CAPITAL UNIVERSITY OF SCIENCE AND
TECHNOLOGY, ISLAMABAD

Proper Orthogonal Decomposition Analysis of Navier Stokes Equations in Different Geometries

by

Ayesha Sagheer

A thesis submitted in partial fulfillment for the
degree of Master of Philosophy

in the
Faculty of Computing
Department of Mathematics

2025

I dedicate my thesis to

My Mother,

The woman who gave me my life and blood and never once gave the idea that I couldn't be anything I wanted to be.

CERTIFICATE OF APPROVAL

Proper Orthogonal Decomposition Analysis of Navier Stokes Equations in Different Geometries

by

Ayesha Sagheer

(Registration No: MMT233013)

THESIS EXAMINING COMMITTEE

S. No.	Examiner	Name	Organization
(a)	External Examiner	Dr. Nabeela Kousar	Air University, Islamabad
(b)	Internal Examiner	Dr. Rashid Ali	CUST, Islamabad
(c)	Supervisor	Dr. Muhammad Sabeel Khan	CUST, Islamabad

Dr. Muhammad Sabeel Khan
Thesis Supervisor
April, 2025

Dr. Muhammad Sagheer
Head
Dept. of Mathematics
April, 2025

Dr. M. Abdul Qadir
Dean
Faculty of Computing
April, 2025

Author's Declaration

I, **Ayesha Sagheer**, hereby state that my MPhil thesis titled "**Proper Orthogonal Decomposition Analysis of Navier Stokes Equations in Different Geometries**" is my work and has not been submitted previously by me for taking any degree from Capital University of Science and Technology, Islamabad or anywhere else in the country/abroad.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw my MPhil Degree.

(Ayesha Sagheer)
Registration No: MMT233103

Plagiarism Undertaking

I solemnly declare that the research work presented in this thesis titled "Proper Orthogonal Decomposition Analysis of Navier Stokes Equations in Different Geometries" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged, and that complete thesis has been written by me.

I understand the zero-tolerance policy of the HEC and Capital University of Science and Technology towards plagiarism. Therefore, I, as an author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after awarding of MPhil Degree, the University reserves the right to withdraw/revoke my MPhil degree and that HEC and the University have the right to publish my name on the HEC/University website on which names of students are placed who submitted plagiarized work.

(Ayesha Sagheer)
Registration No: MMT233013

Acknowledgement

In the name of Allah, the Most Merciful and Beneficent, who created the universe and granted mankind the wisdom and intelligence to unravel its mysteries. I would like to express my sincere gratitude and deep respect to my supervisor, Dr. Muhammad Sabeel Khan, for his unwavering support, insightful guidance, and constant encouragement throughout this research. His constructive feedback, both written and verbal, played a pivotal role in shaping and refining this thesis.

I am deeply grateful to all my teachers for their constant encouragement and for instilling in me the importance of striving for excellence in the study of mathematics. I would also like to express my sincere thanks to CUST for providing such a supportive environment that facilitated my research. Words cannot fully express my gratitude to my parents, **Dr. Muhammad Sagheer** and **Dr. Dur-e-Shehwar Sagheer**, whose endless love, sacrifices, and unwavering faith in me have shaped who I am today. Your constant encouragement and guidance have been my greatest strength. I am deeply thankful for all the sacrifices you've made and for being my pillars of support, every step of the way. This achievement is as much yours as it is mine.

A special thanks goes to my rocks, my sisters, **Khadeeja Sagheer** and **Raabia Sagheer**, who are my constant source of comfort and support. Not a day goes by that I don't thank the universe for giving me the greatest gift of all in the shape of you two. Finally, I want to express my gratitude to all my friends who encouraged me throughout my MPhil degree. To my dearest friend, **Nimra Ahmed**, when life became dreary and the path ahead seemed uncertain, you were there to brighten it all. I would also like to thank my dear friend, **Asma Shafiq**, who encouraged me throughout my MPhil degree and motivated me at every turn.

(Ayesha Sagheer)

Registration No: MMT233013

Abstract

This thesis investigates the application of Proper Orthogonal Decomposition (POD) in fluid dynamics, focusing on the analysis of flow structures within three distinct geometries. The study begins with an introduction to the fundamental concepts of FreeFEM, Singular Value Decomposition (SVD), and POD, providing a foundation for the subsequent analysis. Initially, the thesis reproduces the results from an existing study, where POD was applied to a two-dimensional geometry using the Navier-Stokes equations (NSE). This serves as a benchmark for understanding fluid flow behaviour in simplified geometries. Building upon this groundwork, the primary contribution of the research involves extending the POD analysis to three different geometries with obstacles presented in a series arrangement, where intricate flow patterns characterize the fluid dynamics. The obstacles were set to be circular cylinders and square cylinders. The behaviour of these flows is examined through the extraction of dominant modes, revealing the underlying structure of the turbulent flow in each case. The findings demonstrate the potential of POD as a powerful tool for reducing the dimensionality of the system while retaining the most significant flow features. The results not only provide insight into the fluid flow dynamics in various geometries but also offer a valuable methodology for further investigations in computational fluid dynamics and the optimization of complex flow systems.

Contents

List of Figures

List of Tables

Abbreviations

CST	Cauchy Stress Tensor
ESVD	Economy Singular Value Decomposition
EVP	Eigenvalue Problem
FEM	Finite Element Method
MOS	Method of Snapshots
NSE	Navier Stokes Equations
PDEs	Partial Differential Equations
POD	Proper Orthogonal Decomposition
RTT	Reynolds Transport Theorem
SVD	Singular Value Decomposition
TSVD	Truncated Singular Value Decomposition
WPOD	Weighted Proper Orthogonal Decomposition

Symbols

t	Time
m	Mass
V	Volume
Re	Reynolds number
ρ	Density
ν	Kinematic viscosity
μ	Dynamic viscosity
v	Velocity of fluid flow
σ	Cauchy stress tensor
λ_n	Eigenvalues
$\mathbf{u}_n$	Left singular vectors or POD basis vectors
$\mathbf{v}_n$	Right singular vectors
k	Rank of a matrix
$\mathbb{E}$	Energy Ratio
h	Height of the pipe
w	Width of the pipe
d	Diameter of the circle
r	Radius of the circle
D	Distance between the circles
F_D	Drag Force
F_L	Lift Force
C_D	Drag Coefficient
C_L	Lift Coefficient
p	Pressure

v_{max}	Maximum Velocity
$\bar{v}_{max}$	Mean velocity in x-direction
δ_e	Threshold
l	Length of sides of a square

Chapter 1

Introduction and Literature Survey

Fluid dynamics [1] is a crucial study area across various fields, including engineering, biology, and other sciences. Working knowledge of fluid flow behaviour is essential, especially when designing efficient systems that might predict natural phenomena. When the fluid interacts with complex geometry, modelling can become more complicated, and computational techniques with higher sophistication are required.

The finite element method [2, 3], more popularly known as FEM, is a powerful and efficient numerical technique that is used to solve partial differential equations that govern fluid flows. FEM works by dividing up complex geometries into more minor and more straightforward, discrete elements, which enables the computational model to simulate fluid flow behaviour under many different conditions, accurately. FEM has a wide range of applications and has been successfully employed to solve numerous complex physical problems in fields such as structural mechanics, fluid dynamics, heat transfer, and much more [4–6].

Implementing the finite element method for solving a fluid flow problem can be done through various software and tools [7–10]. Still, amongst a long list of variable options, FreeFEM++ [11] stands out. A brief introduction to FreeFEM++ can be found at https://freefem.org/. FreeFEM++ is an open-source software specially designed to find solutions to PDEs using the finite element method. It offers a friendly environment that helps with efficient computation. FreeFEM++ has a rich library of predefined functions and elements, allowing users to implement the method straight to complex geometries and complicated boundary conditions. Above that, FreeFEM++ supports many other numerical techniques, which enables the integration of other advanced techniques, including proper orthogonal decomposition (POD) and reduced-order modelling (ROM). With FreeFEM++ [12], the accuracy will be enhanced, and fluid flow simulations will become more efficient.

The development of the finite element method can be traced back to 1941, and over the years, this technique has proved its worth. It is highly effective and produces minimal error. However, this approach can be computationally expensive, especially when three-dimensional problems have to be solved.

Full-order models pose many computational challenges, and to resolve them, reduced-order modelling techniques [13] have gained considerable prominence in recent years. One such technique is Proper Orthogonal Decomposition (POD) [14–16], where the system identifies the most dominant modes and extracts them to construct a reduced-order model. This reduced-order model retains the original system's fundamental and essential dynamics, making the computation more cost-efficient. By rescuing the computational requirements, the system can perform more efficient simulations and help with real-time analysis [17].

Notable work has been done on the application of FEM with POD. For further details, interested readers can refer to the key studies and researchers in this area. Studies have been done to generate enrichment functions for generalized finite element methods (GFEM) using the Proper Orthogonal Decomposition (POD) technique, leveraging low-order representations of data to improve solution accuracy [18]. Another study combined Proper Orthogonal Decomposition (POD) with the Finite-Element Method (FEM) to reduce the number of degrees of freedom and overall computing time, applying the technique to both linear and nonlinear problems [19]. Researchers have also explored the application of Proper Orthogonal Decomposition (POD) combined with Finite-Element Method (FEM) in physical problems, such as the advection-dominated diffusion-reaction equations for groundwater contamination prevention [20] and the tropical Pacific Ocean reduced gravity model [21].

In this study,the focus will be on analyzing the flow of the fluid using FEM applied through FreeFEM++. A benchmark geometry introduced by Schäfer et al [22] will serve as the foundational model for understanding the dynamics of the fluid system. Initially, an investigation of the fluid flow behaviour in this geometry will be presented. Subsequently, we will perform a Proper Orthogonal Decomposition (POD) [23] mode analysis to capture the most significant modes of the flow.

This work expands the analysis to three distinct geometries, each featuring cylindrical obstacles arranged in a series configuration. The first two geometries consist of circular cylinders, with the distance between them being 0.4 meters for the first and 0.45 meters for the second. The third geometry features square cylinders with a distance of 0.8 meters between them. The choice of these specific geometries is crucial as the arrangement and spacing of the obstacles significantly influence vortex shedding frequencies and fluid domain fluctuations. Each geometry has been selected to highlight how varying the shape and distance of the obstacles affects the overall fluid flow dynamics and vortex behavior.

The first geometry, with circular cylinders spaced 0.4 meters apart, represents a common configuration where closely spaced obstacles may lead to strong interactions between the vortices shed from each cylinder. The second geometry, with a 0.45-meter gap between circular cylinders, allows for a comparison of how slight variations in spacing impact flow patterns and vortex shedding frequencies. The final geometry, featuring square cylinders with a larger 0.8-meter gap, is crucial for understanding the effect of shape and larger spacing on fluid dynamics. The analysis of these geometries will provide valuable insights into the behavior of flow and the dynamics of vortex shedding in different configurations. Using Proper Orthogonal Decomposition (POD), the most prominent flow features will be extracted for each geometry, identifying key flow modes and helping to understand the fluid behavior in each case.

This approach emphasizes the importance of modal decomposition in fluid dynamics. By analyzing the POD modes across different geometries, we gain a deeper understanding of the underlying flow structures and their influence on the overall dynamics. This study will contribute to the development of reduced-order modelling methods and offer valuable insights into fluid behaviour in varied geometrical configurations, which is fundamental in applications ranging from aerodynamics to industrial fluid systems.

Furthermore, it provides a framework for optimizing designs and improving efficiency in systems where fluid dynamics play a crucial role.

1.1 Thesis Contribution

This thesis makes significant contributions to the field of fluid dynamics by advancing the application of POD together with the well-known finite element method (FEM) using FreeFEM++ in the analysis of fluid flow within various benchmark geometries. This research establishes a solid framework for applying Proper Orthogonal Decomposition (POD) to capture the dominant flow features by systematically studying fluid flow behaviour in these geometries. The integration of POD with FEM enhances computational efficiency while preserving the essential dynamics of the fluid system, enabling more accurate and effective simulations.

This study emphasizes extracting and analysing POD modes from three distinct geometrical configurations, offering valuable insights into the fluid flow characteristics and their underlying dynamics. The results advance the understanding of modal decomposition techniques in fluid dynamics and contribute to developing reduced-order modelling strategies, which can be applied to various liquid flow scenarios. Collectively, this research provides both methodological advancements and practical insights, paving the way for future studies in fluid dynamics, particularly in applications where fluid behaviour in complex geometries is critical.

1.2 Objectives

The primary objectives of this thesis are to improve the understanding and modelling of fluid dynamics through the application of FEM. Specifically, the objectives include the following.

- To apply the finite element method (FEM) using FreeFEM++ to analyze fluid flow in three distinct geometrical configurations.

- To perform a systematic Proper Orthogonal Decomposition (POD) analysis to identify and extract the most significant flow modes in each considered problem.
- To enhance computational efficiency by integrating FEM with POD, enabling the reduced-order modelling of fluid flow across different geometries.
- To investigate the dynamic behaviour of fluid flow by analyzing the key POD modes that capture the essential fluid dynamics of each considered problem.
- To explore the impact of different geometrical configurations on the fluid flow and how POD can be used to summarize and understand these flow characteristics.
- To provide insightful comparisons of the fluid dynamics across the three geometrical configurations, highlighting the differences and similarities in their respective flow modes.
- To contribute to developing efficient modelling techniques in fluid dynamics, particularly for applications requiring real-time or large-scale simulations.
- To examine the broader implications of the findings in the context of practical applications in industries such as aerodynamics, biomedical engineering, and fluid-structure interaction modelling.

1.3 Thesis Layout

This thesis is made up of the following chapters:

- **Chapter 2** presents the mathematical tools necessary to understand the finite element method, including the governing equations of fluid dynamics. It includes an introduction to proper orthogonal decomposition (POD) principles and singular value decomposition (SVD).
- In **Chapter 3**, the relationship between POD and SVD is explored in detail. The chapter emphasizes how SVD facilitates the extraction of dominant modes from the system, providing a theoretical foundation for subsequent modelling approaches.

- In **Chapter 4**, the focus is on implementing the finite element method specifically for the Navier-Stokes equations in the Schäfer et al [22] benchmark geometry. It details the numerical procedures and the validation of the POD modes and presents results from simulations conducted using FreeFEM++.

- With appropriate transformations, in **Chapter 5**, the final chapter, extends the analysis by applying Proper Orthogonal Decomposition (POD) to the Navier-Stokes equations across three different configurations, with a focus on examining the effects on drag and lift coefficients and behaviour of the fluid flow.

- **Chapter 6** contains the conclusion of this work.

The work's references are enumerated in the bibliography.

Chapter 2

Theoretical Background and Mathematical Notions

This chapter will discuss basic concepts, definitions, and governing laws related to fluid dynamics and basic concepts needed to understand Proper Orthogonal Decomposition. The finite element method will also be discussed in detail.

2.1 Mathematical Notions

This section provides a comprehensive overview of the mathematical tools and foundational concepts essential for understanding the subsequent discussions in this thesis. These tools establish the groundwork for analyzing and solving the complex systems central to this work, serving as a bridge between theory and practical application.

2.1.1 Open Domain with Lipschitz Boundary

Let $\Omega \subset \mathbb{R}^d$ be an open domain, where $d \in \{2, 3\}$, and $\partial\Omega$ denotes its boundary. A domain Ω has a local Lipschitz boundary if for each $x \in \partial\Omega$, there exists a neighbourhood U_x such that $\partial\Omega \cap U_x$ is the graph of a Lipschitz continuous function in local coordinates. For our study, $\Omega \subset \mathbb{R}^d$ has a Lipschitz continuous boundary, and $\mathbf{n}$ denotes the outward unit normal vector to $\partial\Omega$.

2.1.2 Some Important Notion

The following are a list of some notions with respect to derivatives and their properties [24].

2.1.2.1 Divergence

The divergence of a vector-valued function $\mathbf{F} = (f_1, f_2, ..., f_n)$ is expressed as

$$\nabla.\mathbf{F} = \frac{\partial f_1}{\partial x_1} + \frac{\partial f_2}{\partial x_2} + ... + \frac{\partial f_n}{\partial x_n},$$

where f_2, f_2 and all the way up to f_n are components of the vector field along the $x_1, x_2, ..., x_n$ directions. Divergence measures the rate at which fluid expands or compresses at a given point. For example, a positive divergence indicates a source of fluid, whereas a negative divergence indicates a sink.

2.1.2.2 Gradient

The gradient of a scalar-valued function $f = f(x_1, x_2, ..., x_n)$ is expressed mathematically as

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, ..., \frac{\partial f}{\partial x_n} \right).$$

The gradient of a scalar-valued function provides valuable information about how the function changes in space. It is a vector that shows both the direction of the steepest increase and the rate of change in that direction.

2.1.2.3 Laplace Operator

The Laplace operator, or the Laplacian, is a second-order differential operator that measures how the value of a scalar-valued function $f(x_1, x_2, \ldots, x_n)$ defined in an n-dimensional space diverges from its average value in a given neighborhood. The Laplace operator is defined as

$$\nabla^2 f = \frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} + ... + \frac{\partial^2 f}{\partial x_n^2}.$$

2.1.3 Bounded Operators

A bounded operator is an operator between two normed vector spaces that maps bounded sets to bounded sets. Consider two normed vector spaces A and B. A linear operator $\mathbf{T} : A \to B$ is said to be a bounded operator if there exists a constant C such that $\forall\ a \in A$

$$||\mathbf{T}(a)||_B \leq C||a||_A$$

where $||\mathbf{T}(a)||_B$ is the norm of $\mathbf{T}(a)$ in the space B and $C||a||_A$ is the norm of a in the space A. A bounded linear operator will also satisfy the linearity condition, i.e.

$$\mathbf{T}(c_1\mathbf{a_1} + c_2\mathbf{a_2}) = c_1\mathbf{T}(\mathbf{a_1}) + c_2\mathbf{T}(\mathbf{a_2}).$$

2.1.4 Function Spaces

A Lebesgue space refers to a measurable space with a measure that allows for the integration of functions. They play a significant role in real analysis, probability theory, and foundational analysis. To formally define the Lebesgue space, $L^p(\Omega)$, where $1 \leq p \leq \infty$, the norm $h : \Omega \to \mathbb{R}$ of a function must be specified as

$$||h||_{L^p(\Omega)} = \left(\int_\Omega |h(x)|^p dx \right)^{\frac{1}{p}},$$

and

$$||h||_{L^\infty(\Omega)} = ess_{x\in\Omega} \sup |h(x)|.$$

The Lebesgue space is defined for p when $1 \leq p \leq \infty$ as

$$L^p(\Omega) = \{\ h : \Omega \to \mathbb{R}|\ \text{h is a measurable function},\ ||h||_{L^p(\Omega)} < \infty\}.$$

These are Banach spaces, which are complete normed vector spaces. When $p = 2$ we get the Hilbert space $L^2(\Omega)$ equipped with an inner product

$$(h, f)_{L^2(\Omega)} = \int_\Omega h(x)f(x)dx.$$

If vector-valued functions are concerned such that $\vec{h}, \vec{f} : \mathbb{R}^d \to \mathbb{R}^d$, the $L^2(\Omega)$ is endowed with the inner product

$$(\vec{h}, \vec{f})_{L^2(\Omega)} = \int_\Omega \vec{h}(x) \cdot \vec{f}(x)dx.$$

For functions taking tensor values (tensor-valued), $H, F : \mathbb{R}^d \to \mathbb{R}^{d\times d}$, the inner product is defined as

$$(H, F)_{L^2(\Omega)} = \int_\Omega H(x) : F(x)dx.$$

Where : denotes the Frobenius inner product. For L^2, the notation $(\cdot, \cdot) := (\cdot, \cdot)_{L^2}$ is used. Let the space we are with be a space of infinitely differentiable functions equipped with compact support. Let it be denoted by $\mathbb{D}_c^\infty$. We call $\mathbf{q} = \nabla f \in [L^2(\Omega)]^d$ the variational gradient of $f \in L^2(\Omega)$ if

$$\int_\Omega \mathbf{q} \cdot \phi dx = -\int_\Omega f(\nabla \cdot \phi)dx$$

for all $\phi \in [\mathbb{D}_c^\infty]^d$. The Sobolev spaces [25], denoted by $\mathbb{W}^{s,t}(\Omega)$, are the main functional spaces used in this thesis. They are crucial in the study of PDEs. They provide a framework to study functions with certain degrees of smoothness and integrability. In this study we are only concerned with Sobolex spaces that contain functions which are one time weakly differentiable, defined as

$$\mathbb{W}^1(\Omega) = \{f \in L^2(\Omega) | \nabla f \in [L^2(\Omega)]^d\}.$$

The $\mathbb{W}^1(\Omega)$ is a Hilbert space with the inner products defined as

$$(f, g)_{\mathbb{W}^1(\Omega)} = (f, g)_{L^2(\Omega)} + (\nabla f, \nabla g)_{L^2(\Omega)}.$$

When homogeneous Dirichlet boundary conditions are taken into account, the space considered will be

$$\mathbb{W}_0^1(\Omega) = \{f \in \mathbb{W}^1(\Omega) | f = 0 \text{ on } \partial\Omega\}.$$

2.2 Linear Algebra Fundamentals for Singular Value Decomposition

This section aims to provide clear definitions and an intuitive understanding of the basic concepts in linear algebra used in Singular Value Decomposition (SVD).

2.2.1 Vector Space

A set V is called a vector space over a field F if the following conditions hold:

1. **Closure under addition:** For all $\mathbf{u}, \mathbf{v} \in V$, the sum $\mathbf{u} + \mathbf{v} \in V$.
2. **Closure under scalar multiplication:** For all $\mathbf{v} \in V$ and $\alpha \in F$, the scalar multiple $\alpha\mathbf{v} \in V$.
3. **Commutativity of addition:** $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ for all $\mathbf{u}, \mathbf{v} \in V$.
4. **Associativity of addition:** $(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$ for all $\mathbf{u}, \mathbf{v}, \mathbf{w} \in V$.
5. **Existence of an additive identity:** There exists an element $\mathbf{0} \in V$ such that $\mathbf{v} + \mathbf{0} = \mathbf{v}$ for all $\mathbf{v} \in V$.
6. **Existence of additive inverses:** For each $\mathbf{v} \in V$, there exists $-\mathbf{v} \in V$ such that $\mathbf{v} + (-\mathbf{v}) = \mathbf{0}$.
7. **Compatibility of scalar multiplication with field multiplication:** $\alpha(\beta\mathbf{v}) = (\alpha\beta)\mathbf{v}$ for all $\alpha, \beta \in F$ and $\mathbf{v} \in V$.
8. **Distributivity of scalar multiplication over vector addition:** $\alpha(\mathbf{u} + \mathbf{v}) = \alpha\mathbf{u} + \alpha\mathbf{v}$ for all $\alpha \in F$ and $\mathbf{u}, \mathbf{v} \in V$.

9. **Distributivity of scalar multiplication over scalar addition:** $(\alpha + \beta)\mathbf{v} = \alpha\mathbf{v} + \beta\mathbf{v}$ for all $\alpha, \beta \in F$ and $\mathbf{v} \in V$.

10. **Existence of a multiplicative identity:** $1\mathbf{v} = \mathbf{v}$ for all $\mathbf{v} \in V$, where 1 is the multiplicative identity in F.

2.2.2 Row, Column, and Null Space

The **row space** of a matrix A is the vector space spanned by the rows of A. For an $m \times n$ matrix A, the row space is a subspace of $\mathbb{R}^n$ (if the matrix has real entries) and is defined as:

$$\text{Row}(A) = \text{Span}\{\text{row}_1(A), \text{row}_2(A), \ldots, \text{row}_m(A)\},$$

where $\text{row}_i(A)$ represents the i-th row of the matrix A [26].

The **column space** of a matrix A is the vector space spanned by the columns of A. For an $m \times n$ matrix A, the column space is a subspace of $\mathbb{R}^m$ (if the matrix has real entries) and is defined as:

$$\text{Col}(A) = \text{Span}\left\{\text{col}_1(A), \text{col}_2(A), \ldots, \text{col}_n(A)\right\},$$

where $\text{col}_i(A)$ represents the i-th column of the matrix A [26].

The **null space** (also popularly known as the **kernel**) of a matrix A is the set of all possible vectors $\mathbf{x}$ such that [26]:

$$A\mathbf{x} = \mathbf{0},$$

where A is an $m \times n$ matrix, $\mathbf{x}$ is an $n \times 1$ vector, and $\mathbf{0}$ is the zero vector in $\mathbb{R}^m$. The null space is a subspace of $\mathbb{R}^n$ (if the matrix has real entries) and is defined as:

$$\text{Null}(A) = \left\{\mathbf{x} \in \mathbb{R}^n \mid A\mathbf{x} = \mathbf{0}\right\}.$$

The **rank** of a matrix A is the dimension of the column space (or the row space) of the matrix. It represents the number matrices of linearly independent columns (or rows) in then $m \times n$ matrix A, the rank is denoted as $\operatorname{rank}(A)$ and is defined as

$$\operatorname{rank}(A) = \dim(\operatorname{Col}(A)) = \dim(\operatorname{Row}(A)),$$

where $\operatorname{Col}(A)$ is the column space of A and $\operatorname{Row}(A)$ is the row space of A [26].

2.2.3 Orthogonal and Orthonormal Matrix

A square matrix A is called **orthogonal** if its rows and columns are orthogonal vectors. In other words, a matrix A is orthogonal if it satisfies the following condition:

$$A^T A = AA^T = I,$$

where A^T is the transpose of A, and I is the identity matrix of the same size as A [27]. Equivalently, the matrix A is orthogonal if its inverse is equal to its transpose:

$$A^{-1} = A^T.$$

A square matrix with orthonormal vectors as rows and columns is known as **orthonormal matrix**. Specifically, a matrix A is orthonormal if the following conditions hold:

- The rows of matrix A together form an orthonormal set of vectors, implying that each row is a unit vector and orthogonal to the others.
- The columns of A form an orthonormal set of vectors, meaning each column is a unit vector and orthogonal to the remaining columns.

In other words, the matrix A is orthonormal if

$$A^T A = AA^T = I,$$

where A^T is the transpose of A, and I is the identity matrix [27].

2.2.4 Diagonalization

Diagonalization and LU decomposition [26, 27] are powerful techniques in linear algebra used to simplify matrix operations and solve systems of linear equations. These methods involve transforming matrices into more manageable forms, either through finding eigenvalues and eigenvectors or by factoring the matrix into a product of triangular matrices, respectively. LU decomposition factorizes a matrix A into the product of two triangular matrices: the lower triangular matrix L and the upper triangular matrix U. That is

$$A = LU$$

such that L is a lower triangular matrix with ones on the diagonal, and U is an upper triangular matrix. This decomposition is particularly useful when the need for solving systems of linear equations arises, as it simplifies the matrix operations by breaking them down into simpler triangular systems. LU factorization can break down complex matrix operations into simpler components, and diagonalization does the same by transforming a matrix into a diagonal form. Diagonalization transforms a square matrix A into a diagonal matrix that will be represented by D using a transformation known as a similarity transformation. A matrix A is diagonalizable if there exists an invertible matrix P such that

$$A = PAP^{-1},$$

where D is a diagonal matrix containing eigenvalues of A and P contains the corresponding eigenvectors. Diagonalization is useful because it simplifies matrix powers and exponential and provides insight into the spectral properties of the matrix. Diagonalization helps understand SVD by providing insight into how a matrix can be represented in terms of its eigenvalues and eigenvectors, similar to how SVD decomposes a matrix into orthogonal components with singular values.

2.2.5 Eigenvalue Problem

Suppose the $C \in M_{k\times k}(\mathbb{R})$. If there exist a non-zero vector $\mathbf{v} \in \mathbb{R}^k$ such that

$$C\mathbf{v} = \lambda\mathbf{v},$$

then the scalar λ is called an eigenvalue of C and $\mathbf{v}$ is called the eigenvector of C corresponding to the eigenvalue [26].

2.2.6 Velocity Matrix

Let matrix **C** represent fluid flow over a certain period.

$$\mathbf{C} = \begin{pmatrix} u(x_1,t_1) & u(x_1,t_2) & \cdots & u(x_1,t_k) \\ u(x_2,t_1) & u(x_2,t_2) & \cdots & u(x_2,t_k) \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ u(x_{k-1},t_1) & u(x_{k-1},t_2) & \cdots & u(x_{k-1},t_k) \\ u(x_k,t_1) & u(x_k,t_2) & \cdots & u(x_k,t_k) \end{pmatrix} \tag{2.1}$$

The matrix represents the velocity $u(x_i, t_j)$ at spatial points x_i in the domain and at different times t_j, where x_i are spatial points, t_j are time instances, and $u(x_i, t_j)$ is the velocity at the spatial point x_i at time t_j. In this matrix, the eigenvalues can provide valuable insights into the behaviour flow. In the matrix itself, each row and column typically corresponds to a specific point or grid cell in the domain, and the matrix entries will represent any fluid quantity at that point over time. The eigenvalues of such a matrix are indicators of flow patterns. The largest ones correspond to the most dominant patterns and might indicate rapid acceleration or expansion. If complex eigenvalues show up, they often indicate oscillatory behaviour; the eigenvectors of a fluid flow matrix can reveal the spatial distribution of the pattern. The eigenvectors with the largest magnitudes (in terms of their eigenvalues) typically represent the dominant modes of the flow. These are the patterns that have the most tremendous significance on overall behaviour. A **snapshot matrix** is a matrix where each column corresponds to the state of a system at a particular time, capturing the system's behavior at that instant. This matrix is commonly used in methods such as Proper Orthogonal Decomposition (POD) to represent time-dependent data, facilitating dimensionality reduction and analysis of dominant modes of the system's evolution over time. The columns of snapshot matrix are referred to as snapshots in this thesis.

2.3 Finite Element Method

The Finite Element Method (FEM) [2] is a powerful numerical technique used to approximate solutions to complex problems in engineering, physics, and applied mathematics. Developed in the mid-20th century, FEM originated from efforts to solve structural mechanics problems, particularly in aerospace engineering. It is grounded in variational calculus and partial differential equations (PDEs), allowing the analysis of the issues involving irregular geometries, heterogeneous materials, and boundary conditions that are difficult to handle analytically.

FEM works by dividing a complex domain into smaller, simpler subdomains called finite elements. The solution is approximated within each element using a set of basis functions, typically polynomials. These elements are connected at nodes, and the weak formulation of the governing equations governs their relationships. The global solution is obtained by assembling and solving a system of algebraic equations derived from the individual elements.

The following outlines the key steps of FEM [28], detailing the critical and systematic process used to approximate solutions:

Step 1: *Discretization* - Divide the problem domain into finite elements by creating a mesh.

Step 2: *Selection of Basis Functions* - Choose appropriate functions (e.g., linear or quadratic polynomials) to approximate the solution within each element.

Step 3: *Formulation of the Weak Form* - Incorporate known constraints (e.g., fixed values or fluxes) to modify the global system.

Step 4: *Application of Boundary Conditions* - Solve the resulting algebraic equations system to determine the nodes' unknown values.

Step 5: *Solution of Equations* - Visualize and interpret the numerical solution, such as stress, temperature, or velocity fields.

FEM is widely used across diverse fields. In structural engineering, it is employed to analyze stresses and deformations in buildings, bridges, and mechanical components. In fluid dynamics, modelling flow behaviour includes complex phenomena like turbulence and heat transfer. FEM is also crucial in electromagnetics for designing and optimizing devices like antennas and motors. Moreover, it plays a vital role in biomechanics, assisting in developing prosthetics and studying biological tissues under stress.

By enabling precise modelling of real-world problems, FEM has become an indispensable tool in scientific research and industrial applications. Its flexibility and adaptability make it a preferred choice for tackling modern challenges in computational modelling.

2.4 Navier Stokes Equation

Before jumping into equations, it is important to recall some of the following fundamental concepts of fluid dynamics.

Density: Represented by ρ, density is defined as the mass of a substance, denoted by m, per unit volume, denoted using V, and density is typically expressed as

$$\rho = \frac{m}{V}.$$

It describes how compact matter is packed within a given space.

Reynolds Number: It is a dimensionless quantity represented by Re that characterizes the ratio of inertial forces to viscous forces in a fluid flow. It determines the flow regime, distinguishing between laminar flow ($Re < 2000$) and turbulent flow ($Re > 4000$). Re is expressed as

$$Re = \frac{\rho v L}{\mu},$$

where L denotes the characteristic length and v denotes the fluid's velocity.

Viscosity: Viscosity measures a fluid's resistance to deformation or flow arising from internal friction between its layers. It is the ratio of shear stress to shear rate and quantifies how easily a fluid flows under applied force.

Kinematic Viscosity: Kinematic viscosity is the measure of a fluid's resistance to flow under gravity, defined as the ratio of dynamic viscosity to fluid density. It is expressed as

$$\nu = \frac{\mu}{\rho}.$$

Dynamic Viscosity: Dynamic viscosity measures a fluid's internal resistance to shear or flow, quantifying how much force is required to move one layer of fluid over another. The symbol μ represents it.

Compressible and Incompressible Flow: Compressible flow refers to fluid flow in which the fluid density changes significantly due to pressure variations, typically occurring at high velocities or in gases. Incompressible flow assumes constant density, where density changes are negligible compared to the pressure and velocity variations often applied to liquids at low speeds.

Now that we know the basic definitions, we can study the Navier-Stokes equations. These equations describe the motion of fluid substances, such as liquids and gases, forming the foundation of fluid dynamics. These nonlinear partial differential equations (PDEs) express the principles of conservation of mass, momentum, and energy within a fluid, accounting for viscosity and external forces. Widely used in engineering and science, they model diverse phenomena ranging from weather patterns and ocean currents to airflow over aircraft wings and blood flow in arteries.

Starting from the fundamental physical conservation principles, particularly those governing the principal of mass and momentum conservation, the Reynolds transport theorem (RTT) demonstrates that the following equation can be used to model the flow of an incompressible, viscous fluid with constant density.

$$\rho\partial_t v + \nabla p - \mu\nabla \cdot (\nabla v + \nabla v^T) + \rho(v \cdot \nabla)v = 0. \tag{2.2}$$

In equation (2.2), v represents a vector-valued function defined as $v : (0, T) \times \Omega \to \mathbb{R}^d$, where $d \in \{2, 3\}$, and p is a scalar-valued function defined as $p : (0, T) \times \Omega \to \mathbb{R}$. Here, ρ is the density of the fluid, ν stands for the kinematic viscosity, and the dynamic viscosity is defined using $\mu = \rho\nu$. Additionally, it is crucial to highlight that the Navier-Stokes equations in (2.2) incorporate the Cauchy stress tensor (CST) for an incompressible and Newtonian fluid, expressed as

$$\sigma = -pI + \mu(\nabla v + \nabla v^T).$$

Chapter 3

Singular Value Decomposition and Proper Orthogonal Decomposition

In this chapter, we introduce the concept of singular value decomposition (SVD) [27] and how it can be used for image compression and other fields. Then, proper orthogonal decomposition is introduced, focusing on the mathematical relationship between POD and SVD. While SVD is a foundational tool in linear algebra for analyzing and approximating matrices, POD leverages its principles to extract dominant patterns from complex datasets, making it indispensable in model reduction and fluid dynamics. Finally, we discuss weighted POD (WPOD), later used for POD mode analysis for fluid flow.

3.1 Singular Value Decomposition (SVD)

Every symmetric matrix A can be expressed as

$$A = PDP^T,$$

where $P \in \mathbb{R}^{n \times n}$ is an orthogonal matrix of eigenvectors of A, D is a diagonal matrix whose diagonal entries are the eigenvalues corresponding to the column vectors of P. This is called an eigenvalue decomposition (EVD). It is a powerful mathematical tool used across various fields, including engineering, physics, and image compression. Its ability to simplify complex problems and reveal underlying structures makes it invaluable in real-world scenarios.

If A is not symmetric, then it does not have an EVD, but it does have a Hessenberg decomposition, i.e

$$A = PHP^T,$$

where P is an orthogonal matrix, and H is a matrix in the upper Hessenberg form. Upper Hessenberg form matrices are $n \times n$ in which all entries above the first sub-diagonal are either real or complex values, and all entries below the sub-diagonal are 0s.

If the eigenvalues of the matrix A are found to be real, then it can have a Schur decomposition, where P is an orthogonal matrix and S is an upper triangular matrix. It can be represented as

$$A = PSP^T.$$

All the above eigenvalue decompositions are extremely useful. They all hold significant importance across various fields and present a variety of real-life applications. However, when they encounter a non-square matrix, they fail to serve. If an $m \times n$ matrix is needed to be decomposed, these techniques will be no good. That is where singular value decomposition [29] will present a much-needed solution.

Let A be a matrix. If it is an $m \times n$ matrix with a rank $k \leq \min(m, n)$, then A can be decomposed as

$$A = \mathbf{U}\mathbb{S}\mathcal{V}^T, \tag{3.1}$$

in which $\mathbf{U}$, $\mathbb{S}$, and $\mathcal{V}^T$ have sizes $m \times m$, $m \times n$, and $n \times n$, respectively.

In this decomposition, $\mathcal{V}^T = [\mathbf{v_1}\ \mathbf{v_2}\ \mathbf{v_3} \cdots \mathbf{v_n}]$ and it orthogonally diagonalizes A^TA. The non-zero diagonal entries of $\mathbb{S}$ are the singular values $\sigma_1 = \sqrt{\lambda_1}$, $\sigma_2 = \sqrt{\lambda_2}\ \cdots, \sigma_k = \sqrt{\lambda_k}$ where $\lambda_1, \lambda_2, \ldots, \lambda_k$ represent the non-zero eigenvalues of A^TA, associated with the column vectors of $\mathcal{V}$. These columns are arranged in descending order according to their corresponding singular values, such that $\sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_k \leq 0$. The matrix

$\mathbf{U}$ orthogonally diagonalizes AA^T such that $\mathbf{U} = [\mathbf{u_1}\ \mathbf{u_2}\ \mathbf{u_3}, ..., \mathbf{u_k}]$ is the orthonormal basis for the column space of A.

Due to the relations

$$A\mathbf{v}_i = \sigma_i \mathbf{u}_i,$$

and

$$A^T\mathbf{u}_i = \sigma_i \mathbf{v}_i,$$

it can be established that $[\mathbf{u}_i]_{i=1}^k$ and $[\mathbf{v}_i]_{i=1}^k$ are the eigenvectors of AA^T and A^TA respectively with eigenvalues $\lambda_i = \sigma_i^2 > 0$ for $1 \le i \le k$.

As the matrix A has a rank k, we can simplify our SVD to a reduced rank, i.e.

$$A = \mathbf{U}^{(k)} \mathbb{D} \left(\mathcal{V}^{(k)}\right)^T = \sum_{i=1}^{k} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \tag{3.2}$$

where $\mathbf{U}^{(k)} \in \mathbb{R}_{m\times k}$ and $\mathcal{V}^{(k)} \in \mathbb{R}_{n\times k}$ are the updated $\mathbf{U}$ and $\mathcal{V}$ that are reduced to k-columns. $\mathbb{D}$ is the diagonal matrix of the non-zero entries of $\mathbb{S}$. If we write $A = [\mathbf{a_1}\ \mathbf{a_2}\ \mathbf{a_3} \cdots \mathbf{a_m}]$, then we can say that

$$\mathbf{a}_j = \sum_{i=1}^{k} \left[D(\mathcal{V}^{(k)})^T\right] \mathbf{u}_i,$$

$$\implies \mathbf{a}_j = \sum_{i=1}^{k} \left[(\mathbf{U}^{(k)})^T \mathbf{U}^{(k)} D(\mathcal{V}^{(k)})^T\right] \mathbf{u}_i,$$

as $(\mathbf{U}^{(k)})^T\mathbf{U}^{(k)} = \mathbb{I} \in \mathbb{R}^{k\times k}$.

$$\implies \mathbf{a}_j = \sum_{i=1}^{k} \left[(\mathbf{U}^{(k)})^T A\right]_{ij} \mathbf{u}_i,$$

$$\implies \mathbf{a}_j = \sum_{i=1}^{k} (\mathbf{u}_i, \mathbf{a}_j)_{\mathbb{R}^m} \mathbf{u}_i.$$

This leads to the Fourier series representation of of $\mathbf{a}_j$ i.e.

$$\mathbf{a}_j = \sum_{i=1}^{k} (\mathbf{a}_j, \mathbf{u}_i)_{\mathbb{R}^m} \mathbf{u}_i.$$

3.1.0.1 SVD: Algorithm

The following algorithm, presented in 1, outlines the steps to compute the SVD of a given matrix A [30].

Algorithm 1 Singular Value Decomposition (SVD) Algorithm

```
1. Input: Matrix $A = [\mathbf{a}_j]_{j=1}^n \in \mathbb{R}^{m \times n}$
2. Output: Singular Value Decomposition of $A$ such that $A = \mathbf{U}\mathbb{S}\mathcal{V}^T$
3. Create the matrices $A' = AA^T$ and $A'' = A^T A$.
4. Steps:
    1. Calculate the eigenvalues of $A' = AA^T$ and the corresponding eigenvectors.
    2. Arrange the eigenvectors of $A' = AA^T$ in descending order of their corresponding eigenvalues and form the matrix $\mathbf{U}$.
    3. Calculate the eigenvalues $\lambda_i$ of $A'' = A^T A$ and compute $\sigma_i = \sqrt{\lambda_i}$.
    4. Create a diagonal matrix $\mathbb{S}$.
    5. Arrange the eigenvectors of $A'' = A^T A$ and form the matrix $\mathcal{V}$.
    6. Express the final result as $A = \mathbf{U}\mathbb{S}\mathcal{V}^T$.
```

3.1.1 Economy and Truncated SVD

In the previous section, when the number of columns of $\mathbf{U}$ and $\mathcal{V}$ were reduced concerning the non-zero diagonal entries of $\mathbb{S}$, the methodology of Economy SVD (ESVD) was utilized to reduce the computational cost and make problem-solving time efficient. Before properly discussing ESVD, taking a closer look at SVD is necessary.

In SVD, $\mathbf{U}$ comprises of k orthonormal columns that span column space of A, and the rest $m-k$ orthonormal columns span the left null space of A. The matrix $\mathbb{S}$ comprises singular values and is padded with zero rows and columns to construct the shape of an $m \times n$ matrix to enable the operations between the three matrices to be valid. And lastly, $\mathcal{V}$ comprises k orthonormal columns span the row space of A and $n-k$ orthonormal columns span the null space of A. That is the configuration we introduced in Equation 3.1.

In ESVD, the $m-k$ columns are removed from $\mathbf{U}$ and the $n-k$ columns are removed from $\mathcal{V}$. This is done so that the padded rows and columns can be removed from $\mathbb{S}$

while the dimensional consistency of the operation remains intact. This turns our SVD into the ESVD in Equation 3.2. It has been proved that ESVD and full SVD result in original A, meaning no data or information is lost.

Truncated Singular Value Decomposition (TSVD) [31] is a dimensionality reduction technique that simplifies a matrix while preserving its essential features. The matrix A made using TSVD approximates the original A. In this technique, the largest k' singular values of the matrix A^TA and AA^T are considered, such that $0 < k' < k$ along with their corresponding left and right singular vectors. The remaining singular values $k - k'$ are discarded along with their respective eigenvectors. This results in the following decomposition

$$A = \mathbf{U}^{(k')}\mathbb{D}_{k'}\left(\mathcal{V}^{(k')}\right)^T = \sum_{i=1}^{k'} \sigma_i \mathbf{u}_i \mathbf{v}_i^T, \tag{3.3}$$

where $\mathbf{U}^{(k')} \in \mathbb{R}_{m\times k'}$ and $\mathcal{V}^{(k')} \in \mathbb{R}_{n\times k'}$ are the updated $\mathbf{U}$ and $\mathcal{V}$ that are reduced to k'-columns. $\mathbb{D}_{k'}$ is the diagonal matrix made up the largest k' singular values.
Discarding small magnitude singular values and their corresponding eigenvectors can save significant space, and the computational costs are lowered. Singular values tend to decay rapidly. If the singular values σ_i are observed in descending order, each singular value has a drastic drop in magnitude. This means that the highest information about the system is in high-magnitude singular values. Let's look at an example.

Example 3.1. *Consider the matrix*

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 9 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 8 & 9 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 7 & 8 & 9 & 1 & 2 & 3 & 4 & 5 & 6 \\ 6 & 7 & 8 & 9 & 1 & 2 & 3 & 4 & 5 \\ 5 & 6 & 7 & 8 & 9 & 1 & 2 & 3 & 4 \\ 4 & 5 & 6 & 7 & 8 & 9 & 1 & 2 & 3 \\ 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & 2 \\ 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 \end{pmatrix}.$$

It can be seen that $A \in \mathbb{R}^{9\times 9}$*. Following the SVD algorithm, the matrix* A^TA *and* AA^T *are computed first. The left singular vectors make up* $\mathbf{U} \in \mathbb{R}^{9\times 9}$ *that orthogonally diagonalizes* AA^T *and the right singular vectors make up* $\mathcal{V} \in \mathbb{R}^{9\times 9}$ *that orthogonally diagonalizes* A^TA*. The singular values of the matrix* A *can be obtained by taking square roots of the eigenvalues of the matrix achieved by doing* A^TA*, and these values are used to construct the diagonal matrix* $\mathbb{S} \in \mathbb{R}^{9\times 9}$*.*

Together, this gives the SVD of A*, i.e.*

$$A = \mathbf{U}\mathbb{S}\mathcal{V}^T.$$

In TSVD, only the singular values with the highest magnitudes are retained, as they contain the most essential information. The line graph in Figure 3.1 clearly shows that the eigenvalues decline rapidly, indicating that most valuable information is concentrated in the first few eigenvalues. Only nine eigenvalues existed in the matrix A presented in this example. However, for larger matrices, the eigenvalues will continue to decrease, making the largest ones increasingly significant for capturing the key features of the data.

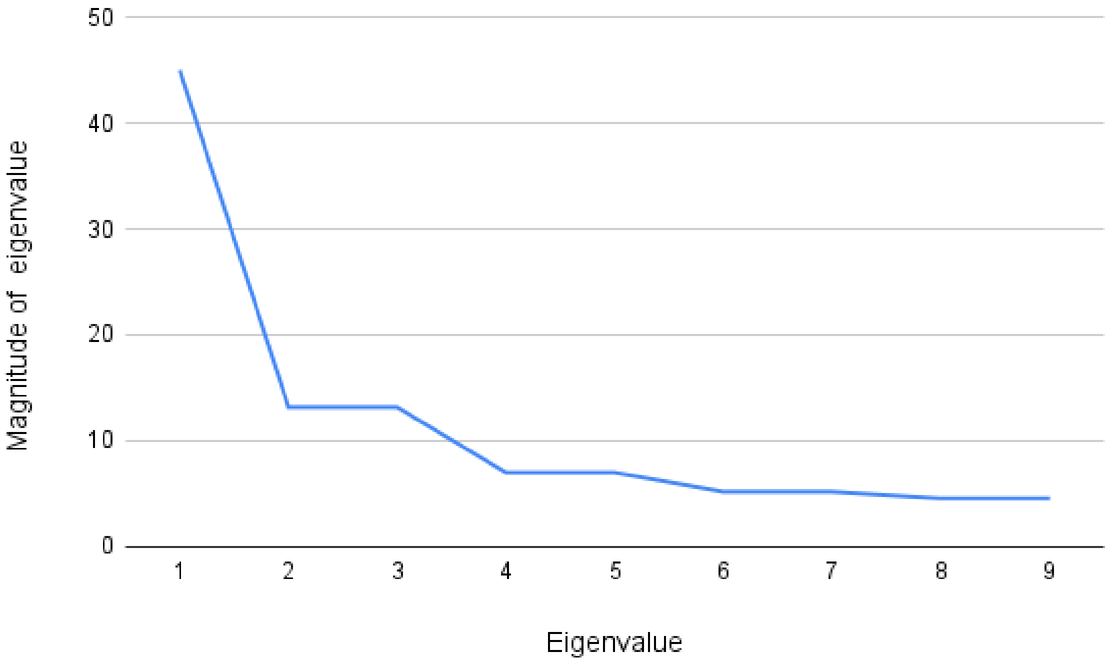


FIGURE 3.1: Eigenvalues of the matrix A in Example 3.1

TSVD is a popular and versatile technique used in various fields for dimensionality reduction, data compression, and noise reduction. It is used for noise reduction as it filters out noise from data in audio and signals. This increases the quality and clarity of the domain. TSVD helps with latent semantic analysis (LSA) in text data analysis to reduce the dimensions of term-document matrices, allowing for the underlying patterns in text. In bioinformatics, when dealing with high-dimension genomic data, TSVD is a tool that bioinformaticians often fall back on. It helps identify the relationship amongst genes and aids with discovering biomarkers, which leads to a better understanding of the biological processes. Similarly, it helps in various real-life applications such as face recognition, data processing [32], noise reduction, and image compression.

3.1.2 Image Compression

With a small demonstration of how images are compressed [33] with the help of SVD and TSVD, the importance of these techniques can be cemented. Given that an image A is considered with the rank k, such that $A \in \mathbb{R}^{m \times n}$. By reducing the rank to k' such that $k' \ll k$, we have the matrix $\hat{A} \in \mathbb{R}^{m \times n}$. Interested readers can turn to [34–36] for more information on image compression.

We are interested in a $\hat{A}$ such that $||A - \hat{A}||$ is minimized. The Eckart-Young Theorem [37] gives us that

$$\min_{rank(\hat{A}=k')} ||A - \hat{A}|| = ||A - A^{(k')}|| = \begin{cases} \sigma_{k'+1} & \text{for the 2-norm ,} \\ \sqrt{\sum_{i=k'+1}^{k} \sigma_i^2} & \text{for the Frobenius norm.} \end{cases} \tag{3.4}$$

The TSVD matrix in Equation (3.2) solves the minimization problem.

Now, let's take a look at an image compression example. We use an image with dimensions 600 × 402 and 241200 pixels. The image is decomposed into three colour channels: red, green and blue. Each channel has values ranging from 0 (representing black) to 255 (representing white). In Figure 3.2, the original image is compared to its TSVD approximations of rank 100, 20, 5, and 3.

Looking closely at the figure, we can observe the effect rank reduction using truncated SVD has on image compression. In Figure 3.2, the first subfigure shows the original image in all clarity. As the rank is reduced to k'=100, at first glance, the integrity of the image remains intact. However, when closely observed, a slight grainy texture can be observed. When the rank is further reduced to 20, image remains visible, but blurring has become apparent. However, despite losing 80 singular values, not much is lost from the picture, and the key components are preserved. Finally, loss is observed when rank is dropped to 5. The image is now unreadable. The image's general colour scheme is visible but has become undetectable. Lastly, further blurring is observed when the rank is reduced to 3, and the image has become highly unreadable. This means that significant information about the image has been lost during compression.

But how do we choose the ideal k'? The hit and trial method is mostly used in practice, considering that the k' value should be large enough such that the energy ratio between $A^{k'}$ and A is bigger than a certain threshold. This means we have to find the smallest k' such that

$$\mathbb{E} := \frac{\sum_{i=1}^{k'} \sigma_i}{\sum_{i=1}^{k} \sigma_i} \geq 0.99. \tag{3.5}$$

The formula in 3.5 describes the energy ratio of the energy $A^{(k')}$ to the energy of A. We want to be as close to 1 as possible. In the case of 3.1, the following tables represent the energy ratio for each value of k'.

TABLE 3.1: Energy ratios for different values of k'

k'	$\mathbb{E}$
1	0.4292
2	0.5547
3	0.6802
4	0.7470
5	0.8137
6	0.8633
7	0.9128
8	0.9564
9	1.0000

This shows that the most significant few eigenvalues, even in a small-scale example, contain the maximum information about the system. When Proper Orthogonal Decomposition is introduced later for Partial Differential Equations, eigenvalues are used instead of using singular values to calculate $\mathbb{E}$. The energy ratios indicate that over 50 % of the energy of the date is conserved for $k' = 2$. Over 80 % of the energy is conserved for $k' = 5$. And over 90 % of the energy is conserved for $k' = 7$. This demonstrates the effectiveness of using a reduced number of modes to approximate the system, as even a small subset of the eigenvalues or singular values retains a significant portion of the system's energy, leading to efficient dimensionality reduction.

3.2 Proper Orthogonal Decomposition (POD)

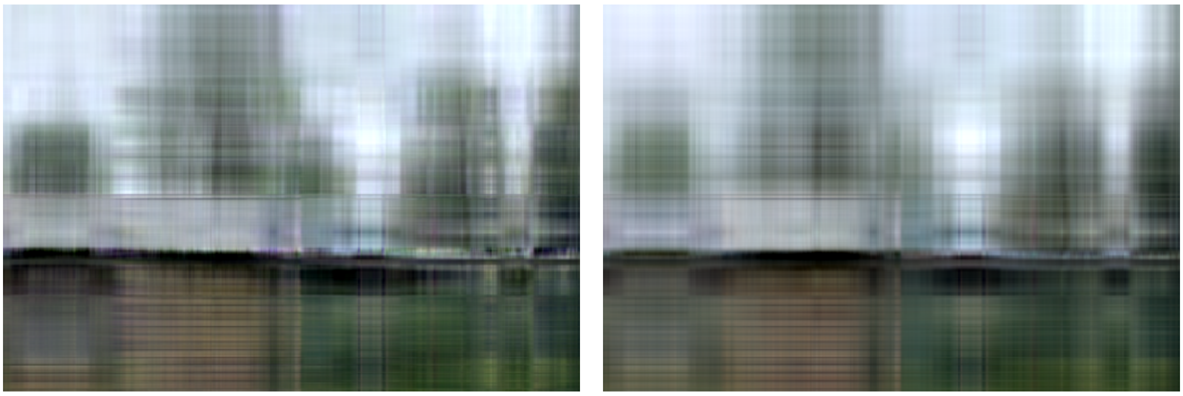
Proper Orthogonal Decomposition (POD) is a mathematical technique used to analyze and reduce the complexity of dynamic systems by identifying the most significant patterns or modes in the data. These modes provide a framework for extracting meaningful information from complex data, facilitating better insights and decision-making. POD involves decomposing a dataset, typically represented as a matrix of observations, into orthogonal components that capture the essential features of the original data. This decomposition is achieved by the application of Singular Value Decomposition (SVD) on the data matrix. It results in a set of orthogonal basis functions, each associated with a singular value that indicates its importance.

POD is handy in various applications, including fluid dynamics, structural analysis, and image processing, as it allows for efficient representation and reconstruction of data while preserving critical information. POD facilitates effective modelling, simulation, and data compression by focusing on the dominant modes, making it a valuable tool in theoretical and applied contexts. POD modes have been used in many other research disciplines and are sometimes referred to as the Karhunen-Loève decomposition [38] in stochastic. It is also used in Principal Component Analysis [39], Hotelling transform [40] and empirical orthogonal functions [41]. Many other research works provide an in-depth analysis of POD and its many other applications [42, 43].

Original Image

Rank 100 image

Rank 20 image



Rank 5 image

Rank 3 image

Figure 3.2: Comparison of the original image with different rank images

Let's explore the derivation of POD through a sequence of optimization problems, highlighting its relationship with singular value decomposition. Our approach involves iteratively computing an orthonormal basis that best approximates the dataset $[\mathbf{a}_i]_{i=1}^{n}$. The vectors produced from this process, denoted as $[\mathbf{u}_i]_{i=1}^{k'}$, will be referred to as the POD basis of rank k'. We will also call these vectors the POD modes or POD basis vectors. In this context, the maximization problem is solved by the first POD basis vector.

$$\max_{\hat{\mathbf{u}}_1 \in \mathbb{R}^m} \sum_{j=1}^{n} |(\mathbf{a}_j, \hat{\mathbf{u}}_1)|_{\mathbb{R}^m}^2 \text{ such that } ||\hat{\mathbf{u}}_1||_{\mathbb{R}^m}^2 = 1. \tag{3.6}$$

Equation (3.6), is the maximization problem, and it can be expressed that the left singular vectors of the matrix $\mathbf{u}_1$, obtained through SVD in Equation 3.1, solves this problem. The maximization problem involves the search for a vector that captures the maximum variance from the data set $\mathbf{a}_i$. By maximizing the expression in 3.7, the direction in which data is changing the most is observed, and the solution $\mathbf{u}_1$ will be one that will capture the most significant features of the data set. The second basis vector of the POD is achieved by finding the correct solution of

$$\max_{\hat{\mathbf{u}}_2 \in \mathbb{R}^m} \sum_{j=1}^{n} |(\mathbf{a}_j, \hat{\mathbf{u}}_2)|_{\mathbb{R}^m}^2 \text{ such that } ||\hat{\mathbf{u}}_2||_{\mathbb{R}^m}^2 = 1, \ \ (\mathbf{u}_1, \hat{\mathbf{u}}_2)_{\mathbb{R}^m} = 1. \tag{3.7}$$

The solution of this maximization problem is the second left singular vector $\mathbf{u}_2$. This process is carried on iteratively. After the first POD basis vector is determined, the process is repeated for $(\mathbf{u}_2, \mathbf{u}_2, \cdots)$, each time ensuring that the new vector is orthogonal to the previous vectors. This produces a set of orthonormal basis vectors that represent the whole dataset. This procedure leads us to the following theorem.

Theorem 3.1. ***POD Basis:*** *Let* $A = [\mathbf{a}_1, \mathbf{a}_2 \cdots, \mathbf{a}_n] \in \mathbb{R}^{m \times n}$ *with rank* $d \leq \min(m, n)$ *have the singular value decomposition of* $Y = \mathbf{U}\mathbb{S}\mathcal{V}^T$ *as shown in Equation 3.1. Then the optimization problem*

$$\max_{\hat{\mathbf{u}}_1, \cdots, \hat{\mathbf{u}}_{k'} \in \mathbb{R}^m} \sum_{i=1}^{k'} \sum_{j=1}^{n} |(\mathbf{a}_j, \hat{\mathbf{u}}_i)|_{\mathbb{R}^m}^2 \ \textit{such that} \ (\hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j)_{\mathbb{R}^m} = \delta_{ij} \ \ \forall\, a \leq i, \ j \geq r, \tag{3.8}$$

is solved by the left singular vectors $[\mathbf{u}_i]_{i=1}^{k'}$ *and it holds that*

$$\arg\max\left(\max_{\hat{\mathbf{u}}_1,\cdots,\hat{\mathbf{u}}_{k'}\in\mathbb{R}^m}\sum_{i=1}^{k'}\sum_{j=1}^{n}|(\mathbf{a}_j,\hat{\mathbf{u}}_i)|^2_{\mathbb{R}^m}\right)=\sum_{i=1}^{k'}\sigma_i^2=\sum_{i=1}^{k'}\lambda_i.$$

The above theorem presents a connection between SVD and POD, and the optimality of the POD basis can be double-checked and verified through the Eckart-Young Theorem (EYT) from Equation (3.4). This means that for all $\hat{A}\in\mathbb{R}^{m\times n}$ with rank=k'

$$||A-A^{(k')}||_F^2=||A-\hat{A}||_F^2. \tag{3.9}$$

The optimal rank approximation of $A^{(k')}$ can be written as

$$A^{(k')}=\mathbf{U}^{(k')}\mathbb{S}^{(k')}\left(\mathcal{V}^{(k')}\right)^T:=\mathbb{B}^{(k')},$$

where $\mathbf{U}^{(k')}=[\mathbf{u_1}\ \ \mathbf{u_2}\ \ \mathbf{u_3}\cdots\mathbf{u_{k'}}]\in\mathbb{R}^{m\times r}$, $\mathcal{V}^{(k')}=[\mathbf{v_1}\ \ \mathbf{v_2}\ \ \mathbf{v_3}\cdots\mathbf{v_{k'}}]\in\mathbb{R}^{m\times r}$, and $\mathbb{S}^{(k')}:=diag(\mathbb{S}_{11},\cdots,\mathbb{S}_{k'k'})$. Substituting this into (3.9) and using (3.4), we get

$$||A-A^{(k')}||_F^2=||A-\mathbf{U}^{(k')}\mathbb{B}^{(k')}||_F^2=\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}-\sum_{l=1}^{k'}\mathbf{U}_{il}^{(k')}\mathbb{B}_{jl}^{k'}|^2,$$

$$\implies ||A-A^{(k')}||_F^2=\sum_{i=1}^{m}\sum_{j=1}^{n}|A_{ij}-\sum_{l=1}^{(k')}(\mathbf{a}_j,\mathbf{u}_l)_{\mathbb{R}^m}\mathbf{u}_{il}^{(k')}|^2,$$

$$\implies ||A-A^{(k')}||_F^2=\sum_{j=1}^{n}||\mathbf{a}_j-\sum_{l=1}^{k'}(\mathbf{a}_j,\mathbf{u}_l)_{\mathbb{R}^m}\mathbf{u}_l||^2_{\mathbb{R}^m}.$$

Now, let us consider a different decomposition of A such that

$$A=\hat{\mathbf{U}}^{(k)}\mathbb{C}^{(k)},$$

where, $\mathbf{U}^{(k)}=[\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_k]\in\mathbb{R}^{m\times k}$, the columns of $[\mathbf{u}_1]_{i=1}^{k'}$ are orthogonal to each other, and $\mathbb{C}_{ij}^{(k')}=(\hat{\mathbf{u}}_i,\mathbf{a}_j)$ for $1\leq i\leq k,\ 1\leq j\leq n$. For this decomposition, if we take $\hat{A}=\hat{\mathbf{U}}^{(k')}\mathbb{C}^{(k')}$, such that $\hat{\mathbf{U}}^{(k')}\in\mathbb{R}^{m\times k'}$ is the matrix composed of the first k' columns of $\mathbf{U}^{(k)}$, and $\mathbb{C}^{(k')}$ is the matrix made up of the first k' rows of $\mathbb{C}^{(k)}$. Using this, we obtain

$$||A - \hat{A}||_F^2 = \sum_{j=1}^{n} ||\mathbf{a}_j - \sum_{l=1}^{k'} (\mathbf{a}_j, \hat{\mathbf{u}}_l)_{\mathbb{R}^m} \hat{\mathbf{u}}_l||^2$$

and after using the EYT from (3.4) for Frobenius norm, the following is achieved

$$\|A - A^{(k')}\|_F^2 = \sum_{j=1}^{n} \left\| \mathbf{a}_j - \sum_{l=1}^{k'} (\mathbf{a}_j, \mathbf{u}_l)_{\mathbb{R}^m} \mathbf{u}_l \right\|^2 \leq \|A - \hat{A}\|_F^2 = \sum_{j=1}^{n} \left\| \mathbf{a}_j - \sum_{l=1}^{k'} (\mathbf{a}_j, \hat{\mathbf{u}}_l)_{\mathbb{R}^m} \hat{\mathbf{u}}_l \right\|_{\mathbb{R}^m}^2 .$$

Now, we have shown that

$$\min_{\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_{k'} \in \mathbb{R}^m} \sum_{j=1}^{n} \left\| \mathbf{a}_j - \sum_{l=1}^{k'} (\mathbf{a}_j, \hat{\mathbf{u}}_i)_{\mathbb{R}^m} \hat{\mathbf{u}}_l \right\|_{\mathbb{R}^m}^2 , \tag{3.10}$$

given that $(\hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j)_{\mathbb{R}^m} = \delta_{ij} \;\; \forall 1 \leq i, \; j \leq k'$.

Equation 3.10 is an optimization-related problem which is equivalent to 3.8. The POD of the basis of rank k' are the modes that have the highest energy, and they are the modes with the minimum error between $\mathbf{a}_j$ and $\mathbf{a}_j^{(k')}$ and these two conditions are satisfied by the first k' left singular vectors of A.

3.2.1 Classical Approach to POD

In real-life applications, computing the singular value decomposition of a matrix A can be computationally expensive and infeasible. That is because the matrix can be very dense, with large values of m and n. As the matrix can represent an audio file, image, database, or even the state of a dynamical system over time, it can result in a heavy matrix whose SVD is impractical. To fix this problem, the $m \times n$ matrix can be converted to an $m \times m$ or $n \times n$ matrix, depending on whether $m << n$ or $n << m$, respectively. By creating a square matrix from a non-square matrix, our problem of finding the POD basis can turn into an eigenvalue problem, which is a much easier problem to solve.

That is the purpose of Lumley's classical method [44] where an $m \times m$ eigenvalue problem is solve with eigenvectors $[\mathbf{u}_i]_{k'}^{i=1}$ and the eigenvalues are $\lambda = \sigma^2$ for $1 \leq i \leq k'$. The eigenvalue problem is

$$AA_T \mathbf{u}_i = \lambda \mathbf{u}_i.$$

However, this method is only used when the number of rows is significantly less than the number of columns, i.e. $m << n$. When doing the POD of a PDE, the number of rows m is the number of degrees of freedom, and the number of columns n are the times steps. Each column represents the positions at a unique time point. This means that the number of rows will exceed the number of columns. Hence, this isn't a suitable for the reduced order modeling (ROM) of PDEs.

In various applications of POD, the method of snapshots (MOS) [45–47] is adopted to produce the best results. The MOS is used in numerical simulations and scientific computing, and it involves capturing the state of a system at different time points to analyse the behaviour over time. Introduced in 1987 by Sirovich, the method of snapshots is widely used for reduced order modeling of PDEs. Firstly, we must solve an $n \times n$ eigenvalue problem.

$$A^T A\mathbf{v}_i = \lambda_i \mathbf{v}_i \quad \text{for } 1 \leq i \leq k'.$$

The right singular vectors of this problem act as the eigenvectors of the EVP $[\mathbf{v}_i]_{i=1}^{k'}$ and the eigenvalues of the matrix are denoted by $\lambda_i = \sigma_i^2$ for $1 \leq I \leq k'$. Utilizing the right singular vectors, we can easily calculate the left singular vectors $[\mathbf{u}_i]_{i=1}^{k'}$, which according to (3.9) are the POD basis vectors. We use the following calculation $\mathbf{u}_i = \frac{1}{\sqrt{\lambda_i}} A\mathbf{v}_i$ for $1 \leq i \leq k'$. We must also decide on the k' that works for the model. In real-world problems, each situation requires a unique k' or minimum rank, for which all necessary information is preserved. This formula allows for easy selection of k'

$$\mathbb{E} := \frac{\sum_{i=1}^{k'} \lambda_i}{\sum_{i=1}^{k} \lambda_i} \geq \delta_e, \tag{3.11}$$

where $\mathbb{E}$ is the notation of energy ratio and the problem's threshold, represented by $\delta_e \in [0, 1]$, is fixed. The purpose of POD is to ensure that maximum energy is being preserved, δ_e should be close to 1. For example, $\delta_e = 0.99$ means the choice of k' is powerful. It should also be noted that, unlike the case of image compression, we use λ values in the energy ratio instead of the singular values λ.

3.2.2 POD: Algorithm

The following algorithm in Table 2 outlines the steps for implementing Proper Orthogonal Decomposition (POD). Each step ensures that the resulting basis captures the essential features while minimizing computational complexity.

Algorithm 2 Algorithm for POD [23, 48]

```
1. Input: The set of snapshots $[\mathbf{a}_j]_{j=1}^{n}$ and the threshold defined by $\delta_e \in [0,1]$.

2. Output: The POD basis $[\mathbf{u}_i]_{i=1}^{k'}$ and eigenvalues $[\lambda_i]_{i=1}^{k'}$ of the matrix.

3. Set $A = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n] \in \mathbb{R}^{m \times n}$.

4. Steps:

    1. Case 1: If $m \approx n$, then
        a) Compute SVD: $[\mathbf{U}, \mathbb{S}, \mathcal{V}] = \mathrm{SVD}(A)$.
        b) Compute $k'$ as the smallest integer such that $\frac{\sum_{i=1}^{k'} \mathbb{S}_{ii}^2}{\sum_{i=1}^{k} \mathbb{S}_{ii}^2} \geq \delta_e$.
        c) Set $\lambda_i = \mathbb{S}_{ii}^2$ and $\mathbf{u}_i = \mathbf{U}$ for $1 \leq i \leq k'$.

    2. Case 2: If $m \ll n$, then
        a) Perform eigenvalue decomposition on $AA^T$: $[\mathbf{U}, \Lambda] = \mathrm{EVD}(AA^T)$, where $AA^T \in \mathbb{R}^{m \times m}$.
        b) Compute $k'$ as in Case 1.
        c) Set $\lambda_i = \Lambda_{ii}^2$ and $\mathbf{u}_i = \mathbf{U}$ for $1 \leq i \leq k'$.

    3. Case 3: If $n \ll m$, then
        a) Perform eigenvalue decomposition on $A^TA$: $[\mathcal{V}, \Lambda] = \mathrm{EVD}(A^TA)$, where $A^TA \in \mathbb{R}^{n \times n}$.
        b) Compute $k'$ as in Case 1.
        c) Set $\lambda_i = \Lambda_{ii}^2$ and $\mathbf{u}_i = \frac{A\mathcal{V}}{\sqrt{\lambda_i}}$ for $1 \leq i \leq k'$.
```

3.3 Proper Orthogonal Decomposition (POD) with Weighted Inner Product

We will work with the following L^2-inner product in the next chapters. Let vectors $\mathbf{u}$ and $\mathbf{v}$ be two functions such that

$$\mathbf{u} := \sum_{j=1}^{m} \mathbf{u}_j^h \phi_j^h \in L^2(\Omega),$$

and

$$\mathbf{v} := \sum_{j=1}^{m} \mathbf{v}_j^h \phi_j^h \in L^2(\Omega),$$

then their L^2-inner product will be defined as

$$(\mathbf{u}, \mathbf{v})_{L^2(\Omega)} = \left(\sum_{j=1}^{m} \mathbf{u}_j^h \phi_j^h, \sum_{j=1}^{m} \mathbf{v}_j^h \phi_j^h \right)_{L^2(\Omega)} = \sum_{i,j=1}^{m} \mathbf{u}_j^h (\phi_j^h, \phi_i^h)_{L^2(\Omega)} \mathbf{v}_i^h = (\mathbf{u}^h)^T N_h \mathbf{v}^h.$$

Let $N_h \in \mathbb{R}^{m \times m}$ denote a mass matrix, where the entries are given by $(N_h)_{ij} = (\phi_j^h, \phi_i^h)_{L^2}$ for the values $1 \leq i, j \leq m$. It is of high importance to note that the L^2-inner product between the functions $\mathbf{u}$ and $\mathbf{v}$ Is equivalent to the weighted inner product of the vectors that represent their coefficients. $\mathbf{u}^h = (\mathbf{u}_1^h, \dots, \mathbf{u}_m^h) \in \mathbb{R}^m$ and $\mathbf{v}^h = (\mathbf{v}_1^h, \dots, \mathbf{v}_m^h) \in \mathbb{R}^m$. We can extend our POD method to Weighted Proper Orthogonal Decomposition (WPOD) using this inner product. This technique enhances the traditional Proper Orthogonal Decomposition (POD) by incorporating a weighting mechanism that addresses specific features or regions of interest in the data. While standard POD identifies dominant modes of variation in a dataset, it treats all data points equally. This can be a limitation in scenarios where certain areas are more significant, such as fluid dynamics or structural analysis. WPOD allows for the prioritization of specific measurements or regions, effectively emphasizing important structures while minimizing the influence of noise or less relevant data. This results in a more accurate and representative decomposition, improving understanding and modelling of complex systems. Let us consider a weighted inner product

$$(\mathbf{u}, \mathbf{v})_W := \mathbf{u}^T W \mathbf{v} = ((\mathbf{u}, W\mathbf{v}))_{\mathbb{R}}^m.$$

For two vectors $\mathbf{u}$ and $\mathbf{v}$, both belonging to the vector space $\mathbb{R}^m$, and considering a symmetric, positive definite matrix W that resides in the space $\mathbb{R}^{m \times m}$, the weighted inner product is used to define the norm $||\mathbf{u}||_W$, which can be expressed as $\sqrt{(\mathbf{u}, \mathbf{u})_W}$ for any vector $\mathbf{u} \in \mathbb{R}^m$. In the particular case where the matrix W is chosen to be the identity matrix in $\mathbb{R}^{m \times m}$, the weighted inner product undergoes a simplification and effectively reduces to the well-known standard Euclidean inner product. Similarly to the POD basis vectors discussed in the previous section, our first maximization problem is solved by the first POD basis vector in this context.

$$\max_{\hat{\mathbf{u}}_1\in\mathbb{R}^m}\sum_{j=1}^{n}|(\mathbf{a}_j,\hat{\mathbf{u}}_1)^W|^2_{\mathbb{R}^m}\ \text{ such that }\ ||\hat{\mathbf{u}}_1||^2_{\mathbb{R}^m}=1. \tag{3.12}$$

In a similar manner, the second POD basis vector is obtained by solving

$$\max_{\hat{\mathbf{u}}_2\in\mathbb{R}^m}\sum_{j=1}^{n}|(\mathbf{a}_j,\hat{\mathbf{u}}_2)_W|^2_{\mathbb{R}^m}\ \text{ such that }\ ||\hat{\mathbf{u}}_2||^2_{\mathbb{R}^m}=1,\ \ (\mathbf{u}_1,\hat{\mathbf{u}}_2)_{\mathbb{R}^m}=1. \tag{3.13}$$

This leads us to the following theorem.

Theorem 3.2. ***(POD basis with weighted inner product:)*** *Let $A=[\mathbf{a}_1,...,\mathbf{a}_n]\in\mathbb{R}^m$ with a rank $k\leq\min(m,n)$, $W\in\mathbb{R}^m$ is a symmetric and positive definite matrix, and $\overline{A}=W^{\frac{1}{2}}A$. Moreover, let $\overline{A}=\overline{\mathbf{U}}\mathbb{S}\overline{\mathcal{V}}^T$ be the singular value decomposition of $\overline{A}$, where $\overline{\mathbf{U}}=[\overline{\mathbf{u}}_1,...,\overline{\mathbf{u}}_m]\in\mathbb{R}^{m\times m}$ and $\overline{\mathbf{V}}=[\overline{\mathbf{v}}_1,...,\overline{\mathbf{v}}_m]\in\mathbb{R}^{n\times n}$ are orthogonal matrices, and $\mathbb{S}\in\mathbb{R}^{m\times n}$ is a diagonal matrix with entries $(\sigma_1,\sigma_2,...,\sigma_k)$ in descending order. Then for $1\leq k'\leq k$ the optimization problem*

$$\max_{\hat{\mathbf{u}}_1,\cdots,\hat{\mathbf{u}}_{k'}\in\mathbb{R}^m}\sum_{i=1}^{k'}\sum_{j=1}^{n}|(\mathbf{a}_j,\hat{\mathbf{u}}_i)_W|^2\ \textit{ such that }\ (\hat{\mathbf{u}}_i,\hat{\mathbf{u}}_j)_W=\delta_{ij}\ \ \forall\ 1\leq i,\ j\geq r, \tag{3.14}$$

is solved using the collection of vectors $\mathbf{u}_i=W^{-\frac{1}{2}}\overline{\mathbf{u}}_i$ for the condition $1\leq i\leq r$. We can hold that

$$\arg\max\left(\max_{\hat{\mathbf{u}}_1,\cdots,\hat{\mathbf{u}}_{k'}\in\mathbb{R}^m}\sum_{i=1}^{k'}\sum_{j=1}^{n}|(\mathbf{a}_j,\hat{\mathbf{u}}_i)_W|^2\right)=\sum_{i=1}^{k'}\sigma_i^2=\sum_{i=1}^{k'}\lambda_i.$$

Following this theorem, we can derive an equivalent relation, i.e.,

$$\min_{\hat{\mathbf{u}}_1,...,\hat{\mathbf{u}}_{k'}\in\mathbb{R}^m}\sum_{j=1}^{n}\left\|\mathbf{a}_j-\sum_{i=1}^{k'}(\mathbf{a}_j,\hat{\mathbf{u}}_i)_W\hat{\mathbf{u}}_i\right\|^2_W, \tag{3.15}$$

given that $(\hat{\mathbf{u}}_i,\hat{\mathbf{u}}_j)_W=\delta_{ij}\ \ \forall 1\leq i,\ j\leq k'$.

If we observe the relation $\overline{A}^T\overline{A}=\overline{A}^TW\overline{A}$ and due to SVD, the MOS will become: Solve an $n\times n$ EVP

$$Y^TAY\overline{\mathbf{v}}_i=\lambda_i\overline{\mathbf{v}}_i,$$

for $1 \leq i \leq k'$ and set

$$\mathbf{u}_i = W^{-\frac{1}{2}}\overline{\mathbf{u}}_i = \frac{1}{\lambda_i}W^{-\frac{1}{2}}\left(\overline{A}\overline{\mathbf{v}}_i\right) = \frac{1}{\sqrt{\lambda_i}}W^{-\frac{1}{2}}W^{\frac{1}{2}}A\overline{\mathbf{v}}_i = \frac{1}{\sqrt{\lambda_i}}A\overline{\mathbf{v}}_i \text{ for } for 1 \leq i \leq k'.$$

It can be observed that the matrix $W^{\frac{1}{2}}$ is not needed for the method of snapshot, i.e.

$$(\mathbf{u}_i, \mathbf{u}_j)_W = \mathbf{u}_i^T W \mathbf{u}_j = \frac{\delta_{ij}\lambda_j}{\sqrt{\lambda_i \lambda_j}} \;\; \forall\, 1 \leq i, j \leq k'.$$

3.3.0.1 WPOD: Algorithm

The following algorithm in Table 3 outlines the steps for implementing Weighted Proper Orthogonal Decomposition (WPOD).

Algorithm 3 Algorithm for WPOD [48]

```
1. Input: Snapshots $[\mathbf{a}_j]_{j=1}^{n} \subset \mathbb{R}^m$, the positive, symmetric, and definite matrix $W \in \mathbb{R}^{m\times m}$, and threshold of the problem $\delta_e \in [0,1]$.
2. Output: The POD basis $[\mathbf{u}_i]_{i=1}^{k'} \subset \mathbb{R}^m$ and eigenvalues $[\lambda_i]_{i=1}^{k'}$.
3. Set $A = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_n] \in \mathbb{R}^{m\times n}$.
4. Steps:
    1. Case 1: If $m \approx n$, then
        a) Calculate $\overline{A} = W^{\frac{1}{2}}A \in \mathbb{R}^{m\times n}$.
        b) Compute SVD: $[\overline{\mathbf{U}}, \mathbb{S}, \overline{\mathcal{V}}] = \text{SVD}(\overline{A})$.
        c) Compute $k'$ as the smallest integer such that $\frac{\sum_{i=1}^{k'} \mathbb{S}_{ii}^2}{\sum_{i=1}^{k} \mathbb{S}_{ii}^2} \geq \delta_e$.
        d) Set $\lambda_i = \mathbb{S}_{ii}^2$ and $\mathbf{u}_i = W^{-\frac{1}{2}}\mathbf{U}$ for $1 \leq i \leq k'$.
    2. Case 2: If $m \ll n$, then
        a) Calculate $\overline{A} = W^{\frac{1}{2}}A \in \mathbb{R}^{m\times n}$.
        b) Compute the eigenvalue decomposition: $[\overline{\mathbf{U}}, \Lambda] = \text{EVD}(\overline{A}\overline{A}^T)$, where $\overline{A}\overline{A}^T \in \mathbb{R}^{m\times m}$.
        c) Compute $k'$ as in Case 1.
        d) Set $\lambda_i = \Lambda_{ii}^2$ and $\mathbf{u}_i = W^{-\frac{1}{2}}\mathbf{U}$ for $1 \leq i \leq k'$.
    3. Case 3: If $n \ll m$, then
        a) Compute the eigenvalue decomposition: $[\overline{\mathcal{V}}, \Lambda] = \text{EVD}(A^T W A)$, where $A^T W A \in \mathbb{R}^{n\times n}$.
        b) Compute $k'$ as in Case 1.
        c) Set $\lambda_i = \Lambda_{ii}^2$ and $\mathbf{u}_i = \frac{A\mathcal{V}}{\sqrt{\lambda_i}}$ for $1 \leq i \leq k'$.
```

Chapter 4

FEM and POD For Navier Stokes Equations Around a Circular Cylinder

This chapter will dive into the finite element simulations of Navier Stokes equation. A well-known textbook on FEM is [49]. The interested reader can find more information in [50] and [51]. Firstly, we will use the Navier Stokes Equations (NSE) in 2D-2 setup from Schäfer et al. experimental research work [22]. We begin by deriving the weak form of the Navier Stokes equations in (2.2). This is achieved by multiplying the equations with a test function, performing integration by parts over the domain Ω, and enforcing the boundary conditions (BCs). Weak form and the Navier Stokes geometry from Schäfer et al. are provided to FreeFEM++ [52], which helps model the fluid flow and provide real-time simulations. Drag and lift coefficient values are collected here, and the fluid flow is analyzed. Finally, resulting modes of the problem [48] are analyzed.

4.1 Problem Setup

The analysis of the 2D-2 from the Navier-Stokes (NS) benchmark examples for numerical simulations is presented by Schäfer et al. [22]. This 2D-2 benchmark problem describes fluid flow going around a circular cylinder. Fluid's flow is characterized as

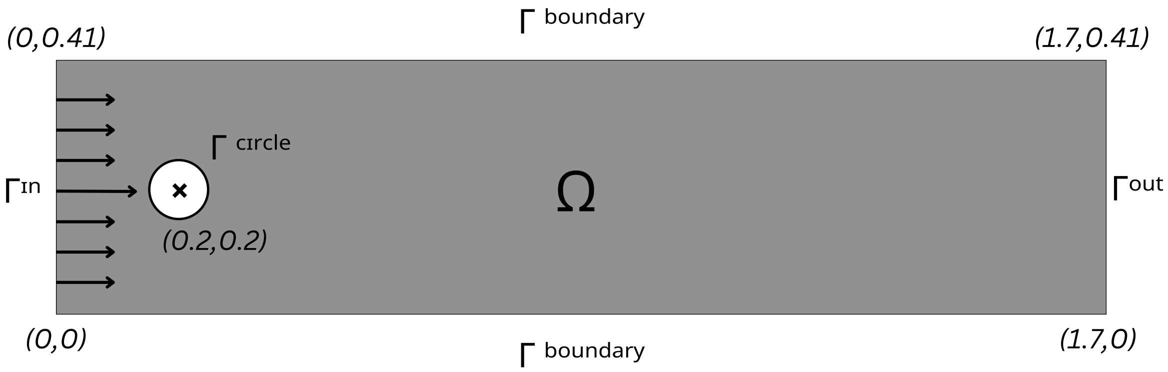


Figure 4.1: Domain $\Omega \subset \mathbb{R}^2$

laminar flow, which shows smooth, orderly, and parallel flow layers, with minimal mixing between the layers. This fluid flow occurs at a low Reynolds number and follows a stable and predictable flow pattern. Examples of fluids that exhibit laminar flow behaviours are water through a narrow pipe or wind rushing over a plane at a low speed. In our case, the fluid is non-stationary but becomes periodic when the flow is fully developed. The following Figure 4.1 depicts the domains Ω and the different boundary components of the domains, which are called $\Gamma^{in}, \Gamma^{out}, \Gamma^{boundary}$ and Γ^{circle}.

The parameters for the benchmark setup are provided in Table 4.1 below:

Table 4.1: Parameters for the Navier-Stokes problem with the domain described in Figure 4.1

Parameter	**Value**
Kinematic Viscosity	$10^{-3}\,\mathrm{m}^2/\mathrm{s}$
Fluid Density	$1\,\mathrm{kg}/\mathrm{m}^3$
Height of Pipe	$0.41\,\mathrm{m}$
Width of Pipe	$1.7\,\mathrm{m}$
Diameter of Circle	$0.1\,\mathrm{m}$

The domain is specified as $\Omega := (0, 1.7) \times (0, 0.41)$, excluding the disk $B_r(x, y)$ with radius $r = 0.05\,\mathrm{m}$, where $B_r(x, y)$ represents a ball centered at (x, y) with radius defined as r. As it can be seen in Figure 4.1, the ball is placed non-symmetrically across the domain, with the distance between the centre of the ball $(0.2, 0.2)$ to the top boundary being 0.21m and the distance between the bottom wall and the centre of the wall being 0.2m. This lack of symmetry by a slight scale will be disruptive and significant in fluid flow structure.

The domain boundary is denoted by $\partial\Omega \subset \mathbb{R}^2$, and we apply the Dirichlet and Neumann boundary conditions (BCs) to the boundary of the domain. Dirichlet boundary condition, also known as the no-slip condition, $v = 0$, is applied on the boundaries $\Gamma^{boundary}$ and Γ^{circle}, where $\Gamma^{boundary} = (0, 1.7) \times \{0\} \cup (0, 1.7) \times \{0.41\}$ and $\Gamma^{circle} = \partial B_r(0.2, 0.2)$. The no-slip condition means that the fluid velocity concerning boundaries $\Gamma^{boundary}$ and Γ^{circle} is zero. A steady parabolic inflow profile is applied on the boundary $\Gamma^{in} = \{0\} \times (0, 0.41)$. This inflow is the inhomogeneous Dirichlet boundary condition, i,e $v = \mathbf{g}$ and is defined using

$$\mathbf{g}(0, y) = \begin{pmatrix} \dfrac{4v_{max} y(h-y)}{h^2} \\ 0 \end{pmatrix}, \tag{4.1}$$

where $v_{max} = 1.5\ m/s$. The outflow boundary $\Gamma^{out} = \{1.7\} \times (0, 0.41)$ has the modified do-nothing condition enforced on the wall Γ^{out} which is defined using

$$\mu \partial_n v - p\mathbf{n} = 0.$$

Classical do-nothing condition, defined as $\sigma.\mathbf{n} = 0$, is typically applied to outflow boundaries and ensures that no artificial forcing is applied to the flow. However, there are specific issues with this condition regarding our problem, including problems with stability and non-physical solutions [53, 54]. That is why it is reasonable to opt for the modified do-nothing condition, which adds an extra term to preserve stability and regulate pressure control.

Given that the circle's diameter is fixed at $d = 0.1\,\mathrm{m}$, and flow's characteristic velocity, which corresponds to the average velocity towards the x-direction along the inflow boundary, is calculated as

$$\begin{aligned}\bar{v}_x(t) :&= \frac{1}{h}\int_0^h v_x(t)\left(t, \begin{pmatrix}0\\ y\end{pmatrix}\right)dy = \frac{1}{h}\int_0^h g_x(0,y)dy \\ &= \frac{1}{h}\int_0^h \frac{6\text{ m/s}.y(h-y)}{h^2}dy = 1\text{m/s}. \end{aligned} \tag{4.2}$$

We can also look at the Reynolds number,

$$Re = \frac{LV}{\nu} = \frac{0.1\text{ m}\cdot\ 1\text{ m/s}}{10^{-3}\text{ m/s}} = 100.$$

Our main relevant quantities are the well known drag coefficient value and lift coefficient value. The drag coefficient is a dimensionless quantity representing the drag force which is acting on a particular object relative to its shape, flow conditions, and size. Drag force is the resistance experienced by an object moving through a fluid, such as air or water, caused by the fluid's viscosity and the object's shape and speed. An example of drag force is the resistance an object would feel while passing through water. The lift coefficient represents the lift force generated by an object relative to its flow condition and surface area. Lift force is the upward force exerted on an object due to the pressure difference between the upper and lower surfaces due to the object's motion through a fluid. An example of lift force is the force experienced by an aeroplane that allows it to lift off the ground.

For the scenario in 4.1, the drag and lift forces will be acting on the circle $B_r(0.2, 0.2)$, based on the interaction with the surrounding fluid. The drag force will act on the circle's surface and oppose the fluid's flow. This drag force won't affect the circle's position, as the circle has been fixed, but it will create a pressure difference on the circle's surface. The lift force is generated due to the pressure difference between a circle's upper and lower surfaces. As the fluid flow is asymmetric due to the pressure difference, a lift force will present perpendicular to the flow direction. The key quantities, namely the drag coefficient (C_D) and the lift coefficient (C_L), can be determined from the drag force (F_D) and lift force (F_L), which can be computed by performing a line integral along the boundary of the circle.

$$\begin{pmatrix} F_D \\ F_L \end{pmatrix} = \int_{\Gamma^{circle}} \sigma.\mathbf{n} ds.$$

The C_D and C_L are expressed as

$$C_D = \frac{2F_D}{\rho d \bar{v}_x^2}, \text{ and } C_L = \frac{2F_L}{\rho d \bar{v}_x^2}. \tag{4.3}$$

4.2 FEM Spaces

Finite element spaces are mathematical constructs that are used in FEM to approximate solutions to PDEs. These spaces are subsets of larger spaces, usually Sobolev space, and are defined with the help of piecewise polynomial functions over a domain. This domain is divided into smaller regions with simpler shapes, such as triangular, tetrahedral, and quadrilaterals [3]. The Navier-Stokes equations form a coupled, mixed system involving

many interrelated variables and equations that describe fluid flow. Upon performing integration by parts on the NSE, the highest derivative of the fluid velocity takes shape of its gradient. For fluid pressure, lower regularity accounts for the Dirichlet boundary conditions in the function space. That is why the following is chosen,

$$V^p = \{v \in [H^1(\Omega)]^2 | v = 0 \text{ on } \Gamma^{in} \cup \Gamma^{boundary} \cup \Gamma^{circle}\}.$$

For the pressure on $\partial\Omega$, homogeneous Neumann boundary conditions are prescribed.

This makes us work with

$$V^p := L^2(\Omega),$$

as the do-nothing condition serves to normalize the pressure in such a way that it guarantees the existence of a unique solution.

The solution of pressure of the fluid flow must also be unique; the inf-sup or the Ladyzhenskaya-Babuška-Brezzi (LBB) condition must be satisfied. The condition is,

$$\inf_{q \in V^p} \sup_{\gamma \in V^v} \frac{(q, \nabla \cdot \gamma)}{\|q\|_{L^2(\Omega)} \|\gamma\|_{H^1(\Omega)}} \geq \zeta > 0. \tag{4.4}$$

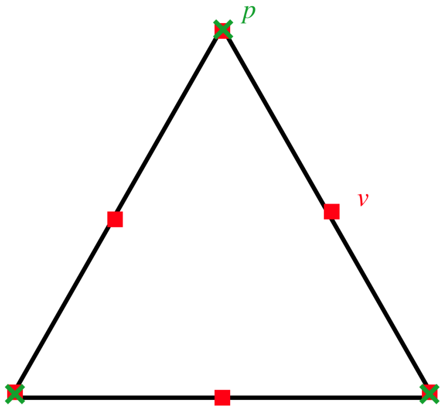


Figure 4.2: Representation of the DOF for the triangular Taylor-Hood element

In this case, V^v and V^p are required to fulfill the stability estimate when they are replaced by their corresponding finite element spaces, $V_h^v \subset V^v$ and $V_h^p \subset V^p$. This implies that the velocity space must be significantly larger than the pressure space. The Taylor-Hood element [49] is a popular choice of finite element method when discretization of mixed elements is required. It satisfies the LBB condition in 4.4, which ensures stability and convergence of velocity and pressure approximations. Taylor Hood elements combine quadratic and linear polynomials for velocity and pressure, respectively. These elements could be triangular or quadrilateral if working with a two-dimensional geometry or tetrahedral or hexahedral elements in three dimensions. In FEM, the degrees of freedom (DOF) refer to the independent variables associated with the finite element approximation. These DOFs define the values at specific points of the chosen element. In the figure 4.2, the degrees of freedom for the velocity's quadratic finite element are indicated by red squares, while the pressure's degrees of freedom are represented by green crosses.

4.3 Weak Formulation

To obtain the weak formulation, which is also referred to as the weak form, of the NSE, we begin by revisiting equation 2.2. We multiply the equation by appropriate test functions $\psi^v \in V^v$ for the velocity field and $\psi^p \in V^p$ for the pressure field. Then, we integrate both terms over the domain Ω, ensuring that the boundary conditions are properly accounted for in the process.

The objective is to determine $\{v, p\} \in \{\mathbf{g}+V^v\}\times V^p$, ensuring that the initial conditions are met. Furthermore, for every time step $t \in (0, T)$, the solution must satisfy the corresponding equation.

$$\rho(\partial_t v, \psi^v) + (\nabla p - \mu\nabla \cdot (\nabla v + \nabla v^T).\psi^v) + \rho((v \cdot \nabla)v, \psi^v) = 0, \quad (4.5)$$

$$(\nabla \cdot v, \psi^p) = 0. \quad (4.6)$$

Using the concept of integration-by-parts, it can be determined that

$$\begin{aligned} &\rho(\partial_t v, \psi^v) + \mu(\nabla v + \nabla v^T, \nabla\psi^v) - (p, \nabla \cdot \psi^v) \\ &- \int_{\Gamma^{out}} [\mu(\nabla v + \nabla v^T) \cdot \mathbf{n} - p\mathbf{n}] \cdot \psi^v \, ds + \rho((v \cdot \nabla)v, \psi^v) = 0. \end{aligned} \quad (4.7)$$

Now we can use the modified do-nothing [53] condition in Equation (4.7)

$$\mu\nabla v \cdot \mathbf{n} - p\mathbf{n} = 0,$$

to get

$$\begin{aligned} &\rho(\partial_t v, \psi^v) + \mu(\nabla v + \nabla v^T, \nabla\psi^v) - (p, \nabla \cdot \psi^v) \\ &- \int_{\Gamma^{out}} \big(\mu\nabla v^T \cdot \mathbf{n}\big) \cdot \psi^v \, ds + \rho((v \cdot \nabla)v, \psi^v) = 0. \end{aligned} \quad (4.8)$$

Now, the time derivative can be discretized with the one-step-θ scheme and then (Eq 4.6) and (4.8) can be added to get the much-needed semi-linear form of the equation.

$$\begin{aligned} A(U)(\psi) :=& \rho(v, \psi^v) + k\theta\mu(\nabla v + \nabla v^T, \nabla\psi^v) - (k\theta(p, \nabla \cdot \psi^v) \\ &+ k\theta \int_{\Gamma^{out}} [\mu\nabla v^T \cdot \mathbf{n}] \cdot \psi^v \, ds + k\theta\rho((v \cdot \nabla)v, \psi^v) + k(\nabla \cdot v, \psi^p). \end{aligned} \quad (4.9)$$

This would also lead to the right-hand side functional

$$\begin{aligned} F(\psi) :=& \rho(v^{n-1}, \psi^v) - k(1-\theta)\mu(\nabla v^{n-1} + \nabla(v^{n-1})^T, \nabla\psi^v) + k(1-\theta)(p^{n-1}, \nabla \cdot \psi^v) \\ &+ k(1-\theta) \int_{\Gamma^{out}} \big(\mu\nabla(v^{n-1})^T \cdot \mathbf{n}\big) \cdot \psi^v ds - k(1-\theta)\rho((v^{n-1} \cdot \nabla)v^{n-1}, \psi^v). \end{aligned} \quad (4.10)$$

In equations (4.9) and (4.10), we introduce $U := \{v, p\}$, where U belongs to the set $\{\mathbf{g} + V^v\} \times V^p$, and define $\psi := \{\psi^v, \psi^p\}$ as an element of the space $V^v \times V^p$. Within this framework, the quantities $v := v^n$ and $p := p^n$ represent the unknown solutions that need to be computed at the current time step. On the other hand, v^{n-1} and p^{n-1} correspond to the values that were obtained during the previous time step. It is important to note that in the numerical implementation, the pressure term is implicitly handled. This is made clear by the presence of the term $-k\theta(p, \nabla \cdot \psi^v)$ in the semi-linear formulation of (4.9), while the term $(p^{n-1}, \nabla \cdot \psi^v)$ is conspicuously absent from the right-hand side functional of (4.10).

4.4 Numerical Results of FEM

The finite element method's input mesh was designed to have 27736 triangles and 14168 vertices. The Figure 4.3 shows the resulting mesh. The time step was comparatively coarse until the flow was fully developed, with $\Delta t = 0.05\ s$. After the passage of time and the development of fluid flow, the time step was finer at $\Delta t = 0.005\ s$. This time step was continued till the final time was reached, and the fluid simulation was stopped when the time t reached 35 s. This resulted in total of 4770 snapshots from the problem. All the FEM simulations and other related results were produced using FreeFEM++.

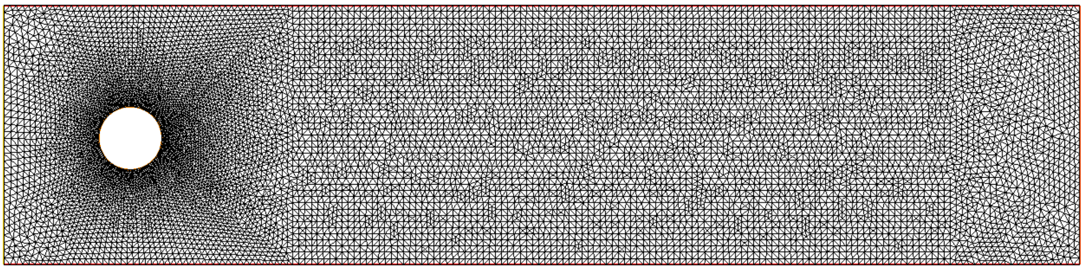


FIGURE 4.3: Input Mesh for FEM simulations

After observing the fluid flow simulation, the quantities of interest were calculated for the fluid flow over the last two seconds of the simulation, i.e. $t = 25$ to $t = 27$. These include the maximum drag coefficient (max C_D) and the maximun lift coefficient (max C_L).

TABLE 4.2: max (C_D) outcomes from the FEM solution achieved from the 2D-2 benchmark geometry

Time	Our Results of max (C_D)	Benchmark Results of max (C_D) [22]
10	3.48002	3.2152
18	3.4136	3.1605
34	3.45976	3.203
67	3.46276	3.2171

Table 4.2 represents our results and shows that they are similar to the results produced by Schafer et al. for max (C_D), only with slight fluctuations. Table 4.3, represents the comparison between the maximal lift coefficient between results from our FEM simulations in FreeFEM++ and experimental values from the Schafer/Turek research.

TABLE 4.3: max (C_L) outcomes from the FEM solution achieved from the 2D-2 benchmark geometry

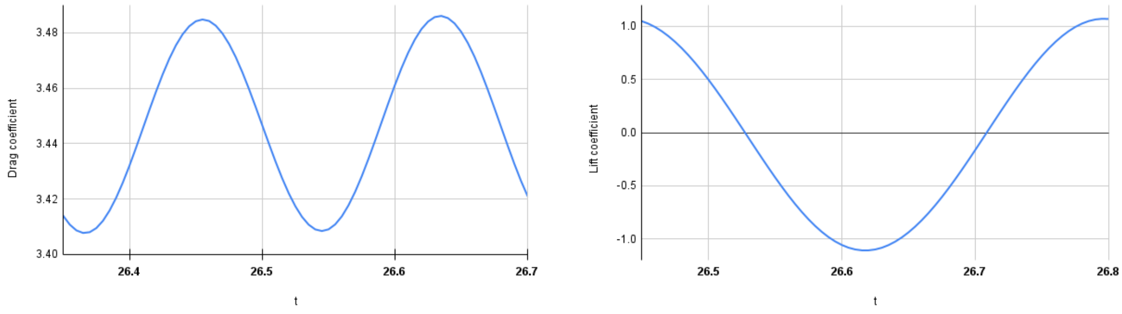
Time	Our Results of (max C_L)	Benchmark Results of (max C_L) [22]
10	0.97979	0.9028
18	0.724994	0.8026
34	1.04438	0.9223
67	1.051002	0.9591

Figure 4.5, illustrates the results of the FEM simulations for the benchmark problem. In the initial moments of the simulation, the fluid flow has not yet reached a steady state, which is apparent in the snapshots of the velocity magnitude as the fluid enters

the domain. During this phase, the flow follows a straight path along the x-direction. The flow remains smooth and free from disturbances. As time progresses, a vortex begins to form and move circle's right side. After approximately 25 seconds, the fluid flow becomes fully developed and exhibits periodic behavior. One complete cycle corresponds to the time interval between the consecutive two peaks in the lift coefficient's plot. We present snapshots of the fluid flow velocity magnitudes at two distinct points: when the lift reaches its maximum at $t = 26.395\,\text{s}$ and when the lift is at its minimum at $t = 26.56\,\text{s}$.

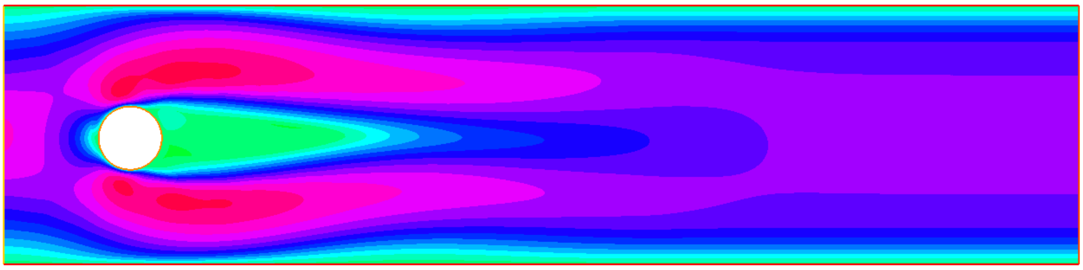
The graphs in Figure 4.4 show clear oscillations of C^D and C^L caused by vortex shedding in fluid flow around the circle. These vortices form alternately on either side of the circle and similarly shed on either side. This creates fluctuating forces. In Figure ??), the drag coefficient attains its maximum value when time reaches $t = 26.46\ s$ and then goes down to the minimum value after a few seconds. This pattern keeps repeating indefinitely, cementing that the drag force exhibits oscillatory and periodic behaviour. In Figure ??, one period of fluid flow can be observed, i.e. between one lift coefficient peak and the other. Lift forces can also be observed to be periodic and oscillatory.

Building upon these observations, it is important to note that the periodic oscillations of both drag and lift coefficients are indicative of a coupled vortex shedding process. The regular shedding of vortices creates a fluctuating wake behind the circle, which induces alternating low-pressure and high-pressure regions around the object, leading to the observed oscillatory behavior in both drag and lift. Furthermore, the amplitude and frequency of these oscillations can be influenced by factors such as the flow velocity, the shape of the object, and the Reynolds number, which controls the flow regime.

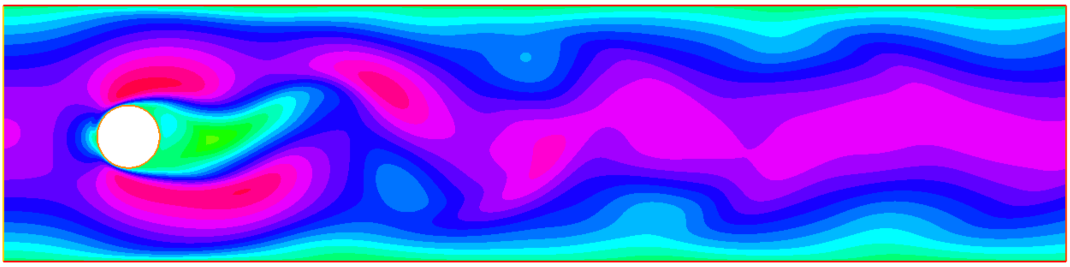


Drag coefficient (C_D) Lift coefficient (C_L)

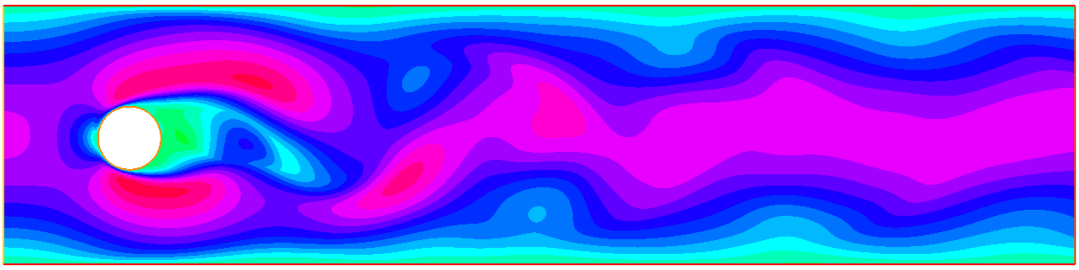
FIGURE 4.4: Drag and Lift coefficient of the FEM solutions over one period of flow



Solution at $t = 0.75\ s$



Solution at $t = 26.395\ s$ - maximal lift



Solution at $t = 26.56\ s$ - minimal lift

FIGURE 4.5: Here are snapshots illustrating the magnitude of the velocity, which are derived from the finite element method (FEM) solution computed using the FreeFEM++ software.

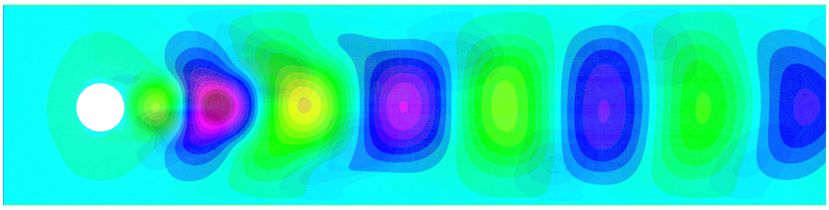
4.5 POD of Velocity

Using FEM in the previous section, 401 snapshots were computed from the initial time of $t = 25\ s$ to the final time of $t = 27\ s$ with the time step of $\Delta t = 0.005\ s$. This section will discuss POD vectors for velocity in depth and present an analysis of the topic. The interested reader can learn more about the topic in [55–57].

As explained in the Algorithm 3, the velocity modes can be computed using the MOS. To compute the POD modes, the first step is to center the snapshots by subtracting the mean flow. This involves calculating the average of all snapshots and then subtracting this mean from each individual snapshot y_i. Once the snapshots are centered, proper orthogonal decomposition (POD) can be applied to the correlation matrix derived from these adjusted snapshots, allowing us to extract the dominant modes of the flow.

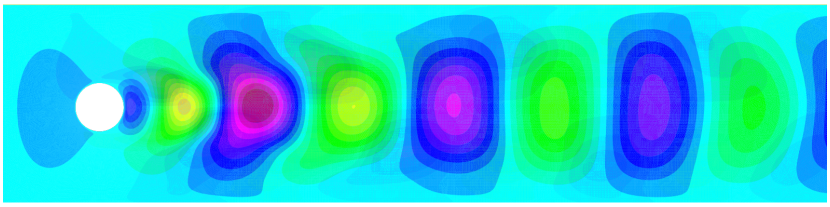
It can be clearly observed that the eigenvalues associated with the correlation matrix decrease rapidly, which validates the fact that the first few largest eigenvalues contain all the significant information about the fluid flow properties. For PO in this section, we set the threshold to be $\delta = 0.99$, which means that we will only select the eigenvalues up to a point 99% for energy to be conserved, after which all the modes will be neglected as their contributions to the system are minimal. The first five basis vectors of velocity met this threshold, and the following Figure shows the magnitude corresponding to the first five POD modes. As can be seen in Figure 4.6, the first mode of velocity captures the dominant fluid flow feature, such as the formation of vortices, and we can observe the strong coherent patterns in the figure depicting significant energy contributions. We can also observe that the first mode exhibits symmetry along the x-direction, and maximum and minimum velocities are alternating — this accounts for the periodic nature of the fluid flow. The second POD mode is similar to the first one. It is symmetric along the x-direction and shows alternating maximum and minimum velocities; however, the distance between the maximum and minimum velocities decreases compared to the first mode.

The third POD mode develops a two-layer fluid formation, where both layers have a distinct build but are still connected. This shows that intermediate-order modes will capture the intricate details of the fluid flow, such as the sheer layers and the flow transition going from uniform motion in the x direction to oscillatory motion throughout the domain. It shows that the fluid, despite moving in the positive x-direction, also oscillates in the y-direction. This results in the development of more complex vortex structures. We can also observe that the POD mode is anti-symmetric, further developing the fluid's alternating and periodic nature. Like the third mode, the fourth mode has a double-layer formation with an anti-symmetric nature. But unlike the third mode, we can see that the layers now split from the middle, forming two separate layers. Finally, in the fifth mode, we can observe both the layers joining into a single formation with alternating maximum and minimum velocities.



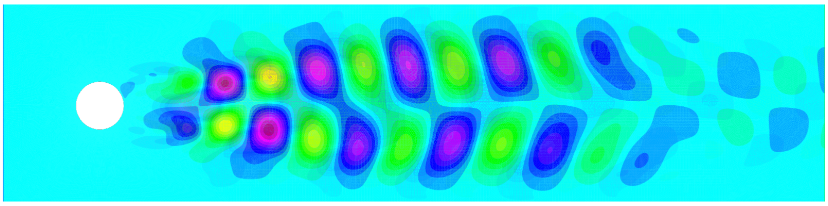
(a)

First velocity mode magnitude



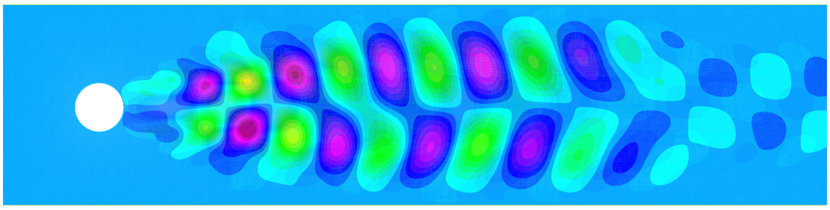
(b)

Second velocity mode magnitude



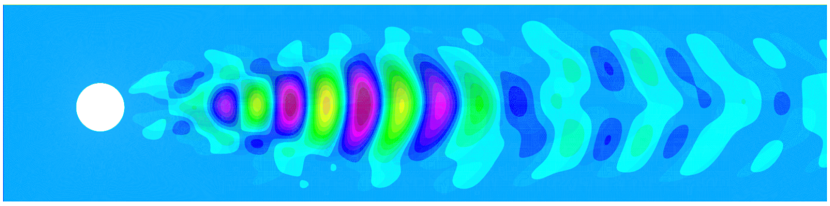
(c)

Third velocity mode magnitude



(d)

Fourth velocity mode magnitude



(e)

Fifth velocity mode magnitude

Figure 4.6: The magnitude obtained from the first five velocity modes derived from the Navier-Stokes equations

Chapter 5

FEM and POD Analysis of Three Different Configurations

In this chapter, we extend the application of POD analysis to three different configurations, each of which presents unique flow characteristics and challenges. By comparing the resulting modes across these configurations, we aim to uncover the influence of geometric factors on the flow structures and highlight the versatility of POD in addressing varied fluid dynamics problems. The insights gained from this comparative approach will contribute to a deeper understanding of the interplay between geometry and flow behaviour, with potential implications for design optimization and predictive modeling.

Before looking into each scenario, let us fix a few parameters that will remain the same across all three sections. The following 5.1 table presents a list of them.

Table 5.1: Parameters used in the analysis

Parameter	**Value**
Kinematic Viscosity	$10^{-3}\,\mathrm{m^2/s}$
Fluid Density	$1\,\mathrm{kg/m^3}$
Reynolds Number	100
Maximum Velocity	$1.5\,\mathrm{m/s}$
Mean Velocity in x-direction	$1\,\mathrm{m/s}$
Height of Pipe	$0.41\,\mathrm{m}$

5.1 Two-dimensional flow around two circular cylinders placed in a series

For numerical simulations, we will consider a domain with two circular cylinders [58] arranged in series, with two different distances between the centre of the circles. After testing various distances between the obstacles, I selected $0.4; m$ and $0.45; m$ as they were the minimum separations where distinct vortex shedding occurred at both obstacles. Distances smaller than these values resulted in the obstacles behaving as a single entity, leading to the formation of a single vortex instead of two separate vortices. Then for each, FEM is applied, using FreeFEM++, to get real-time simulations and drag and lift values. Finally, the POD modes are presented and analyzed.

5.1.1 Case 1: When the distance between the cylinders is $D = 0.4$ m

In this case, 2-dimensional laminar flow is described around two circular cylinders, with 0.4 m spacing between the center of the cylinders. The following Figure 5.1 depicts the domain known as Ω and the different components of boundary of the domain Γ^{in}, Γ^{out}, and $\Gamma^{boundary}$.

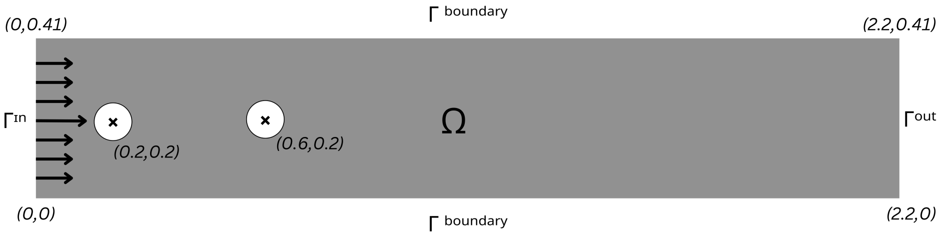


Figure 5.1: Two circular cylinders in series, with center-to-center distance $D = 0.4$

The parameters of this configuration are given as a list in the Table 5.2 below:

Table 5.2: Parameters used in the analysis of Case 1

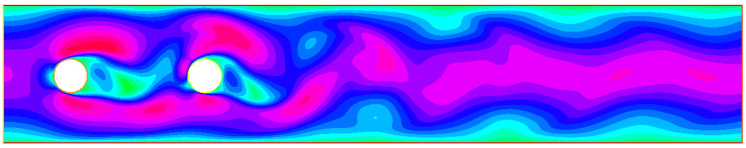
Parameter	**Value**
Width of pipe	2.2 m
Diameter of circle 1	0.1 m
Diameter of circle 2	0.1 m
Distance between center of circles	$D = 4d_1 = 4d_2 = 0.4\,\mathrm{m}$

The domain in Figure 5.1 is defined using $\Omega := (0, 2.2) \times (0, 0.41)$ excluding $B_{r_1}(0.2, 0.2)$ and $B_{r_2}(0.6, 0.2)$, where r_1 and r_2 are both 0.05 m. On the boundary of the domain, $\partial\Omega$, Dirichlet BC and Neumann BC are applied, which are similar to the ones described in section 4.1, where the homogeneous Dirichlet boundary conditions $v = 0$ is applied on the boundary $\Gamma^{boundary}$ and the boundaries of the circle, and the inhomogeneous Dirichlet boundary condition (4.1) is applied on the boundary Γ^{in}. The modified do-nothing boundary condition is applied on Γ^{out}.

Using the known information, the aim is to present simulations of the fluid flow in the domain and use those simulations to find the key quantities, i.e. C_D (drag force) and C_L (lift force), using the formulas in (4.1) and (4.3). For FEM simulations, the time step was taken to be 0.05 s for the first 3.5 seconds. This is the time the fluid starts oscillating and forming the first vortex. For the rest of the duration, from 3.5 to 27 seconds, the time step was changed to 0.005, as a course time step would give results with higher accuracy. The following figure in 5.2 shows the screenshots from the fluid flow simulations. From these FEM simulations, the quantities of interest were computed, and they are displayed in the table 5.3 below.

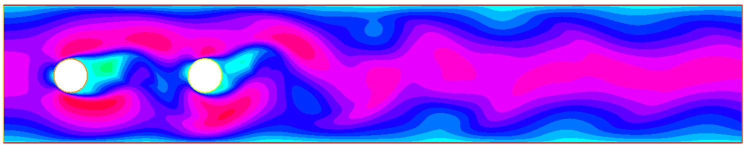
Table 5.3: Quantities of interest in the analysis of Case 1

Quantity of Interest	**Time**	**Value**
Max C_D on cylinder 1	26.720	3.507
Max C_L on cylinder 1	26.145	1.337
Max C_D on cylinder 2	26.680	3.094
Max C_L on cylinder 2	26.450	3.540



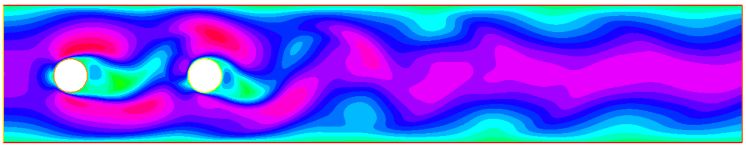
(a)

Solution at $t = 26.72\ s$ - Maximal drag on cylinder 1



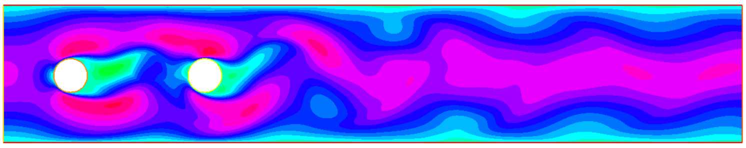
(b)

Solution at $t = 26.145\ s$ - Maximal lift on cylinder 1



(c)

Solution at $t = 26.68\ s$ - Maximal drag on cylinder 2

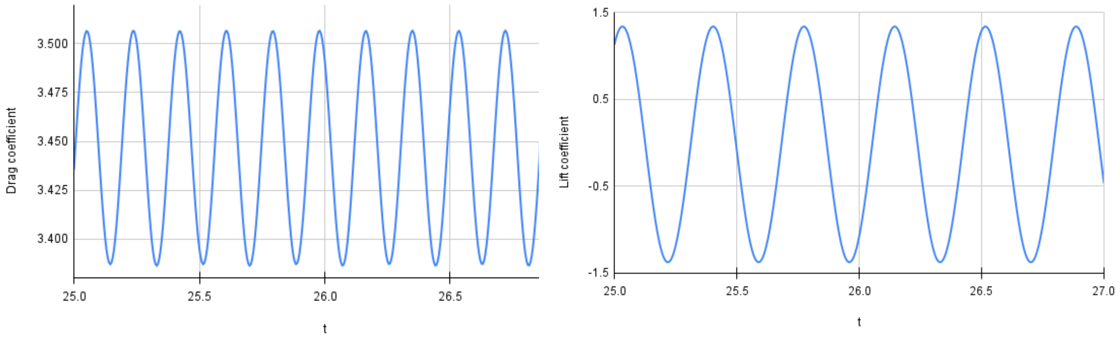


(d)

Solution at $t = 26.45\ s$ - Maximal lift on cylinder 2

Figure 5.2: Snapshots of velocity magnitude from FEM solutions in FreeFEM++ at time stamps of maximum lift and drag forces.

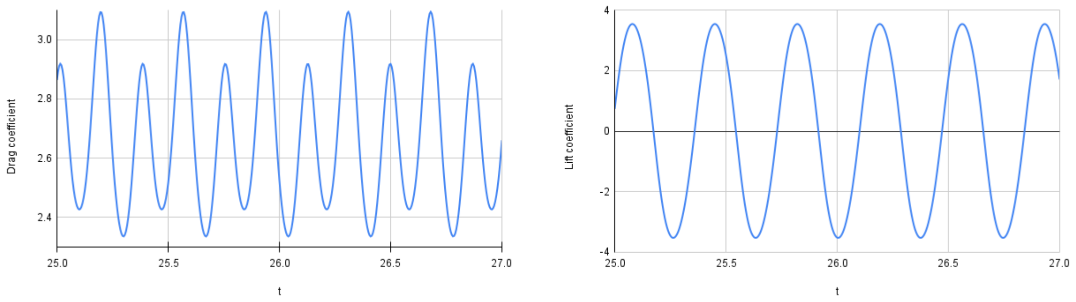
The following graphs in (5.3) and (5.4) show the oscillatory behaviour of the lift and drag forces on both cylinders. The drag and lift forces exhibit periodic behaviour. The frequency of the oscillations in the drag coefficient of cylinder 1, is the highest. The drag coefficient for cylinder 2, shows a different behaviour compared to the other three graphs. Despite being periodic and oscillatory like the others, it follows a distinct pattern, with each period having two local maximums and minimums and a local minimum between every two maximum values. The lift coefficients in cylinder 1, are always positive and follow a pattern, and the lift coefficient values in cylinder 2, are also periodic.



Drag Coefficient (C_D) over varying time

Lift coefficient (C_L) over varying time

Figure 5.3: Plots of drag and lift coefficients on cylinder 1 from FEM solutions versus time between $t = 25\ s$ to $t = 27\ s$ for the configuration 5.1



Drag Coefficient (C_D) over varying time

Lift coefficient (C_L) over varying time

Figure 5.4: Plots of drag and lift coefficients on cylinder 2 from FEM solutions versus time between $t = 25\ s$ to $t = 27\ s$ for the configuration 5.1

Now that an in-depth review of the FEM simulations has been done, we can proceed with the analysis of the POD modes of the FEM results. By employing the snapshot method, as detailed and demonstrated in 3, we are able to apply proper orthogonal decomposition (POD) to the correlation matrix. This matrix is constructed from the centred snapshots, which are obtained through FEM simulations conducted using FreeFEM++. The overall energy associated with the problem, which represents the cumulative sum of eigenvalues derived from the correlation matrix, was computed and found to be 485.221. Now, let us focus on analyzing the largest, most significant eigenvalues of the correlation matrix, as they provide crucial insight into the dominant structures and patterns present in the fluid flow.

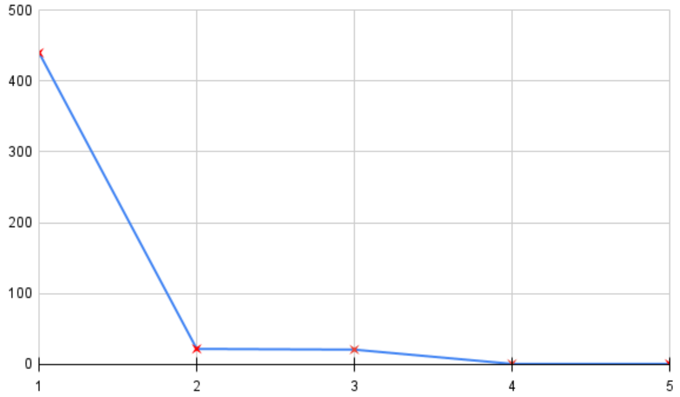


Figure 5.5: The five largest eigenvalues of the correlation matrix characterizing velocity in the NSE context, computed for the geometry in 5.1.

Figure 5.5 depicts the rapid decay of the first five eigenvalues associated with the correlation matrix of velocity. Shedding light on the successive five eigenvalues, 6 to 10, will give a clear idea of how the highest eigenvalues contain most of the information about the matrix. It is observed that the eigenvalues associated with the correlation matrix associated with the velocity exhibit a sharp decline, which further supports the validity of the claim that the matrix can be approximated using the largest eigenvalues and their corresponding most significant eigenvectors. A conclusion can be drawn regarding the most significant POD modes by examining the ten largest eigenvalues and analyzing their corresponding partial energies. These partial energies are calculated as the cumulative sum of the m first eigenvalues, along with the associated energy ratios.

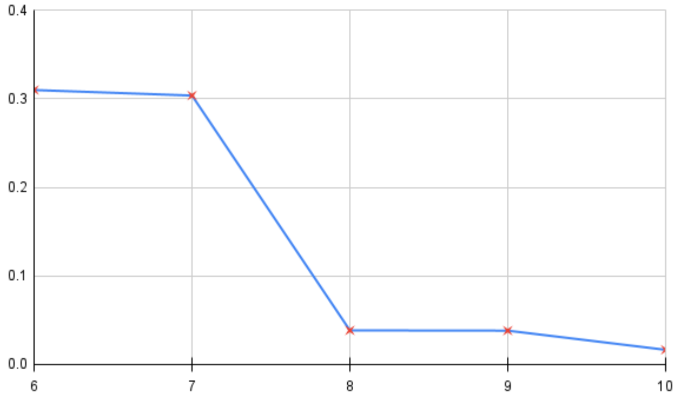
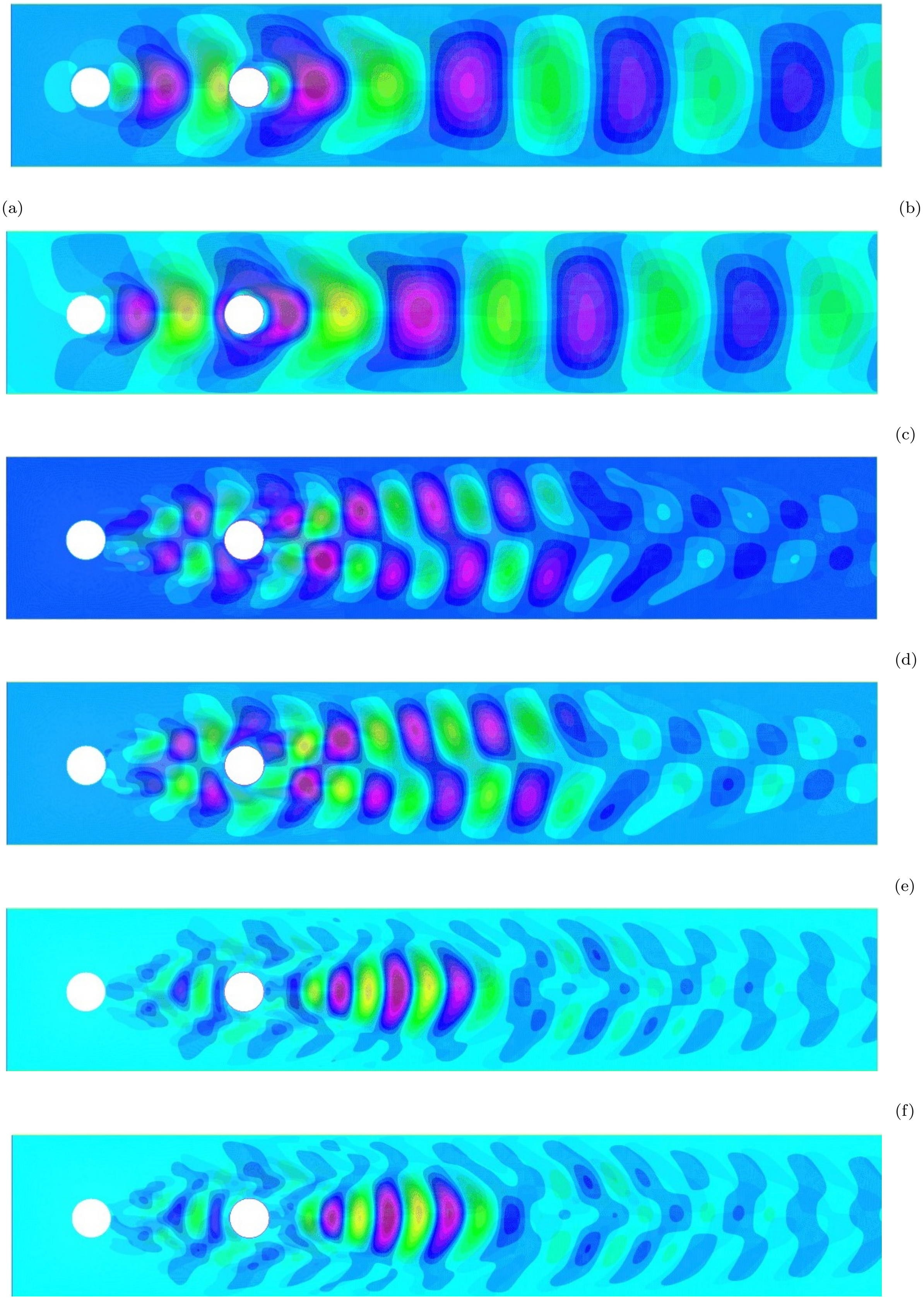


Figure 5.6: The eigenvalues from the fifth to the tenth largest of the correlation matrix in 5.1.

Table 5.4: A comparison of the ten most significant eigenvalues, along with associated partial energies, energy ratios, and the corresponding energy ratios expressed as percentages, is presented for the velocity field in the Navier-Stokes equation (NSE).

Size of POD (m)	m^{th} **eigenvalue**	**Partial energy**	**Energy ratio**	**Energy ratio %**
1	439.680	439.680	0.906	90.614
2	22.034	461.714	0.952	95.155
3	20.989	482.703	0.995	99.481
4	0.922	483.625	0.997	99.671
5	0.861	484.486	0.999	99.848
6	0.310	484.795	0.999	99.912
7	0.304	485.099	0.999	99.975
8	0.039	485.138	0.999	99.983
9	0.038	485.176	0.999	99.991
10	0.017	485.192	0.999	99.994

In Figure 5.5 and 5.6, along with the data presented in Table 5.4, it is observed that the partial energies provide an excellent approximation of the velocity field's total energy. It can be seen that the energy ratios are approaching 100 %. If we choose $\delta_e = 90$ %, the first basis vector will do. In this problem, our choice for the threshold will be $\delta_e = 99.9$ %, for which we will need six basis vectors. We will observe the first six POD modes' magnitudes in the following figure.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 5.7: Magnitude of the six most significant velocity POD modes from the NSE for configuration 5.1. (a) First mode, (b) Second mode, (c) Third mode, (d) Fourth mode, (e) Fifth mode, (f) Sixth mode.

In the Figure 5.7, we can observe that the first two modes display a single layer formation and are showcasing symmetric about the line forming at the height $h = 0.2$. In contrast to the first mode, we can see a greater distance between the maximum and minimum velocities in the second mode. In the third mode, a double-layer formation can be observed with an anti-symmetric nature and alternating maximum and minimum velocities. The fourth mode is quite similar, but the double-layer formation can be seen to come together. In the fifth mode, we can see a reappearance of the single-layer formation, and the sixth mode, despite being uncannily similar to the fifth mode, showcases some minor differences from the fifth mode. The reason is that the fifth and the sixth partial energies are also almost the same value, so after the sixth mode, 99.9 % of the energy is conserved, and this mode is a good depiction of the nature of the fluid flow between the time $t = 25$ s to $t = 27$ s.

5.1.2 Case 2: When the two cylinders are placed apart at $D = 0.45$ m

In this case, we will be looking at a domain similar to the one in 5.1, but the difference here is that we will be taking the distance between the centre of the circles to be 0.4 m. The figure 5.8 represents the domain Ω for this case and shows that boundary components that are Γ^{in}, Γ^{out}, and $\Gamma^{boundary}$.

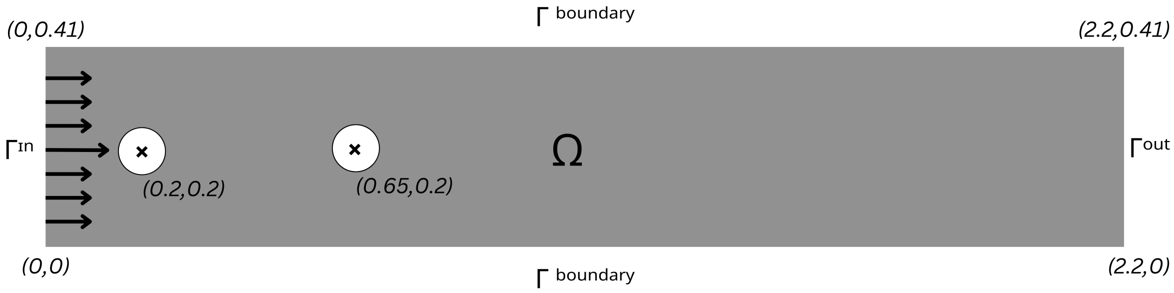


Figure 5.8: Domain with two circular cylinders in series, center-to-center distance $D = 0.45$

The parameters of this configuration are listed in the Table 5.5. The width of the domain is 2.2 m and the height is 0.41 m. Two cylinders are placed inside the domain with their centers at the points $(0.2, 0.2)$ and $(0.65, 0.2)$. These circles are called $B_r(0.2, 0.2)$ and $B_r(0.65, 0.2)$ respectively, where r describes the radius of the circles that is 0.05 m. Dirichlet and Neumann BCs are applied on the domain's boundary, denoted as $\partial\Omega$. The homogeneous boundary condition, $v = 0$, is applied to the $\Gamma^{boundary}$ and the walls of the cylinders. The inhomogeneous boundary condition as described in (4.1) is applied on the inflow boundary, Γ^{in}. And the modified do-nothing boundary condition is applied on the boundary Γ^{out}. These conditions simulate the fluid flow in the given domain in Figure 5.8.

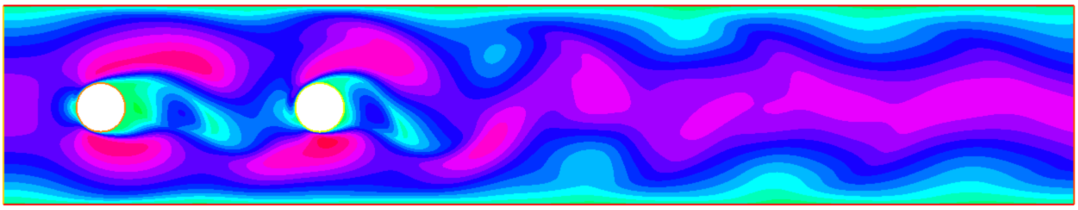
Table 5.5: Parameters used in the analysis of Case 2

Parameter	Symbol	Value
Width of pipe	w	$2.2m$
Diameter of circle 1	d_1	$0.1m$
Diameter of circle 2	d_2	$0.1m$
Distance between center of circles	D	$4.5d_1 = 4.5d_2 = 0.4m$

The first 3.5 seconds were taken for the simulation using the time step of 0.05 . That is when the vortices start to form, so after that, the time step is finer at $t = 0.005$ s. The following figure shows snapshots from the fluid flow simulations at four different time stamps. The maximum drag and lift coefficient magnitude values are displayed in the table 5.6 below. These values represent the peak forces exerted on the object at specific instances during the oscillatory cycle, highlighting the extreme conditions the object experiences during vortex shedding. By analyzing these maximum coefficients, we can gain a deeper understanding of the intensity and frequency of fluid-induced forces.

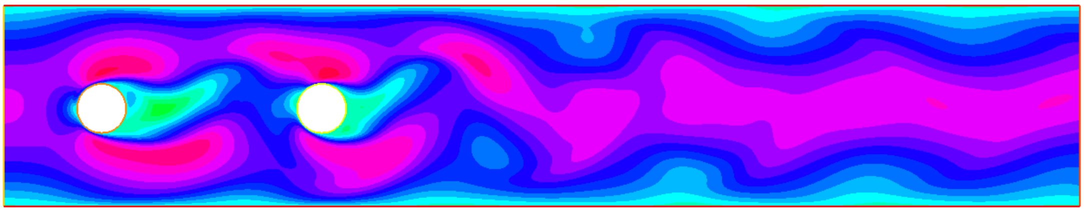
Table 5.6: Quantities of interest in the analysis of Case 2

Quantity of Interest	Time	Value
Max C_D on cylinder 1	25.325	3.499
Max C_L on cylinder 1	25.125	1.267
Max C_D on cylinder 2	25.515	3.294
Max C_L on cylinder 2	25.940	3.668



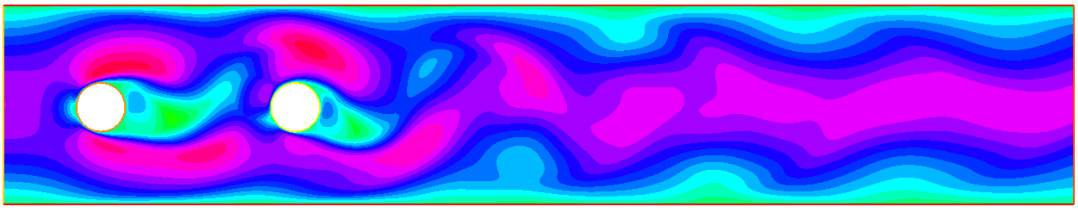
(a)

Solution at $t = 25.325\ s$ - maximal drag on cylinder 1



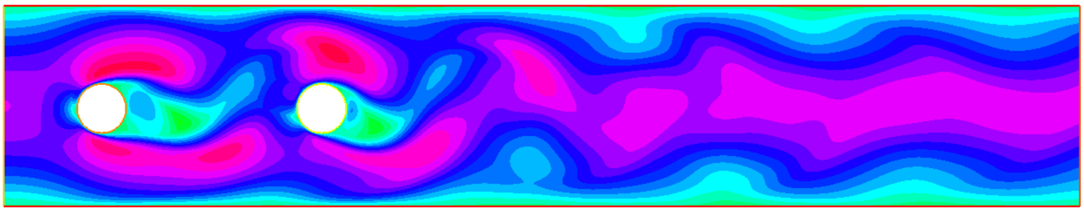
(b)

Solution at $t = 25.125\ s$ - maximal lift on cylinder 1



(c)

Solution at $t = 26.68\ s$ - maximal drag on cylinder 2

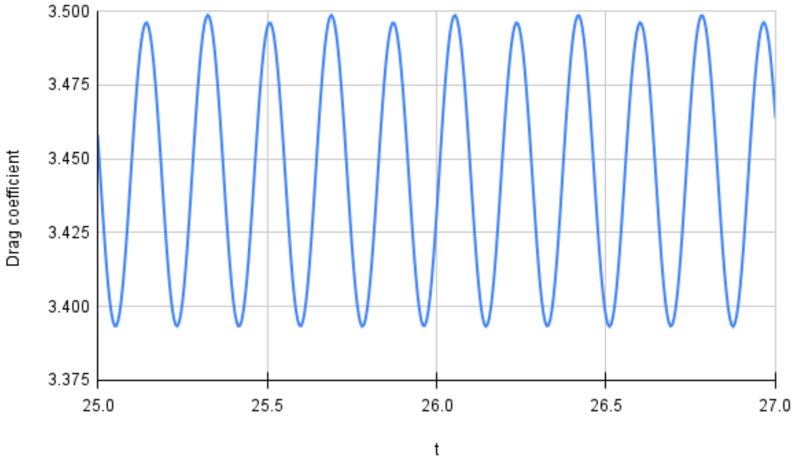


(d)

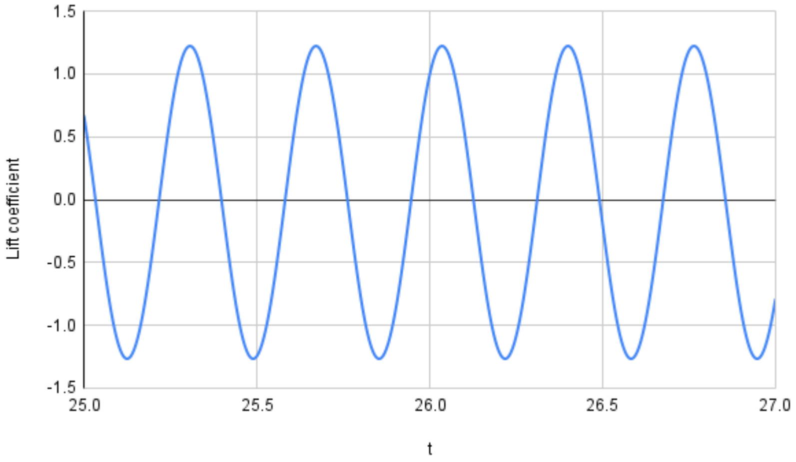
Solution at $t = 25.94\ s$ - maximal lift on cylinder 2

Figure 5.9: Velocity magnitude snapshots from FEM solutions for NSE at lift and drag maxima for 5.8.

In the graphs below, 5.10 and 5.11, it can be seen that the lift and drag coefficient exhibit oscillatory behaviour on both cylinders in the domain. The drag frequency for cylinder 1, is the highest out of all four. The drag graph for cylinder 2, stands out among the four because of its irregular shape but still exhibits periodic behaviour. Lift for cylinders 1 and 2, exhibit sinusoidal behaviour, but the lift graph for cylinder 2, has a higher amplitude than that for cylinder 1.

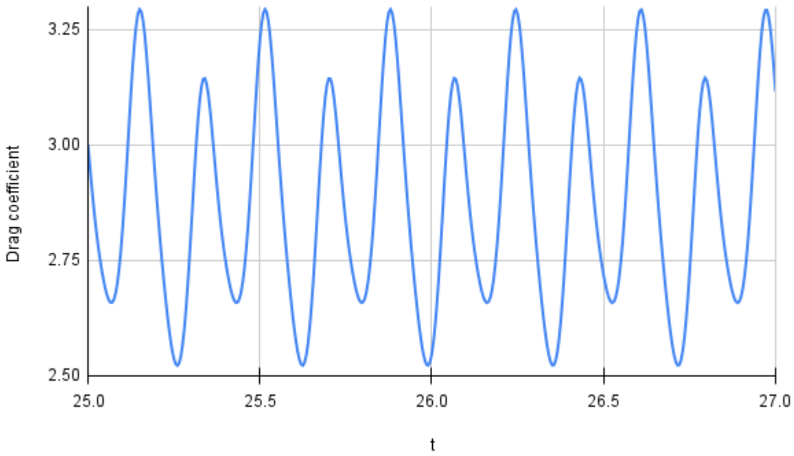


Drag coefficient (C_D) plotted over varying time

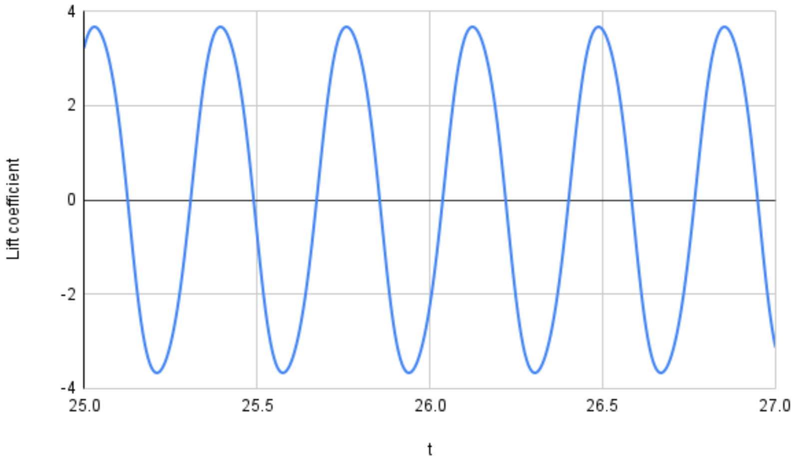


Lift coefficient (C_L) plotted over varying time

FIGURE 5.10: Drag and lift coefficient plots on cylinder 1 vs. time for $t = 25$ s to $t = 27$ s for 5.8



Drag coefficient (C_D) plotted against time



Lift coefficient (C_L) plotted against time

Figure 5.11: Drag and lift coefficient plots on cylinder 2 vs. time for $t = 25\ s$ to $t = 27\ s$ for 5.8

Following the review of the FEM simulations presented above, we now turn our attention to examining the POD modes derived from the FEM results. Using the snapshot method from Table 3, we apply POD on the correlation matrix of the centred snapshots achieved through FEM. The eigenvalue decomposition gives us 401 eigenvalues and eigenvectors for the matrix, and the sum of these eigenvalues gives the total energy of the eigenvalue problem, which is 485.535. The maximum contribution of this total energy comes from the first few most significant eigenvalues. The abrupt decline of the eigenvalues can be seen in Figure 5.12.

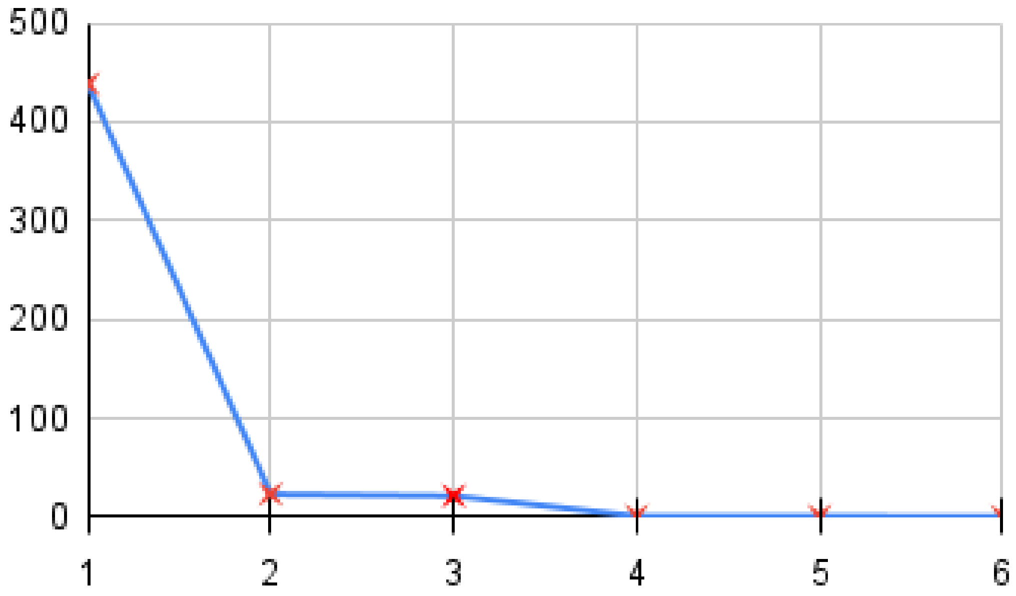


Figure 5.12: The six largest eigenvalues of the correlation matrix for the NSE velocity field in 5.8.

Using this total energy, partial energy, which the formula of energy ratios given using

$$\text{Energy ratio} = \frac{\text{Partial Energy}}{\text{Total Energy}} \tag{5.1}$$

we can analyze the eigenvalues of the correlation matrix.

From the Table 5.7, we can conclude that the partial energies offer an excellent approximation of the total energy of the problem. Furthermore, the energy ratio percentages are converging towards 100%, indicating the dominance of the largest modes in capturing the total energy. The size of the POD basis vectors is determined by using selection of the smallest subset of POD modes for which the cumulative energy ratio meets a predefined threshold. By referring to the table below, we can analyze how the size of the velocity POD basis vectors is influenced by the selection of the threshold value, δ_e.

In Table 5.8, it can be observed that, depending on the selected threshold for the energy ratio of the velocity, only the first few POD modes are required to capture the desired amount of energy. Only the first POD basis vector is enough for $\delta_e = 90$ %. We need

Table 5.7: Comparison of the ten largest eigenvalues, partial energies, and energy ratios for the NSE velocity field.

Size of POD (m)	m^{th} **eigenvalue**	**Partial energy**	**Energy ratio**	**Energy ratio %**
1	438.382	438.382	0.903	90.288
2	23.267	461.649	0.951	95.080
3	21.443	483.092	0.995	99.497
4	0.882	483.974	0.997	99.678
5	0.838	484.812	0.999	99.851
6	0.292	485.104	0.999	99.911
7	0.289	485.393	0.999	99.971
8	0.045	485.438	0.999	99.980
9	0.045	485.483	0.999	99.989
10	0.020	485.502	0.999	99.993

the first two basis vectors for the threshold $\delta_e = 95$ %. For $\delta_e = 99$ %, we need the first three basis vectors, and finally, to meet the threshold of $\delta_e = 99.9$ %, the first six basis vectors are needed. Using this, we can observe and analyse the magnitudes of the first six POD velocity modes.

Table 5.8: The size of the POD basis depending on the threshold δ_e for the energy ratio of the velocity is presented below.

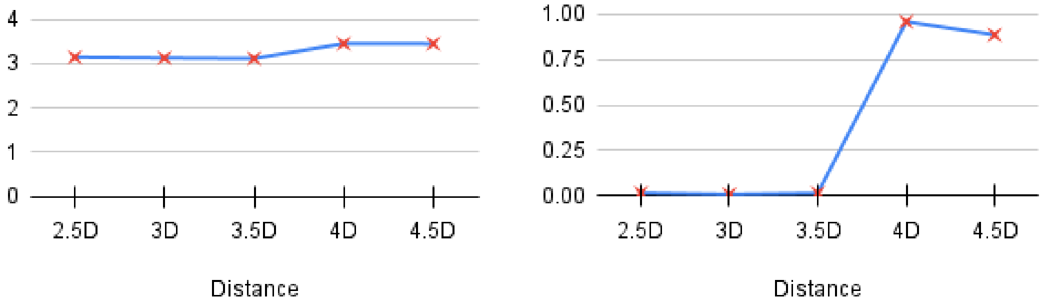
δ_e	POD size	Energy ratio percentage
90 %	1	90.28844471
95 %	2	95.08043704
99 %	3	99.4967613
99.9 %	6	99.91126016

Similarly to Figure 5.7, in Figure 5.14, we can observe that the first two modes display a single-layer formation. This formation is symmetric about the x-axis. The second mode features greater distances between the maximum and minimum velocities than the first. In the third mode, a double-layer form with an anti-symmetric nature and alternating maximum and minimum velocities can be observed to be antisymmetric.

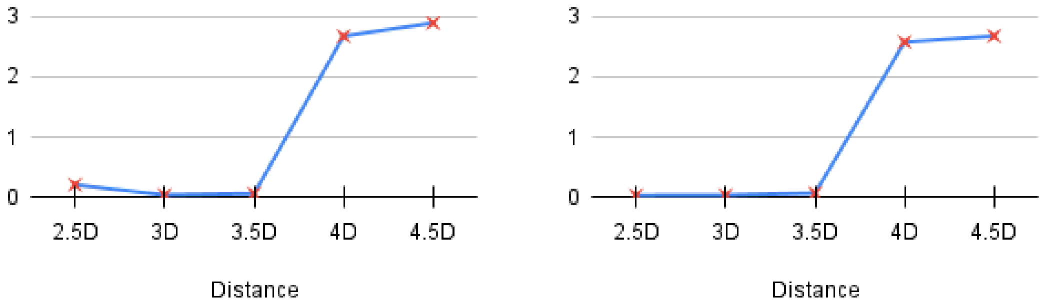
The fourth mode is quite similar, but the double-layered structure can be seen to come together at the ends to form a uni-layer. In the fifth mode, we can see a reappearance of the uni-layer formation, and the sixth mode, despite being uncannily similar, showcases some minor differences from the fifth mode.

We have done an in-depth analysis with two different distances between the circular cylinders, i.e. 0.4 m and 0.45 m, and now we can observe the mean squared averages of the lift and drag coefficient when we change the distance between the two circular cylinders. The following image displays the behaviour of the mean squared average lift and drag coefficient plotted against the distance between the cylinders.

The behavior observed in Figure 5.13 can be attributed to the interaction between the vortices shed by each cylinder as the distance between them changes.



Drag and Lift for cylinder 1



Drag and Lift for cylinder 2

FIGURE 5.13: Drag and Lift coefficients on cylinders 1 and 2 plotted against distance between the cylinders

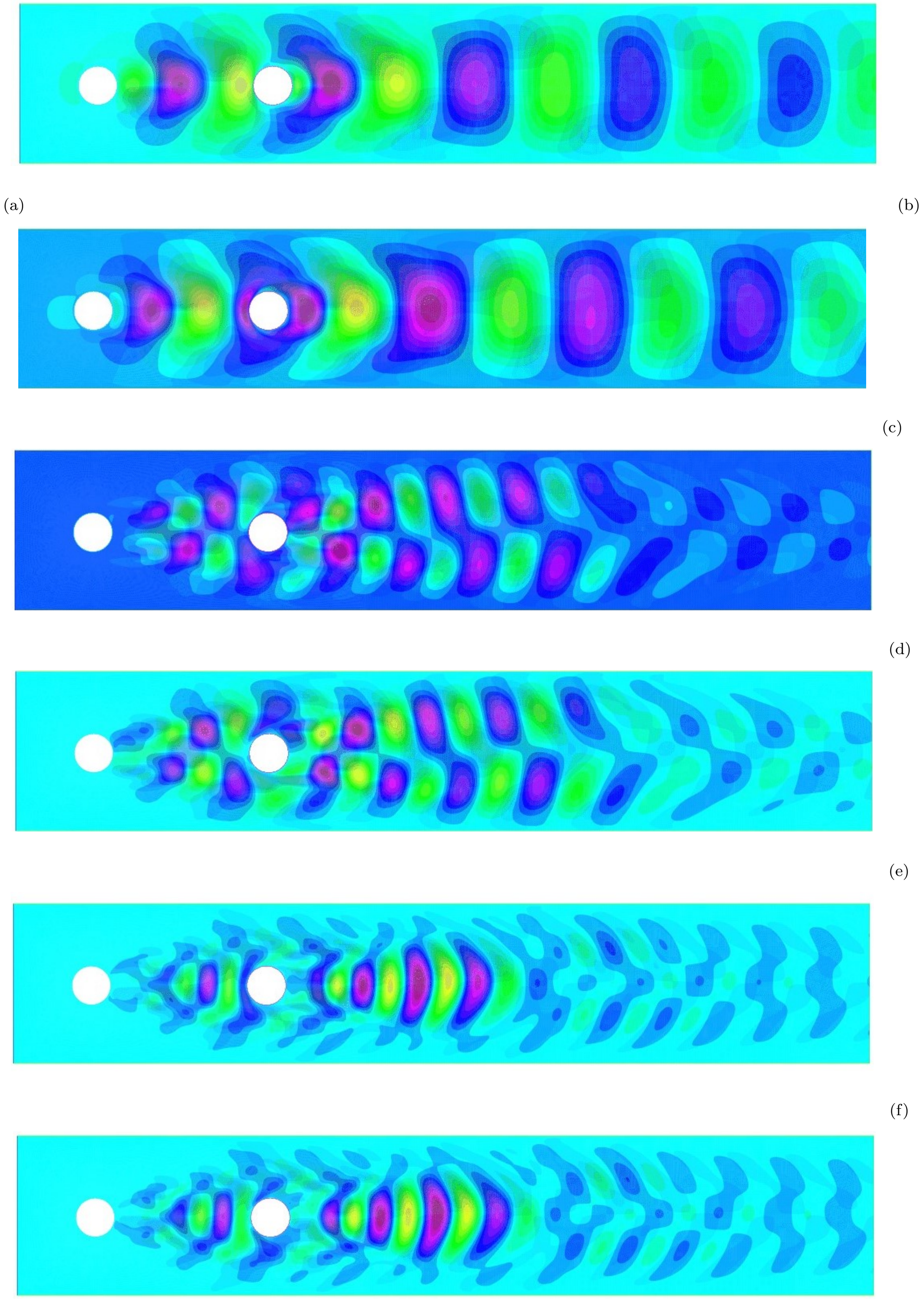


Figure 5.14: The magnitude of the six most significant velocity POD modes for configuration 5.8. (a) First mode, (b) Second mode, (c) Third mode, (d) Fourth mode, (e) Fifth mode, (f) Sixth mode.

For cylinder 1, at closer distances (e.g., $2.5D$, $3D$, $3.5D$), the flow interaction between the two cylinders is weak, resulting in low lift and drag coefficients due to the limited disturbance in the flow field. However, when the distance reaches $4D$, the vortices from each cylinder start to interact more strongly, causing the flow separation to become more chaotic, leading to a peak in both drag and lift. The slight decrease at $4.5D$ suggests that the vortex interaction begins to diminish as the distance increases further, leading to a more stable flow configuration.

For the second cylinder, the same initial trend is observed where the coefficients are low due to weak vortex interactions at smaller distances. As the distance between the cylinders increases to $4D$, the interaction between their wakes increases, causing a sudden spike in the lift and drag coefficients. The highest values are observed at $4.5D$, where the vortex shedding from both cylinders is most synchronized, maximizing the aerodynamic forces on the second cylinder. The spike in both coefficients corresponds to the increased flow disturbance caused by the strong wake interactions at this optimal distance.

5.2 Two-dimensional flow around two square cylinders set in a series

In this case, two-dimensional laminar flow is considered in a domain with two square cylinders [59, 60]. The Figure 5.15 displays the domain and it's boundaries Γ^{in}, Γ^{out}, and $\Gamma^{boundary}$. The distance between the centres of two squares is taken to be 0.4 m

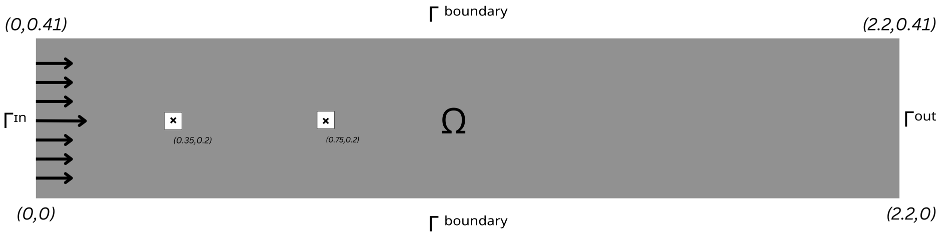


Figure 5.15: Domain with two square cylinders in series, center-to-center distance $D = 0.4$

The details and parameters of this configuration are provided in the Table 5.9 below:

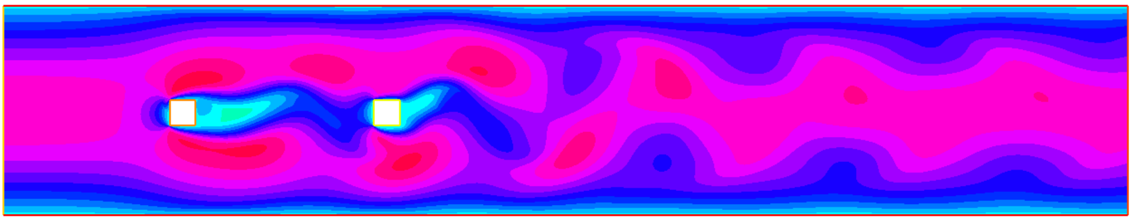
Parameter	Symbol	Value
Width of pipe	w	2.2 m
Length of square 1	l_1	0.05 m
Length of square 2	l_2	0.05 m
Distance between center of squares	D	$8l_1 = 8l_2 = 0.4$ m

Table 5.9: Parameters used in the analysis

The boundary $\Gamma^{boundary}$ is taken to be on the top and bottom of the domain, where $\Gamma^{boundary} = (0, 2.2) \times \{0\} \cup (0, 2.2) \times \{0.41\}$, $\Gamma^{in} = \{0\} \cup (0, 0.41)$, and $\Gamma^{out} = \{1.7\} \times (0, 0.41)$. The squares in the domains are centred at $(0.35, 0.2)$ and $(0.75, 0.2)$, and they have boundaries of 0.05 m on each side. A steady parabolic inflow profile is applied on the inflow boundary, Γ^{in}. This inflow is described using the inhomogeneous Dirichlet boundary condition, i.e. $v = g$, and the g is described using (4.1). In this set-up, v_{max} is taken to be 1.5 m/s. The modified do-nothing condition is enforced on the outflow boundary, and the no-slip condition is enforced on the boundary of the squares and the $\Gamma^{boundary}$. For the fluid flow, the mean velocity in the x direction v_x is 1 m/s, and the Reynolds number is $Re = 100$.

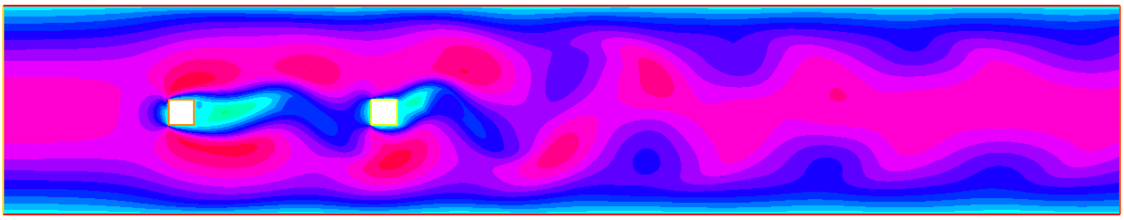
Our main quantities of interest throughout this problem are the drag and lift coefficient and the POD basis vectors that can be used to calculate the POD modes. For the drag and lift coefficient, we apply FEM to the NSEs that govern the fluid flow in the domain. Using FreeFEM++, the fluid flow simulations in the given domain were generated, which aided in the drag and lift coefficient calculations. For this fluid flow simulation, similar to the previous set-up with a circular cylinder, the time stamp was 0.05 s until the vortices started to form. Once vortex production began, the time step was taken to be courses for the next 23.5 s.

Within the time interval from 25 s to 27 s, the fluid flow simulations were mainly targeted for further analysis, as by this time stamp, the fluid flow was fully developed and reached maximum stability. The following figures show the fluid flow simulation at the maximum drag and lift coefficient values.



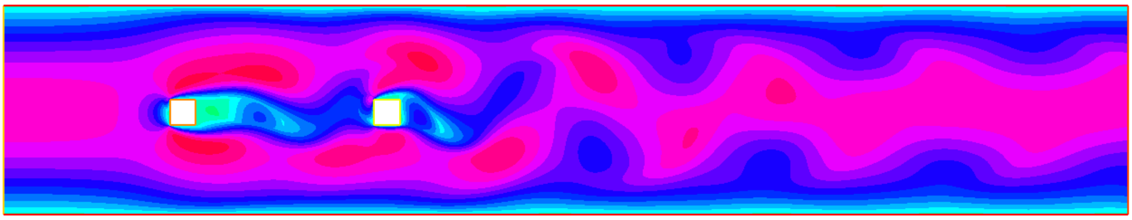
(a)

Solution at $t = 25.675\ s$ - maximal drag on cylinder 1



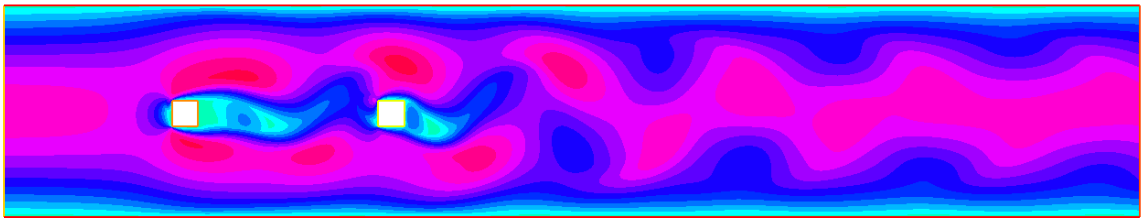
(b)

Solution at $t = 26.235\ s$ - maximal lift on cylinder 1



(c)

Solution at $t = 26.43\ s$ - maximal drag on cylinder 2



(d)

Solution at $t = 26.375\ s$ - maximal lift on cylinder 2

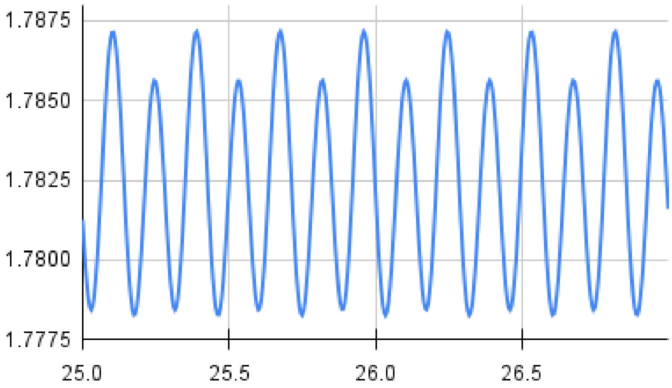
Figure 5.16: Velocity magnitude snapshots from FEM solutions for NSE at lift and drag maxima for 5.8.

From the FEM simulations, shown in 5.16, the quantities of interest were computed and are recorded in the table below.

Quantity of Interest	Time	Value
Max C_D on cylinder 1	25.6750	1.7872
Max C_L on cylinder 1	26.2350	0.2094
Max C_D on cylinder 2	26.4300	1.6363
Max C_L on cylinder 2	26.3750	1.4969

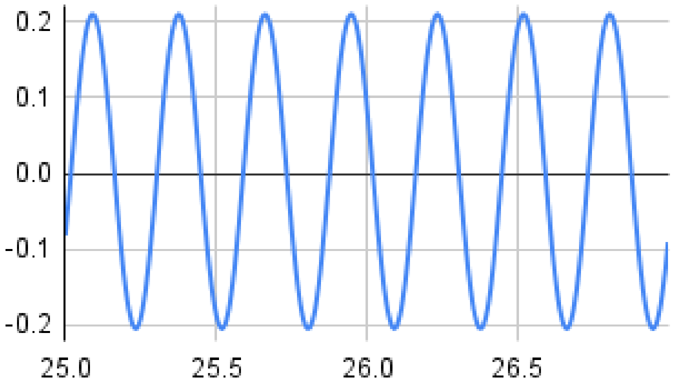
TABLE 5.10: Quantities of interest used in the analysis

The following Figures 5.17 and 5.18 show the periodic and repetitive behaviour of the lift and drag coefficients on both cylinders.



(a)

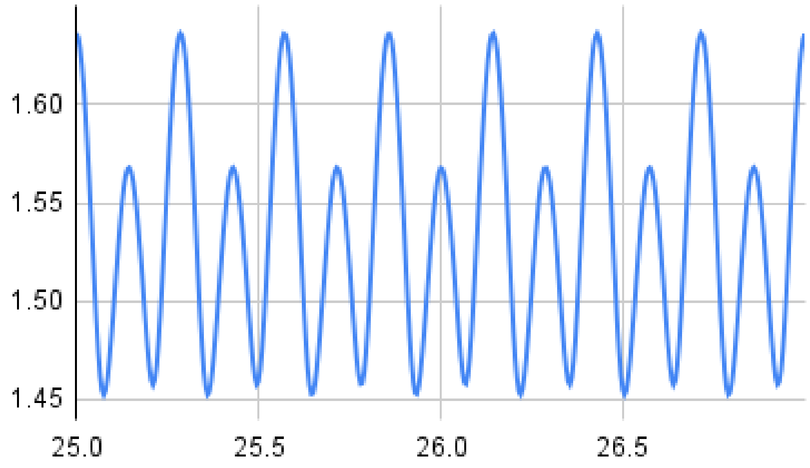
Drag Coefficient (C_D) plotted against time



(b)

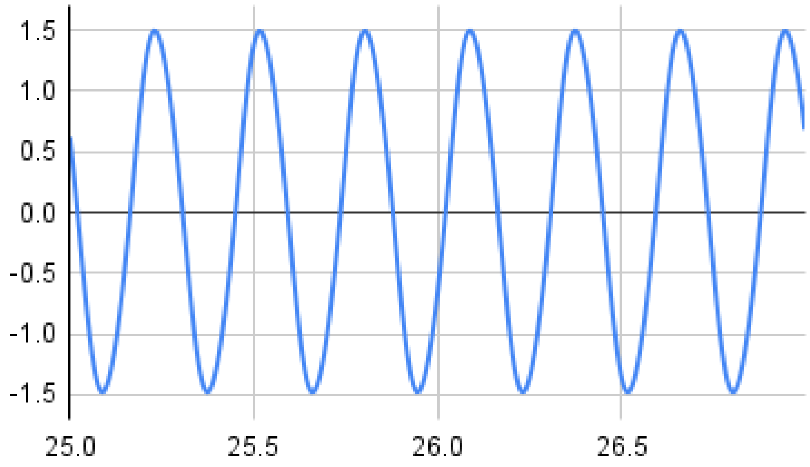
Lift coefficient (C_L) plotted against time

FIGURE 5.17: Drag and lift coefficient plots on square cylinder 1 vs. time for $t = 25\ s$ to $t = 27\ s$ for configuration 5.15



(a)

Drag Coefficient (C_D) plotted against time



(b)

Lift coefficient (C_L) plotted against time

FIGURE 5.18: Drag and lift coefficient plots on square cylinder 2 vs. time for $t = 25\ s$ to $t = 27\ s$ for configuration 5.15

The following quantity of interest in this problem is the POD basis vectors. Using this snapshot method and the same techniques used in the previous problem setups, the velocity matrix achieved from FEM can be analyzed. After applying proper orthogonal decomposition, the system's total energy was 434.82. Using this total energy, analysis of the other eigenvalues of the problem can be done, but before that, it is crucial to observe the decay of the eigenvalues.

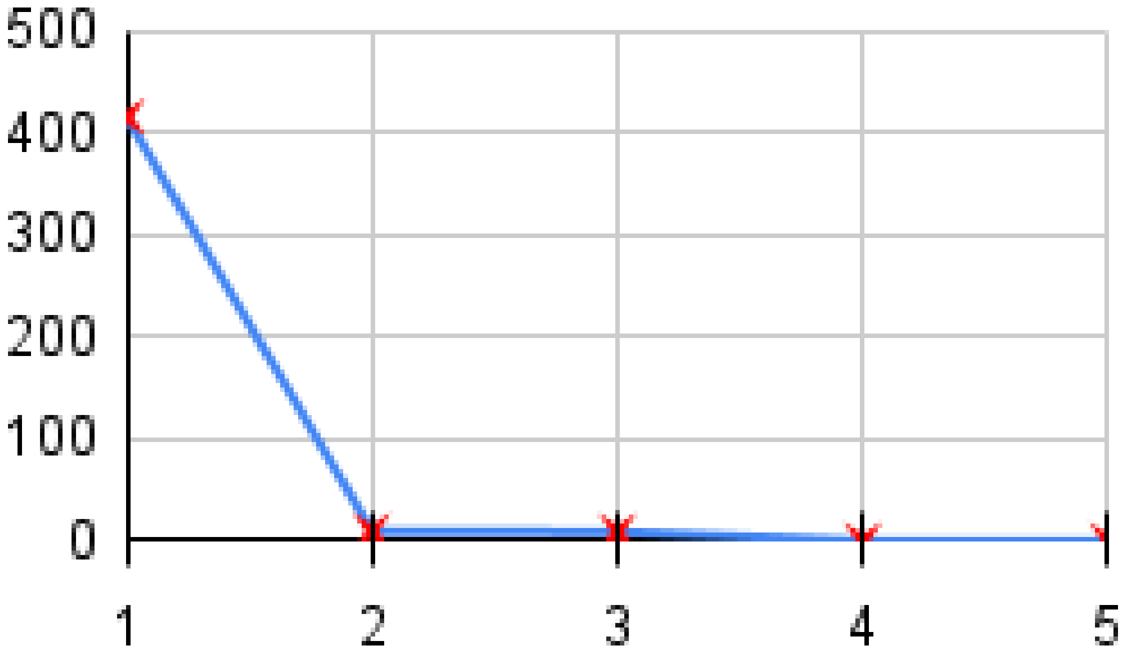


Figure 5.19: The five most significant eigenvalues of the NSE velocity field for the configuration in 5.15.

In the Figure 5.15, the abrupt decay of the eigenvalue magnitude can be observed. This sheds light on the fact that the largest few eigenvalues hold the most helpful information about the properties and nature of the velocity in the fluid flow. Now, we can look at the eigenvalues, partial energies, energy ratio, and velocity solutions percentages.

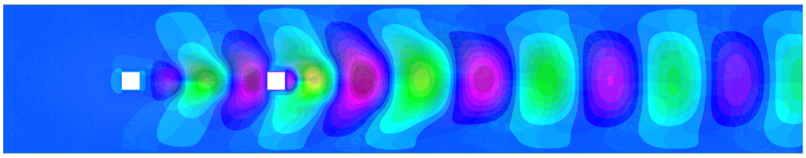
From the energy ratio, it is evident that the highest energy is contained in the first eigenvalue and its corresponding eigenvector. That eigenvector of the correlation matrix is called the POD mode 1. After that, 97 % of the energy is contained in the first two POD modes, which is a significant percentage. And 99.9 % of the energy is conserved

Table 5.11: A comparison of the ten largest eigenvalues, along with their corresponding partial energies, energy ratios, and the energy ratios expressed as percentages, is presented for the velocity field of the NSE.

POD size (m)	m^{th} **eigenvalue**	**Partial energy**	**Energy ratio**	**Energy ratio %**
1	415.329	415.329	0.955	95.517
2	9.635	424.964	0.977	97.733
3	9.112	434.076	0.998	99.829
4	0.278	434.353	0.999	99.893
5	0.266	434.620	0.999	99.954

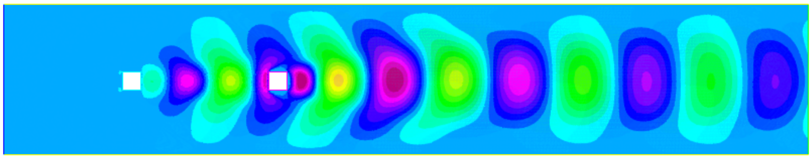
until the fifth mode. The threshold set for this problem is $\delta_e = 99.9$ %, and according to that, the first five POD modes of the velocity problem are enough for the analysis of the fluid flow in the domain. The following figure shows the magnitudes of the first five POD modes [61].

In the diagram 5.20, it is evident that the first two modes exhibit a single-layer configuration, with symmetry about the line at height $h = 0.2$. The second mode shows a more significant separation between the maximum and minimum velocities than the first. The third mode reveals a double-layer structure with an anti-symmetric pattern, where the maximum and minimum velocities alternate. The fourth mode is quite similar, but the double-layer formation appears to merge here. In the fifth mode, we observe the reemergence of the single-layer structure, while the sixth mode, although closely resembling the fifth, displays slight differences. This mode accurately represents the fluid flow characteristics between the times $t = 25\ s$ and $t = 27\ s$. The progression from symmetric to anti-symmetric flow patterns highlights the dynamic nature of the system and the influence of geometric factors on the fluid behavior. These modes provide valuable insights into the flow structures at different temporal stages, which can be crucial for understanding vortex shedding and aerodynamic forces.



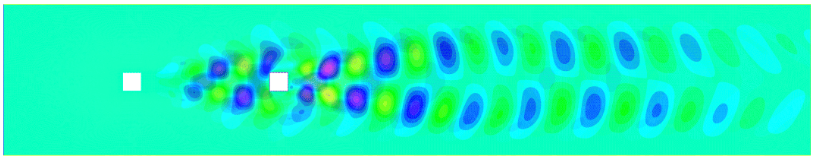
(a)

First velocity mode magnitude



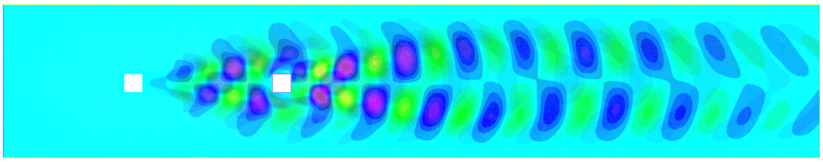
(b)

Second velocity mode magnitude



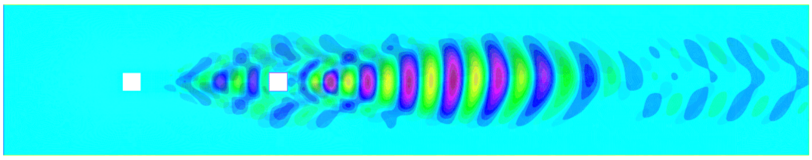
(c)

Third velocity mode magnitude



(d)

Forth velocity mode magnitude



(e)

Fifth velocity mode magnitude

FIGURE 5.20: Magnitudes of the first six most significant velocity POD modes from the NSE for the configuration in 5.15.

Chapter 6

Conclusion

This thesis presented an in-depth investigation into the application of Proper Orthogonal Decomposition (POD) to the Navier-Stokes equations across three distinct geometries. The study began by introducing the fundamental principles of POD, followed by its application to the benchmark geometry established by Schäfer et al., where the method was validated against known results. The geometries studied included two with circular cylinders placed at distances of 0.4 m and 0.45 m, and a third with square cylinders spaced 0.8 m apart. All geometries were subjected to Finite Element Method (FEM) simulations, and the results were analyzed using POD. The analysis provided insights into the fluid flow characteristics within these geometries, with a particular focus on the calculation of eigenvalues, POD modes, drag and lift coefficients, and energy contributions of each mode.

The results demonstrated the power of POD in extracting the dominant flow features, particularly the patterns associated with vortex shedding, which played a significant role in the fluid dynamics of the systems. In the first two geometries, the drag and lift coefficient graphs exhibited periodic behavior. However, cylinder 2, influenced by vortex shedding from the first cylinder, showed greater asymmetry, leading to higher lift forces on the second cylinder. The periodic nature of the flow was clearly evident in these geometries, where the interaction between the cylinders influenced the wake structure and the resulting lift and drag forces. In the third geometry, involving the square cylinder, irregular vortex shedding from the sharp edges caused drag asymmetry. The turbulent wake behind the square cylinder led to more frequent vortex formation, contributing to higher fluctuations in the lift force and inducing oscillations at a higher frequency. These observations highlight the complex interactions between geometry and fluid dynamics, where vortex shedding plays a crucial role in the resulting aerodynamic forces.

POD analysis across all three geometries revealed that the first 5 to 6 modes captured over 99.9% of the total energy, effectively representing the dominant flow features such as velocity fluctuations and vortex shedding. This result underscores the efficiency of POD in capturing the essence of fluid dynamics with a reduced computational burden. The reduction in modes, while maintaining accuracy, suggests that POD is a powerful tool for simplifying complex flow analyses without compromising critical details. Moreover, the calculated lift and drag coefficients provided valuable insights into the fluid-structure interactions, which are essential for optimizing designs, particularly in minimizing drag or maximizing lift in aerodynamic and hydrodynamic applications.

In the future, this work can be extended to investigate more complex and irregular geometries, such as those involving turbulent flows or unsteady boundary conditions. The application of POD to such flows could reveal new insights into the dynamic behavior of fluid systems. Additionally, integrating POD with machine learning techniques could lead to the development of surrogate models, which would enable real-time flow predictions in engineering simulations. Finally, the energy analysis and mode extraction techniques employed in this study could be applied to optimize designs in fields such as aerodynamics, hydrodynamics, and renewable energy, where understanding dominant flow structures is crucial for improving performance and efficiency. This study thus provides a foundation for further research into efficient and accurate methods for analyzing fluid dynamics, with potential applications in various engineering disciplines.

Bibliography

[1] George Keith Batchelor. *An introduction to fluid dynamics.* Cambridge university press, 2000.

[2] Young W Kwon and Hyochoong Bang. *The finite element method using MATLAB.* CRC press, 2018.

[3] Junuthula Narasimha Reddy. An introduction to the finite element method. *New York*, 27(14), 1993.

[4] Muhammad Sabeel Khan. Analytical results for the equilibrium configurations of a three-phase elastic material. *AIP Advances*, 9(2), 2019.

[5] Muhammad Sabeel Khan, M Asif Memon, Ilyas Khan, and Sayed M Eldin. Finite element based direct and iterative approach to investigate a magneto-micropolar flow through a rectangular channel. *Alexandria Engineering Journal*, 75:55–66, 2023.

[6] Muhammad Sabeel Khan. Computational heat transfer analysis of liquid food in a heat exchanger with varying stirrer settings in a nonclassical continuum framework. *Numerical Heat Transfer, Part B: Fundamentals*, pages 1–20, 2024.

[7] Juan C Heinrich. *The finite element method: basic concepts and applications with MATLAB, MAPLE, and COMSOL.* CRC press, 2017.

[8] Anders Logg, Kent-Andre Mardal, and Garth Wells. *Automated solution of differential equations by the finite element method: The FEniCS book*, volume 84. Springer Science & Business Media, 2012.

[9] Alexander G Kuzmin. Using featool multiphysics software for cfd education. *International Journal of Open Information Technologies*, 12(1):114–117, 2024.

[10] Long Chen. ifem: an innovative finite element methods package in matlab. *Preprint, University of Maryland*, 20, 2008.

[11] Frédéric Hecht, Olivier Pironneau, A Le Hyaric, and K Ohtsuka. Freefem++ manual. *Laboratoire Jacques Louis Lions*, pages 1–188, 2005.

[12] Roberto Font and Francisco Peria. The finite element method with freefem++ for beginners. *Electronic Journal of Mathematics & Technology*, 7(4), 2013.

[13] René Pinnau. Model reduction via proper orthogonal decomposition. In *Model order reduction: theory, research aspects and applications*, pages 95–109. Springer, 2008.

[14] Muruhan Rathinam and Linda R Petzold. A new look at proper orthogonal decomposition. *SIAM Journal on Numerical Analysis*, 41(5):1893–1925, 2003.

[15] Anindya Chatterjee. An introduction to the proper orthogonal decomposition. *Current science*, pages 808–817, 2000.

[16] Gal Berkooz. Observations on the proper orthogonal decomposition. *Studies in Turbulence*, pages 229–247, 1992.

[17] Sanjeev Sanghi and Nadeem Hasan. Proper orthogonal decomposition and its applications. *Asia-Pacific Journal of Chemical Engineering*, 6(1):120–128, 2011.

[18] W Aquino, JC Brigham, CJ Earls, and N Sukumar. Generalized finite element method using proper orthogonal decomposition. *International journal for numerical methods in engineering*, 79(7):887–906, 2009.

[19] Adam Fic, Ryszard A Białecki, and Alain J Kassab. Solving transient nonlinear heat conduction problems by proper orthogonal decomposition and the finite-element method. *Numerical Heat Transfer, Part B: Fundamentals*, 48(2):103–124, 2005.

[20] Mehdi Dehghan, Baharak Hooshyarfarzin, and Mostafa Abbaszadeh. Numerical simulation based on a combination of finite-element method and proper orthogonal decomposition to prevent the groundwater contamination. *Engineering with Computers*, 38(Suppl 4):3445–3461, 2022.

[21] Zhendong Luo, Jiang Zhu, Ruiwen Wang, and I Michael Navon. Proper orthogonal decomposition approach and error estimation of mixed finite element methods for the tropical pacific ocean reduced gravity model. *Computer Methods in Applied Mechanics and Engineering*, 196(41-44):4184–4195, 2007.

[22] Michael Schäfer, Stefan Turek, Franz Durst, Egon Krause, and Rolf Rannacher. *Benchmark computations of laminar flow around a cylinder.* Springer, 1996.

[23] Julien Weiss. A tutorial on the proper orthogonal decomposition. In *AIAA aviation 2019 forum*, page 3333, 2019.

[24] Howard Anton, Irl Bivens, and Stephen Davis. *Calculus early transcendentals.* Chichester, 2013.

[25] Robert A Adams and John J Fournier. *Sobolev spaces*, volume 140. Elsevier, 2003.

[26] Howard Anton and Chris Rorres. *Elementary linear algebra: applications version.* John Wiley & Sons, 2013.

[27] Gilbert Strang. *Introduction to linear algebra.* SIAM, 2022.

[28] Vishal Jagota, Aman Preet Singh Sethi, and Khushmeet Kumar. Finite element method: an overview. *Walailak Journal of Science and Technology (WJST)*, 10 (1):1–8, 2013.

[29] Ben Noble, James W Daniel, et al. *Applied linear algebra*, volume 3. Prentice-Hall Englewood Cliffs, NJ, 1977.

[30] Gene H Golub and Charles F Van Loan. *Matrix computations.* JHU press, 2013.

[31] Antonella Falini. A review on the selection criteria for the truncated svd in data science applications. *Journal of Computational Mathematics and Data Science*, 5: 100064, 2022.

[32] Nastaran Chitsaz, Ali Azarnivand, and Shahab Araghinejad. Pre-processing of data-driven river flow forecasting models by singular value decomposition (svd) technique. *Hydrological Sciences Journal*, 61(12):2164–2178, 2016.

[33] L Cordier and M Bergmann. Réduction de dynamique par décomposition orthogonale aux valeurs propres (pod), 2006.

[34] Ranjeet Kumar, Utpreksh Patbhaje, and Anil Kumar. An efficient technique for image compression and quality retrieval using matrix completion. *Journal of King Saud University-Computer and Information Sciences*, 34(4):1231–1239, 2022.

[35] Kohei Inoue, Kenji Hara, and Kiichi Urahama. Matrix principal component analysis for image compression and recognition. *Proc. 1st Joint Ws. Mach. Perc. Rob*, pages 115–120, 2005.

[36] Peng Yi-Gang, Suo Jin-Li, DAI Qiong-Hai, and XU Wen-Li. From compressed sensing to low-rank matrix recovery: theory and applications. *Acta Automatica Sinica*, 39(7):981–994, 2013.

[37] Nicholas J Higham. *Matrix nearness problems and applications.* University of Manchester. Department of Mathematics, 1988.

[38] Kyunghee Han, Pantelis Z Hadjipantelis, Jane-Ling Wang, Michael S Kramer, Seungmi Yang, Richard M Martin, and Hans-Georg Müller. Functional principal component analysis for identifying multivariate patterns and archetypes of growth, and their association with long-term cognitive development. *PloS one*, 13(11): e0207073, 2018.

[39] Ian T Jolliffe. *Principal component analysis for special types of data.* Springer, 2002.

[40] Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

[41] RÄNÜ Preisendorfer. Principal component analysis in meteorology and oceanography. *Elsevier Sci. Publ.*, 17:425, 1988.

[42] Stefan Volkwein. Model reduction using proper orthogonal decomposition. *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz. see http://www. uni-graz. at/imawww/volkwein/POD. pdf*, 1025, 2011.

[43] Michel Bergmann and Laurent Cordier. Contrôle optimal par réduction de modèle pod et méthode à région de confiance du sillage laminaire d'un cylindre circulaire. *mécanique & industries*, 8(2):111–118, 2007.

[44] John Leask Lumley. The structure of inhomogeneous turbulent flows. *Atmospheric turbulence and radio wave propagation*, pages 166–178, 1967.

[45] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of applied mathematics*, 45(3):561–571, 1987.

[46] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. ii. symmetries and transformations. *Quarterly of Applied mathematics*, 45(3):573–582, 1987.

[47] Lawrence Sirovich. Turbulence and the dynamics of coherent structures, parts i, ii and iii. *Quart. Appl. Math.*, pages 561–590, 1987.

[48] Julian Roth. Proper orthogonal decomposition for fluid mechanics problems, 2021.

[49] Vivette Girault and Pierre-Arnaud Raviart. *Finite element methods for Navier-Stokes equations: Theory and Algorithms.* Springer-Verlag, Berlin; New York, 1986.

[50] Rolf Rannacher. Finite element methods for the incompressible navier-stokes equations. In *Numerical Methods for Fluid Dynamics*, pages 191–293. Birkhäuser Basel, Basel, 2000.

[51] Thomas Wick. Lecture notes in "modeling, discretization, optimization, and simulation of fluid-structure interaction", Nov 2019. Unpublished lecture notes.

[52] Frédéric Hecht. New development in freefem++. *Journal of numerical mathematics*, 20(3-4):251–266, 2012.

[53] Rolf Rannacher Heywood, John G. and Stefan Turek. Artificial boundaries and flux and pressure conditions for the incompressible navier–stokes equations. *International Journal for Numerical Methods in Fluids*, 22(5):325–352, 1996.

[54] Thomas Wick. *Numerical Methods for Partial Differential Equations.* 01 2022. doi: 10.15488/11709.

[55] David Reese Wells. *Stabilization of POD-ROMs.* PhD thesis, 2015.

[56] Swetlana Giere. *Numerical and analytical aspects of pod-based reduced-order modeling in computational fluid dynamics.* PhD thesis, 2016.

[57] Alfonso Caiazzo, Traian Iliescu, Volker John, and Swetlana Schyschlowa. A numerical investigation of velocity-pressure reduced order models for incompressible flows. *Journal of Computational Physics*, 259:598–616, Feb 2014. doi: 10.1016/j.jcp.2013.12.004.

[58] Thomas F Geyer. Experimental evaluation of cylinder vortex shedding noise reduction using porous material. *Experiments in Fluids*, 61:1–21, 2020.

[59] Raheela Manzoor, Rida Habib, Noreen Nadeem, and Shazia Kalsoom. Numerical analysis of flow past over a triangular rod through passive control method. *Physics of Fluids*, 36(3), 2024.

[60] Jingcheng Hao, Siva Ramalingam, Md Mahbub Alam, Shunlin Tang, and Yu Zhou. Pod analysis of the wake of two tandem square cylinders. *Fluids*, 9(9):196, 2024.

[61] Chao Xia, Zheng Wei, Haidong Yuan, Qiliang Li, and Zhigang Yang. Pod analysis of the wake behind a circular cylinder coated with porous media. *Journal of Visualization*, 21:965–985, 2018.