}$

- 1: **if** $m \approx n$ **then**
- 2: Compute singular value decomposition $[\Psi, \Sigma, \Phi] = \text{svd}(\mathbf{Y})$
- 3: Compute $\kappa = \min\{\kappa \in \mathbb{N} \mid \mathcal{E}(\kappa) = \sum_{i=1}^{\kappa} \Sigma_{ii}^2 / \sum_{i=1}^r \Sigma_{ii}^2 \geq \delta_{\mathcal{E}}, \forall 1 \leq \kappa \leq r\}$.
- 4: Set $\lambda_i = \Sigma_{ii}^2$ and $\psi_i = \Psi_{:,i} \in \mathbb{R}^m, \forall 1 \leq i \leq \kappa$ $\triangleright i = 0, 1, \dots$
- 5: **else if** $m \ll n$ **then**
- 6: Compute eigen value decomposition $[\Psi, \Lambda] = \text{eig}(\mathbf{Y}^\top \mathbf{Y})$ $\triangleright \mathbf{Y}^\top \mathbf{Y} \in \mathbb{R}^{m \times m}$
- 7: Compute $\kappa = \min\{\kappa \in \mathbb{N} \mid \mathcal{E}(\kappa) = \sum_{i=1}^{\kappa} \Lambda_{ii} / \sum_{i=1}^r \Lambda_{ii} \geq \delta_{\mathcal{E}}, \forall 1 \leq \kappa \leq r\}$.
- 8: Set $\lambda_i = \Lambda_{ii}$ and $\psi_i = \Psi_{:,i} \in \mathbb{R}^m, \forall 1 \leq i \leq \kappa$ $\triangleright i = 0, 1, \dots$
- 9: **else if** $n \ll m$ **then**
- 10: Compute eigen value decomposition $[\Phi, \Lambda] = \text{eig}(\mathbf{Y} \mathbf{Y}^\top)$ $\triangleright \mathbf{Y} \mathbf{Y}^\top \in \mathbb{R}^{n \times n}$
- 11: Compute $\kappa = \min\{\kappa \in \mathbb{N} \mid \mathcal{E}(\kappa) = \sum_{i=1}^{\kappa} \Lambda_{ii} / \sum_{i=1}^r \Lambda_{ii} \geq \delta_{\mathcal{E}}, \forall 1 \leq \kappa \leq r\}$.
- 12: Set $\lambda_i = \Lambda_{ii}$ and $\psi_i = \mathbf{Y} \Phi_{:,i} / \sqrt{\lambda_i} \in \mathbb{R}^m, \forall 1 \leq i \leq \kappa$ $\triangleright i = 0, 1, \dots$
- 13: **end if**

rate. To use this method, data snapshots from a dynamical system at different times are collected. The benefits of this method are its simplicity and lack of assumptions about the system. The only cost is performing a SVD on the snapshot matrix created from the data. DMD can be used and interpreted in several ways. More specifically, three main tasks can be enabled by the DMD algorithm; Diagnostics (especially in complex fluid systems), State estimation and future-state prediction, and Control.

2.3.2 Algorithm

As proposed by [5], we will follow the basic DMD formulation. Considering a data matrix $\mathbf{V}_1^n \in \mathbb{R}^{m \times n}$ as

$$\mathbf{V}_1^n = \begin{bmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_n \\ | & | & & | \end{bmatrix} \quad (18)$$

As $\mathbf{v}_{i=1,2,\dots,n}$ represents the i^{th} vector in the sequence. The time step between two consecutive snapshots can be denoted as Δt .

First, it is assumed that there exists a linear mapping operator, denoted by \mathbf{A} , which links the flow field at a particular time, referred to as \mathbf{v}_i , with the flow field at the following time, denoted as \mathbf{v}_{i+1} , that

$$\mathbf{v}_{i+1} = \mathbf{A} \mathbf{v}_i \quad (19)$$

As more snapshots are taken and the data sequence provided by the first snapshot captures the main characteristics of the physical process, it is reasonable to assume that after a certain number of snapshots, the vectors given by 18 will become linearly dependent. This means that adding more flow fields to the data sequence will not improve the vector space spanned by \mathbf{V} . When this limit is reached, the vector \mathbf{v}_n can be expressed as a linear combination of the previous vectors \mathbf{v}_i , where i ranges from 1 to $n - 1$, and these previous vectors are linearly independent. The vector \mathbf{v}_n can be represented as

$$\mathbf{v}_N = a_1 \mathbf{v}_1 + a_2 \mathbf{v}_2 + a_3 \mathbf{v}_3 + \dots + a_{N-1} \mathbf{v}_{n-1} + \mathbf{r} \quad (20)$$

It can be represented in a matrix format as

$$\mathbf{v}_n = \mathbf{V}_1^{n-1} \mathbf{a} + \mathbf{r} \quad (21)$$

where, $\mathbf{a}^\top = [a_1, a_2, \dots, a_{n-1}] \in \mathbb{R}^{n-1}$, and $\mathbf{V}_1^{n-1} \in \mathbb{R}^{m \times (n-1)}$, and $\mathbf{r} \in \mathbb{R}^m$ is the residual vector. According to [27] Eq. 21 becomes,

$$\begin{aligned} \mathbf{A} \mathbf{V}_1^{n-1} &= \mathbf{V}_2^n = \mathbf{V}_1^{n-1} \mathbf{S} + \mathbf{r} \mathbf{e}^\top \\ \mathbf{A} \begin{bmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_{n-1} \\ | & | & & | \end{bmatrix} &= \begin{bmatrix} | & | & & | \\ \mathbf{v}_2 & \mathbf{v}_2 & \dots & \mathbf{v}_n \\ | & | & & | \end{bmatrix} = \begin{bmatrix} | & | & & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{V}_1^{n-1} \mathbf{a} \\ | & | & & | \end{bmatrix} + \mathbf{r} \mathbf{e}^\top \end{aligned} \quad (22)$$

where S is a designed matrix defined as

$$S = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & a_1 \\ 1 & 0 & 0 & 0 & \dots & a_2 \\ 0 & 1 & 0 & 0 & \dots & a_3 \\ 0 & 0 & 1 & \ddots & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 & a_{n-2} \\ 0 & 0 & 0 & \dots & 1 & a_{m-1} \end{bmatrix} \quad (23)$$

The eigenvalues of matrix S are a good approximation for the eigenvalues of the original matrix. There are some decompositions of the data based on the companion matrix S like the one in Arnoldi method [3] which is often unable to extract more than the first or first two dominant dynamic modes, and that it is considered ill-conditioned, meaning that it can be highly sensitive to small changes in input data, which can result in unreliable output. So, a more robust implementation is used. This implementation results in a matrix \tilde{A} that includes all of the necessary information through a similarity transformation from the original matrix A . First, a preprocessing step for the data is conducted using the singular value decomposition of the data $V_1^{n-1} = U\Sigma W^\top$, where $U \in \mathbb{C}^{m \times r}$, $\Sigma \in \mathbb{C}^{r \times r}$, $W \in \mathbb{C}^{(n-1) \times r}$ and r is the reduced SVD approximation rank to V_1^{n-1} . U represents the POD modes and the columns of U and W are orthonormal. The SVD step performs a low rank truncation of the data. From 22 and using pseudoinverse,

$$A = V_2^n (V_1^{n-1})^\dagger = V_2^n W \Sigma^{-1} U^\top \quad (24)$$

For more computational efficiency, \tilde{A} is computed which is the $r \times r$ projection of the full matrix A onto POD modes.

$$\tilde{A} = U^\top A U = U^\top V_2^n W \Sigma \quad (25)$$

Using the unitary transform of

$$Pv = \tilde{v} \quad (26)$$

The matrix \tilde{A} represents a low dimensional linear model of the dynamical system on POD coordinates.

$$v_{i+1} = \tilde{A}v_i \quad (27)$$

The high dimensional state can be reconstructed as

$$V_1^{n-1} = U \tilde{V}_1^{n-1} \quad (28)$$

Then, the eigendecomposition of \tilde{A} is computed as

$$\tilde{A}\Xi = \Xi\Lambda \quad (29)$$

Where Ξ are the eigenvectors and Λ are the corresponding eigenvalues arranged in a diagonal matrix.

Finally, the eigendecomposition of A can be reconstructed from Ξ and Λ , where the eigenvalues of A are given by Λ and the eigenvectors of A are given by the columns of Φ

$$\Phi = V_2^n W \Sigma^{-1} \Xi \quad (30)$$

These eigenvectors are proven in [24] to be exact eigenvectors of the matrix A . Once the approximations of the eigenvalues and eigenvectors with low rank are available, it is possible to create a future solution projection for all future time periods as

$$V(t) \approx \sum_{i=1}^r \Phi \exp(\omega_i t) b = \Phi \exp(\Omega t) b \quad (31)$$

where,

$$\omega_n = \frac{\ln \lambda_n}{\Delta t}$$

$$\Omega = \text{diag}(\omega_i)$$

$b :=$ is the initial amplitudes

Eq. 31 can be interpreted as the least square fit or regression for the sampled data. The matrix A is constructed by minimizing $\|v_i - Av_{i-1}\|^2$ across all the snapshots. The remaining parameter to be computed is b_n . taking the initial snapshot v_n at $t = 0$, then from Eq. 31, $v_1 = \Phi b$. then,

$$b = \Phi^\dagger v_1 \quad (32)$$

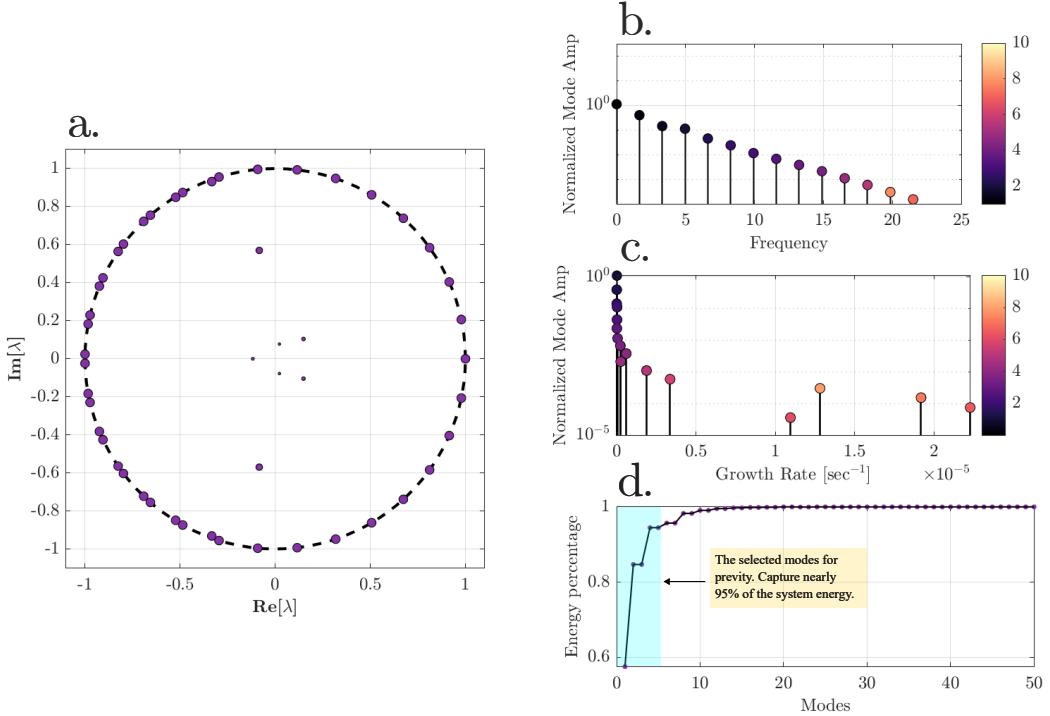


Figure 3: (a) Demonstrate the eigenvalue of the system; the size indicates the corresponding value. (b) Shows most dominant mode frequency. (c) Shows the growth rate of each mode. (d) Shows the energy ranking of the system characterized by Eq. 33

2.3.3 DMD Energy Selection

Since the entire idea of (Reduced Order Models) ROMs is based on the idea of predicting the system dynamics based on the projection of the full order dynamics onto a relatively small number of bases (i.e. modes), mode selection plays a crucial role in ROMs. Despite the fact that the DMD generates modes, each of which has a different oscillation frequency and growth/decay rate, it lacks an intrinsic sorting criterion. In order to choose the most effective modes, user-specific rules must be established. Here, we examine some of the methods for doing so [28].

One way to choose the DMD most dominant modes are presented in [29] as

$$\begin{aligned} E_i &= \frac{1}{T} \int_0^T \|\exp(\sigma_i t) b_i\| dt \\ &= \|b_i\| \frac{\exp(\sigma_i T) - 1}{\sigma_i T} \end{aligned} \quad (33)$$

And with this we can sort the energy for DMD in a heuristic way, then we can use Eq. 17 to choose the modes that contains most of the system energy. Figure 3 present the mode energies as well as the growth rate and the frequency of each mode.

3 Examples and Discussion

3.1 Example: flow over a cylinder

Here we utilize both POD and DMD to analyze the flow past a cylindrical object. A classic problem in fluid dynamics where vortices are developed behind the cylinder to form what-so-called Kármán vortex street. Here the two-dimensional Navier-Stokes equations are simulated using the immersed boundary projection method (IBPM) solver developed by [30]. With a discretized domain having mesh of (199, 499) point in x and y respectively.

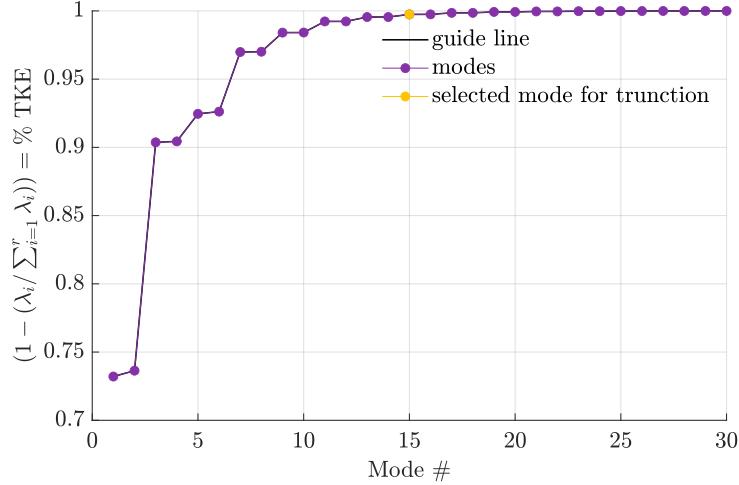


Figure 4: POD energy and modes selection.

3.1.1 Analysis using POD

$$\mathcal{E}(\kappa) := \frac{\sum_{i=1}^{\kappa} \lambda_i}{\sum_{i=1}^r \lambda_i} \geq 0.99 \quad (34)$$

This result in first 15 modes of the system. This means only 15 out of total 89351 modes capture nearly 0.991% of the Total Kinetic Energy of the system. This can be shown in Fig. 4. A significant reduction of system basis is obvious. Consequently, a significant reduction in computation time can be sensed as well after using the POD modes to discretize the original PDE. So, Instead of solving 89351 ordinary differential equation at each time step, only 15 are needed to be solved in order to capture 99.1% of system. Huge computation cost and time will be indeed saved.

Furthermore, a visualization of the first 15 modes of the system can be seen in Fig. 5. We can note the decay of the system modes and the small contribution as we increase the number of modes. Here we appreciate the orthogonality of the POD basis that offer a hierarchical energy ranking based on the most dominant singular/eigen value. So the modes are ranked as significance from mode number 1 to mode number 15.

3.1.2 Analysis using DMD

Results are shown in Figures 3 and 6. The results shows the modes energy, frequency behaviour, and growth rate. Which is the main objective of DMD that Schmid highlighted in his original Paper [5].

A Appendix

A.1 POD in Generalized Fourier Basis

Starting with a data matrix denoted as $\mathbf{Y} \in \mathbb{R}^{n \times m}$

$$\mathbf{Y} = [\mathbf{y}_1 \dots \mathbf{y}_m] \quad (35)$$

this matrix can be represented by the truncated SVD formulation as

$$\mathbf{Y} = \tilde{\Psi} \tilde{\Sigma} \tilde{\Phi}^\top \quad (36)$$

so, the column space of \mathbf{Y} can be represented in terms of r linearly independent columns of $\tilde{\Psi}$, and the coefficient of the expansion for the columns of \mathbf{y}_j in the basis $\{\psi\}_{i=1}^r$ are given by the j -th column of $\tilde{\Psi} = \tilde{\Sigma} \tilde{\Phi}^\top$

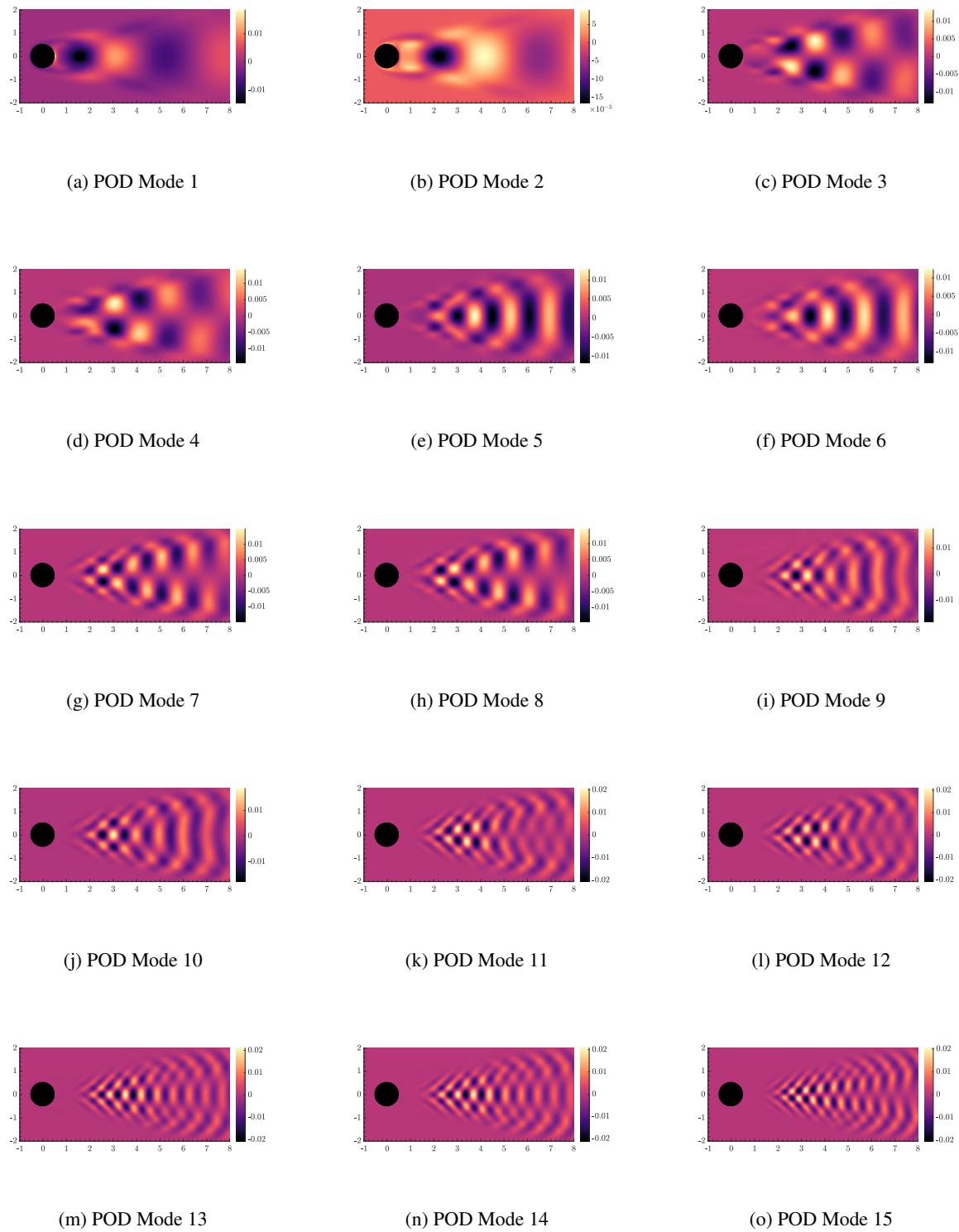


Figure 5: First fifteen POD Modes.

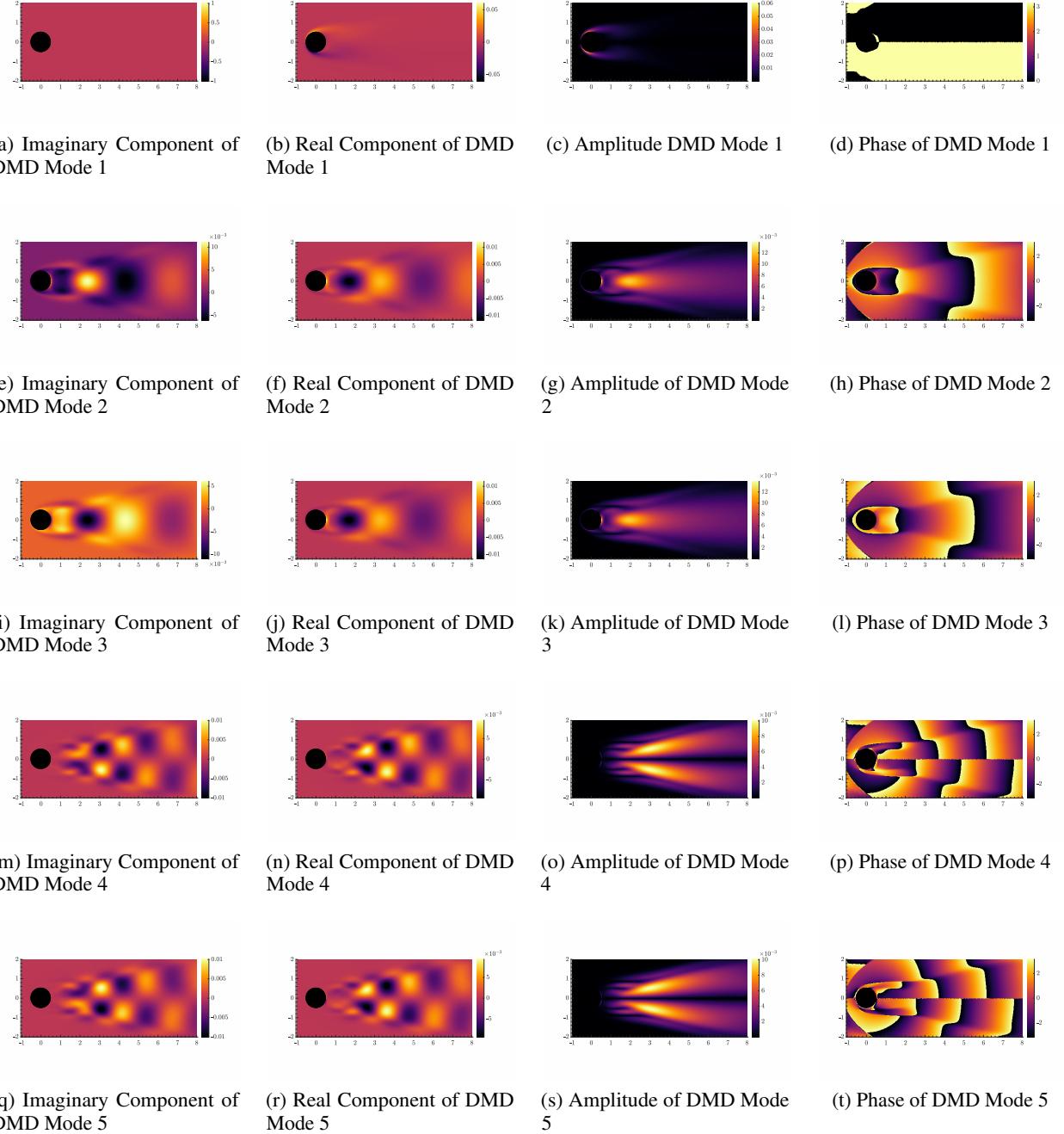


Figure 6: Columns from left to right: (1) is the imaginary part of each mode; (2) is the real part of each mode; (3) is the mode amplitude; (4) is the phase of each mode. Notice in Figure 3(d) the energy modes that captures energy system come in complex-conjugate pairs. So, notice here in this graph; column (1) and (4) for modes couple mode 2 and 3 and 4 and 5; the values which contain imaginary parts are reversed unlike columns (2) and (3) which contain the real part of the mode and the amplitude respectively. A nice observation to verify the energy selection. Also, From observation, mode 1 is obviously the mean flow which has no oscillatory motion as seen in (a).

since ψ_i vectors are orthonormal, then

$$\begin{aligned}
 \mathbf{y}_i &= \sum_{i=1}^r \tilde{\mathbf{T}}_{ij} \psi_i \\
 &= \sum_{i=1}^r \left[\tilde{\Psi}^\top \tilde{\Psi} \tilde{\Sigma} \right]_{ij} \psi_i \\
 &= \sum_{i=1}^r \left[\tilde{\Psi}^\top \tilde{\Psi} \tilde{\Sigma} \tilde{\Phi}^\top \right]_{ij} \psi_i \\
 &= \sum_{i=1}^r \left[\tilde{\Psi}^\top \mathbf{Y} \right]_{ij} \psi_i \\
 &= \sum_{i=1}^r \langle \psi_i, \mathbf{y}_i \rangle \psi_i
 \end{aligned} \tag{37}$$

So, at the end, we have reached to the general Fourier series representation of \mathbf{y}_i and it has the form of

$$\mathbf{y}_i = \sum_{i=1}^r \langle \mathbf{y}_i, \psi_i \rangle \psi_i = \sum_{i=1}^r a_i \psi_i \tag{38}$$

References

- ¹C. W. Rowley and S. T. Dawson, “Model reduction for flow analysis and control”, Annual Review of Fluid Mechanics **49**, 387–417 (2017).
- ²K. 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 Journal **55**, 4013–4041 (2017).
- ³P. Benner, S. Gugercin, and K. Willcox, “A survey of projection-based model reduction methods for parametric dynamical systems”, SIAM review **57**, 483–531 (2015).
- ⁴J. L. Lumley, *Stochastic tools in turbulence. volume 12. applied mathematics and mechanics*, tech. rep. (PENNSYLVANIA STATE UNIV UNIVERSITY PARK DEPT OF AEROSPACE ENGINEERING, 1970).
- ⁵P. J. Schmid, “Dynamic mode decomposition of numerical and experimental data”, Journal of fluid mechanics **656**, 5–28 (2010).
- ⁶L. N. Trefethen, *Spectra and pseudospectra: the behaviour of non-normal matrices and operators* (Springer, 1999).
- ⁷G. H. Golub and C. F. Van Loan, *Matrix computations* (JHU press, 2013).
- ⁸S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: machine learning, dynamical systems, and control* (Cambridge University Press, 2022).
- ⁹P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley, *Turbulence, coherent structures, dynamical systems and symmetry* (Cambridge university press, 2012).
- ¹⁰J. Guckenheimer and P. Holmes, *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, Vol. 42 (Springer Science & Business Media, 2013).
- ¹¹B. Noble, J. W. Daniel, et al., *Applied linear algebra*, Vol. 477 (Prentice-Hall Englewood Cliffs, NJ, 1977).
- ¹²M. Loève, *Probability theory* (Courier Dover Publications, 2017).
- ¹³I. Jolliffe, “Principal component analysis”, Encyclopedia of statistics in behavioral science (2005).
- ¹⁴H. Hotelling, “Analysis of a complex of statistical variables into principal components.”, Journal of educational psychology **24**, 417 (1933).
- ¹⁵R. Preisendorfer, “Principal component analysis in meteorology and oceanography”, Elsevier Sci. Publ. **17**, 425 (1988).
- ¹⁶S. Volkwein, “Proper orthogonal decomposition: theory and reduced-order modelling”, Lecture Notes, University of Konstanz **4**, 1–29 (2013).
- ¹⁷C. Gräßle, M. Hinze, and S. Volkwein, “Model order reduction by proper orthogonal decomposition”, in *Model order reduction: volume 2: snapshot-based methods and algorithms* (De Gruyter, 2021), pp. 47–96.
- ¹⁸G. Berkooz, P. Holmes, and J. L. Lumley, “The proper orthogonal decomposition in the analysis of turbulent flows”, Annual review of fluid mechanics **25**, 539–575 (1993).

- ¹⁹C. Eckart and G. Young, “The approximation of one matrix by another of lower rank”, *Psychometrika* **1**, 211–218 (1936).
- ²⁰L. Sirovich, “Turbulence and the dynamics of coherent structures. i. coherent structures”, *Quarterly of applied mathematics* **45**, 561–571 (1987).
- ²¹L. Sirovich, “Turbulence and the dynamics of coherent structures. ii. symmetries and transformations”, *Quarterly of Applied mathematics* **45**, 573–582 (1987).
- ²²L. Sirovich, “Turbulence and the dynamics of coherent structures. iii. dynamics and scaling”, *Quarterly of Applied mathematics* **45**, 583–590 (1987).
- ²³J. N. Kutz, *Data-driven modeling & scientific computation: methods for complex systems & big data* (Oxford University Press, 2013).
- ²⁴J. H. Tu, “Dynamic mode decomposition: theory and applications”, PhD thesis (Princeton University, 2013).
- ²⁵J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems* (SIAM, 2016).
- ²⁶I. Mezić, “Spectral properties of dynamical systems, model reduction and decompositions”, *Nonlinear Dynamics* **41**, 309–325 (2005).
- ²⁷A. Ruhe, “Rational krylov sequence methods for eigenvalue computation”, *Linear Algebra and its Applications* **58**, 391–405 (1984).
- ²⁸S. E. Ahmed, O. San, D. Bistrian, and I. Navon, “Sketching methods for dynamic mode decomposition in spherical shallow water equations”, in Aiaa scitech 2022 forum (2022), p. 2325.
- ²⁹J. Kou and W. Zhang, “An improved criterion to select dominant modes from dynamic mode decomposition”, *European Journal of Mechanics-B/Fluids* **62**, 109–129 (2017).
- ³⁰K. Taira and T. Colonius, “The immersed boundary method: a projection approach”, *Journal of Computational Physics* **225**, 2118–2137 (2007).