Reduced Order Modeling of Geophysical Flows Using Physics-based and Data-driven Modeling Techniques

By

Sk. Mashfiqur Rahman

Bachelor of Science in Mechanical Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
March, 2016

Submitted to the Faculty of the
Graduate College of
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
July, 2019

Reduced Order Modeling of Geophysical Flows Using Physics-based and Data-driven Modeling Techniques

Thesis Approved:

Omer San

Thesis Advisor

Arvind Santhanakrishnan

Rushikesh Kamalapurkar

ACKNOWLEDGMENTS

All praises are due to Almighty for the opportunities and success I have been blessed with. I would like to thank my advisor, Dr. San for his watchful guidance, motivations, and patience. Without his valuable insights and contributions, none of this thesis work would have been possible. I would like to express my heartily gratitude towards Dr. Santhanakrishnan and Dr. Kamalapurkar for being a part of my defense committee. In addition, I would like to thank my labmates for always being supportive to me throughout the progress of my MS degree. Thanks to my family, friends, the fellow graduate students of MAE, OSU, Bangladeshi students association (BSA, OSU), respected faculties and staffs of OSU, and those who are directly or indirectly involved with this thesis work for their encouragement. Finally, I am grateful to MAE department and graduate college of Oklahoma State University for providing me the opportunity and financial assistance to pursue my MS degree in this prestigious institution. I would also like to extend thanks to Oklahoma NASA EPSCOR and the US DOE Office of Science for their financial support.

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Dedicated to my loving parents, and my brother.

Name: Sk. Mashfiqur Rahman

Date of Degree: July, 2019

Title of Study: REDUCED ORDER MODELING OF GEOPHYSICAL FLOWS USING PHYSICS-BASED AND DATA-DRIVEN MODELING TECHNIQUES

Major Field: Mechanical & Aerospace Engineering

Abstract: The growing advancements in computational power, algorithmic innovation, and the availability of data resources have started shaping the way we numerically model physical problems now and for years to come. Many of the physical phenomena, whether it be in natural sciences and engineering disciplines or social sciences, are described by a set of ordinary differential equations or partial differential equations which is referred as the mathematical model of a physical system. High-fidelity numerical simulations provide us valuable information about the flow behavior of the physical system by solving these sets of equations using suitable numerical schemes and modeling tools. However, despite the progress in software engineering and processor technologies, the computational burden of high-fidelity simulation is still a limiting factor for many practical problems in different research areas, specifically for the large-scale physical problems with high spatio-temporal variabilities such as atmospheric and geophysical flows. Therefore, the development of efficient and robust algorithms that aims at achieving the maximum attainable quality of numerical simulations with optimal computational costs has become an active research question in computational fluid dynamics community. As an alternative to existing techniques for computational cost reduction, reduced order modeling (ROM) strategies have been proven to be successful in reducing the computational costs significantly with little compromise in physical accuracy. In this thesis, we utilize the state of the art physics-based and data-driven modeling tools to develop efficient and improved ROM frameworks for large-scale geophysical flows by addressing the issues associated with conventional ROM approaches. We first develop an improved physics-based ROM framework by considering the analogy between dynamic eddy viscosity large eddy simulation (LES) model and truncated modal projection, then we present a hybrid modeling approach by combining projection based ROM and extreme learning machine (ELM) neural network, and finally, we devise a fully data-driven ROM framework utilizing long short-term memory (LSTM) recurrent neural network architecture. As a representative benchmark test case, we consider a two-dimensional quasi-geostrophic (QG) ocean circulation model which, in general, displays an enormous range of fluctuating spatial and temporal scales. Throughout the thesis, we demonstrate our findings in terms of time series evolution of the field values and mean flow patterns, which suggest that the proposed ROM frameworks are robust and capable of predicting such fluid flows in an extremely efficient way compared to the conventional projection based ROM framework.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

CHAPTER I

Introduction

Modeling of large-scale turbulent flows has been subject to an active research for over the past half century due to its impact on a variety of applications, such as ocean modeling, weather forecasting, aeronautical applications, energy harvesting and so on. The reliable prediction, control and diagnostics of the chaotic nature of these turbulence phenomena have always been a challenging task from the very beginning of turbulence studies and until now, a plethora of theoretical and computational studies have been dedicated to understand the characteristics of turbulence. High-fidelity numerical simulations like direct numerical simulation (DNS) are capable of capturing the broad range of active spatio-temporal scales in turbulent motions as the full energy spectra of turbulence can be resolved down to the Kolmogorov scale using a very fine grid resolution. Due to the continuous growth of the computer power, we now can obtain DNS of a number of simple turbulent flow problems. However, the amount of computational resources required to resolve large-scale turbulence is still unmanageable in most of the cases and even if some of the scales can be resolved, the process can turn out to be extremely inefficient and computationally expensive. It should be noted as well that the continuous progress in computer power and performance following Moore's law has started to become saturated in the recent years. As a result, for obvious reasons, a lot of active research projects in different fields are devoted to develop efficient and robust modeling algorithms that enhance the capability of maximum attainable quality of numerical simulations with optimal computational costs. Among these, traditional computational fluid dynamics

(CFD) researches were mostly based on low-fidelity or coarse grained models by introducing modeling assumptions and often required a closure model to compensate the effects of truncated scales. Reynolds averaged Navier-Stokes (RANS) and large eddy simulation (LES) approaches are the most common coarse-grained models, although there exist many alternatives. In these approaches, the similar high-fidelity model is used with a reduced number of grid points. As an alternative to the existing traditional modeling techniques, reduced order modeling (ROM) strategies have been proven to be successful in reducing computational expenses greatly and have got a significant attention to CFD research community. In conventional snapshot-based ROM approach, the existing data snapshots, obtained from either experiments, field measurements or high-dimensional numerical simulations, are utilized to generate a lower dimensional system with an acceptable range of accuracy through a variety of mathematical operations. There are a several established ways to obtain ROM where each has their own strengths and weaknesses. Rowley and Dawson have reviewed some of the most popular ROM techniques, such as proper orthogonal decomposition (POD), dynamic mode decomposition (DMD), eigensystem realization algorithms (ERA) and so on, along with the similarities and discrepancies between them (Rowley and Dawson, 2017). In this thesis, we focus mainly on the improvement in the performance of POD based ROM by addressing their limitations using physics-based and data-driven modeling ideas.

Developing ROMs of nonlinear dynamical systems has been introduced to the computational fluid dynamics as a computational cost reduction approach and there have been a number of ROM techniques proposed over the years. These ROM techniques have been utilized for a wide variety of applications related to, for example, flow control, data assimilation in weather and climate modeling, uncertainty quantification, and so on. Among the different variants of ROM strategies, the Galerkin-projection combined with proper orthogonal decomposition (POD) based ROMs have become

widely popular in various areas. POD is a mathematical technique to extract the dominant statistical characteristics from a complex dynamical system by identifying the modes carrying most of the information of overall system (Lumley, 1967; Benner et al., 2015; Rowley and Dawson, 2017). These few selected POD modes have the capability of representing the fine-scale details of the true physics accurately. A detailed literature review on Galerkin-projection based ROM and different updates of POD techniques are provided in the following chapters. In general, POD uses the data obtained from experiments, field measurements or high-fidelity numerical simulations and generates an orthonormal set of spatial basis vectors describing the main directions (modes) by which the flow is represented optimally (Berkooz et al., 1993). The most energetic modes are kept to generate the reduced order system while the other modes are truncated. Indeed, there are several limitations in conventional ROM strategies, for example, getting diverging solution due to the effect of truncated modes. Often the truncated modes contribute to the evolving dynamics of complex multidimensional turbulent flows like geophysical flows. Several other issues related to POD and projection based ROM and some possible solutions proposed in different recent literature are discussed with references in the later chapters.

In this thesis work, we present three different ideas to develop an efficient and robust POD based ROM framework using three distinct type of modeling approaches: physics-based (intrusive) approach, hybrid (semi non-intrusive) approach and data-driven (fully non-intrusive) approach. To be specific, the main goal is to capture the true flow behavior using minimal amount of POD modes with optimal computational expenses. In physics-based approaches, the online ROM computation is done using some parts (or whole) of the high-fidelity model, i.e., the underlying governing equation is required to generate the ROM. In contrast, the data-driven modeling only requires the high-fidelity data snapshots to generate the ROM, i.e., no prior information about the physical system or model is required. In the data-driven approach, we replace

the physics-based Galerkin projection with any state-of-the-art data-driven tools to make the whole system fully non-intrusive or data-driven. Until recently, purely data-driven machine learning (ML) algorithms have been considered one of the most promising fields to benefit greatly from existing datasets. In the past few years, artificial neural networks (ANNs) and other ML techniques have started to make an impact in the turbulence modeling community (Raissi et al., 2019; Maulik et al., 2018; Faller and Schreck, 1997; San and Maulik, 2018b; Wang et al., 2019; Moosavi et al., 2015; Kani and Elsheikh, 2017). Interested readers are directed to Refs. (Brunton et al., 2019; Kutz, 2017; Durbin, 2018; Duraisamy et al., 2018; Gamboa, 2017) for more information on ML in fluid mechanics, specifically turbulence modeling. Both physics-based and data-driven modeling strategies have some positive and negative aspects. This has led to the development of a another category of ROM framework, known as hybrid modeling approach, where both physics-based and data-driven approaches are combined in such a way that a significantly enhanced performance can be obtained than the performance of component models alone. A detailed literature review and discussion on physics-based, hybrid, and data-driven ROM techniques are presented in the following chapters.

To evaluate the performance of any ROM framework, the selection of test problem is particularly very crucial. This is because the development of ROM starts to become very challenging for large-scale turbulent flow systems with complex, nonlinear spatio-temporal scales interactions. As a result, an increased number of retained modes are required to obtain a physical and stable solution. However, increasing number of modes beyond some threshold would increase the computational cost of solving ROMs or even exceeds the cost of the original full-order model. Hence, a better ROM model should be capable of resolving the scales of such flow problems with a minimal amount of POD modes. To test our proposed frameworks, we consider the barotropic vorticity equation (BVE) representing the single-layer quasi-geostrophic (QG) ocean

circulation model as a representative of large-scale geophysical flow problems. It is well-known that much of the world's ocean circulation is wind-driven in large-scale. Furthermore, there is an apparent connection between ocean currents and climate dynamics since the oceans are a significant part of the climate system. Due to the difference in heat capacity and density between atmospheric air and ocean water, the ocean currents are comparatively quite slow with respect to the movement of air masses in the atmosphere. The slowness of ocean current means a long timescale motions which regulate a connection between the typically shorter temporal motions of the atmosphere with the long time evolution of the climate system. It should be noted that the ocean circulation has effects on the weather as well on shorter timescales. Hence, wind-driven flows of mid-latitude ocean basins have been studied frequently by modelers using idealized single- and double-gyre wind forcing, which helps in understanding various aspects of ocean dynamics, including the role of mesoscale eddies and their effect on mean circulation (Holm and Nadiga, 2003; San et al., 2011). However, modeling the vast range of spatio-temporal scales of the oceanic flows with all the relevant physics has always been challenging. As a result, the numerical simulation of oceanic and atmospheric flows still requires approximations and simplifications of the full model. The BVE model is a simplified version of the full-fledged geophysical flow equations by considering the hydrostatic balance, geostrophic balance, the β-plane approximation, and horizontal eddy viscosity parameterizations. To avoid complexity, we perform all the analyses in a single test problem only to quantify the predictive performance of the proposed ROM frameworks with respect to conventional projection based ROM and full order model simulation results. Throughout this document, we assess and quantify the performance of the models using mean flow fields, L_2-error norms, and time series response of field values for different flow conditions. The nature and physical aspects of the wind-driven QG turbulence are detailed with references in the later chapters.

The individual chapters of this thesis apart from the introduction and conclusions are dedicated to an individual ROM development idea based on different modeling approaches. A brief summary of the core chapters of this document is presented below:

1. In chapter 2, we present a physics-based, dynamic closure modeling approach to stabilize the projection-based ROMs for the forced-dissipative QG systems. We propose an eddy viscosity closure approach to stabilize the resulting surrogate model considering the analogy between LES and truncated modal projection. Our efforts, in particular, include the translation of the dynamic subgrid-scale model into our ROM setting by defining a test truncation similar to the test filtering in LES (Rahman et al., 2019a).

2. In chapter 3, we put forth a robust hybrid ROM framework for the QG system by introducing a weighting parameter between the Galerkin projection (physics-based contribution) and extreme learning machine neural network architecture (data-driven contribution) (Rahman et al., 2018).

3. In chapter 4, we propose a fully data-driven ROM framework for large-scale QG systems which exploits the time series prediction capability of long short-term memory (LSTM) recurrent neural network architecture such that: (i) in the training phase, the LSTM model is trained on the modal coefficients extracted from the high-resolution data snapshots using POD transform, and (ii) in the testing phase, the trained model predicts the modal coefficients for the total time recursively based on the initial time history. The mean flow fields and time series response of the field values are then reconstructed from the predicted modal coefficients by using an inverse POD transform (Rahman et al., 2019b).

CHAPTER II

A Dynamic Closure Modeling Framework for Model Order Reduction of Geophysical Flows[†]

2.1 Abstract

In this paper, a dynamic closure modeling approach has been derived to stabilize the projection-based reduced order models in the long-term evolution of forced-dissipative dynamical systems. To simplify our derivation without losing generalizability, the proposed reduced order modeling (ROM) framework is first constructed by Galerkin projection of the single-layer quasi-geostrophic equation, a standard prototype of large-scale general circulation models, onto a set of dominant proper orthogonal decomposition (POD) modes. We then propose an eddy viscosity closure approach to stabilize the resulting surrogate model considering the analogy between large eddy simulation (LES) and truncated modal projection. Our efforts, in particular, include the translation of the dynamic subgrid-scale model into our ROM setting by defining a test truncation similar to the test filtering in LES. The a posteriori analysis shows that our approach is remarkably accurate, allowing us to integrate simulations over long time intervals at a nominally small computational overhead.

2.2 Introduction

High-fidelity numerical simulations are crucial for reliable predictions, control and diagnostics. Thanks to the huge advancement in computational power, in terms of speed (e.g., number of arithmetic operations per second) and memory, computational

[†]The chapter is adapted from Rahman et al., Physics of Fluids 31.4 (2019): 046602.

fluid dynamics has witnessed substantial development during the last century. This includes using much finer numerical resolution and fewer approximations. However, there are still some situations in which computational resources cannot meet the requirements for feasible simulations. This is evident, especially in numerical weather predictions, where accurate simulations are solely applicable to regional weather models, while solution of global models is still restricted to relatively coarse grids (Kalnay, 2003; Powers et al., 2017).

Although the progress in computer power and performance has been adequately following Moore's law (Moore, 1965, 1975) during the past decades, it is becoming increasingly obvious in worldwide semiconductor industry that it is nearing its end (Powell, 2008; Waldrop, 2016; Kumar, 2018). Therefore, the development of efficient algorithms that elevate the maximum attainable quality of numerical simulations with the available resources, or at least reduce the computational cost of traditional simulations, has become a must. The latter is particularly important when multiple forward simulations are required, like those encountered in inverse problems (Daescu and Navon, 2007; Navon, 2009; Cao et al., 2007; He et al., 2011; Houtekamer and Mitchell, 1998, 2001; Bennett, 2005; Evensen, 2009; Law and Stuart, 2012; Buljak, 2011). Reduced order modeling (ROM), also known as model order reduction, is such a way of representing high-dimensional systems with much lower-dimensional (but dense) systems, resulting in substantial reduction in computational cost while keeping output quality within acceptable range (Quarteroni et al., 2015; Taira et al., 2017). This is feasible due to the fact that most high-dimensional complex systems, basically follow low-dimensional characteristic dynamics. For example, complex fluid flows often consist of superposition of spatially or temporally developing coherent structures, either growing/decaying with a specific rate, oscillating with constant frequency or containing the largest possible kinetic energy. The evolution of such structures is responsible for the bulk mass, momentum, and energy transfer. Therefore, the development of

reduced order surrogate models through extracting these underlying characteristics, would be an effective way to reduce the computational cost of numerical simulations and address more complex problems.

Among different ROM techniques, snapshot-based projection methods are particularly important where the time response of a system, either recorded from experiments or high-fidelity numerical simulations and given a certain input, is assumed to contain the essential behavior of that system. Proper orthogonal decomposition (POD), also known as principal component analysis, is a widely popular technique for snapshot-based ROMs (Antoulas et al., 2001a; Chinesta et al., 2011; Benner et al., 2015; Rowley and Dawson, 2017), and it has been introduced to the fluid dynamics community as a mathematical technique to extract coherent structures from turbulent flow fields (Lumley, 1967). The spatiotemporal biorthogonal decomposition, a variant of POD, has been introduced to recover the temporal information associated with the POD modes (Aubry et al., 1991; Aubry, 1991). More recently, a spectral POD (SPOD) (Sieber et al., 2016) has been applied to extract low-dimensional compression of the full order model (FOM) extending POD to account for temporal dynamics in addition to energetic optimality. Although the spatial SPOD modes are no longer orthonormal, it was shown that the norm of the spatial modes gives further insights into the data set and reduces POD in the limiting case. A frequency domain form of POD (also called as SPOD) has been also introduced in order to account for the statistical variability of turbulent flows and a connection between their approach and resolvent analysis has been established. Loiseau et al. (Loiseau et al., 2018) have also introduced manifold modeling approach as a potentially viable compression model. They showed that a two-dimensional manifold is more accurate than an expansion with using 50 POD modes for the transient cylinder wake.

In general, POD computes a set of orthonormal basis vectors which describe the main directions (modes), by which the given dataset is characterized, in the L_2 sense

(Berkooz et al., 1993). Based on the energy cascade between different modes, the most energetic POD modes are selected to generate a reduced order system. POD coupled with the Galerkin projection has been considered an efficient approach to generate ROMs for linear and nonlinear systems (Ito and Ravindran, 1998; Iollo et al., 2000; Rowley et al., 2004; Pinnau, 2008; Sachs and Volkwein, 2010; Noack et al., 2017; Puzyrev et al., 2019). It has been applied in a large number of problems involving fluid flows during the past few decades where the governing equations are projected onto these selected modes. While the high-dimensional approximation using standard discretization techniques often generates spaces with millions of degrees of freedoms, the reduced spaces spanned by ROMs are typically of order 100 or smaller (Milk et al., 2016). In practice, ROM approximation can lead to speedups of several orders of magnitude as well as great reduction in memory requirements. In fluid dynamics applications, the resulting dense system consists of triadic interactions due to the quadratic nonlinearity with an order of $O(R^3)$ computational load, where R refers to the retained number of modes.

However, it has been noted that truncated modes often contribute to the evolving dynamics of complex multidimensional turbulent flows, especially encountered in geophysical systems (Lassila et al., 2014), resulting in diverging POD-based solution. This leads to either increasing the number of selected modes to better embed the underlying system, or sacrificing results quality. The latter is, of course, unacceptable and would ruin the reliability and applicability of such models. On the other hand, increasing number of modes beyond some threshold, would increase the computational cost of solving ROMs in a way that approaches, or even exceeds, the cost of the original full-order model. Moreover, Rempfer (Rempfer, 2000) has shown that a complete set of POD modes is not sufficient for a POD-Galerkin model to reproduce the full order dynamics accurately, even for many non-turbulent flows. He found that adding just small perturbations to the flow field (like those created by a small

numerical error in the integration of the ODEs) would result in a ROM that, in general, may not faithfully represent the full order model anymore, and the dynamics of the POD-Galerkin model could show instabilities while the true dynamics of the system is stable. Noack et al. (Noack et al., 2003) also reported the same observations. They proposed including an extra 'shift-mode' that represents the shift of short-term averaged flow away from the POD space such that the Galerkin approximation also includes an accurate representation of the unstable steady solution.

Numerical instabilities due to the truncation in Galerkin models have been addressed considering turbulence models (Aubry et al., 1988; Rempfer, 1994; Rempfer and Fasel, 1994b,a; Cazemier et al., 1998; Protas et al., 2015). Generally, POD modes resolve the production much better than the dissipation, leading to an excess production of turbulent kinetic energy in the POD subspace (Cordier et al., 2013). Several studies have approached this weak dissipation through the introduction of eddy-viscosity terms. Sirisup and Karniadakis (Sirisup and Karniadakis, 2004) proposed a dissipative model based on a spectral viscosity diffusion convolution operator to improve the long-term predictions of Galerkin-based ROMs. In their approach viscosity amplitude decreases with the mode number, and it was shown to guarantee a non-oscillatory behavior except for some negligible bounded oscillations. Cordier et al. (Cordier et al., 2013) reported that constant eddy viscosity, even carefully calibrated, is nonphysical and would lead to incorrect scaling characteristics. They proposed an improved nonlinear eddy-viscosity model using data assimilation techniques (4D-Var in particular). Having a globally defined eddy viscosity coefficient might imply that the coherent structures of a high Reynolds number flow are represented by a fully resolved low Reynolds number computation. For example, Östh et al. (Östh et al., 2014) have used modal eddy viscosities to model high Reynolds number Ahmed body wake.

In this paper, we aim at proposing an automated framework to produce stable

Galerkin projection based reduced order models, using a sufficiently small number of modes, without sacrificing much accuracy through the introduction of closure ideas (Cordier et al., 2013; Sirisup and Karniadakis, 2004; Östh et al., 2014; Couplet et al., 2003; Kalb and Deane, 2007; Kalashnikova and Barone, 2010; Wang et al., 2011, 2012; Borggaard et al., 2011; Akhtar et al., 2012; Amsallem and Farhat, 2012; Balajewicz and Dowell, 2012; San and Iliescu, 2014, 2015; Östh et al., 2014; Protas et al., 2015; Wells et al., 2017). Dynamic eddy viscosity based closure models have been applied in large eddy simulation (LES) area to provide numerical stabilization as well as statistical fidelity preservation using an explicit test filtering procedure (Germano et al., 1991; Vreman, 2004). The filtering operation that is centred in LES has been less investigated in the ROM community. Therefore, extending state-of-the-art LES methodologies to the ROM field has been considered as a promising approach (Xie et al., 2017, 2018b,a; Mohebujjaman et al., 2019). Wang et al.(Wang et al., 2012) developed the variational multiscale and dynamic Smagorinsky models for the POD-based ROMs of structurally dominated turbulent flows, where the POD filtering has been implemented in the reconstructed space. Using dynamic LES and ROM analogy, the chief novelty of this paper is to introduce a "test truncation" mechanism, which is implemented in the reduced order space (i.e., aiming at obtaining an efficient ROM with a small computational overhead), to generate stable, self-adapted, dynamic ROMs for estimating long term dynamics of forced-dissipative turbulent flows.

To the authors' knowledge, the application of such dynamic closure models in ROMs is very limited, and the current work is an effort for such incorporation. Following LES ideology, the eddy viscosity concept is thought to provide an efficient framework to account for the unrepresented scales due to intense mode truncation. Eddy viscosity is computed on the fly using test truncation idea, similar to test filtering in LES. To assess our idea, the barotropic vorticity equation is selected as our test bed. It is a widely used mathematical model to study the forced-dissipative large scale ocean

circulation problems, also known as the single-layer quasi-geostrophic (QG) model, first introduced by Jule Charney (Charney, 1949). We found that the proposed dynamic closure approach gives stabilized results over longer time intervals, compared with regular ROMs, with a negligible computational overhead. While presented in the vorticity-streamfunction formulation, we highlight that our modular approach can be easily generalized to the projection-based reduced order models in primitive variables.

The rest of the paper is organized as follows: Sec. 2.3 describes the governing equations briefly for the adopted test bed to generate snapshots; Sec. 2.4 is devoted to the description and derivation of the dynamic eddy viscosity closure in ROM; in Sec. 2.5, we present and discuss our results of the proposed framework; and Sec. 2.6 provides a summary of this study and the conclusions drawn from it. The numerical discretization schemes used for spatial and temporal derivative approximations as well as the generation and selection criteria of POD modes are specified in the Appendix at the end of this manuscript.

2.3 Barotropic Vorticity Equation Model

Atmospheric and oceanographic flows often take place over horizontal length scales, much larger than their vertical length scale. Therefore, they can be adequately described by using the shallow water equations (SWE). The single layer two-dimensional QG equation is an approximation of the SWE, filtering the inertia-gravity waves, under the following assumptions (Vallis, 2006):

- Rossby number, Ro is small, such that inertial forces are an order of magnitude smaller than the Coriolis and pressure forces,
- Horizontal scale of motion is the same order of magnitude as the deformation scale, implying that the variations in fluid depth are small compared to its total depth,

- Variations in the Coriolis parameter are small,
- The timescale is the advective timescale, hence the gravity waves, which evolve on a short timescale, are filtered out.

Much of the world's ocean circulation is wind-driven in large-scale. Therefore, wind-driven flows of mid-latitude ocean basins have been studied by modelers using idealized single- and double-gyre wind forcing which helps in understanding various aspects of ocean dynamics, including the role of mesoscale eddies and their effect on mean circulation. The barotropic vorticity equation (BVE) describing the single-layer QG equation with dissipative and forcing terms is one of the most commonly used models for the double-gyre wind-driven geophysical flows (Majda and Wang, 2006).

The BVE model is a simplified version of the more general primitive equations used in operational weather forecast centers (Kalnay, 2003), making it a suitable model for testing new ideas. Detailed discussions on different underlying mechanisms and formulations have been presented in literature (Holland and Rhines, 1980; Munk and Wunsch, 1982; Griffa and Salmon, 1989; Cummins, 1992; Greatbatch and Nadiga, 2000; Nadiga and Margolin, 2001). In dimensionless vorticity-streamfunction formulation, using β-plane assumption reasonable for most oceanic flows, the forced-dissipative BVE can be written as follows (Rahman et al., 2018):

$$\frac{\partial \omega}{\partial t} + J(\omega, \psi) - \frac{1}{\mathrm{Ro}} \frac{\partial \psi}{\partial x} = \frac{1}{\mathrm{Re}} \nabla^2 \omega + \frac{1}{\mathrm{Ro}} \sin(\pi y), \tag{2.1}$$

where ∇^2 refers to the Laplacian in two-dimensions, ω and ψ are the kinematic vorticity and streamfunction, respectively, defined as:

$$\omega = \nabla \times \mathbf{u}, \tag{2.2}$$

$$\mathbf{u} = \nabla \times \psi \hat{k}, \tag{2.3}$$

where $\mathbf{u}$ is the two-dimensional velocity field and $\hat{k}$ refers to the unit vector perpendicular to the horizontal plane. The nonlinear advection term in Eq. (2.1) is given by the Jacobian

$$J(\omega, \psi) = \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y}. \tag{2.4}$$

Eq. (2.1) has two dimensionless parameters, Reynolds number, Re and Rossby number, Ro, which are related to the characteristic length and velocity scales in the following way:

$$\mathrm{Re} = \frac{VL}{\nu}, \quad \mathrm{Ro} = \frac{V}{\beta L^2}, \tag{2.5}$$

where ν is the horizontal eddy viscosity of the BVE model and β is the gradient of the Coriolis parameter at the basin center ($y = 0$). L is the basin length scale and V is the velocity scale, also known as the Sverdrup velocity(Sverdrup, 1947), and is given by

$$V = \frac{\tau_0}{\rho H}\frac{\pi}{\beta L}, \tag{2.6}$$

where τ_0 is the maximum amplitude of the double-gyre wind stress, ρ is the mean fluid density, and H is the mean depth of the ocean basin.

Despite not being explicitly represented in Eq. (2.1), there are two important relevant physical parameters, the Rhines scale, δ_I and the Munk scale, δ_M which are the nondimensional boundary layer thicknesses for the inertial and viscous (Munk) layer of the basin geometry, respectively. As a physical interpretation of these parameters in BVE model, δ_I accounts for the strength of nonlinearity and δ_M is a measure of dissipation strength. δ_I and δ_M can be defined as

$$\frac{\delta_I}{L} = \left(\frac{V}{\beta L^2}\right)^{\frac{1}{2}}, \quad \frac{\delta_M}{L} = \left(\frac{\nu}{\beta L^3}\right)^{\frac{1}{3}} \tag{2.7}$$

and are related to Ro and Re by the following relations

$$\frac{\delta_I}{L} = (\mathrm{Ro})^{\frac{1}{2}}, \quad \frac{\delta_M}{L} = \left(\frac{\mathrm{Ro}}{\mathrm{Re}}\right)^{\frac{1}{3}}. \tag{2.8}$$

Finally, in order to satisfy the incompressibility constraint, the vorticity and stream-function are related through the following Poisson equation:

$$\nabla^2 \psi = -\omega. \tag{2.9}$$

Following (Holm and Nadiga, 2003; San et al., 2011), a four-gyre circulation problem is considered as a benchmark for oceanic flow to generate numerical data. Since ocean circulation models where the Munk and Rhines scales are close to each other, like the QG model, remain time dependent rather being converged to a steady state as time approaches to infinity (Medjo, 2000), numerical computations of these models are conducted in a statistically steady state, also known as the quasi-stationary state. Hence, in our study, we utilize numerical schemes suited for simulation of such type of ocean models and for long-time integration. In our full order model (FOM) simulations, we use a second-order accurate kinetic energy and enstrophy conserving Arakawa finite difference scheme (Arakawa, 1966). The derivatives in the linear terms are also approximated using the standard second-order finite differences. Our time advancement scheme is given by the classical total variation diminishing third-order accurate Runge-Kutta scheme (Gottlieb and Shu, 1998). Details of the numerical schemes are given in Appendix.

To close the problem, boundary and initial conditions need to be specified. Following previous studies, we use slip boundary condition for the velocity, which implies homogeneous Dirichlet boundary condition for the vorticity. Also, the impermeability boundary condition forces homogeneous Dirichlet boundary condition for the

streamfunction:

$$\omega|_\Gamma = \psi|_\Gamma = 0, \tag{2.10}$$

where Γ refers to all boundary coordinates. As an initial state, we start our computations from a quiescent state (i.e., $\omega_{t=0} = \psi|_{t=0} = 0$) and integrate the model until a statistically steady state is obtained, i.e., the wind forcing, dissipation, and Jacobian (eddy flux of potential vorticity) balance each other.

2.4 Dynamic Closure Modeling for Reduced Order Models

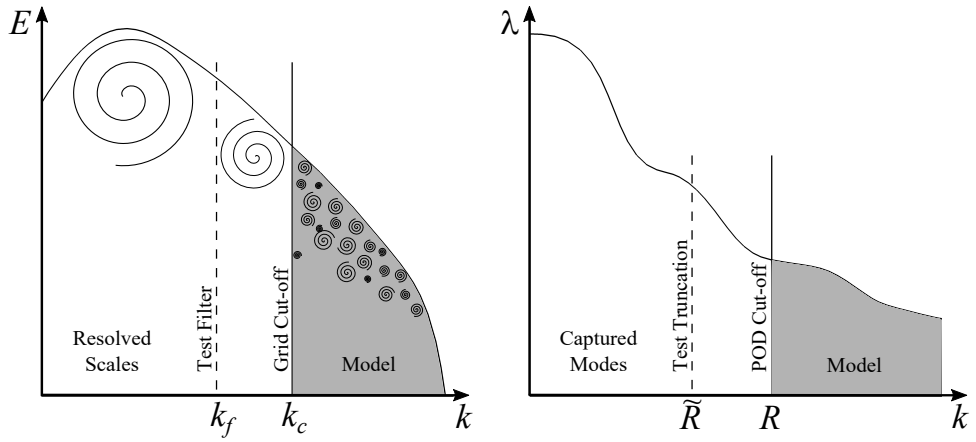


Figure 2.1: Closure modeling analogy between large eddy simulation (left) and reduced order modeling (right) where higher k index refers to smaller scales. In LES, k_c and k_f refer to the grid cut-off and test filtering scales, whereas in ROM, R and $\tilde{R}$ indicate the number of POD modes retained in the model and test truncation process, respectively.

The main idea of this paper is that the effect of truncated modes can be approximated dynamically. To illustrate this online ROM closure idea, we first rewrite the governing equation as

$$\frac{\partial \omega}{\partial t} = -J(\omega, \psi) + \frac{1}{\mathrm{Ro}}\frac{\partial \psi}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\omega + \frac{1}{\mathrm{Ro}}\sin(\pi y), \tag{2.11}$$

where we approximate the prognostic variable (i.e., kinematic vorticity in this case)

using the most energetic R POD modes

$$\omega(x,y,t) = \bar{\omega}(x,y) + \sum_{k=1}^{R} \alpha_k(t)\phi_k(x,y), \tag{2.12}$$

where $\bar{\omega}(x,y)$ refers to the mean vorticity field of the training data set of snapshots, $\alpha_k(t)$ is the kth time-dependent coefficient and $\phi_k(x,y)$ is the kth spatial POD mode for the fluctuating vorticity field. The derivation of such POD modes is detailed in Appendix. We note that the POD modes are orthonormal (both orthogonal and normalized), i.e.,

$$\int_{\Omega} \phi_i(x,y)\phi_j(x,y)dxdy = \delta_{ij}, \tag{2.13}$$

where Ω is the entire spatial domain and δ_{ij} is the Kronecker delta defined by

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{2.14}$$

To simplify our notation we use the following angle-parenthesis definition for the inner product

$$\int_{\Omega} f(x,y)g(x,y)dxdy = \langle f; g \rangle, \tag{2.15}$$

and hence $\langle \phi_i; \phi_j \rangle = \delta_{ij}$. Since the vorticity and streamfunction are related through the kinematic relationship given by Eq. (2.9), we can expand the streamfunction using the same time-dependent coefficient,

$$\psi(x,y,t) = \bar{\psi}(x,y) + \sum_{k=1}^{R} \alpha_k(t)\theta_k(x,y), \tag{2.16}$$

where $\bar{\psi}(x,y)$ refers to the mean streamfunction field of the training data set of snapshots, and $\theta_k(x,y)$ is the kth spatial POD mode for the streamfunction, which

can be obtained through solving the following Poisson equations (offline computing):

$$\nabla^2\bar{\psi}(x,y) = -\bar{\omega}(x,y), \tag{2.17}$$

$$\nabla^2\theta_k(x,y) = -\phi_k(x,y), \quad k = 1,2,...,R. \tag{2.18}$$

We note that the POD modes for streamfunction do not need to be orthonormal since the streamfunction is not a prognostic variable. Substituting Eq. (2.12) and Eq. (2.16) into Eq. (2.11), an orthogonal Galerkin projection is then performed by multiplying Eq. (2.11) with the spatial POD modes $\phi_k(x,y)$, and integrating over the entire domain Ω. The resulting dense dynamical system for α_k can be written as

$$\frac{d\alpha_k}{dt} = \mathfrak{B}_k + \sum_{i=1}^{R}\mathfrak{L}_k^i\alpha_i + \sum_{i=1}^{R}\sum_{j=1}^{R}\mathfrak{N}_k^{ij}\alpha_i\alpha_j, \quad k = 1,2,...,R, \tag{2.19}$$

where the predetermined model coefficients can be computed by the following numerical integration (offline computing)

$$\begin{aligned}
\mathfrak{B}_k &= \Big\langle -J(\bar{\omega},\bar{\psi}) + \frac{1}{\mathrm{Ro}}\frac{\partial\bar{\psi}}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\bar{\omega} + \frac{1}{\mathrm{Ro}}\sin(\pi y); \phi_k\Big\rangle, \\
\mathfrak{L}_k^i &= \Big\langle -J(\bar{\omega},\theta_i) - J(\phi_i,\bar{\psi}) + \frac{1}{\mathrm{Ro}}\frac{\partial\theta_i}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\phi_i; \phi_k\Big\rangle, \\
\mathfrak{N}_k^{ij} &= \Big\langle -J(\phi_i,\theta_j); \phi_k\Big\rangle.
\end{aligned} \tag{2.20}$$

To complete the dynamical system given by Eq. (2.19), the initial conditions for $\alpha_k(t)$ may be obtained by the following projection

$$\alpha_k(t_0) = \big\langle \omega(x,y,t_0) - \bar{\omega}(x,y); \phi_k\big\rangle, \tag{2.21}$$

where $\omega(x,y,t_0)$ is the vorticity field specified at initial time t_0. The standard Galerkin projection ROM given by Eq. (2.19), denoted as ROM-G in this study, often yields unstable solutions when the largest R modes might not adequately capture the

system's dynamics (Couplet et al., 2003; Kalb and Deane, 2007; Bergmann et al., 2009; Kalashnikova and Barone, 2010; Wang et al., 2011; Carlberg et al., 2011; Wang et al., 2012; Borggaard et al., 2011; Akhtar et al., 2012; Amsallem and Farhat, 2012; Balajewicz and Dowell, 2012; Balajewicz et al., 2013; Lassila et al., 2013; San and Iliescu, 2014, 2015; Baiges et al., 2015; Wells et al., 2017; Xie et al., 2017). As we will illustrate in our numerical examples, this is particularly true for turbulent flows. Using an eddy viscosity approach (e.g., see FIG. 2.1 for the analogy between ROM and LES that is first illustrated by Akhtar et al. (Akhtar et al., 2012)), the stabilization of the ROM can be achieved by adding a regularization term to the governing equation (San and Iliescu, 2015)

$$\frac{\partial \omega}{\partial t} = -J(\omega, \psi) + \frac{1}{\mathrm{Ro}}\frac{\partial \psi}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\omega + \frac{1}{\mathrm{Ro}}\sin(\pi y) + \nu_e \nabla^2 \omega, \tag{2.22}$$

where ν_e refers to the eddy viscosity. To account for the effects of the truncated modes, following the similar Galerkin projection approach we may obtain a regularized ROM model

$$\frac{d\alpha_k}{dt} = \mathfrak{B}_k + \tilde{\mathfrak{B}}_k + \sum_{i=1}^{R}(\mathfrak{L}_k^i + \tilde{\mathfrak{L}}_k^i)\alpha_i + \sum_{i=1}^{R}\sum_{j=1}^{R}\mathfrak{N}_k^{ij}\alpha_i\alpha_j, \tag{2.23}$$

where the additional two terms can be written as

$$\begin{aligned} \tilde{\mathfrak{B}}_k &= \langle \nu_e \nabla^2 \bar{\omega}; \phi_k \rangle, \\ \tilde{\mathfrak{L}}_k^i &= \langle \nu_e \nabla^2 \phi_i; \phi_k \rangle. \end{aligned} \tag{2.24}$$

The free stabilization parameter ν_e may be simply considered as a given empirical constant (Aubry et al., 1988; Wang et al., 2012). This empirical eddy viscosity idea may be improved by supposing that the amount of dissipation is not identical for all the POD modes (Rempfer, 1991; Cazemier, 1997; San and Iliescu, 2014). It has been, however, shown that finding an optimal value for this parameter significantly improves

the predictive performance of ROMs (San and Iliescu, 2014, 2015). Therefore, the main novelty of the present study is the derivation of an automated approach to estimate this ν_e parameter dynamically at each time step (online computing). An alternative dynamic determination of ν_e has been presented by San and Maulik using a supervised neural network approach (San and Maulik, 2018a). However, our effort in this paper aims at developing a mathematical model based on the idea of the "test truncation", translating the idea of the test filtering approach (Germano et al., 1991) in dynamic LES subgrid-scale models into the ROM setting.

To demonstrate our approach, we first utilize the test truncation to Eq. (2.23) considering less number of modes $\tilde{R}$ (i.e., $\tilde{R} < R$)

$$\frac{d\tilde{\alpha}_k}{dt} = \mathfrak{B}_k + \tilde{\mathfrak{B}}_k + \sum_{i=1}^{\tilde{R}} (\mathfrak{L}_k^i + \tilde{\mathfrak{L}}_k^i)\tilde{\alpha}_i + \sum_{i=1}^{\tilde{R}}\sum_{j=1}^{\tilde{R}} \mathfrak{N}_k^{ij}\tilde{\alpha}_i\tilde{\alpha}_j, \tag{2.25}$$

where the $\tilde{\alpha}_k$ refers to our approximation for k^{th} time-dependent coefficient on a test truncated space and we subtract the resulting model given by Eq. (2.25) from the original model given by Eq. (2.23) to yield the difference equation at each k^{th} mode

$$\begin{aligned} &\sum_{i=1}^{R} (\mathfrak{L}_k^i + \tilde{\mathfrak{L}}_k^i)\alpha_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \mathfrak{N}_k^{ij}\alpha_i\alpha_j \\ &- \sum_{i=1}^{\tilde{R}} (\mathfrak{L}_k^i + \tilde{\mathfrak{L}}_k^i)\tilde{\alpha}_i - \sum_{i=1}^{\tilde{R}}\sum_{j=1}^{\tilde{R}} \mathfrak{N}_k^{ij}\tilde{\alpha}_i\tilde{\alpha}_j = \frac{d\alpha_k}{dt} - \frac{d\tilde{\alpha}_k}{dt}, \end{aligned} \tag{2.26}$$

and we can approximate Eq. (2.26) by the modal scale similarity hypothesis

$$\begin{aligned} &\sum_{i=1}^{R} \mathfrak{L}_k^i\alpha_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \mathfrak{N}_k^{ij}\alpha_i\alpha_j - \sum_{i=1}^{\tilde{R}} \mathfrak{L}_k^i\alpha_i - \sum_{i=1}^{\tilde{R}}\sum_{j=1}^{\tilde{R}} \mathfrak{N}_k^{ij}\alpha_i\alpha_j \\ &= -\sum_{i=1}^{R} \tilde{\mathfrak{L}}_k^i\alpha_i + \sum_{i=1}^{\tilde{R}} \tilde{\mathfrak{L}}_k^i\alpha_i, \end{aligned} \tag{2.27}$$

where we assume that $\tilde{\alpha}_k \approx \alpha_k$ when we use the eddy viscosity closure. Therefore,

using the definitions given by Eq. (2.24), we can rewrite Eq. (2.27) as

$$H_k = \nu_e M_k, \tag{2.28}$$

where

$$\begin{aligned} H_k &= \sum_{i=\tilde{R}+1}^{R} \mathfrak{L}_k^i \alpha_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \mathfrak{N}_k^{ij} \alpha_i \alpha_j - \sum_{i=1}^{\tilde{R}}\sum_{j=1}^{\tilde{R}} \mathfrak{N}_k^{ij} \alpha_i \alpha_j \\ M_k &= -\sum_{i=\tilde{R}+1}^{R} \tilde{L}_k^i \alpha_i, \end{aligned} \tag{2.29}$$

in which the predetermined coefficients of $\tilde{L}_k^i$ can be given (offline computing)

$$\tilde{L}_k^i = \langle \nabla^2 \phi_i ; \phi_k \rangle, \tag{2.30}$$

where we assume that ν_e is treated as constant locally in Eq. (2.24), i.e., frozen eddy viscosity field hypothesis expressed by $\tilde{\mathfrak{L}}_k^i = \nu_e \tilde{L}_k^i$. Similar to the approach driven by Lilly (Lilly, 1992) for LES, we propose a least-squares based estimation for ν_e in Eq. (2.28) where we define the error at each mode, $E_k = H_k - \nu_e M_k$. Once we square this error term

$$E_k^2 = H_k^2 - 2\nu_e H_k M_k + \nu_e^2 M_k^2, \tag{2.31}$$

then the eddy viscosity coefficient in our ROM model can be computed by minimizing the sum of square errors with respect to the free model parameter ν_e to obtain

$$\frac{\partial(\sum_{k=1}^{\tilde{R}} E_k^2)}{\partial(\nu_e)} = -2\sum_{k=1}^{\tilde{R}} H_k M_k + 2\nu_e \sum_{k=1}^{\tilde{R}} M_k M_k. \tag{2.32}$$

The right-hand side of the above equation becomes zero when the error is minimized

to give us finally the following expression for the eddy viscosity coefficient

$$\nu_e = \frac{\sum_{k=1}^{\tilde{R}} H_k M_k}{\sum_{k=1}^{\tilde{R}} M_k M_k}, \tag{2.33}$$

where H_k and M_k are computed by Eq. (2.29) at each time step (online computing). To provide always a positive eddy viscosity in our ROM simulations, we have also applied the following clipping rule (Sagaut, 2006)

$$\nu_e = \max\Big(0, \frac{\sum_{k=1}^{\tilde{R}} H_k M_k}{\sum_{k=1}^{\tilde{R}} M_k M_k}\Big). \tag{2.34}$$

This completes the derivation of our dynamic closure model for ROM settings. We denote our proposed model as ROM-D (i.e., Eq. (2.23) equipped with Eq. (2.34)). We note that our dynamic closure framework can be applied to modal eddy viscosities (e.g., see Östh et al. (Östh et al., 2014) on the need for a nonlinear subscale turbulence term in POD models) by replacing Eq. (2.32) with the modal pendant, a topic we will investigate further in future studies.

2.5 Numerical Experiments

In this section, we primarily focus on the comparative performance of the standard ROM-G model and our proposed ROM-D model in estimating the flow behavior at different flow conditions. In addition, we also assess the robustness and prediction capability of the ROM-D model through the extrapolatory prediction and sensitivity tests. To produce the ideal data set and simulation results for our evaluations, we select the single-layer QG model as our benchmark test case which has been appeared in numerous studies as test problem (Cushman-Roisin and Beckers, 2011; Holm and Nadiga, 2003; San et al., 2011; Cummins, 1992; Greatbatch and Nadiga, 2000). Indeed, the QG test problem comes with a great challenge of capturing wide range of scales and complex flow behavior on coarse spatial resolutions, for example, resolving the

four-gyre circulation (in the time mean) (Greatbatch and Nadiga, 2000), which makes this problem a suitable test bed to evaluate the capability of ROMs.

In the current study, we present our performance evaluations of the aforementioned reduced order methodologies based upon four distinct numerical experiments. In the first experiment, we investigate the predictive performance of both modeling frameworks at lower (Re, Ro) combination using the data snapshots extracted from a 256×512 resolution FOM simulation at the same flow condition. Next, we perform the similar analysis for comparatively higher (Re, Ro) combination in the second experiment. Finally, in the last two experiments, we test the extrapolatory predictive performance of the models using the data snapshots collected at higher/lower (Re, Ro) combination to predict a flow field at lower/higher (Re, Ro) combination, respectively. Throughout the analyses, we utilize the true projection of the FOM simulations as the baseline or reference for all the relative comparisons between ROMs.

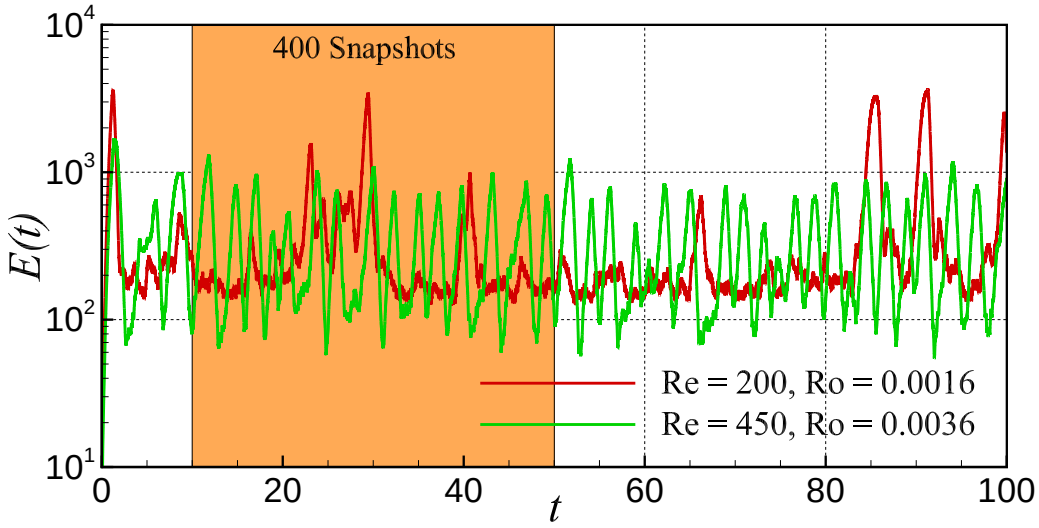


Figure 2.2: Graphical representation of the snapshots selection (shaded in orange) from the time histories of total kinetic energy for various (Re, Ro) combinations.

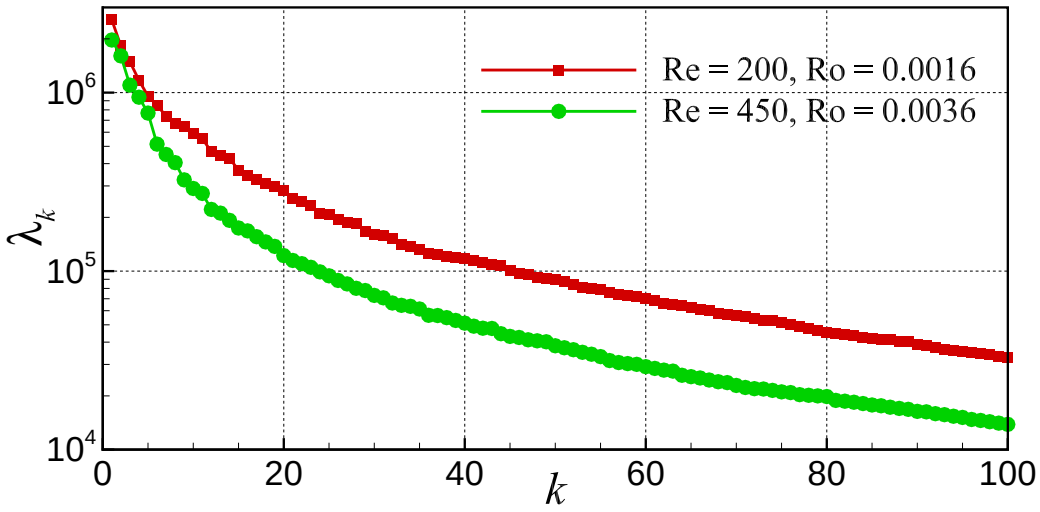


Figure 2.3: The decay of the POD eigenvalues. Note that only the first 100 largest eigenvalues (out of 400) are shown.

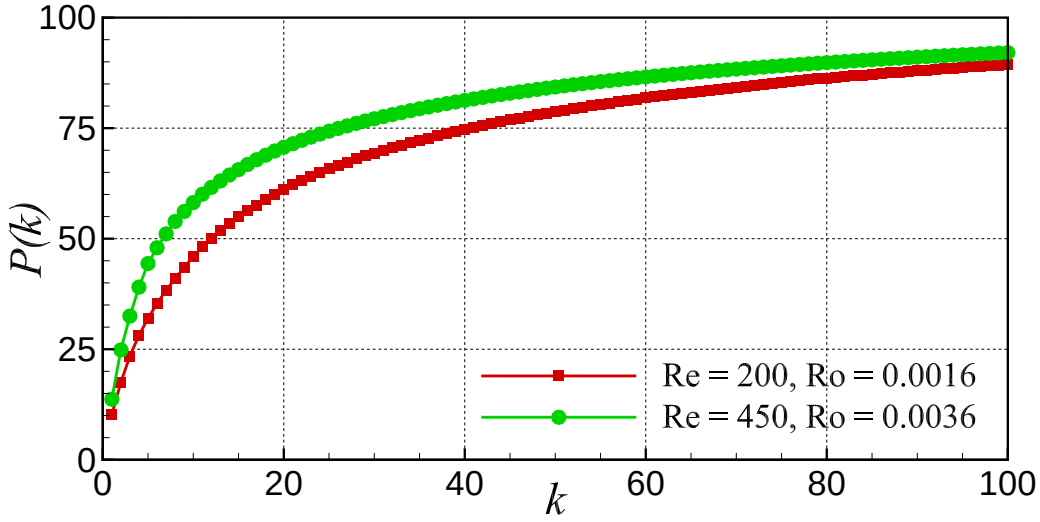


Figure 2.4: The relative information content showing the POD percentage energy accumulation with respect to modal index.

2.5.1 FOM simulation and data snapshots collection

The computational domain of our test problem is $(x, y) \in [0, 1] \times [-1, 1]$. The FOM simulation is conducted starting from $t = 0$ to $t = 100$ using a fixed time step of $\Delta t = 2.5 \times 10^{-5}$ on a Munk layer resolving 256×512 grid resolution (i.e., consisting of about four grid points in the Munk scale, i.e., $\delta_M/L = 0.02$). In FIG. 2.2, we present the plots of time histories of the basin integrated total kinetic energy for both (Re, Ro) combinations, which can be calculated by:

$$E(t) = \frac{1}{2} \int_{\Omega} \left(\left(\frac{\partial \psi}{\partial x} \right)^2 + \left(\frac{\partial \psi}{\partial y} \right)^2 \right) dxdy. \tag{2.35}$$

In general, the time evolution of the kinetic energy for QG model shows an initial short transient interval followed by the statistically steady state. As we can see from the time series plots, both (Re, Ro) combinations show a similar trend with the higher (Re, Ro) combination showing comparatively steady state fluctuations (with larger amplitude) and the lower (Re, Ro) combination showing comparatively more unsteady fluctuations. As shown in FIG. 2.2, we store 400 snapshots from $t = 10$ to $t = 50$ to collect data snapshots at statistically steady state after the initial transient period.

To get an idea of the energy captured by the POD modes for different flow conditions, we plot the decay of the POD eigenvalues in FIG. 2.3. We also present the percentage of energy accumulation with respect to the POD modes in FIG. 2.4. Here, we compute the percentage modal energy using the relative information content formula given by (Gunzburger, 2012):

$$P(k) = \left(\frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{N} \lambda_j} \right) \times 100, \tag{2.36}$$

where the number of snapshots is set to $N = 400$ in our study. It is apparent in the percentage modal energy plot that 50 POD modes capture around 80% of total energy of the system for lower (Re, Ro) combination whereas capture around 85%

of total energy of the system for higher (Re, Ro) combination. Since we have seen a comparatively more steady fluctuations in the time series plot of the total energy for higher (Re, Ro) combination, it is expected to accumulate more energy in less POD modes in higher (Re, Ro) combination case. Surely, if we increase the number of modes, the captured percentage modal energy will increase for both cases. Regardless of that, we represent most of our simulation results up to 80 modes in this study. FIG. 2.5 shows the instantaneous vorticity fields for Experiment I and II, and gives a visualization of the vigorous eddying nature of this test problem. The mixing between outer and inner-gyre in the instantaneous fields demonstrates that the flow is in the turbulent state and a strong distortion of the vorticity contours can be noticed in both experiments. As outlined in Table 2.1, we also include two out-of-sample experiments to assess the performance of the proposed method.

Table 2.1: Numerical experiments for various (Re, Ro) combinations.

Experiment	Data collection (Re, Ro)	Prediction (Re, Ro)
I	(200, 0.0016)	(200, 0.0016)
II	(450, 0.0036)	(450, 0.0036)
III	(450, 0.0036)	(200, 0.0016)
IV	(200, 0.0016)	(450, 0.0036)

2.5.2 Experiment I: Both data collection and prediction at Re = 200, Ro = 0.0016

In Experiment I, we collect our training snapshots data from a 256×512 resolution FOM simulation at Re = 200 and Ro = 0.0016. FIG. 2.6 shows the streamfunction contours, averaged in time, obtained by FOM, ROM-G (with R = 10, 20, 30, 40 and 50) and ROM-D (with R = 10 and 20), respectively. Although the dynamics of instantaneous flows is chaotic and fluctuates in time, the time-averaged contour plot shows the four-gyre circulation pattern for FOM simulation as displayed in

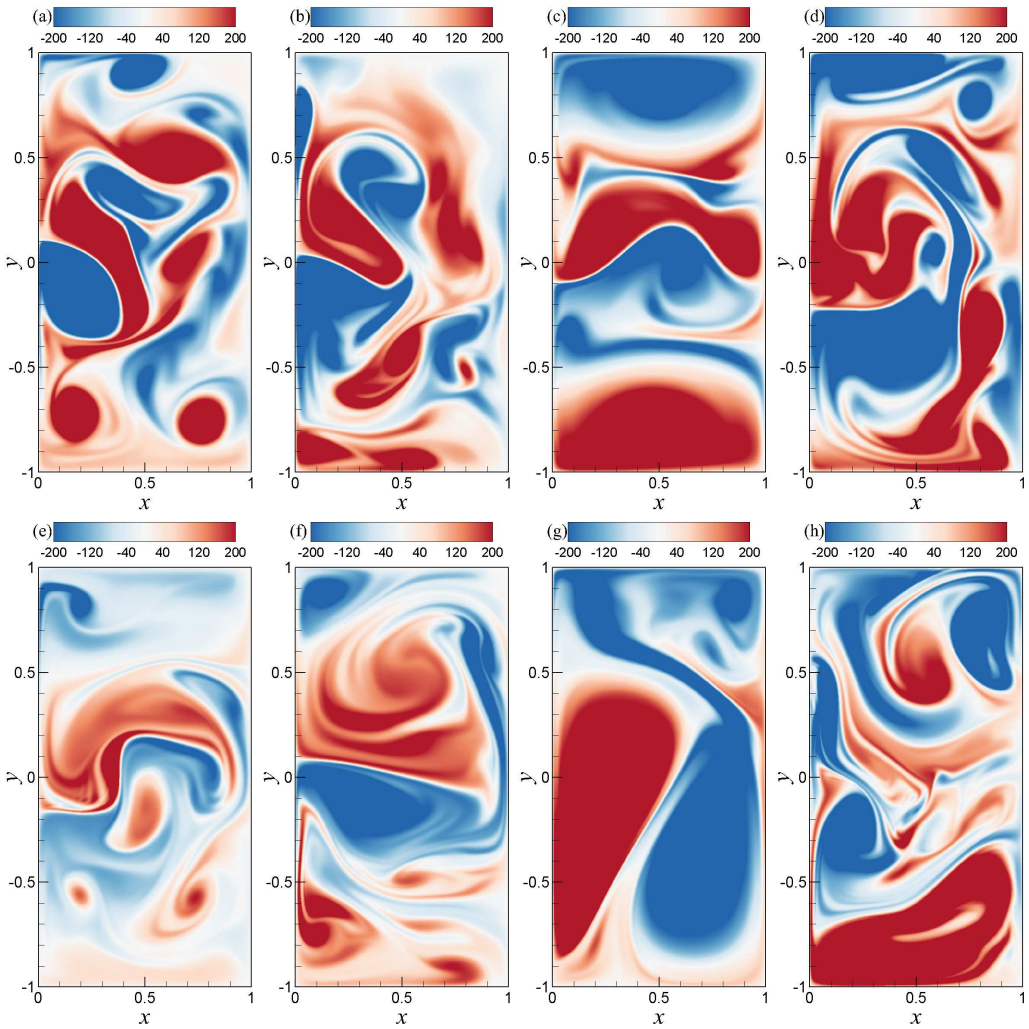


Figure 2.5: Some snapshots from the training data set showing the instantaneous vorticity fields. (a) Experiment I ($t = 10$); (b) Experiment I ($t = 20$); (c) Experiment I ($t = 30$); (d) Experiment I ($t = 40$); (e) Experiment II ($t = 10$); (f) Experiment II ($t = 20$); (g) Experiment II ($t = 30$); (h) Experiment II ($t = 40$).

FIG. 2.6. This four-gyre pattern indicates that the model reaches a state of turbulent equilibrium where the two inner gyres circulation are similar as the the wind stress curl forcing while the outer gyres correspond to the northern and southern gyres found in geostrophic turbulence experiments (i.e., time-averaged field data is obtained by averaging between $t = 10$ and $t = 100$). The circulations of the two gyres in both inner and outer regions can be seen in the figure where the red color represents the circulation in positive direction (i.e., counter-clock wise) and the blue color indicates the circulation in opposite direction.

Since we conduct the experiment for this test problem in highly turbulent regime, i.e., turbulence with weak dissipation, the four-gyre circulation pattern is expected when the flow reaches statistically steady state and the simulation result is consistent with the existing literature (Greatbatch and Nadiga, 2000). The physical interpretation of this four-gyre circulation can be the wind stress curl represented by the two inner-gyres is equilibriated by the two outer-gyres driven by the eddy flux of potential vorticity. In FIG. 2.6, we can also observe the nonphysical flow prediction in the contour plots for ROM-G solutions with lower R (Note the range in the legend for $R = 30$, $R = 20$ and $R = 10$). For $R = 40$, the ROM-G gives a good prediction of the FOM solution and the $R = 50$ result is comparatively even better. The proposed ROM-D solutions, however, show a better estimation of the FOM solution for only $R = 10$ and $R = 20$ than the ROM-G solution with $R = 50$. We can clearly visualize the presence of four-gyre circulation in reasonable range for only $R = 20$ which indicates the balance between the wind stress curl and the divergence of the eddy potential vorticity flux. A point should be noted here that we utilize $\Delta t = 2.5 \times 10^{-4}$ for all of our ROM-G and ROM-D computations since ROMs are free of the stability constraint (even though time step is set to $\Delta t = 2.5 \times 10^{-5}$ for the FOM simulation).

In another analysis, we plot the time series evolution of the first modal coefficient for Experiment I in FIG. 2.7 which show the true data projection of the FOM simulation

in each row, the ROM-G solutions with $R = 10$, $R = 20$, $R = 30$, $R = 40$ and $R = 50$ in consequent rows starting from the first row, and the ROM-D solution with $R = 10$ and $R = 20$ in the last two rows, respectively. We extend the time series plot to 100 to show the range of prediction capability of different models. In the plots, the orange colored part of the true projection data represents the training zone, the black colored part represents the extended zone, and the blue colored line in each figure indicates the solution obtained by the ROMs. Similar to the findings from mean contour plots, we can observe that the ROM-G model with $R = 50$ produces the result closest to the FOM solution although a clear gap is present between the true projection and ROM-G solutions for all the modes up to $R = 50$. In contrast, the ROM-D solutions for $R = 10$ and $R = 20$, undoubtedly, show better performance than ROM-G solutions and an almost overlap can be noticed between ROM-D with $R = 20$ and the true projection. Similar statistical observations can be seen for other modal coefficients (not shown here due to clarity and space limitation reasons). To investigate the robustness of the proposed ROM-D model, we perform a sensitivity test with respect to the dynamic model parameter, $\tilde{R}$ for different test truncation, ΔR values. In our ROM-D framework formulation, we define $\Delta R = R - \tilde{R}$ as a modeling parameter analogous to the test filter strength in LES. We must mention here that we use $\Delta R = 3$ value in all of our experiments (except the sensitivity analyses where we vary the ΔR value) in this study.

For sensitivity analysis, we first present the time series of the first modal coefficient for Experiment I with different (ΔR, R) combinations of the proposed ROM-D model in FIG. 2.8. As we can see, the results do not vary much statistically for any R with different ΔR combinations. However, it seems that there are some correlations between ΔR and the amplitude of the fluctuations for ROM-D with $R = 10$. On the other hand, the time series of ROM-G with $R = 20$ on the bottom last row reveals that the both $R = 10$ and $R = 20$ combined with different ΔR values ROM-D models

exhibit a better estimation of the true projection than the ROM-G solution. Since we take into account the truncated modes in ROM-D models, the results improve significantly with respect to the ROM-G model with lower R.

We further illustrate the sensitivity analysis based on the mean streamfunction contour plots in FIG. 2.9 which clearly show that the $R = 20$ for different ΔR give a very promising prediction of the FOM solution. Furthermore, results do not vary much qualitatively with respect to different ΔR. These results indicate the ROM-D model for $R = 20$ is robust in predicting the true solution for this experiment. Moreover, we provided our quantitative assessments of the ROM-G and ROM-D models for Experiment I in Table 2.2 showing the computational overhead and L_2-norm error for the mean streamfunction field. Although the dynamic methodology requires additional computational overhead, it can be easily seen that this computational overhead will be always small (i.e., CPU time of ROM-D $<$ 2 times CPU time of ROM-G), since $\tilde{R}$ is less than R, and we formulate our dynamic approach at the reduced order space using the time dependant coefficients without a cost related to field reconstruction. Furthermore, it is evident that we can achieve more accuracy for ROM-G model with the increment of R and computational time (for example, ROM-G with $R = 80$ gives 1.18×10^{-1} of accuracy in 1741.60 seconds). However, the similar order of accuracy can be obtained by the ROM-D with $R = 10$ and $\Delta R = 4$ in around 224 times speedup. Additionally, ΔR in ROM-D model gives us a freedom (combining with higher R) to increase the accuracy of the solution with a little increase in computational time.

2.5.3 Experiment II: Both data collection and prediction at Re = 450, Ro = 0.0036

For Experiment II, we carry out similar analyses as Experiment I using higher (Re, Ro) combination. Since the flow is in highly turbulent regime for this experiment, we expect to observe the four-gyre circulation again in the mean field contours because

Table 2.2: Quantitative assessments for Experiment I demonstrating the CPU time in seconds for ROM simulations (using computational time step $\Delta t = 2.5 \times 10^{-4}$), and L_2-norm error for the mean streamfunction field (with respect to FOM). Note that the CPU time for the FOM simulation is about 135 hours (between $t = 0$ and $t = 100$), where computational time step is set $\Delta t = 2.5 \times 10^{-5}$ due to the CFL restriction of numerical stability for our explicit forward model on the resolution of 256×512. Offline computing time for solving the eigensystem to find POD modes is about 21 minutes (including about 8 seconds (per 10 modes) for performing numerical integration to calculate the predetermined coefficients). Note that $\Delta R = R - \tilde{R}$.

	CPU (s)	$\|\|\psi_{\text{ROM}} - \psi_{\text{FOM}}\|\|^2$
Galerkin ROM		
ROM-G ($R = 80$)	1741.60	1.18×10^{-1}
ROM-G ($R = 60$)	723.88	1.35×10^{0}
ROM-G ($R = 50$)	432.35	1.17×10^{1}
ROM-G ($R = 40$)	287.59	1.19×10^{1}
ROM-G ($R = 30$)	94.54	1.11×10^{2}
ROM-G ($R = 20$)	29.71	3.74×10^{2}
ROM-G ($R = 10$)	4.43	1.07×10^{6}
Dynamic ROM		
ROM-D ($R = 10$, $\Delta R = 4$)	7.75	2.39×10^{-1}
ROM-D ($R = 10$, $\Delta R = 3$)	8.58	2.67×10^{-1}
ROM-D ($R = 10$, $\Delta R = 2$)	9.75	1.69×10^{0}
ROM-D ($R = 20$, $\Delta R = 4$)	68.45	5.25×10^{-2}
ROM-D ($R = 20$, $\Delta R = 3$)	72.72	3.15×10^{-2}
ROM-D ($R = 20$, $\Delta R = 2$)	77.26	7.72×10^{-2}

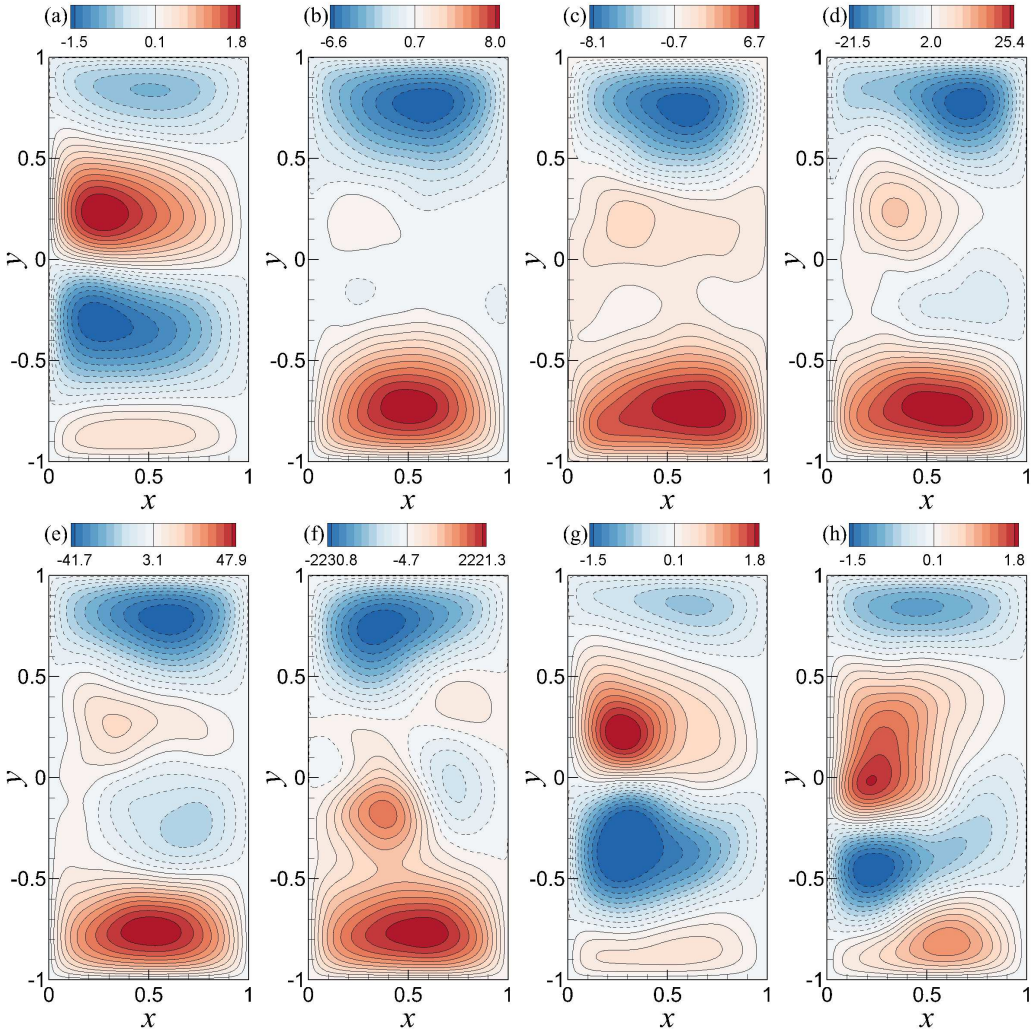


Figure 2.6: Mean streamfunction contours for Experiment I (for training snapshots between $t = 10$ and $t = 50$). (a) FOM at a resolution of 256×512; (b) ROM-G with $R = 50$ modes; (c) ROM-G with $R = 40$ modes; (d) ROM-G with $R = 30$ modes; (e) ROM-G with $R = 20$ modes; (f) ROM-G with $R = 10$ modes; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$; (h) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$.

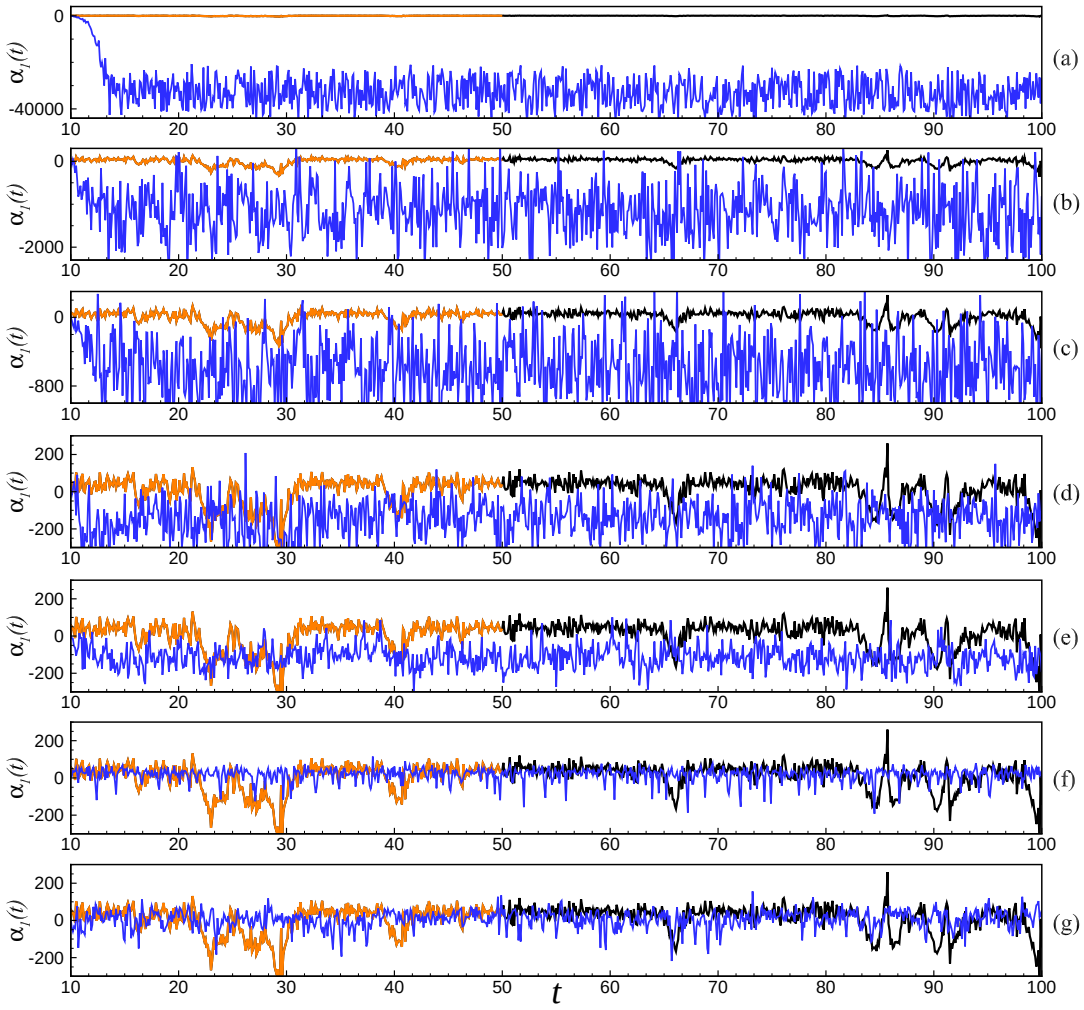


Figure 2.7: Time series of the first modal coefficient for Experiment I. (a) ROM-G with $R = 10$ modes; (b) ROM-G with $R = 20$ modes; (c) ROM-G with $R = 30$ modes; (d) ROM-G with $R = 40$ modes; (e) ROM-G with $R = 50$ modes; (f) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$. True projection data is underlined in each figure with orange (training zone) and black (extended zone).

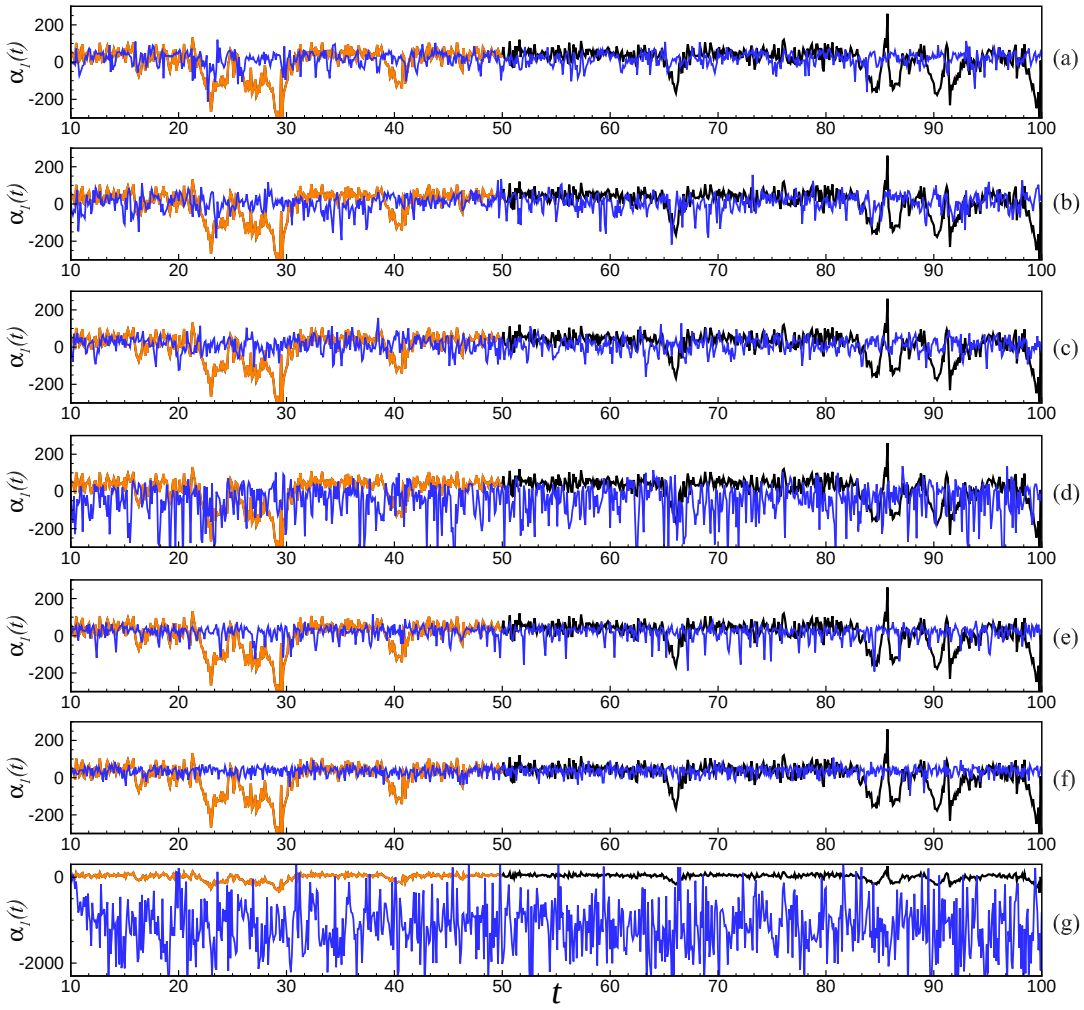


Figure 2.8: A sensitivity test with respect to the dynamic model parameter $\tilde{R}$ showing the time series of the first modal coefficient for Experiment I. (a) ROM-D ($R = 20$) with $\Delta R = 2$; (b) ROM-D ($R = 20$) with $\Delta R = 3$; (c) ROM-D ($R = 20$) with $\Delta R = 4$; (d) ROM-D ($R = 10$) with $\Delta R = 2$; (e) ROM-D ($R = 10$) with $\Delta R = 3$; (f) ROM-D ($R = 10$) with $\Delta R = 4$; (g) ROM-G with $R = 20$ modes. Note that $\Delta R = R - \tilde{R}$. True projection data is presented in each figure with orange (training zone) and black (extended zone).

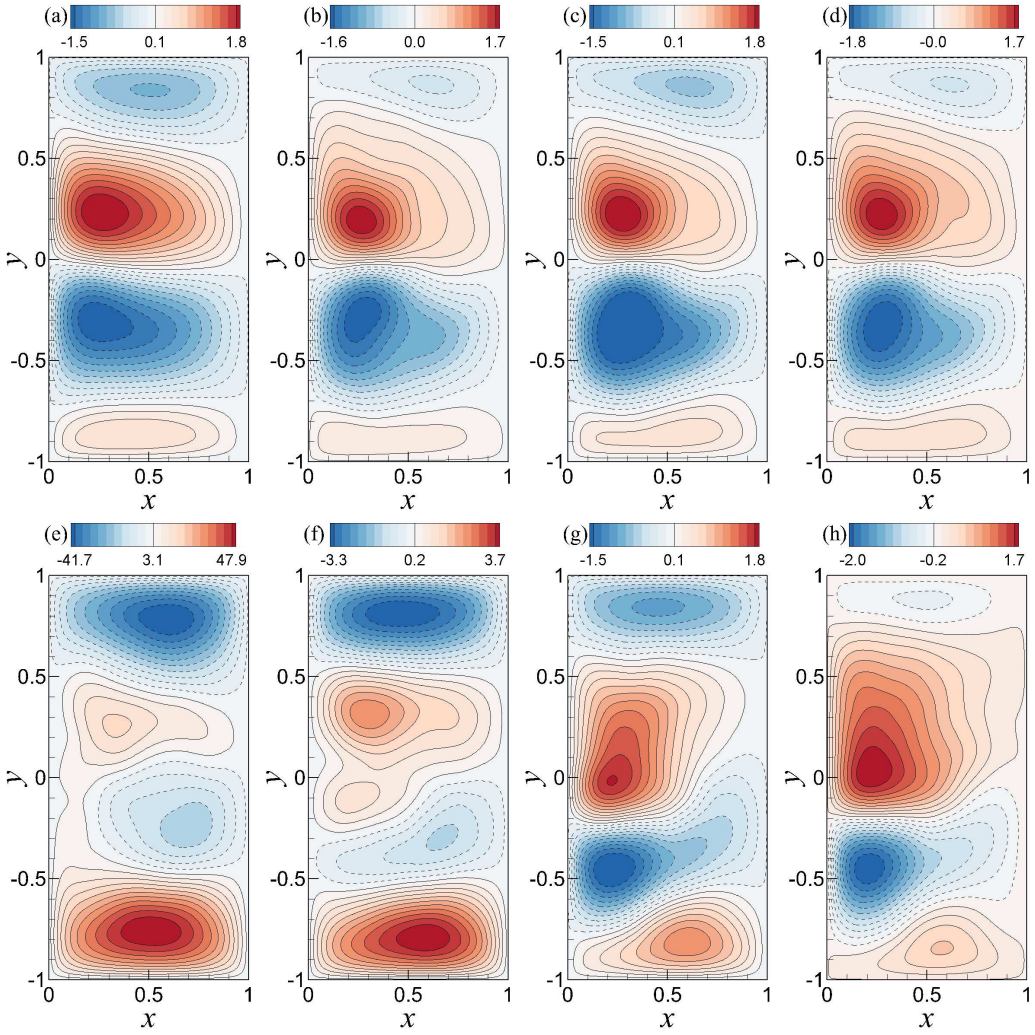


Figure 2.9: A sensitivity test with respect to the dynamic model parameter $\tilde{R}$ showing the mean streamfunction contours for Experiment I. (a) FOM at a resolution of 256×512; (b) ROM-D ($R = 20$) with $\Delta R = 2$; (c) ROM-D ($R = 20$) with $\Delta R = 3$; (d) ROM-D ($R = 20$) with $\Delta R = 4$; (e) ROM-G with $R = 20$ modes; (f) ROM-D ($R = 10$) with $\Delta R = 2$; (g) ROM-D ($R = 10$) with $\Delta R = 3$; (h) ROM-D ($R = 10$) with $\Delta R = 4$. Note that $\Delta R = R - \tilde{R}$.

the eddy flux of potential vorticity balances the vorticity input from the wind stress forcing. Based on the visualizations of the mean streamfunction contour plots in FIG. 2.10, Experiment II reveals that the ROM-G gives a physical result from $R = 30$ although $R = 20$ and $R = 10$ show a nonphysical two-gyre circulation instead of four-gyre circulation (i.e., due to the instability of the POD-G approach with such number of retained modes). In contrast, the ROM-D models with $R = 10$ and $R = 20$ exhibit a good estimation of the FOM solution whereas the $R = 20$ shows the visible four-gyre circulation. The time series evolution for the first modal coefficient plots in FIG. 2.11 gives us a better statistical view on the comparative performance of both models. As we can see in the first two rows of the figure that the solutions obtained by ROM-G with $R = 10$ and $R = 20$ modes become nonphysical after a while whereas the true physics suggests a statistically steady flow field after $t = 10$. At the same time range, the $R = 30$, $R = 40$ and $R = 50$ modes for ROM-G give a statistically steady and satisfactory prediction of the true solution which is consistent with the findings of the mean streamfunction contour plots. On the other hand, the ROM-D model with both $R = 10$ and $R = 20$ modes give an excellent prediction of the truth compared to the ROM-G solutions in lower modes. As we have seen in the POD analysis in FIG. 2.2, Experiment II displays a statistically more stable time series evolution (compared to Experiment I), and it is expected that comparatively more energy will be accumulated in lower R for this case.

Next, we perform the sensitivity tests for Experiment II and it can be observed in FIG. 2.12 that the ROM-D model is showing consistent predictions for even $R = 10$ invariant to the change in the value of ΔR. Nonetheless, the ROM-G with $R = 20$ prediction is nonphysical yet again after certain time. Similarly, the mean streamfunction contours in FIG. 2.13 indicates the ROM-D model is very robust for $R = 20$ and also, showing a good prediction for $R = 10$. In Table 2.3, we report the computational time and L_2-norm error of the ROM-G and ROM-D model simulation

results for Experiment II. Similar to Experiment I, we can obtain equivalent order of accuracy as ROM-G ($R = 80$) in ROM-D ($R = 10$, $\Delta R = 4$) with around 219 times reduction in overall computational time. It is also apparent that we can gain more accurate solutions for $R = 20$ with different ΔR using ROM-D model.

Table 2.3: Quantitative assessments for Experiment II demonstrating the CPU time in seconds for ROM simulations (using computational time step $\Delta t = 2.5 \times 10^{-4}$), and L_2-norm error for the mean streamfunction field (with respect to FOM). Note that the CPU time for the FOM simulation is about 130 hours (between $t = 0$ and $t = 100$), where computational time step is set $\Delta t = 2.5 \times 10^{-5}$ due to the CFL restriction of numerical stability for our explicit forward model on the resolution of 256×512. Offline computing time for solving the eigensystem to find POD modes is about 22 minutes (including about 8 seconds (per 10 modes) for performing numerical integration to calculate the predetermined coefficients). Note that $\Delta R = R - \tilde{R}$.

	CPU (s)	$\|\|\psi_{\text{ROM}} - \psi_{\text{FOM}}\|\|^2$
Galerkin ROM		
ROM-G ($R = 80$)	1752.79	3.84×10^{-1}
ROM-G ($R = 60$)	722.02	3.33×10^{-1}
ROM-G ($R = 50$)	554.14	3.63×10^{-1}
ROM-G ($R = 40$)	218.06	4.33×10^{-1}
ROM-G ($R = 30$)	96.00	9.99×10^{-1}
ROM-G ($R = 20$)	29.32	9.87×10^{2}
ROM-G ($R = 10$)	4.36	5.59×10^{3}
Dynamic ROM		
ROM-D ($R = 10$, $\Delta R = 4$)	8.00	3.55×10^{-1}
ROM-D ($R = 10$, $\Delta R = 3$)	8.96	7.82×10^{-1}
ROM-D ($R = 10$, $\Delta R = 2$)	9.51	1.51×10^{0}
ROM-D ($R = 20$, $\Delta R = 4$)	68.32	3.02×10^{-1}
ROM-D ($R = 20$, $\Delta R = 3$)	72.70	2.88×10^{-1}
ROM-D ($R = 20$, $\Delta R = 2$)	77.27	1.22×10^{-1}

2.5.4 Experiment III: Data collection at Re = 450, Ro = 0.0036, prediction at Re = 200, Ro = 0.0016

The first two experiments clearly address the improvements we can achieve through the dynamic closure ROM approach for the prediction of the flow field within the range of training data. However, to further investigate the prediction capability of the proposed ROM-D model for out-of-sample flow condition, we perform an extrapolatory

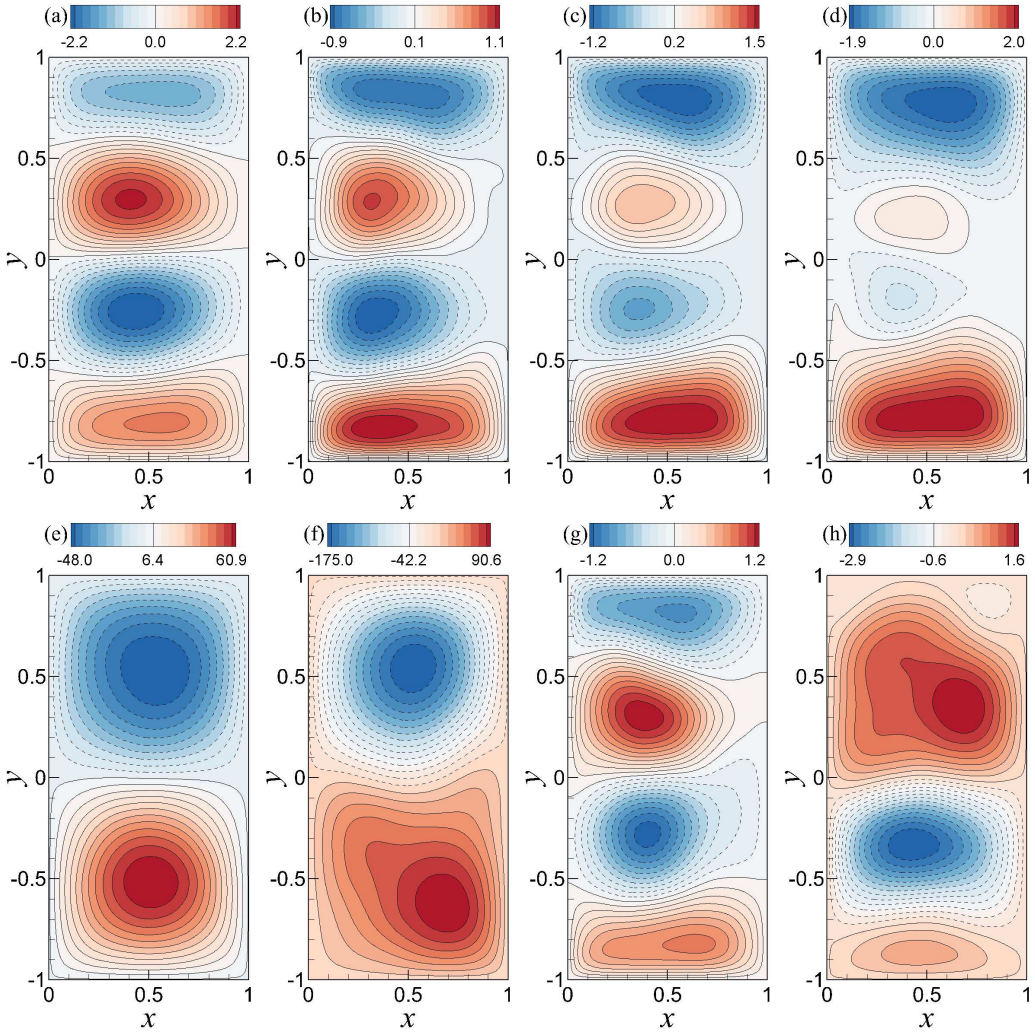


Figure 2.10: Mean streamfunction contours for Experiment II (for training snapshots between $t = 10$ and $t = 50$). (a) FOM at a resolution of 256×512; (b) ROM-G with $R = 50$ modes; (c) ROM-G with $R = 40$ modes; (d) ROM-G with $R = 30$ modes; (e) ROM-G with $R = 20$ modes; (f) ROM-G with $R = 10$ modes; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$; (h) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$.

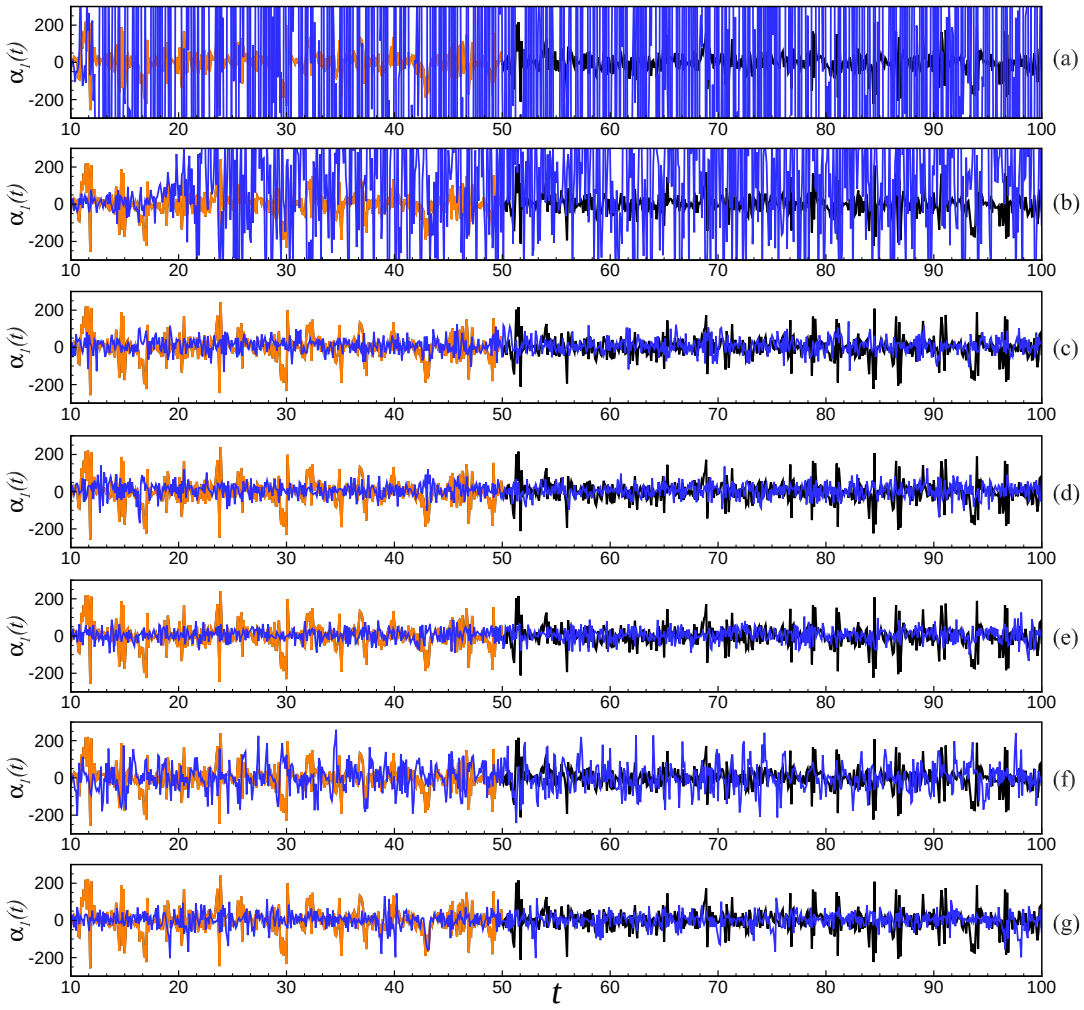


Figure 2.11: Time series of the first modal coefficient for Experiment II. (a) ROM-G with $R = 10$ modes; (b) ROM-G with $R = 20$ modes; (c) ROM-G with $R = 30$ modes; (d) ROM-G with $R = 40$ modes; (e) ROM-G with $R = 50$ modes; (f) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$. True projection data is underlined in each figure with orange (training zone) and black (extended zone).

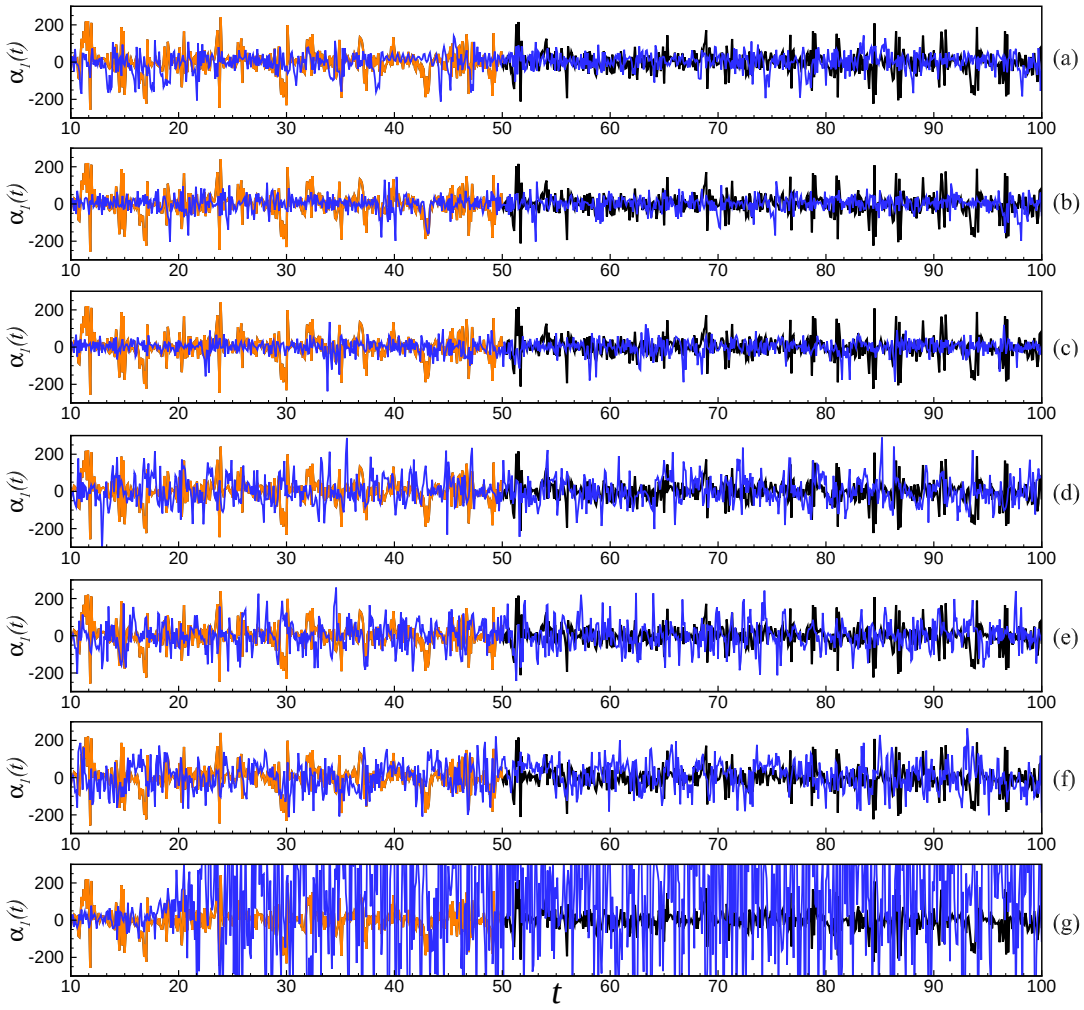


Figure 2.12: A sensitivity test with respect to the dynamic model parameter $\tilde{R}$ showing the time series of the first modal coefficient for Experiment II. (a) ROM-D ($R = 20$) with $\Delta R = 2$; (b) ROM-D ($R = 20$) with $\Delta R = 3$; (c) ROM-D ($R = 20$) with $\Delta R = 4$; (d) ROM-D ($R = 10$) with $\Delta R = 2$; (e) ROM-D ($R = 10$) with $\Delta R = 3$; (f) ROM-D ($R = 10$) with $\Delta R = 4$; (g) ROM-G with $R = 20$ modes. Note that $\Delta R = R - \tilde{R}$. True projection data is presented in each figure with orange (training zone) and black (extended zone).

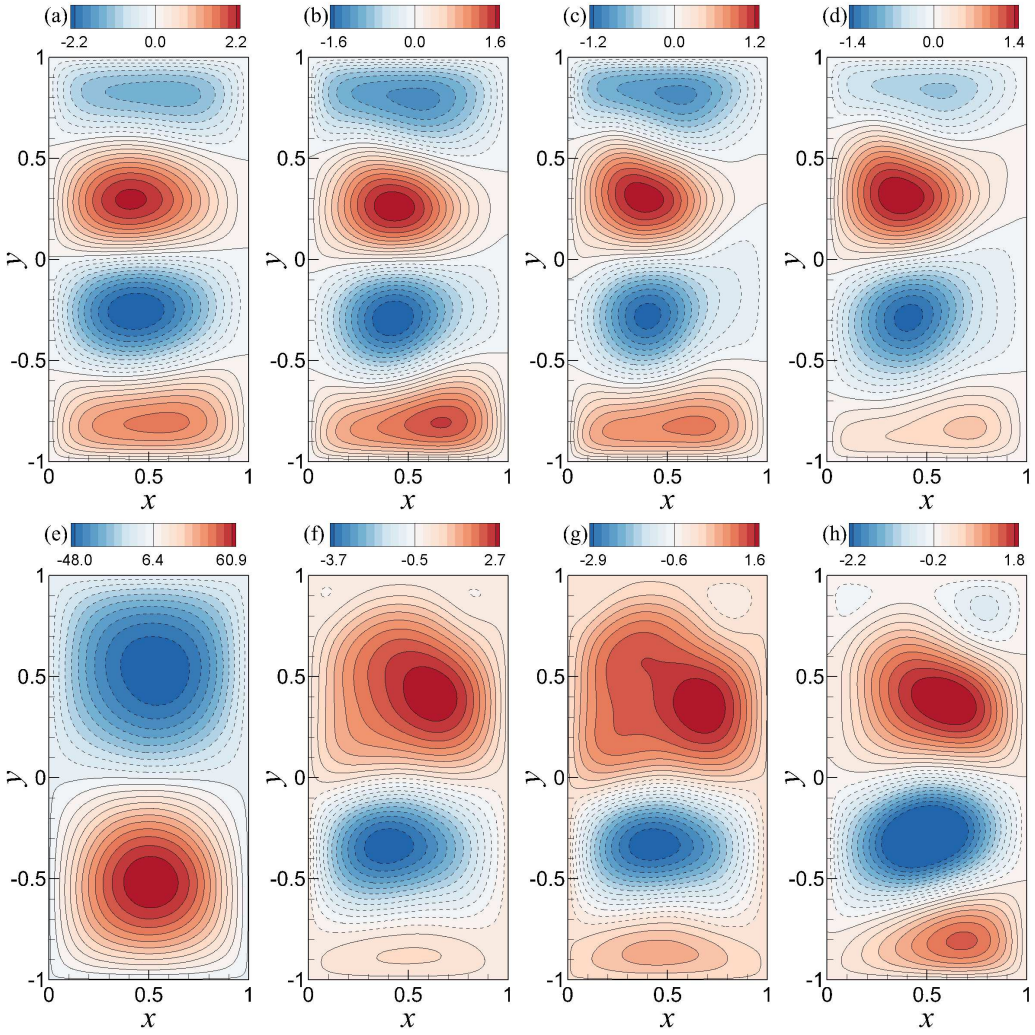


Figure 2.13: A sensitivity test with respect to the dynamic model parameter $\tilde{R}$ showing the mean streamfunction contours for Experiment II. (a) FOM at a resolution of 256×512; (b) ROM-D ($R = 20$) with $\Delta R = 2$; (c) ROM-D ($R = 20$) with $\Delta R = 3$; (d) ROM-D ($R = 20$) with $\Delta R = 4$; (e) ROM-G with $R = 20$ modes; (f) ROM-D ($R = 10$) with $\Delta R = 2$; (g) ROM-D ($R = 10$) with $\Delta R = 3$; (h) ROM-D ($R = 10$) with $\Delta R = 4$. Note that $\Delta R = R - \tilde{R}$.

predictive performance analysis where we collect the training data for POD basis function generation at higher (Re, Ro) combination, i.e., at (Re = 450, Ro = 0.0036) and then, test to predict the flow field at (Re = 200, Ro = 0.0016). Indeed, it should be challenging since we have seen the analyses in the previous sections that the (Re = 200, Ro = 0.0016) combination introduces comparatively uneven fluctuations than the higher (Re, Ro) flow condition. As a result, the lower (Re, Ro) flow condition requires higher POD modes to capture greater percentage of energy of the system.

The mean streamfunction plots for this set up is presented in FIG. 2.14 which show the FOM simulation result at (Re = 200, Ro = 0.0016) on the top left corner of the figure. In the same figure, we can also observe that the physical four-gyre pattern is captured by the standard ROM-G with $R = 50$ and $R = 40$. However, retaining the lower R number of modes might fail to capture the true physics. On the contrary, the ROM-D model shows a hint of capturing the four-gyre for $R = 20$. It is evident from the figure that the ROM-D with lower R values clearly showing a better prediction than the ROM-G with higher R values. The sensitivity of ROM-D with respect to ΔR is also shown for two values of the $R = 20$ case. Eventually, we present the time series evolution of the first modal coefficient plots for Experiment III in FIG. 2.15 to get a lucid idea of the underlying physics. It can be observed that the ROM-D with $R = 10$ exhibits the fluctuations with larger amplitude, but the $R = 20$ yields a comparatively better estimation. Also, we can see that the estimations of ROM-G models are showing a good prediction for $R = 40$ and $R = 50$. Even so, if we only compare the performance of both ROM-G and ROM-D for the same retained number of modes (e.g., $R = 20$), we can clearly identify the difference in performance.

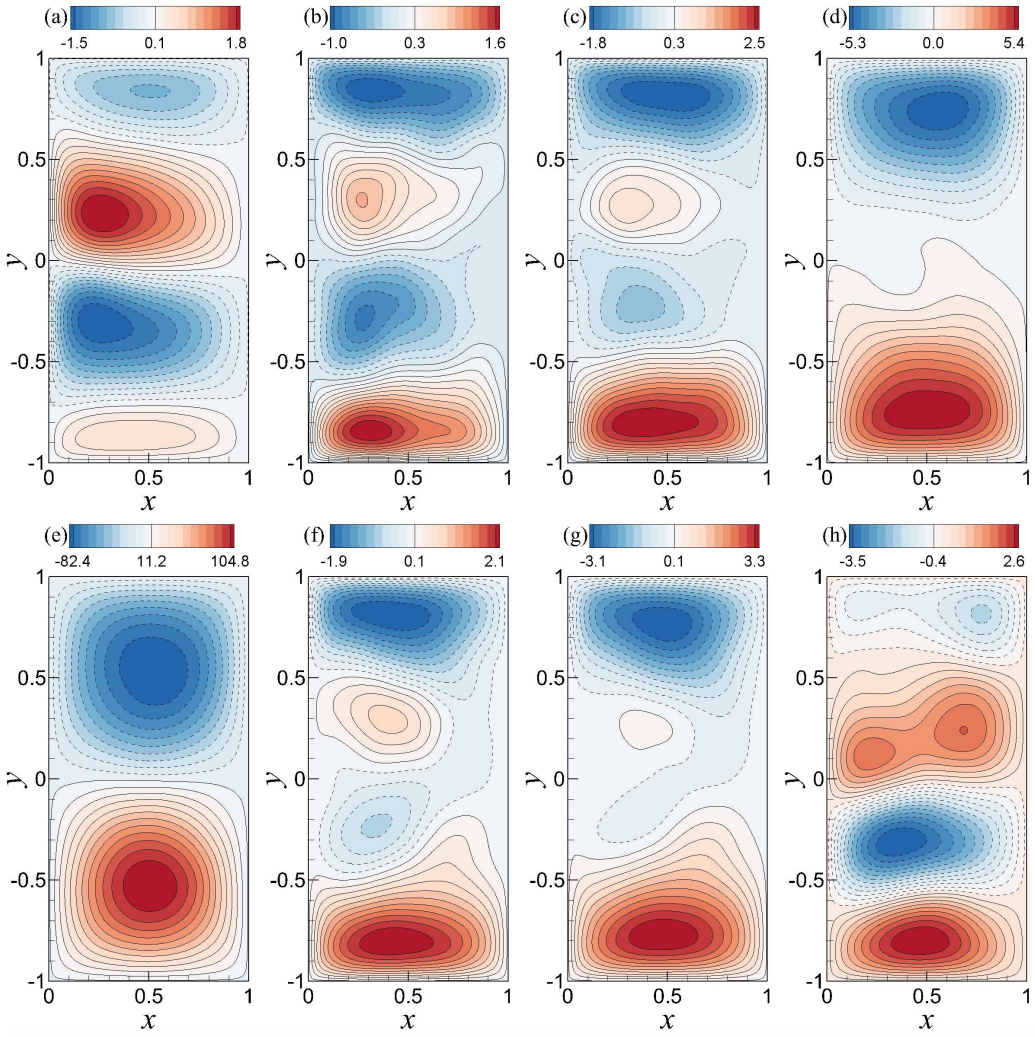


Figure 2.14: Mean streamfunction contours for Experiment III showing the extrapolatory predictive performance at Re = 200, and Ro = 0.0016 (i.e, using POD basis functions and mean fields associated with the training data obtained at Re = 450, and Ro = 0.0036). (a) FOM at a resolution of 256×512; (b) ROM-G with $R = 50$ modes; (c) ROM-G with $R = 40$ modes; (d) ROM-G with $R = 30$ modes; (e) ROM-G with $R = 20$ modes; (f) proposed ROM-D with $R = 20$ modes and $\Delta R = 4$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$; (h) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$.

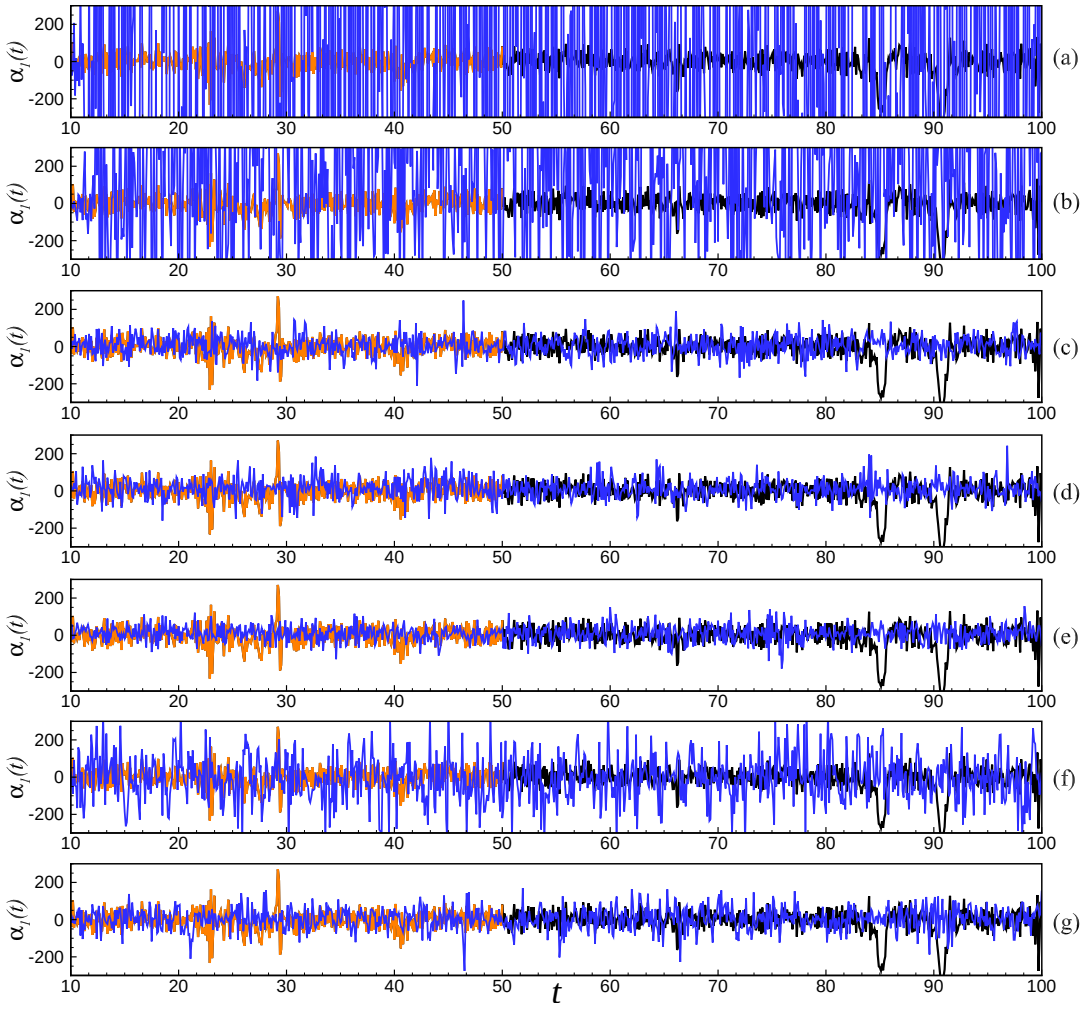


Figure 2.15: Time series of the first modal coefficient for Experiment III showing the extrapolatory predictive performance at Re = 200, and Ro = 0.0016 (i.e, using POD basis functions and mean fields associated with the training data obtained at Re = 450, and Ro = 0.0036). (a) ROM-G with $R = 10$ modes; (b) ROM-G with $R = 20$ modes; (c) ROM-G with $R = 30$ modes; (d) ROM-G with $R = 40$ modes; (e) ROM-G with $R = 50$ modes; (f) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$. True projection data is underlined in each figure with orange (training zone) and black (extended zone).

2.5.5 Experiment IV: Data collection at Re = 200, Ro = 0.0016, prediction at Re = 450, Ro = 0.0036

Our final experiment is based on the prediction at (Re = 450, Ro = 0.0036) flow configuration using the data snapshots at (Re = 200, Ro = 0.0016). Following the similar analyses to Experiment III, here we put an effort to understand the extrapolatory predictive behavior of the ROM-D model for the opposite test condition than the previous experiment. In FIG. 2.16, we can see in the mean streamfunction contours that the ROM-D model is showing a good prediction of the FOM solution for both $R = 10$ and $R = 20$ by capturing the four-gyre circulation pattern. Additionally, ROM-G with higher R also displays an accurate prediction of the true solution. The time series evolution plots in FIG. 2.17 shows that the ROM-D solutions are showing a slight phase shift from true projection states. However, the amplitude of the fluctuation for ROM-D with $R = 10$ is comparatively smaller, unlike the scenario in Experiment III, than the true projection fluctuation amplitudes. Though we observe a very good prediction of the true solution using ROM-G with $R = 50$, we can see the the ROM-D predictions for lower R values are impressive compared to the ROM-G solutions for $R = 10$, $R = 20$ and $R = 30$.

2.6 Summary and Conclusions

In this work, we put forth a dynamic closure modeling framework for reduced order models (ROM-D) based on the idea of test truncation, analogous to the test filtering in dynamic eddy viscosity model in LES, in order to stabilize the ROM for systems with complex flow dynamics. Our closure is based on a global eddy viscosity model dynamically obtained from POD coefficients at each time step, and follows the idea as dynamic subgrid model of LES with two modal resolution levels. Previously, it has been shown that the stabilization of ROM for large scale turbulent flows can be achieved using an eddy viscosity parameter with an optimal value that can improve

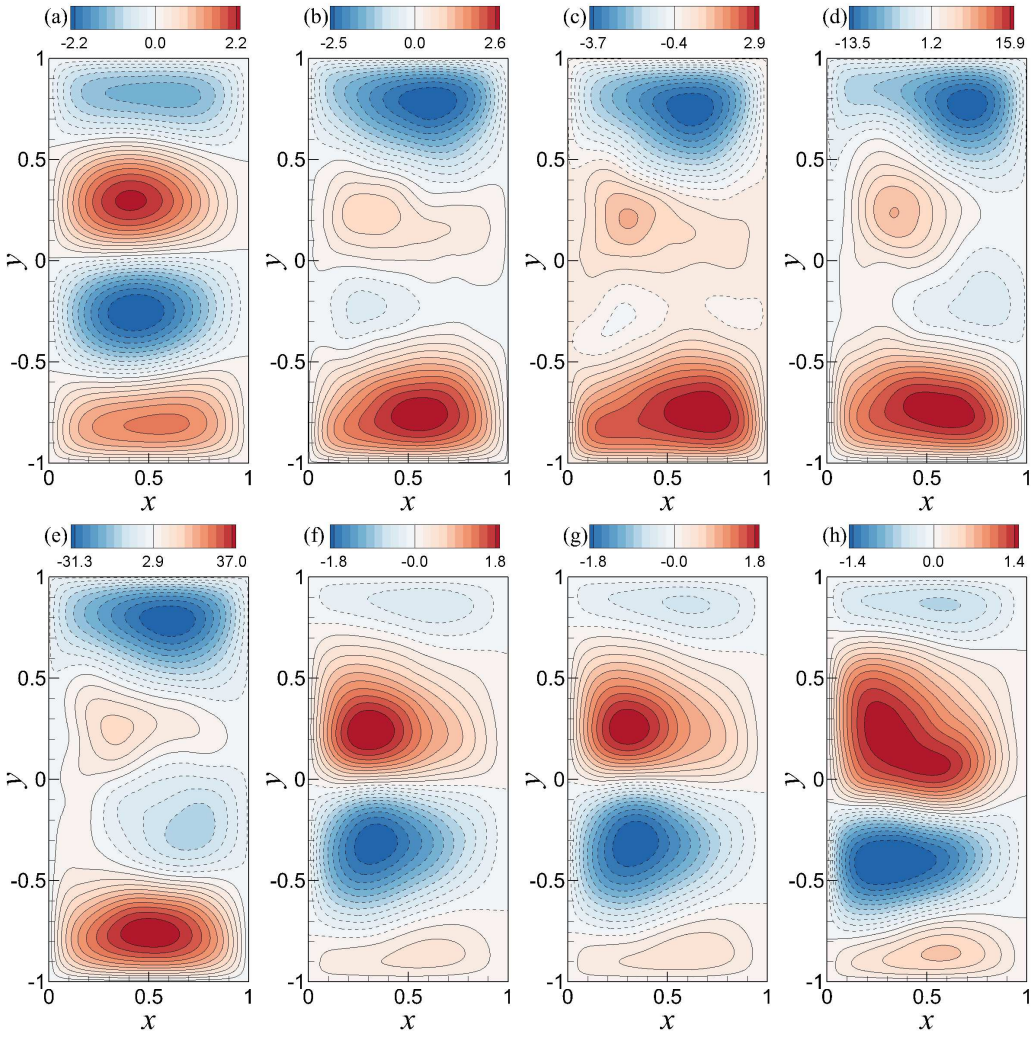


Figure 2.16: Mean streamfunction contours for Experiment IV showing the extrapolatory predictive performance at Re = 450, and Ro = 0.0036 (i.e, using POD basis functions and mean fields associated with the training data obtained at Re = 200, and Ro = 0.0016). (a) FOM at a resolution of 256×512; (b) ROM-G with $R = 50$ modes; (c) ROM-G with $R = 40$ modes; (d) ROM-G with $R = 30$ modes; (e) ROM-G with $R = 20$ modes; (f) proposed ROM-D with $R = 20$ modes and $\Delta R = 4$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$; (h) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$.

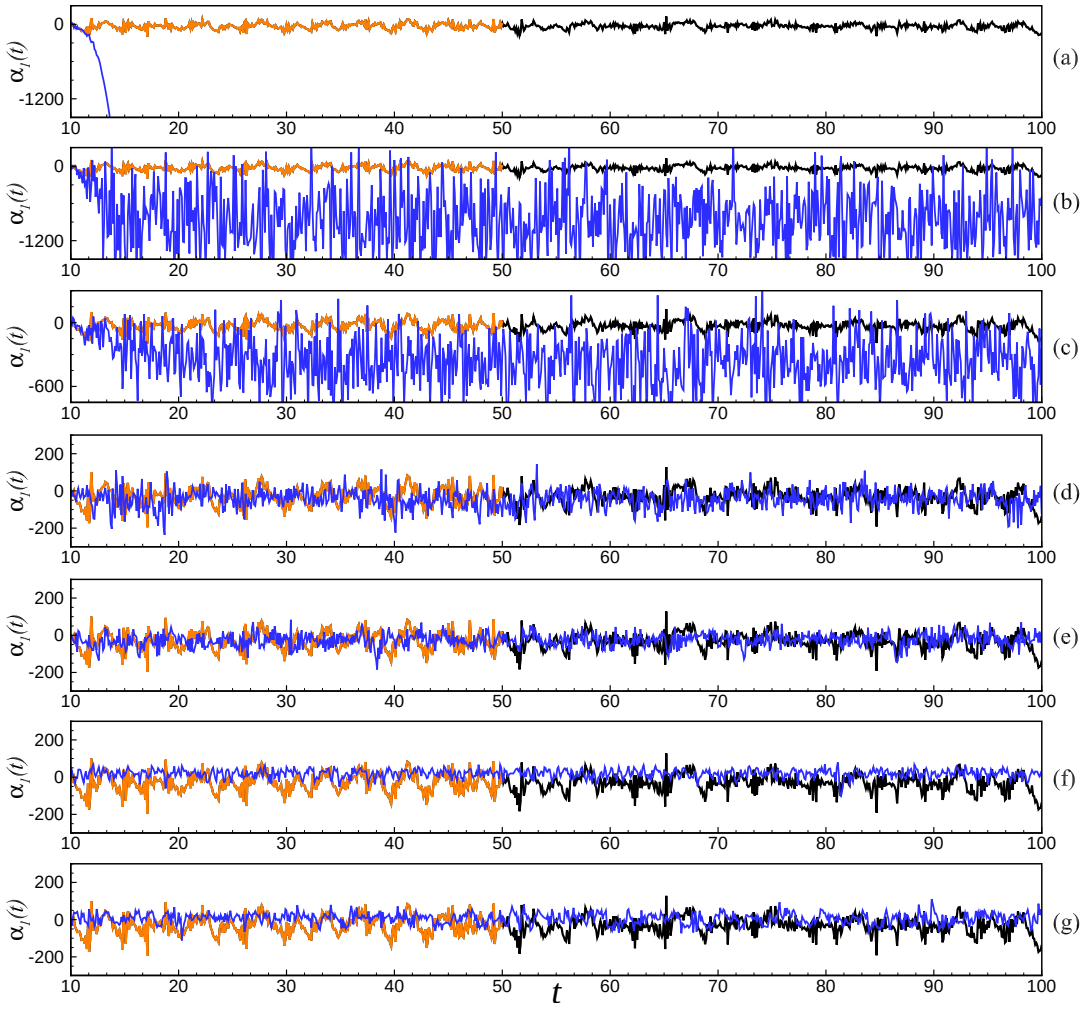


Figure 2.17: Time series of the first modal coefficient for Experiment IV showing the extrapolatory predictive performance at Re = 450, and Ro = 0.0036 (i.e, using POD basis functions and mean fields associated with the training data obtained at Re = 200, and Ro = 0.0016). (a) ROM-G with $R = 10$ modes; (b) ROM-G with $R = 20$ modes; (c) ROM-G with $R = 30$ modes; (d) ROM-G with $R = 40$ modes; (e) ROM-G with $R = 50$ modes; (f) proposed ROM-D with $R = 10$ modes and $\Delta R = 3$; (g) proposed ROM-D with $R = 20$ modes and $\Delta R = 3$. Note that $\Delta R = R - \tilde{R}$. True projection data is underlined in each figure with orange (training zone) and black (extended zone).

the predictive performance of the model significantly (San and Iliescu, 2015). Inspired by the LES and ROM analogy, we devise the proposed ROM-D framework which takes into account the stabilization parameter in an automated fashion at each time step and dynamically stabilizes the system of any flow condition without any external computation of optimal eddy viscosity. Since the test filtering is done at the projected reduced order space (i.e, test truncation), the computational overhead associated with this secondary test filtering operation is relatively minor. This ensures that the proposed ROM-D is not significantly more expensive than the standard ROM-G. Through a series of numerical experiments, we show that the predictive performance of the ROM-D framework is not only promising but also consistent for different flow conditions. All performance evaluations of the ROM-D framework are done with respect to the standard ROM-G models with various levels of complexity. As a benchmark test case for our numerical investigations, we consider the BVE describing the single-layer QG ocean model. For data snapshots collection and comparison purpose, we use the Munk layer resolving FOM simulation results obtained at 256×512 resolution.

We perform our numerical assessments based upon two different flow conditions. Based on the existing literature and our findings in FOM simulations, the BVE model driven by two-gyre wind forcing shows a four-gyre circulation pattern in time mean once the model reaches the turbulent equilibrium state under these conditions. Physically, this means the balance between the wind stress curl forcing and the eddy flux of potential vorticity. For this reason, we investigate both mean streamfunction contour and time series evolution (after the statistically steady state achieved) plots to get a clear understanding on the simulation results in capturing the four-gyre pattern or attaining the statistically steady state. Owing to the wide range of temporal and spatial scales of the QG test problem, it is observed in the POD analysis that 50 POD modes ($R = 50$) capture around 80% of total energy of the system for lower (Re, Ro) combination whereas capture around 85% of total energy of the system for higher

(Re, Ro) combination. For the experiments on both flow conditions, we come to the main conclusion that the ROM-D model with $R = 10$ predicts the true solution with a same order accuracy as a ROM-G solution with $R = 80$. This implies considerable savings in terms of both storage needs and computing time.

Since the ROM-D model includes dissipative contributions from truncated modes through the stabilization parameter, it is expected to capture a greater percentage of the energy in the system using lower R values in ROM-D model. As a result, we can see a huge reduction in overall computational overhead for equally accurate solution using ROM-D model. Moreover, the higher value of R combined with different values of ΔR in ROM-D exhibits more gain in accuracy of the predictions. To demonstrate the robustness of the ROM-D model, we also perform the sensitivity analysis on both flow conditions which display a consistent prediction for $R = 20$ irrespective to different ΔR values. Finally, we test the extrapolatory predictive performances for both ROM frameworks which reveal a better prediction of FOM solution by the ROM-D model than the ROM-G model for same value of R. Based on the numerical experiments and analyses above, the ROM-D framework has a great potential for efficient model order reduction of complex turbulent flow problems.

2.7 Appendix

2.7.1 Time advancement scheme

We apply a conservative finite difference formulation for our numerical framework in the current study. Using method of lines, we cast the governing equation given by Eq. (2.1) in the following semi-discretized ordinary differential equations form:

$$\frac{d\omega_{i,j}}{dt} = \pounds_{i,j}, \tag{2.37}$$

where subscripts i and j represent the discrete spatial indices in x- and y-directions, respectively. Here, $\pounds_{i,j}$ denotes the discrete spatial derivative operators. For our time advancement scheme, we apply a third order Runge-Kutta scheme given as (Gottlieb and Shu, 1998):

$$\begin{aligned}
\omega_{i,j}^{(1)} &= \omega_{i,j}^{(n)} + \Delta t \pounds_{i,j}^{(n)}, \\
\omega_{i,j}^{(2)} &= \tfrac{3}{4}\omega_{i,j}^{(n)} + \tfrac{1}{4}\omega_{i,j}^{(1)} + \tfrac{1}{4}\Delta t \pounds_{i,j}^{(1)}, \\
\omega_{i,j}^{(n+1)} &= \tfrac{1}{3}\omega_{i,j}^{(n)} + \tfrac{2}{3}\omega_{i,j}^{(2)} + \tfrac{2}{3}\Delta t \pounds_{i,j}^{(2)},
\end{aligned} \tag{2.38}$$

where $\Delta t = 2.5 \times 10^{-5}$ for our FOM simulations to satisfy the numerical stability criteria through the Courant–Freidrichs–Lewy (CFL) number. Similarly, the ROM equation can be rewritten as

$$\frac{d\alpha_k}{dt} = \mathfrak{R}_k, \tag{2.39}$$

where $\mathfrak{R}_k$ refers to the right-hand-side of Eq. (2.23). We use the same algorithm to compute $\alpha_k^{(n+1)}$ from $\alpha_k^{(n)}$

$$\begin{aligned}
\alpha_k^{(1)} &= \alpha_k^{(n)} + \Delta t \mathfrak{R}_k^{(n)}, \\
\alpha_k^{(2)} &= \tfrac{3}{4}\alpha_k^{(n)} + \tfrac{1}{4}\alpha_k^{(1)} + \tfrac{1}{4}\Delta t \mathfrak{R}_k^{(1)}, \\
\alpha_k^{(n+1)} &= \tfrac{1}{3}\alpha_k^{(n)} + \tfrac{2}{3}\alpha_k^{(2)} + \tfrac{2}{3}\Delta t \mathfrak{R}_k^{(2)},
\end{aligned} \tag{2.40}$$

where we specify larger time step, $\Delta t = 2.5 \times 10^{-4}$, since there is no such CFL stability constraints in the time advancement of ROMs. Although not shown here, we have verified that stable and accurate solutions can still be obtained by much bigger Δt using the ROM-D model.

2.7.2 Numerical discretizations

The source term, $\pounds_{i,j}$ in Eq. (2.37) includes nonlinear convective terms as well as the linear rotational and diffusive terms which can be written as:

$$\begin{aligned}\pounds_{i,j} = &-J(\omega_{i,j}, \psi_{i,j}) \\ &+ \frac{1}{\mathrm{Ro}}\frac{\partial \psi_{i,j}}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\omega_{i,j} + \frac{1}{\mathrm{Ro}}\sin(\pi y_{i,j}),\end{aligned} \tag{2.41}$$

where we use the standard second-order central finite difference schemes for linear terms. Therefore, the derivative operators in Eq. (2.41) can be written in following discrete form:

$$\frac{\partial \psi_{i,j}}{\partial x} = \frac{\psi_{i+1,j} - \psi_{i-1,j}}{2\Delta x}, \tag{2.42}$$

$$\begin{aligned}\nabla^2\omega_{i,j} = &\frac{\omega_{i+1,j} - 2\omega_{i,j} + \omega_{i-1,j}}{\Delta x^2} \\ &+ \frac{\omega_{i,j+1} - 2\omega_{i,j} + \omega_{i,j-1}}{\Delta y^2},\end{aligned} \tag{2.43}$$

where Δx and Δy are the step sizes in x- and y-directions, respectively. For the modeling of nonlinear term, we use second-order Arakawa scheme (Arakawa, 1966) to avoid computational instabilities arising from nonlinear interactions for the Jacobian term, $J(\omega_{i,j}, \psi_{i,j})$ in Eq. (2.41), defined as

$$J(\omega_{i,j}, \psi_{i,j}) = \frac{1}{3}(J_1 + J_2 + J_3), \tag{2.44}$$

where the discrete parts of the Jacobians have the following forms:

$$\begin{aligned}J_1 = &\frac{1}{4\Delta x \Delta y}[(\omega_{i+1,j} - \omega_{i-1,j})(\psi_{i,j+1} - \psi_{i,j-1}) \\ &- (\omega_{i,j+1} - \omega_{i,j-1})(\psi_{i+1,j} - \psi_{i-1,j})],\end{aligned} \tag{2.45}$$

$$J_2 = \frac{1}{4\Delta x \Delta y}[\omega_{i+1,j}(\psi_{i+1,j+1} - \psi_{i+1,j-1}) - \omega_{i-1,j}(\psi_{i-1,j+1} - \psi_{i-1,j-1}) - \omega_{i,j+1}(\psi_{i+1,j+1} - \psi_{i-1,j+1}) - \omega_{i,j-1}(\psi_{i+1,j-1} - \psi_{i-1,j-1})], \tag{2.46}$$

$$J_3 = \frac{1}{4\Delta x \Delta y}[\omega_{i+1,j+1}(\psi_{i,j+1} - \psi_{i+1,j}) - \omega_{i-1,j-1}(\psi_{i-1,j} - \psi_{i,j-1}) - \omega_{i-1,j+1}(\psi_{i,j+1} - \psi_{i-1,j}) - \omega_{i+1,j-1}(\psi_{i+1,j} - \psi_{i,j-1})]. \tag{2.47}$$

We refer the article by San and Illiescu (San and Iliescu, 2015) for reader's interest on the derivation of higher-order Arakawa schemes. In addition, we require to solve an elliptic Poisson equation at each substep in the time integration to find the streamfunction from the updated vorticity values which is the most computationally heavy part of our QG model solver. Since our computational domain is simple and uniform in this study, we utilize the fast Fourier transform (FFT) method to solve the Poisson equation. In our framework, not to deviate from the main focus of this paper, we use the FFT algorithm given by Press et al. (Press et al., 1992) to compute the forward and inverse sine transforms. The formulation of our FFT algorithm can be found in the referred article (San and Iliescu, 2015). We must note here that even though an FFT based Poisson solver reduces the overall computational burden of elliptic equations, it is limited to simple geometries on structured grids. Alternatively, a multigrid Poisson solver might be a better option for more complex basin problems which works well in complex geometries (San and Staples, 2013b).

2.7.3 Numerical integration

As stated before, we take the inner products of two quantities to compute the POD modes in our ROM frameworks which is done by taking the integral of the product over the domain Ω. For the two-dimensional numerical integration over Ω, we use the fourth-order accurate Simpson's $1/3^{rd}$ rule (Moin, 2010; Hoffman and Frankel, 2001). To illustrate the integration technique, we can consider $f(x, y) = u(x, y)v(x, y)$ so that it can be written as follows

$$\begin{aligned}\langle u; v\rangle &= \int_{\Omega} f(x, y)dxdy \\ &= \frac{\Delta y}{3}\sum_{j=1}^{N_y/2}\Big(\hat{f}_{2j-1} + 4\hat{f}_{2j} + \hat{f}_{2j+1}\Big),\end{aligned} \tag{2.48}$$

where

$$\begin{aligned}\hat{f}_j &= \frac{\Delta x}{3}\sum_{i=1}^{N_x/2}(f_{2i-1,j} + 4f_{2i,j} + 4f_{2i+1,j}), \\ j &= 1, 2, ..., N_y.\end{aligned} \tag{2.49}$$

Here, N_x and N_y are the number grid points in both x- and y-directions, respectively. For valid implementation of Simpson's $1/3^{rd}$ rule, the number of intervals in the x- and y-directions should be even. Also, the uniform interval sizes in each direction is assumed in the above formulae.

2.7.4 Proper orthogonal decomposition modes

A number of snapshots of the 2D vorticity field, denoted as $\omega(x, y, t_n)$, are stored at consecutive times t_n for $n = 1, 2, \ldots, N$. The time-averaged field, called "base flow", can be computed as

$$\bar{\omega}(x, y) = \frac{1}{N}\sum_{n=1}^{N}\omega(x, y, t_n). \tag{2.50}$$

The mean-subtracted snapshots, also called as anomaly or fluctuation fields, are then computed as the difference between the instantaneous field and the mean field

$$\omega'(x, y, t_n) = \omega(x, y, t_n) - \bar{\omega}(x, y). \tag{2.51}$$

This subtraction has been common in ROM community, and it guarantees that ROM solution would satisfy the same boundary conditions as full order model (Chen et al., 2012). This anomaly field procedure can be also interpreted as a mapping of snapshot data to its origin.

For generating the POD modes, we are following the standard method of snapshots proposed by Sirovich (Sirovich, 1987) as an efficient numerical procedure to save time in solving the eigenvalue problem necessary for POD, when the data dimension is much bigger than number of snapshots. First, an $N \times N$ snapshot data matrix $\mathbf{A} = [a_{ij}]$ is computed from the inner product of mean-subtracted snapshots

$$a_{ij} = \langle \omega'(x, y, t_i); \omega'(x, y, t_j) \rangle \tag{2.52}$$

Then, an eigen decomposition of $\mathbf{A}$ is performed as

$$\mathbf{AV} = \mathbf{V}\boldsymbol{\Lambda} \tag{2.53}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix whose entries are the eigenvalues λ_k of $\mathbf{A}$, and $\mathbf{V}$ is a matrix whose columns $\mathbf{v}_k$ are the corresponding eigenvectors. In our computations, we use the eigensystem solver based on the Jacobi transformations since $\mathbf{A}$ is a symmetric positive definite matrix (Press et al., 1992). It should be noted that eigenvalues need to be arranged in a descending order (i.e., $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$), for proper selection

of the POD modes. The POD modes ϕ_k are then computed as

$$\phi_k(x,y) = \frac{1}{\sqrt{\lambda_k}} \sum_{n=1}^{N} v_k^n \omega'(x,y,t_n) \tag{2.54}$$

where v_k^n is the nth component of the eigenvector $\mathbf{v}_k$. The scaling factor, $1/\sqrt{\lambda_k}$, is to guarantee the orthonormality of POD modes, i.e., $\langle \phi_i; \phi_j \rangle = \delta_{ij}$.

CHAPTER III

A Hybrid Approach for Model Order Reduction of Barotropic Quasi-Geostrophic Turbulence[†]

3.1 Abstract

We put forth a robust reduced-order modeling approach for near real-time prediction of mesoscale flows. In our hybrid-modeling framework, we combine physics-based projection methods with neural network closures to account for truncated modes. We introduce a weighting parameter between the Galerkin projection and extreme learning machine models and explore its effectiveness, accuracy and generalizability. To illustrate the success of the proposed modeling paradigm, we predict both the mean flow pattern and the time series response of a single-layer quasi-geostrophic ocean model, which is a simplified prototype for wind-driven general circulation models. We demonstrate that our approach yields significant improvements over both the standard Galerkin projection and fully non-intrusive neural network methods with a negligible computational overhead.

3.2 Introduction

The modeling of large-scale turbulent flow problems has observed a continuous change over the years with the rapid growth of the computational power, technological breakthrough, algorithmic innovation, and the availability of data resources. Many of these physical phenomena are described by a set of ordinary differential equations (ODEs) or partial differential equations (PDEs) to construct the mathematical model

[†]The chapter is adapted from Rahman et al., Fluids 3.4 (2018): 86.

of a physical system. Thus far, many computational approaches are based on solving these sets of equations of a mathematical model after applying suitable discretization schemes and numerical tools which may be referred to as a physics-based solution approach. Despite the advances in software engineering and processor technologies, the computational burden of high-fidelity simulation is still a limiting factor for many practical problems in different areas of research, specifically for the large-scale physical problems with high spatiotemporal variabilities such as turbulence modeling of geophysical flows, weather forecasting, and climate modeling. Indeed, low-fidelity models such as large eddy simulation (LES) (Sagaut, 2006) and Reynolds-averaged Navier–Stokes (RANS) (Wilcox, 1998) are introduced with additional approximations to neglect some of the physical aspects for closure modeling, and, as a result, the computational cost is reduced. However, these techniques require parameter calibration to approximate the true solution to any degree of confidence and may thus increase costs related to model validation and benchmark data generation.

As a first attempt to perform efficient physics-based surrogate modeling, projection based implementations of model order reduction strategies have proven to be successful in reducing computational cost significantly with little compromise in physical accuracy, and have been extensively utilized in the literature to address challenging fluid flow problems. In contrast to the physics-based modeling technique, the non-intrusive, data-driven modeling has recently become extremely popular in turbulence modeling because of the abundance of data resources from both high-fidelity simulations and experiments (Durbin, 2018; Duraisamy et al., 2018). Until recently, fully non-intrusive artificial intelligence and machine learning algorithms have been considered one of the most promising fields to benefit greatly from existing datasets. Because of their black box nature in predicting the solution, the data-driven techniques are yet to be fully accepted in the community. This has led to the development of a new and fast emerging hybrid modeling approach (i.e., *hybrid analytics*) that combines a physics-

based modeling strategy with a data-driven model to develop a robust scheme with significantly enhanced potential for performance than the component models alone. In this paper, we propose a hybrid framework to develop a robust and improved reduced-order model by combining a standard projection based reduced-order modeling with a simple artificial neural network architecture. This approach is applied to a single-layer quasi-geostrophic ocean model.

The study of ocean circulation, specifically the wind-driven circulation, is one of the most demanding areas of research from engineering and scientific points of view because of the energy potential in the motions of water masses within oceans in the form of heat and strong currents as well as the connection between ocean currents and climate dynamics (Munk, 1950; Pelc and Fujita, 2002; Esteban and Leary, 2012). Since ocean circulation is a significant part of climate systems, it is extremely important to study and understand the underlying physics to get the benefits from the rich potential of oceans. There have been many efforts put on developing models for ocean dynamics in the last few decades (McWilliams, 2006; Berloff, 2015; San et al., 2011; Berloff, 2016; Miller, 2007; Kondrashov et al., 2018). However, it is computationally challenging to model the ocean dynamics because of its large range of spatiotemporal scales and sporadic random transitions between coexisting eddies and vortices (Lynch, 2008; Laurie and Bouchet, 2015). In our study, we use a two-dimensional single layer quasi-geostrophic ocean model as a test case for our proposed reduced-order model and consider the barotropic vorticity equation (BVE) with dissipative and forcing terms which is one of the most commonly used models for the double-gyre wind-driven geophysical flows. The BVE model is a simplified version of the full-fledged geophysical flow equations by considering the hydrostatic balance, geostrophic balance, the β-plane approximation, and horizontal eddy viscosity parameterizations. However, the computational cost of simulation and analysis of the BVE model is high which makes this model a suitable test case to investigate an improved and robust reduced-order

model.

Reduced-order modeling (ROM) techniques have been commonly introduced to reduce the computational cost of obtaining high-fidelity simulation results by generating a lower dimensional system with a similar range of accuracy through a variety of mathematical operations. Based on the approach, there have been a considerable number of promising reduced-order modeling techniques introduced, such as proper orthogonal decomposition (POD), Koopman mode decomposition (KMD), dynamic mode decomposition (DMD), reduced bases, etc. A detailed overview of such techniques and recent developments can be found in Refs. (Rowley and Dawson, 2017; Taira et al., 2017). To predict the flow behavior in reduced-order space, we utilize the POD framework to generate the POD basis function and then use the Galerkin projection to find a set of ODEs that yield a low dimensional dynamical system (Ito and Ravindran, 1998). Even though the Galerkin projection based ROMs have been successfully used in various areas of work (Lucia et al., 2004; Brenner et al., 2012; Iollo et al., 2000; Freno and Cizmas, 2014), they are limited by modeling errors and numerical instabilities due to the truncation of POD modes for complex systems, such as the ocean circulation system considered in this study. See Refs. (San and Iliescu, 2015; Behzad et al., 2015; Cizmas et al., 2008; Bistrian and Navon, 2015; Behzad et al., 2018; Bistrian and Navon, 2017) to find some recent developments made to improve physics-based ROM techniques by stabilizing the numerical instabilities and taking into account the truncated mode contributions. Furthermore, Table 3.1 summarizes our descriptions for classifying several ROM approaches.

In the present study, we utilize an artificial neural network (ANN) to develop our ROM framework. Neural networks, one of many machine learning (ML) strategies, have become an area of interest for researchers from diverse fields of study because of their self-adaptive and flexible nature, while approximating complex functions of any physical process with a high degree of accuracy. Even though ML tools such as ANN

Table 3.1: A classification of reduced-order modeling approaches.

Approaches	**Comments**
Fully non-intrusive models	Data-based modeling; data could be generated by experimental measurements (observational data) or high-fidelity numerical simulations (synthetic data); no need to know the underlying physical system or model (no need to have access to a full order model generating data).
Semi non-intrusive models	It is a mixed approach with offline intrusive and online non-intrusive models; in addition to snapshot data, we should have access to the high-fidelity model to generate our surrogate model (ROM); after built, ROM stays on a reduced subspace, and we do not need to have access to a high-fidelity model during ROM computations; we often use a projection approach (with truncation) to obtain a dense low-order system.
Intrusive models	We need to have access to some parts (or whole) of a high-fidelity model during online ROM computations; sparse sampling or interpolation approaches might be incorporated; multilevel, multigrid, adaptive mesh refinement and dynamic time stepping approaches to accelerate high-fidelity simulations can be considered in this category.
Coarse-grained models	It is a special case of intrusive modeling; reduction approach might utilize a similar high-fidelity model with a reduced computational complexity (i.e., fewer grid points in LES or RANS approaches); they often need a closure model to compensate the effects of truncated scales; closure effects can be embedded into numerics as well.

have been utilized in flow control community for the last couple of decades (Gautier et al., 2015; Lee et al., 1997), the data-driven idea has recently been popular in computational fluid dynamics community for modeling and solving complex fluid flow problems (Gamahara and Hattori, 2017; Duraisamy et al., 2015; Maulik and San, 2017; San and Maulik, 2018a; Wu et al., 2018; Moosavi et al., 2015). These data-driven models can be developed either in a black box fashion or in combination with existing models. We utilize a simple single hidden layer feedforward network (SLFN), known as extreme learning machine (ELM), in this paper for both fully data-driven model and hybrid model. The resolved reduced-order model coefficients and full order simulation data are used to train the ELM architecture. The descriptive idea on ELM and its implementations can be found in Refs. (Huang et al., 2006; Huang and Chen, 2007; Xiang et al., 2014).

The purpose of this paper is to develop a robust and efficient ROM framework using a hybrid approach, and test the performance of the proposed model with respect to the component models, i.e., physics-based and data-driven models. Recently, several promising works have been conducted to develop more efficient and improved reduced-order model frameworks using the hybridization of traditional projection based ROMs and data-driven ROMs (Williams et al., 2013; Xie et al., 2018a; Mohebujjaman et al., 2018). The proposed hybrid framework is developed by combining two component models, the standard projection based ROM and an ANN framework. Our main motivation of this study comes from addressing the strengths and weaknesses of the component models to find a bridge between both models to benefit from the strength of both approaches. Combining an imperfect physics-based model with a data-driven technique to get a hybrid scheme is not an entirely new idea in scientific research (Wan et al., 2018; Pathak et al., 2018). Even though the hybridization approach varies with respect to the physical problem, selection of ML architecture, approach to find the solution, and the associated physics-based models, the main idea remains similar

for most of the works. We then choose a challenging test problem for model order reduction for validation and comparison purpose, and perform a number of statistical analyses on the hybrid model as well as on the component models. The results of our tests suggest that the proposed model is robust and can be extremely useful for complex flow problems.

The rest of this manuscript is outlined as follows: in Section 3.3, we review the underlying governing equation of the physical problem considered in this study to generate data bank in full order space for POD basis construction and training the ANN architecture as well as the numerical schemes used for full order simulation. We briefly discuss the standard Galerkin projection based ROM approach and fully non-intrusive data-driven ROM approach in Sections 3.4 and 3.5, respectively. In Section 3.6, we introduce our proposed hybrid ROM framework. In Section 3.7, we perform a variety of analyses to assess the performance and viability of the proposed model with respect to the performance of its physics-based component or data-driven component alone. We also include the full order simulation results as reference. Finally, we summarize our findings in Section 3.8.

3.3 Full Order Model (FOM)

In FOM, we solve the underlying governing equations using a suitable numerical model and generate the data bank of snapshots for ROM. In this section, we introduce the quasi-geostrophic ocean model as a test case and provide a brief description of the numerical schemes employed in this study.

3.3.1 Quasi-geostrophic (QG) ocean model

Since it is well established that much of the world's ocean circulation is wind-driven in large-scale, and oceanic mesoscale processes are also extremely prevalent in the large-scale oceanic circulations, studies of wind-driven circulation using an idealized double

gyre wind forcing have become increasingly important in understanding various aspects of ocean and climate dynamics, such as the role of mesoscale eddies and their effect on mean circulation (Laurie and Bouchet, 2015; Siedler et al., 2013; Byrne et al., 2016; Holland, 1978; Hogg et al., 2009). However, the numerical simulation of oceanic and atmospheric flows, up to this point, still requires approximations and simplifications of the full model to resolve some of the enormous range of spatial and temporal scales of the full form of the general circulation models. Even though remarkable advances in developing simplified models have been made during the last few decades in producing increasingly accurate results (McWilliams, 2006; Hua and Haidvogel, 1986), an additional computational challenge is in being able to implement them in long time integration such as those required by the climate modeling (Ghil et al., 2008; Lynch, 2008). Therefore, with this in mind, we consider the simple single-layer QG model, also known as barotropic vorticity equation (BVE) that can capture the random inter-decade and inter-annual transitions in large-scale ocean basins (Charney et al., 1950). This mathematical model shares many analogies with the two-dimensional Euler equation and Navier–Stokes equation (Bouchet and Simonnet, 2009) and has been extensively used over the years to describe various aspects of the largest scales of turbulent geophysical fluid dynamics (Cushman-Roisin and Beckers, 2011; Majda and Wang, 2006; Vallis, 2017). The dimensionless BVE for QG single-layer ocean model can be written as (Greatbatch and Nadiga, 2000; San and Maulik, 2018a)

$$\frac{\partial \omega}{\partial t} + \mathrm{J}(\omega, \psi) = \frac{1}{\mathrm{Ro}} \frac{\partial \psi}{\partial x} + \frac{1}{\mathrm{Re}} \nabla^2 \omega + \frac{1}{\mathrm{Ro}} \sin(\pi y), \tag{3.1}$$

where the left hand side terms account for local acceleration and nonlinear advection, respectively, and the right hand side terms represent the rotational, dissipative, and forcing effects, respectively, in the governing equation. Here, ω is the kinematic

vorticity, expressed as

$$\omega = \nabla \times \mathbf{u}, \tag{3.2}$$

where $\mathbf{u}$ is the velocity vector. For two-dimensional flow in the x-y plane, the kinematic vorticity becomes

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \tag{3.3}$$

Here, the flow velocity components can be found from the stream function, ψ, using the following definition:

$$\mathbf{u} = \nabla \times \psi \hat{k}. \tag{3.4}$$

In Cartesian co-ordinates,

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}. \tag{3.5}$$

The kinematic equation connecting the vorticity and stream function can be found by substituting the velocity components in terms of stream function in Equation (3.3), which forms the following divergence-free constraint satisfying Poisson equation:

$$\nabla^2 \psi = -\omega, \tag{3.6}$$

where ∇^2 is the standard two-dimensional Laplacian operator. The dimensionless formulation of BVE in Equation (3.1) contains Reynolds number (Re) and Rossby number (Ro), two dimensionless numbers which are related to the physical parameters

and non-dimensional variables in the following way:

$$\mathrm{Re} = \frac{VL}{\nu}, \ \mathrm{Ro} = \frac{V}{\beta L^2}, \tag{3.7}$$

where ν is the horizontal eddy viscosity coefficient for ocean basin and β is the Rossby parameter. It should be noted that the BVE model presented here uses the beta-plane approximation, valid for most mid-latitude simplified ocean basins, i.e., the Coriolis parameter to account for Earth's rotational effect is approximated by $f = f_0 + \beta y$, where f_0 is the constant mean Coriolis parameter and β is the gradient of the Coriolis parameter at the basin center. The characteristic horizontal length scale, L (for the non-dimensional purpose) is the basin dimension in x-direction in this study and the reference Sverdrup velocity scale, V, is given by

$$V = \frac{\tau_0}{\rho H} \frac{\pi}{\beta L}, \tag{3.8}$$

where τ_0 is the maximum amplitude of the sinusoidal double-gyre wind stress, and ρ and H are the reference fluid density and depth of the ocean basin, respectively. In our study, we consider the benchmark four-gyre circulation problem where the time average of the statistically steady equilibrium state exhibits a four-gyre circulation pattern for relatively high Re instead of the symmetric double-gyre standard structure (Nadiga and Margolin, 2001; Greatbatch and Nadiga, 2000; San et al., 2011). Based on reference literature (San, 2016; San and Staples, 2013a; San et al., 2011; Özgökmen and Chassignet, 1998; Holm and Nadiga, 2003), we use slip boundary condition for the velocity, which translates into homogeneous Dirichlet boundary condition for the vorticity $\omega|_\Gamma = 0$ and corresponding $\psi|_\Gamma = 0$, where Γ symbolizes all the Cartesian boundaries. We initiate our computations from a quiescent state (i.e., $\omega = 0$ and $\psi = 0$) and integrate Equation (3.1) until a statistically steady state is obtained.

3.3.2 Numerical schemes

Since for most of the physically relevant ocean circulation models, such as the QG models, the solutions remain in a quasi-stationary state as time goes to infinity (Medjo, 2000), schemes designed for the numerical integration of such phenomena should be suited for quasi-stationary behavior of the solutions as well as for the long time integration. In our study, we use the second-order central finite difference schemes in linear terms. For the modeling of nonlinear term, we use second-order skew-symmetric, energy- and enstrophy-conserving Arakawa scheme (Arakawa, 1966) to avoid computational instabilities arising from nonlinear interactions for the Jacobian term, $J(\omega_{i,j}, \psi_{i,j})$ in Equation (3.1), defined as

$$J(\omega_{i,j}, \psi_{i,j}) = \frac{1}{3}(J_1(\omega_{i,j}, \psi_{i,j}) + J_2(\omega_{i,j}, \psi_{i,j}) + J_3(\omega_{i,j}, \psi_{i,j})), \tag{3.9}$$

where

$$\begin{aligned} J_1(\omega_{i,j}, \psi_{i,j}) =& \frac{1}{4\Delta x \Delta y}[(\omega_{i+1,j} - \omega_{i-1,j})(\psi_{i,j+1} - \psi_{i,j-1}) \\ & - (\omega_{i,j+1} - \omega_{i,j-1})(\psi_{i+1,j} - \psi_{i-1,j})], \end{aligned} \tag{3.10}$$

$$\begin{aligned} J_2(\omega_{i,j}, \psi_{i,j}) =& \frac{1}{4\Delta x \Delta y}[\omega_{i+1,j}(\psi_{i+1,j+1} - \psi_{i+1,j-1}) - \omega_{i-1,j}(\psi_{i-1,j+1} - \psi_{i-1,j-1}) \\ & - \omega_{i,j+1}(\psi_{i+1,j+1} - \psi_{i-1,j+1}) - \omega_{i,j-1}(\psi_{i+1,j-1} - \psi_{i-1,j-1})], \end{aligned} \tag{3.11}$$

$$\begin{aligned} J_3(\omega_{i,j}, \psi_{i,j}) =& \frac{1}{4\Delta x \Delta y}[\omega_{i+1,j+1}(\psi_{i,j+1} - \psi_{i+1,j}) - \omega_{i-1,j-1}(\psi_{i-1,j} - \psi_{i,j-1}) \\ & - \omega_{i-1,j+1}(\psi_{i,j+1} - \psi_{i-1,j}) - \omega_{i+1,j-1}(\psi_{i+1,j} - \psi_{i,j-1})]. \end{aligned} \tag{3.12}$$

Here, Δx and Δy are the step size in x- and y-directions, respectively. The detailed derivation of higher-order Arakawa schemes can be found in Ref. (San and Iliescu, 2015). To implement the Runge–Kutta scheme, we use the method of lines to cast

our system in the following form:

$$\frac{d\omega_{i,j}}{dt} = \pounds(\omega_{i,j}). \tag{3.13}$$

Here, subscripts i and j represent the discrete spatial indices in x and y directions, respectively. $\pounds(.)$ denotes the discrete spatial derivative operators. For our time advancement scheme, we use the optimal third order accurate total variation diminishing Runge–Kutta (TVDRK3) scheme given as (Gottlieb and Shu, 1998):

$$\left.\begin{aligned} \omega_{i,j}^{(1)} &= \omega_{i,j}^{(n)} + \pounds(\omega_{i,j}^{(n)})\Delta t, \\ \omega_{i,j}^{(2)} &= \frac{3\omega_{i,j}^{(n)} + \omega_{i,j}^{(1)} + \pounds(\omega_{i,j}^{(1)})\Delta t}{4}, \\ \omega_{i,j}^{(n+1)} &= \frac{\omega_{i,j}^{(n)} + 2\omega_{i,j}^{(2)} + 2\pounds(\omega_{i,j}^{(2)})\Delta t}{3}, \end{aligned}\right\} \tag{3.14}$$

where Δt is the adaptive time step, which can be computed at the end of each time step satisfying the stability criteria through the Courant–Freidrichs–Lewy (CFL) number. Finally, taking advantage of the simple Cartesian domain, an efficient fast Fourier transform (FFT) method is utilized for the solution of the elliptic Poisson equation which is the most computationally expensive part of the QG models (Miller, 2007). Details of the aforementioned numerical methods used for this study can be found in Refs. (San, 2016; San et al., 2011).

3.4 Galerkin Projection Based Reduced-Order Model (ROM-GP)

Similar to the area of interest in the present study, many areas of physical science and engineering often require iterative computations, data assimilation, and finer scale simulation for large dynamical systems to resolve the underlying physical phenomena (Wedi, 2014; Brunton and Noack, 2015; Navon et al., 1992; Kalnay, 2003). The efficiency of these large-scale simulations decreases with increasing computational complexity

of the full order models due to the accumulation of the less significant intermediate results. Hence, the reduced-order models (ROMs) have come into wide use in large-scale simulations to replace large dynamical systems with lower dimensional systems having a similar range of validity and input/output characteristics. Proper orthogonal decomposition (POD), also referred as Karhunen–Loève transform (KLT), principal component analysis (PCA) or singular value decomposition (SVD), is a widely applied snapshot-based ROM technique in science and engineering applications with large dimensional systems of nonlinear ODEs or PDEs (Holmes et al., 2012; Lumley, 1970; Kosambi, 2016). In this section, we develop a standard projection methodology for Equation (3.1) where a POD-GP ROM framework is developed from the field variables, ω and ψ on the flow domain, and Ω based on the method of snapshots (Sirovich, 1987). The snapshots are obtained from the stored sample solution of FOM. The main procedure of ROM in the current study consists of generating low-dimensional orthogonal basis functions from snapshots and next, performing Galerkin projection to obtain the ROM. For POD basis construction, we first obtain N number of snapshots and denote the vorticity field, $\omega(x, y)$ as $\omega(x, y, t_n)$ for $n = 1, 2, ..., N$ at pseudo-time $t = t_n$. Then, we decompose the solution field by following way into a time-invariant mean, $\bar{\omega}(x, y)$ and a fluctuating part, $\omega'(x, y, t)$ to represent the full field with only fluctuating part (Noack et al., 2003):

$$\omega(x, y, t) = \bar{\omega}(x, y) + \omega'(x, y, t), \tag{3.15}$$

for $x, y \in \Omega$ where Ω stands for the two-dimensional domain and the mean of the snapshot is

$$\bar{\omega}(x, y) = \frac{1}{N} \sum_{n=1}^{N} \omega(x, y, t_n). \tag{3.16}$$

The following algorithm describes the POD-GP procedure:

1. POD basis construction:

   * Construct a data correlation matrix of the fluctuating part, $\mathbf{A} = [\alpha_{ij}]$ from the snapshots where $\alpha_{ij} = \int_\Omega \omega'(x, y, t_i)\omega'(x, y, t_j)dxdy$. Here, i and j refer to the snapshot indices.

   * Compute the optimal POD basis functions by solving $\mathbf{A\Gamma} = \mathbf{\Gamma\Lambda}$, where $\mathbf{\Lambda} = \text{diag}[\lambda_1, ..., \lambda_Q]$ is the diagonal eigenvalue matrix and $\mathbf{\Gamma} = [\gamma^1, ..., \gamma^N]$ refers to right eigenvector matrix whose columns are eigenvectors of $\mathbf{A}$. In general, most of the subroutines for solving above eigenvalue equation give Γ with all of the eigenvectors normalized to unity.

   * Using the eigenvalues $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_N$ stored in a descending order in the diagonal matrix, $\mathbf{\Lambda}$, define the orthogonal POD basis functions for the vorticity field as

$$\phi_k(x, y) = \frac{1}{\sqrt{\lambda_k}} \sum_{n=1}^{N} \gamma_n^k \omega'(x, y, t_n), \tag{3.17}$$

     where λ_k is the kth eigenvalue, γ_n^k is the nth component of the kth eigenvector, and $\phi_k(x, y)$ is the kth POD mode. We must mention that the eigenvectors must be normalized in such a way that the basis functions satisfy the orthogonality condition.

   * Obtain the kth mode for the stream function, $\varphi_k(x, y)$, utilizing the linear dependence between stream function and vorticity given by Equation (3.6): $\nabla^2\varphi_k = -\phi_k$.

   * Span the fluctuating component of the field variables into the POD modes

by doing the separation of variable as

$$\omega'(x,y,t) = \sum_{k=1}^{N} a_k(t)\phi_k(x,y), \tag{3.18}$$

$$\psi'(x,y,t) = \sum_{k=1}^{N} a_k(t)\varphi_k(x,y), \tag{3.19}$$

where $a_k(t)$ is the time-dependent modal coefficients and $\phi_k(x,y)$ and $\varphi_k(x,y)$ refer to the POD modes. It should be noted that the same $a_k(t)$ accounts for both stream function and vorticity based on the kinematic relation given by Equation (3.6).

* Retain R modes where $R << N$, such that these R largest energy containing modes correspond to the largest eigenvalues $(\lambda_1, \lambda_2, ..., \lambda_R)$. The resulting full expression for the field variables can be written as

$$\omega(x,y,t) = \bar{\omega}(x,y) + \sum_{k=1}^{R} a_k(t)\phi_k(x,y), \tag{3.20}$$

$$\psi(x,y,t) = \bar{\psi}(x,y) + \sum_{k=1}^{R} a_k(t)\varphi_k(x,y). \tag{3.21}$$

2. Galerkin projection to obtain ROM:

* Perform an orthogonal Galerkin projection by multiplying the governing equation with the POD basis functions and integrating over the domain Ω (Rapún and Vega, 2010), which yields the following dynamical system for a_k:

$$\frac{da_k}{dt} = \zeta_k + \sum_{i=1}^{R} \eta_k^i a_i + \sum_{i=1}^{R}\sum_{j=1}^{R} \chi_k^{ij} a_i a_j, \tag{3.22}$$

where

$$\zeta_k = \langle \frac{1}{\mathrm{Re}}\nabla^2\bar{\omega} + \frac{1}{\mathrm{Ro}}\sin(\pi y) + \frac{1}{\mathrm{Ro}}\frac{\partial\bar{\psi}}{\partial x} - J(\bar{\omega},\bar{\psi}), \phi_k\rangle,$$
$$\eta_k^i = \langle \frac{1}{\mathrm{Re}}\nabla^2\phi_i + \frac{1}{\mathrm{Ro}}\frac{\partial\varphi_i}{\partial x} - J(\bar{\omega},\varphi_i) - J(\phi_i,\bar{\psi}), \phi_k\rangle,$$
$$\chi_k^{ij} = \langle -J(\phi_i,\varphi_j), \phi_k\rangle.$$

We can define the right hand side of Equation (3.22) as r_k^{GP} to express the equation in the following form:

$$\frac{da_k}{dt} = r_k^{\mathrm{GP}}. \tag{3.23}$$

The system given by above algorithm consists of R coupled ODEs for modal coefficients, a_k which can be solved numerically by any suitable time integration scheme. Additionally, the POD basis functions and ROM coefficients can be precomputed from the data, which makes the system more efficient. Defining the vorticity field, $\omega(x, y, t_0)$ at time t_0, a complete specification of the dynamical system given by Equation (3.22) can be calculated by the following projection of the initial condition:

$$a_k(t_0) = \langle \omega(x, y, t_0) - \bar{\omega}(x, y), \phi_k\rangle. \tag{3.24}$$

It is worth mentioning that there are occasions in our POD-GP ROM framework where we compute the inner product of two functions $f = f(x, y)$ and $g = g(x, y)$ as

$$\langle f, g\rangle = \int_\Omega f(x, y)g(x, y)dxdy. \tag{3.25}$$

In the present study, we compute the inner products by taking the integral of the product over the domain, Ω using the dual integration method for Simpson's $1/3^{\mathrm{rd}}$ rule. We refer to Ref. (Hoffman and Frankel, 2001) for the details of the integration

technique.

3.5 Artificial Neural Network Based Non-Intrusive Reduced-Order Model (ROM-ANN)

The study of artificial neural networks (ANN) was introduced in the 1990s as a multilayered perceptron based on the idea of mimicking the functions of neurons in human brain (Hagan et al., 1996). ANNs consist of layers possessing a predefined number of unit cells called neurons and connections between the neurons called weights to establish a map between an input layer and an output layer. Other than the input and output layers, an ANN architecture also contains at least one hidden layer which is independent of the training data. The learning can be supervised or unsupervised. In unsupervised learning, the network extracts data features without knowing the target output and input label (Jang et al., 1997). On the other hand, we have used a supervised learning network named single-hidden layer feedforward neural networks (SLFNs) based on the precomputed inputs from ROM-GP model which discovers the mapping between inputs and outputs from provided labeled sample. Mathematically, the input layer distributes input signals $x_p : p = 1, 2, ..., P \in \mathbb{R}^P$ to the Q dimensional hidden layer feature space (i.e., Q is the number of hidden neurons). Let $y_j : j = 1, 2, ..., J \in \mathbb{R}^J$ represent the outputs from J number of output neurons. $c_q^p \in \mathbb{R}^{Q \times P}$ are the weights connecting the neurons in input and hidden layers, $\beta_j^q \in \mathbb{R}^{J \times Q}$ are the output weights between the hidden and output layers, and $\epsilon_q \in \mathbb{R}^Q$ and $\xi_j \in \mathbb{R}^J$ are the random biases operating as thresholds for the hidden and output layers. Then, the feed-forward operation can be written as:

$$y_j = G\left[\xi_j + \sum_{q=1}^{Q} \beta_j^q F\left(\sum_{p=1}^{P} c_q^p x_p + \epsilon_q\right)\right], \tag{3.26}$$

where differentiable functions F and G are the transfer or activation functions (Hornik et al., 1989; Lapedes and Farber, 1988) which can be log-sigmoid, hyperbolic tangent sigmoid, or linear transfer function. The training process of ANN is adjusting the weights to reproduce the desired outputs when the given inputs are fed forward. In the present study, we use Extreme Learning Machine (ELM) architecture which randomly chooses weights for the hidden nodes and bias terms and analytically determines the output weights of SLFNs (Huang et al., 2011). Huang et al. (Huang et al., 2006) first introduced ELM with a goal to achieve a learning speed thousands time faster than the traditional feedforward network learning algorithm while producing a better generalization performance. Later, several extensions of general ELM method are developed in different works (Huang and Chen, 2007; Zhou et al., 2015; Kasun et al., 2013). In our study, the ANN architecture is trained by utilizing an ELM approach which requires no biases in the output layer ($\xi = 0$) as well as the weights (c_q^p) and biases (ϵ_q) are initialized randomly from a uniform distribution between -1 and $+1$ and no longer modified. Therefore, β_j^q weights are only unknown to be determined in this case. We note that we utilized the tan-sigmoid activation function for the hidden layer neurons given by

$$F(x) = \frac{1 - \exp(-2x)}{1 + \exp(+2x)}, \tag{3.27}$$

and a linear activation function for the output layer neurons which is $G(x) = x$. Finally, from Equation (3.26), the output of ELM can be written as for N sample training data examples,

$$y_j^n = \sum_{q=1}^{Q} \beta_j^q F\left(\sum_{p=1}^{P} c_q^p x_p^n + \epsilon_q\right), \tag{3.28}$$

where $x_p^n \in \mathbb{R}^{P\times N}$ and $y_j^n \in \mathbb{R}^{J\times N}$ refer to training input–output data pairs. To illustrate further, we can show our problem in following matrix form:

$$\begin{bmatrix} y_1^1 & . & . & . & . & y_1^N \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ y_J^1 & . & . & . & . & y_J^N \end{bmatrix}_{(J\times N)} = \begin{bmatrix} \beta_1^1 & . & . & \beta_1^Q \\ . & . & . & . \\ . & . & . & . \\ \beta_J^1 & . & . & \beta_J^Q \end{bmatrix}_{(J\times Q)} \begin{bmatrix} (H_1^1)^T & . & . & . & . & (H_1^N)^T \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ (H_Q^1)^T & . & . & . & . & (H_Q^N)^T \end{bmatrix}_{(Q\times N)}, \tag{3.29}$$

where the superscript "T" refers to "Transpose". In our study, the value of J is always 1 since there is only one output neuron. $\mathbf{H}^T \in \mathbb{R}^{Q\times N}$ on the right hand side of the above equation is given by

$$\mathbf{H}^T_{(Q\times N)} = F\left(\begin{bmatrix} \epsilon_1 & . & . & . & . & \epsilon_1 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ \epsilon_Q & . & . & . & . & \epsilon_Q \end{bmatrix}_{(Q\times N)} + \begin{bmatrix} C_1^1 & . & C_1^P \\ . & . & . \\ . & . & . \\ C_Q^1 & . & C_Q^P \end{bmatrix}_{(Q\times P)} \begin{bmatrix} x_1^1 & . & . & . & . & x_1^N \\ . & . & . & . & . & . \\ x_P^1 & . & . & . & . & x_P^N \end{bmatrix}_{(P\times N)} \right). \tag{3.30}$$

By taking the transpose of Equation (3.29), the solution for the weights can be computed by

$$\begin{bmatrix} \beta_1^1 & . & . & \beta_1^J \\ . & . & . & . \\ . & . & . & . \\ \beta_Q^1 & . & . & \beta_Q^J \end{bmatrix}_{(Q\times J)} = \mathbf{H}^{\dagger}_{(Q\times N)} \begin{bmatrix} (y_1^1)^T & . & . & (y_1^J)^T \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ . & . & . & . \\ (y_N^1)^T & . & . & (y_N^J)^T \end{bmatrix}_{(N\times J)}. \tag{3.31}$$

Here, $\mathbf{H}^{\dagger} \in \mathbb{R}^{Q \times N}$ is the pseudo-inverse of $\mathbf{H} \in \mathbb{R}^{N \times Q}$ and "†" refers to "pseudo-inverse". Since $N \geq Q$ for $\mathbf{H}$ matrix, we can apply following single value decomposition (SVD) to $\mathbf{H}$ to compute the pseudo-inverse,

$$\begin{bmatrix} H_1^1 & . & H_1^Q \\ . & . & . \\ . & . & . \\ H_N^1 & . & H_N^Q \end{bmatrix}_{(N\times Q)} = \begin{bmatrix} U_1^1 & . & U_1^Q \\ . & . & . \\ . & . & . \\ U_N^1 & . & U_N^Q \end{bmatrix}_{(N\times Q)} \begin{bmatrix} \Sigma_1^1 & . & 0 \\ . & . & . \\ 0 & . & \Sigma_Q^Q \end{bmatrix}_{(Q\times Q)} \begin{bmatrix} (V_1^1)^T & . & (V_1^Q)^T \\ . & . & . \\ (V_Q^1)^T & . & (V_Q^Q)^T \end{bmatrix}_{(Q\times Q)}, \tag{3.32}$$

$$\begin{bmatrix} (H_1^1)^{\dagger} & . & (H_1^N)^{\dagger} \\ . & . & . \\ (H_Q^1)^{\dagger} & . & (H_Q^N)^{\dagger} \end{bmatrix}_{(Q\times N)} = \begin{bmatrix} V_1^1 & . & V_1^Q \\ . & . & . \\ V_Q^1 & . & V_Q^Q \end{bmatrix}_{(Q\times Q)} \begin{bmatrix} (\Sigma_1^1)^{\dagger} & . & 0 \\ . & . & . \\ 0 & . & (\Sigma_Q^Q)^{\dagger} \end{bmatrix}_{(Q\times Q)} \begin{bmatrix} (U_1^1)^T & . & (U_1^N)^T \\ . & . & . \\ (U_Q^1)^T & . & (U_Q^N)^T \end{bmatrix}_{(Q\times N)}, \tag{3.33}$$

where $\mathbf{U}$ and $\mathbf{V}$ are column-orthogonal and orthogonal matrices, and Σ is a diagonal matrix whose elements (i.e., Σ_q^q where $q = 1, 2, \ldots, Q$) are non-negative and called singular values. To minimize the numerical instability introduced while calculating the inverse of Σ by taking the reciprocal of each non-zero element, a well-known Tikhonov-type regularization (Cordier et al., 2010) is used as

$$(\Sigma_q^q)^{\dagger} = \frac{\Sigma_q^q}{(\Sigma_q^q)^2 + \tau}. \tag{3.34}$$

We choose $\tau = 10^{-12}$ for the present study which indicates the trade-off between the least-squares error and the penalty term for regularization (Cancelliere et al., 2017). The unknown weights can be calculated by using Equation (3.31).

In the present study, our ELM layers and architecture are devised with five inputs

($P = 5$) and a single output ($J = 1$) mechanism, as shown in Figure 3.1. Here, "LT" refers to "linear term", "NT" refers to "nonlinear term", and "TP" refers to "true projection" in the figure. The inputs are resolved ROM-GP variables in each time integration step which can be denoted from Equation (3.22) as

$$a_k = \langle \omega - \bar{\omega}, \phi_k \rangle, \tag{3.35}$$

$$r_k^{\mathrm{LT}} = \sum_{i=1}^{R} \eta_k^i a_i, \tag{3.36}$$

$$r_k^{\mathrm{NT}} = \sum_{i=1}^{R} \sum_{j=1}^{R} \chi_k^{ij} a_i a_j, \tag{3.37}$$

$$r_k^{\mathrm{GP}} = \zeta_k + r_k^{\mathrm{LT}} + r_k^{\mathrm{NT}}, \tag{3.38}$$

and the single true solution output for training set can be expressed as

$$r_k^{\mathrm{TP}} = \left\langle -\mathrm{J}(\omega, \psi) + \frac{1}{\mathrm{Ro}} \frac{\partial \psi}{\partial x} + \frac{1}{\mathrm{Re}} \nabla^2 \omega + \frac{1}{\mathrm{Ro}} \sin(\pi y), \phi_k \right\rangle, \tag{3.39}$$

which is the Galerkin projection to the FOM given by Equation (3.1). We denote the ANN predicted solution output as r_k^{ANN} for the rest of this paper. Using the ANN prediction, r_k^{ANN}, we can compute the modal coefficients, a_k, for fully non-intrusive ROM using the following equation similar to Equation (3.23):

$$\frac{da_k}{dt} = r_k^{\mathrm{ANN}}, \tag{3.40}$$

where r_k^{ANN} is computed using Equation (3.28) in our neural network model deployment. We use the trained weights from the model to get the outputs (denoted as $\mathbf{Y}$, i.e., $y_1 = r_k^{\mathrm{ANN}}$ in our study) for corresponding inputs of each time integration steps. We note that we normalize the computed ROM variables before using them as inputs for ELM training set to stabilize the system from any kind of numerical errors or biases,

and to ensure that all the input variables are at a similar range. The normalized input sets in our study can be expressed as

$$\mathbf{X} = \left[k, \frac{a_k}{||\mathbf{a}||}, \frac{r_k^{\mathrm{LT}}}{||\mathbf{r}^{\mathrm{LT}}||}, \frac{r_k^{\mathrm{NT}}}{||\mathbf{r}^{\mathrm{NT}}||}, \frac{r_k^{\mathrm{GP}}}{||\mathbf{r}^{\mathrm{GP}}||}\right], \tag{3.41}$$

where

$$||\mathbf{g}|| = \sqrt{\frac{1}{R}\sum_{k=1}^{R}(g_k)^2}, \quad \text{where } g = \mathbf{a}, \mathbf{r}^{\mathrm{LT}}, \mathbf{r}^{\mathrm{NT}}, \mathbf{r}^{\mathrm{GP}}. \tag{3.42}$$

Finally, we normalize our training set input and output in $(\max, \min) \in [+1, -1]$ limit as

$$\tilde{\mathbf{X}} = \frac{2\mathbf{X} - (\mathbf{X}_{max} + \mathbf{X}_{min})}{(\mathbf{X}_{max} - \mathbf{X}_{min})}, \tag{3.43}$$

$$\tilde{\mathbf{Y}} = \frac{2\mathbf{Y} - (\mathbf{Y}_{max} + \mathbf{Y}_{min})}{(\mathbf{Y}_{max} - \mathbf{Y}_{min})}. \tag{3.44}$$

The physical output, $\mathbf{Y}$ is then recovered from the ANN predicted output, $\tilde{\mathbf{Y}}$, by following unnormalization method for the normalized data:

$$\mathbf{Y} = \frac{\tilde{\mathbf{Y}}(\mathbf{Y}_{max} - \mathbf{Y}_{min}) + (\mathbf{Y}_{max} + \mathbf{Y}_{min})}{2}. \tag{3.45}$$

This normalization ensures that the input variables lie in the data range of the tan-sigmoid activation function, i.e., -1 to $+1$ (Huang and Babri, 1998; Sun et al., 2008).

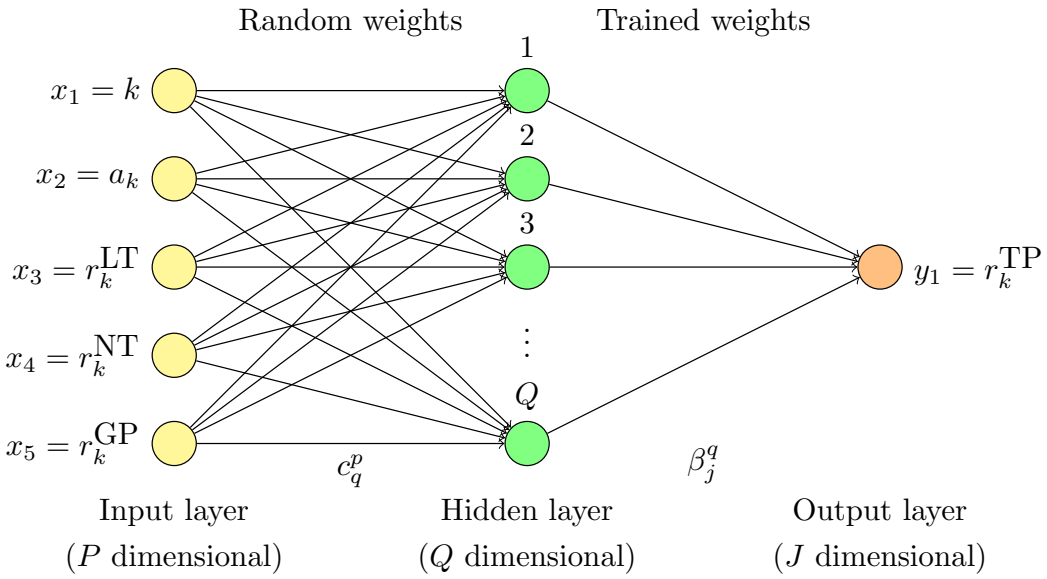
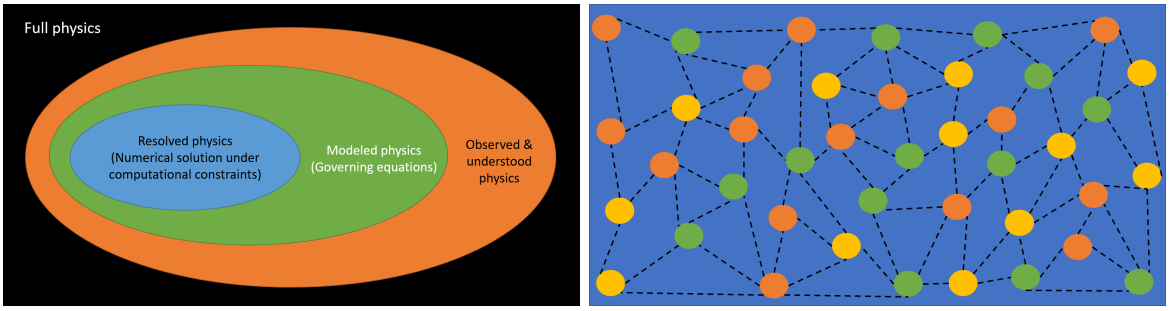


Figure 3.1: Schematic representation of the extreme learning machine (ELM) neural network architecture utilized for the data-driven reduced-order modeling framework in this study.

3.6 Hybrid Modeling (ROM-GP + ROM-ANN) Based Reduced-Order Model (ROM-H)

Hybrid modeling, as a combination of physics-based and data-driven modeling, is a fast emerging modeling approach in fluids community since both of these distinct modeling approaches are well-established in the community at this point. As illustrated in the left part of Figure 3.2, the traditional physics-based modeling approach can be described as understanding a part of the full physics (full physics is represented by the black colored region whereas the orange colored region displays the understood physics), developing a mathematical understanding through equations (green colored region in the figure), and finally resolving some part of the true physics (represented by the blue colored region) by using further assumptions and computational constraints. On the other hand, data-driven modeling works as a black box where we make an estimation of the true

solution by using available observed data snapshots without having any knowledge of the governing physics. For model order reduction, the physics-based approach is widely applied as mentioned in previous sections, and nowadays, with the rapid growth of the availability of data, data-driven approach has become a popular route for reduced-order modeling. A comparative representation of these two approaches has been outlined in Table 3.2. Considering the pros and cons of both sides, our hybrid reduced-order modeling (ROM-H) comes up with the idea to address the negative aspects of both approaches, whereas the positive aspects can be leveraged simultaneously. Similar to any kind of hybrid approach, the primary goal of this paper is to establish a robust hybrid ROM framework which aims at making the model more efficient as compared to either the physics-based or data-driven ROM models.



(a) (b)

Figure 3.2: A schematic representation of the two distinct modeling approaches: (a) physics-based modeling approach; and (b) data-driven modeling approach.

Table 3.2: Physics-based ROM vs. data-driven ROM.

	Physics-Based Modeling (ROM-GP)		**Data-Driven Modeling (ROM-ANN)**
+	Solid foundation based on physics, first principles and reasoning (high interpretability)	−	Thus far, most of the algorithms have worked as black boxes (low interpretability)
−	Difficult to assimilate very long-term historical/archival data into the computational models	+	Takes into account long-term historical/archival data and experiences
−	Sensitive and susceptible to numerical instability due to a range of reasons (boundary conditions, uncertainties in the input parameters and meshing)	+	Once the model is trained, it is very stable for making predictions
+	Errors/uncertainties can be bounded and estimated	−	Not quite possible to bound errors/uncertainties
+	Less biases	−	Bias in data is reflected in the prediction
+	Generalizes well to new problems with similar physics	−	Poor generalization on unseen problems

We build our ROM-H framework by introducing a free parameter to dynamically account for the contribution of the physics-based ROM and data-driven ROM. The ROM-H methodology can be summarized by following key steps:

Step 1 (offline): Generate a set of basis functions ϕ_k for $k = 1, 2, 3, ..., R$ from the snapshot data obtained from FOM.

Step 2 (offline): Apply Galerkin projection to compute the coefficients required for ROM-GP.

Step 3 (offline): Train the ELM network using the resolved ROM-GP coefficients and true projections datasets.

Step 4 (online): Compute a_k by solving the following ordinary differential equations in reduced-order space:

$$\frac{da_k}{dt} = (1-\eta) r_k^{\mathrm{GP}} + \eta r_k^{\mathrm{ANN}}, \tag{3.46}$$

where we define η as a free weighting parameter in a range of $0 \leq \eta \leq 1$ to establish a relationship between the standard ROM-GP and non-intrusive ROM-ANN models. If $\eta = 0$, we recover Equation (3.23), whereas, for $\eta = 1$, Equation (3.40) can be obtained.

Step 5 (offline): Obtain the full order solution by transferring data from reduced-order space by using Equation (3.20) as a post-processing task if needed.

The hybridization between the physics-based and data-driven based ROM happens in Step 4, where, in each time step, r_k^{GP} is calculated using Equation (3.23), and r_k^{ANN} is obtained from Equation (3.40). The parameter η gives us a freedom to benefit from both modeling techniques mentioned above rather than fully depending on a single modeling technique, and, as we show in Section 3.7, the hybrid model with optimal contribution from both modeling approaches gives a better prediction of the true solution than the individual modeling approaches. The limiting values of η will result in the standard ROM-GP and ROM-ANN model solutions, respectively, and any value between the limits of 0 to 1 will give a solution with contribution from both ROM-GP and ROM-ANN models. Based on the working physical problem, we assume that either more physics-based model contribution or more data-driven model contribution can produce a better estimation of the truth. Keeping this in mind, instead of forcing an arbitrary value to specify η for hybridization, we apply the following dynamic

optimization strategy based on the values of r_k^{GP} and r_k^{ANN} in each time integration steps:

$$\eta = \left| \tanh \left(\frac{||\mathbf{r}^{\text{GP}}|| - ||\mathbf{r}^{\text{ANN}}||}{||\mathbf{r}^{\text{ANN}}||} \right) \right|. \tag{3.47}$$

Here, we utilize the absolute value of hyperbolic tangent function, which always produces a value for η between 0 and 1. If the values of r_k^{GP} and r_k^{ANN} are closely related ($||\mathbf{r}^{\text{GP}}|| \cong ||\mathbf{r}^{\text{ANN}}||$), the value of η parameter will be 0, i.e., it indicates the use of ANN prediction for that particular case will be redundant. Similarly, if the values are far away, the ANN prediction contribution will be more dominant. However, we emphasize that the estimation technique we utilized here for η calculation is on an ad-hoc basis for the present study. On this hybrid framework, the process of finding the optimal η can be a potential topic for another research direction where optimization techniques such as trust-region method, conjugate gradient methods, or quasi-Newton methods can be used (Wright and Nocedal, 1999; Lin et al., 2008).

3.7 Numerical Results

For validation purpose, our reduced-order model resulting from the hybridization of the standard ROM-GP approach and fully non-intrusive ROM-ANN approach has been used to model a four-gyre barotropic ocean circulation problem. In this section, we present the numerical results and analyses demonstrating the computation and prediction performance of our proposed hybrid framework as well as the component models (ROM-GP and ROM-ANN) and FOM simulation for comparisons. We select this particular test problem (as many other studies do (Cushman-Roisin and Manga, 1995; Holm and Nadiga, 2003; San et al., 2011; Greatbatch and Nadiga, 2000; Cummins, 1992)) because of its complexity induced by the symmetric double-gyre wind forcing developing a four-gyre circulation pattern which makes this problem challenging

enough to assess the viability of proposed models. Based on the following results, it is apparent that the ROM-H model provides a better estimation of the true solution as compared to the individual component model solutions. Indeed, it was shown before that adding a stabilization to the standard ROM-GP improves the model prediction and resolve more mean dynamics (San and Iliescu, 2015). However, our proposed hybrid model idea is data-driven without taking any additional phenomenological arguments (e.g., eddy viscosity) to stabilize the model, and can be applied to most of the flow phenomena. On the other hand, we would like to note that the ROM-GP model can be cured with nonlinear modal eddy viscosities, i.e., with an energy-flow correction (Östh et al., 2014) or other closure approaches (Wang et al., 2012; San and Iliescu, 2014).

3.7.1 Case setup specifications for FOM simulations

In the present work, we consider the dimensionless form of the BVE describing the QG problem in the Cartesian domain $(x, y) \in [0, 1] \times [-1, +1]$ as a reference. The FOM simulation is conducted starting from $t = 0$ to $t = 60$ using a fixed time step of $\Delta t = 1 \times 10^{-4}$ on a Munk layer resolving 128×256 grid resolution. We collect 900 data snapshot from time $t = 15$ to $t = 60$ to avoid collecting data from initial transition period for POD basis generation and ELM training, i.e., the sampling time Δs is 5×10^{-2}. We use dimensionless parameters Re $= 25, 100, 400$ and constant Ro $= 3.6 \times 10^{-3}$ for all the simulations throughout the paper.

3.7.2 Analysis of the standard ROM-GP method

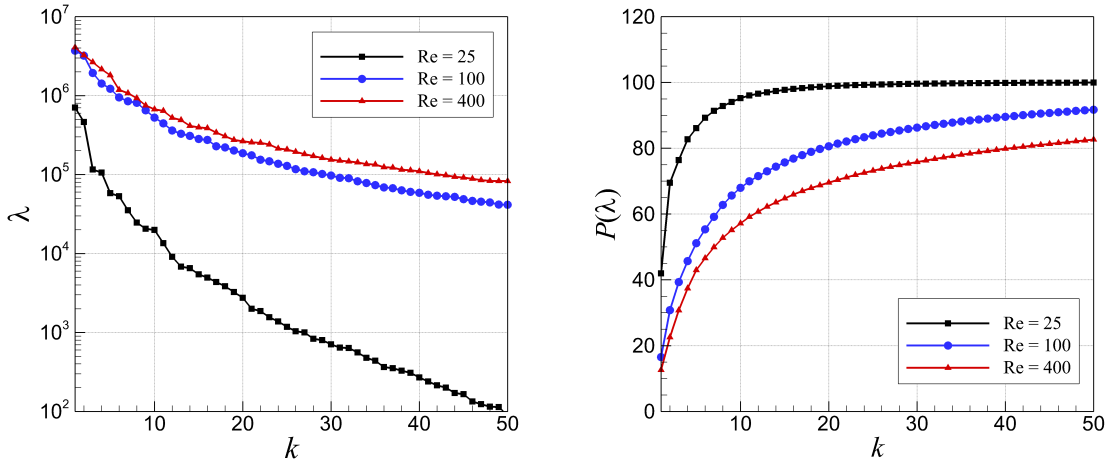
We first construct the POD data correlation matrix using the snapshots data from FOM simulation to generate the POD basis function for all the ROM approach in this paper. Hence, it is important to observe the characteristics and distribution of the eigenvalues with respect to modal index as presented in Figure 3.3. To estimate

the total kinetic energy accumulation of the system, we compute the percentage of each eigenvalue by

$$P(\lambda_k) = \left(\frac{\sum_{j=1}^{k} \lambda_j}{\sum_{j=1}^{N_s} \lambda_j} \right) \times 100, \tag{3.48}$$

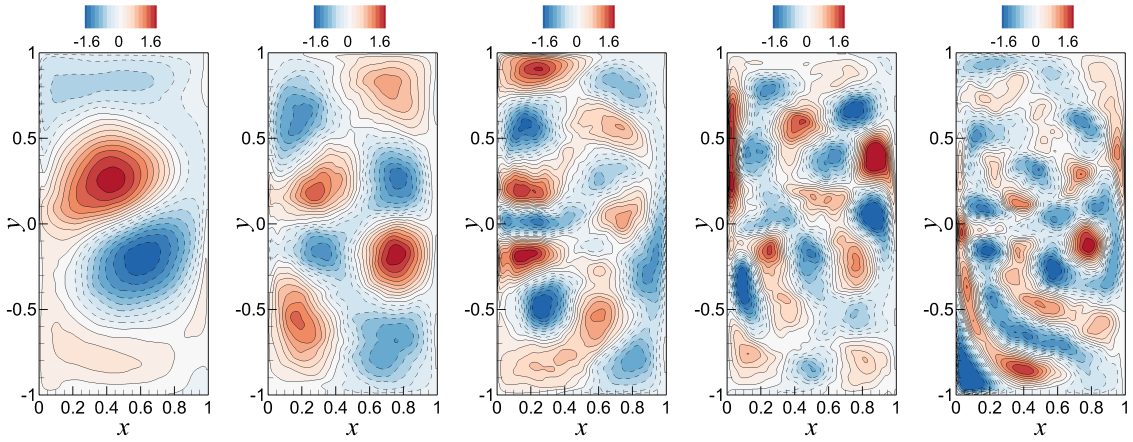
where $N_s = 900$ is the total number of snapshots. Since higher Re incorporates more stratification or, to be specific, as more energy is added to the system, more modes are required to capture majority percentage of the energies in the system. Figure 3.3 shows that, while only around 10 modes is enough to capture most of the system's kinetic energy for Re = 25, a large percentage of energies (close to 85% and 75%) are accumulated in the first 30 and 40 modes for Re = 100 and Re = 400 cases, respectively. However, the percentage of captured energy is lower in higher Re cases which indicates the challenges associated with model order reduction of this test case in higher Re. These findings are supported by Figure 3.4 where we present the first few POD basis functions, the most important ones regarding the captured information, for Re = 100. We can identify the increase in the equal number of smaller structures (maxima and minima) which correspond to higher POD indices. Using more POD bases will result in better approximation until the corresponding eigenvalue stagnation point reaches. Based on the POD analysis, we develop the rest of our studies using POD modes up to 40 to incorporate most of the physics within the system.

Figures 3.5–3.7 show the mean stream function field evolution with respect to increasing modes, R using standard ROM-GP approach for Re $= 25, 100, 400$, respectively. It is apparent in Figure 3.5 that, for Re = 25 case, even $R = 10$ shows a good prediction of the FOM solution. We can also notice in the figure that the mean stream function plot shows a consistent result for $R = 20$ with respect to $R = 10$ result, but we get a worse result for $R = 30$. This can be explained as an over-fitting, as we have seen in Figure 3.3 that, for Re = 25, $R = 20$ modes contain more than 99% of the system's total kinetic energy. It has also been noted in (Giere



(a) (b)

Figure 3.3: Proper orthogonal decomposition (POD) analysis by using 900 equally distributed snapshots for Re $= 25, 100, 400$: (a) Eigenvalue spectrum of the correlation matrix, $\mathbf{A}$; and (b) Eigenvalue percentage energy accumulation with respect to modal index.



(a) (b) (c) (d) (e)

Figure 3.4: Contour plots of some illustrative examples of POD basis functions for Re $= 100$ generated by the method of snapshots: (a) $\varphi_1(x, y)$; (b) $\varphi_{10}(x, y)$; (c) $\varphi_{20}(x, y)$; (d) $\varphi_{30}(x, y)$; and (e) $\varphi_{40}(x, y)$.

et al., 2015) that adding more modes might yield worse solutions due to the noise in the system. The inaccurate result for $R = 30$ is understandable in a sense that the previous modes contain most of the energies (more than 99%) of the system which is considered a sufficient total kinetic energy for the optimal number of modes in most of the POD studies (Cordier et al., 2010). Even though $R = 40$ modes shows a very good estimation of FOM result, it is recommended not to use more than the optimal number of POD modes to avoid instabilities in the solution field. This occurrence can also be seen in Figures 3.8 and 3.9 for Re = 25. On the other hand, for Re = 100 case, $R = 10$ shows a nonphysical two-gyre circulation pattern in Figure 3.6 similar to the plots of coarse truncated modes ($R = 10$ and $R = 20$) for Re = 400 case in Figure 3.7. We further illustrate the effect of R and Re for ROM-GP approach in the prediction of true solution in Figures 3.8 and 3.9. For the purpose of comparison, we include the FOM data projection to reduced-order space using corresponding POD basis functions for Re $= 25, 100, 400$. In Figure 3.8, we show the non-dimensional time evolution of the first modal coefficient, $a_1(t)$ in reduced-order space for different Re and R. The plot supports the conclusions drawn from the POD analysis and the mean stream function plots that for Re = 100, 10 POD modes provide an unphysical result. For even higher Re = 400, 20 POD modes show the numerical instabilities as we can see a sudden burst of random oscillations in much higher scales. To put extra emphasis on our findings, we present the results for tenth modal coefficient in Figure 3.9 where we can observe the same scenario as Figure 3.8 on a different scale. We can notice that the scales for tenth modal coefficient are much smaller than the first modal coefficient time series which is expected for higher modes, but the performance of the ROM-GP model is consistent.

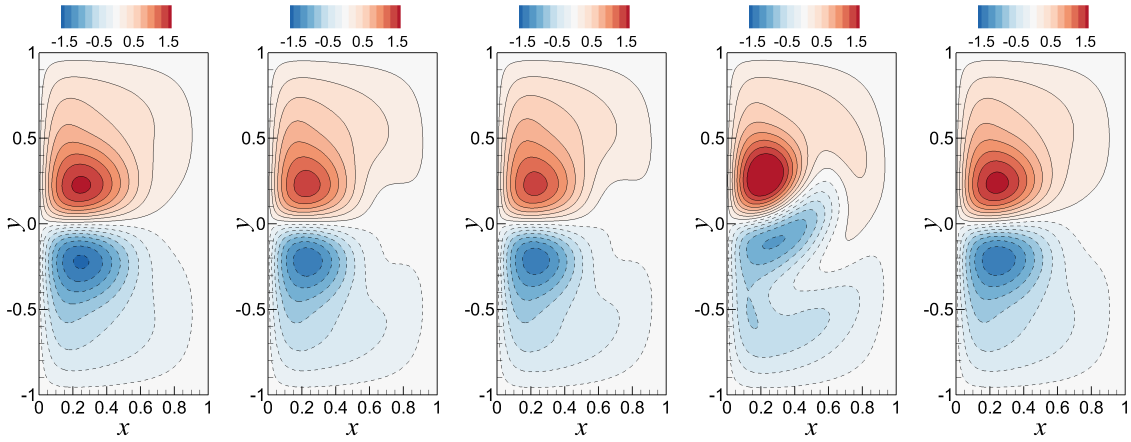


Figure 3.5: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and standard ROM-GP simulations for Re $= 25$: (a) ψ_{FOM} at a resolution of 128×256; (b) $\psi_{\text{ROM-GP}}$ with $R = 10$ modes; (c) $\psi_{\text{ROM-GP}}$ with $R = 20$ modes; (d) $\psi_{\text{ROM-GP}}$ with $R = 30$ modes; and (e) $\psi_{\text{ROM-GP}}$ with $R = 40$ modes. Note that a stable and well-estimated solution can be found using $R = 10$ modes and adding more modes might yield worse solutions due to a possible over-fitting as occurred in $R = 30$ case.

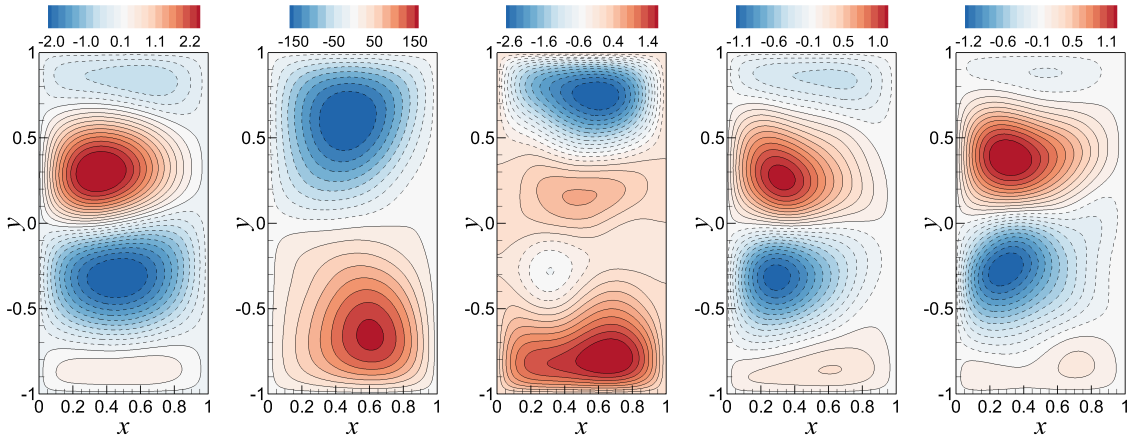


Figure 3.6: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and standard ROM-GP simulations for Re $= 100$: (a) ψ_{FOM} at a resolution of 128×256; b) $\psi_{\text{ROM-GP}}$ with $R = 10$ modes; (c) $\psi_{\text{ROM-GP}}$ with $R = 20$ modes; (d) $\psi_{\text{ROM-GP}}$ with $R = 30$ modes; and (e) $\psi_{\text{ROM-GP}}$ with $R = 40$ modes. Note that a stable and well-estimated solution can be found within $R = 30$ or 40 modes for this case.

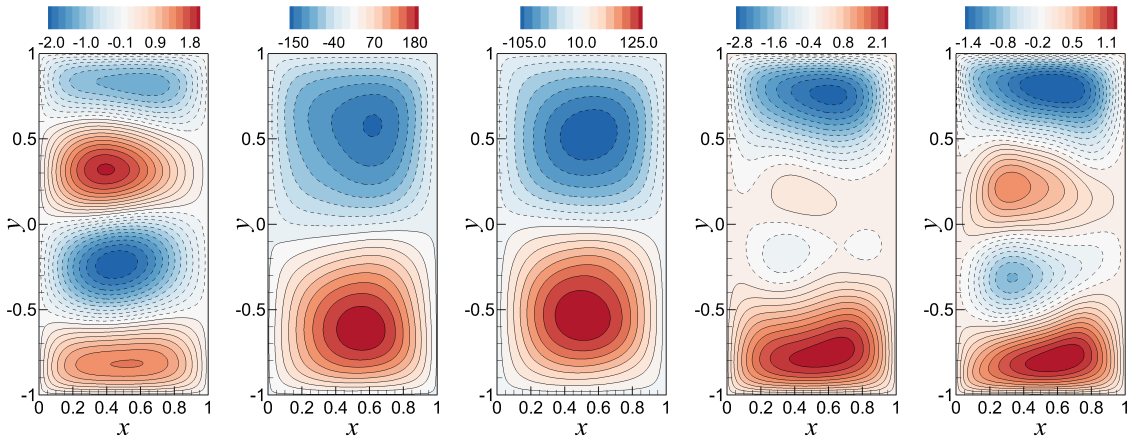


Figure 3.7: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and standard ROM-GP simulations for Re $= 400$: (a) ψ_{FOM} at a resolution of 128×256; (b) $\psi_{\text{ROM-GP}}$ with $R = 10$ modes; (c) $\psi_{\text{ROM-GP}}$ with $R = 20$ modes; (d) $\psi_{\text{ROM-GP}}$ with $R = 30$ modes; and (e) $\psi_{\text{ROM-GP}}$ with $R = 40$ modes. Note that a stable and well-estimated solution can be found within $R = 40$ modes for this case.

3.7.3 Assessments of the prediction performance of ROM-GP, ROM-ANN, ROM-H

We observed the inability of the standard ROM-GP model to estimate the correct physics using coarse truncated modes ($R = 10$ or 20) for Re $= 400$ in Section 3.7.2. For this reason, we consider $R = 10$ modes only for the rest of the analyses to assess the predictive performances of the models. As we can see in Figure 3.10 for Re $= 25$, all models capture the two-gyre pattern, which is consistent with the true solution as well as previous observations. Similar to Section 3.7.2, the ROM-GP model fails to capture four-gyre pattern for Re $= 100$ and $R = 10$ combination, as shown in Figure 3.11, but the ROM-ANN and ROM-H models exhibit a successful prediction. Nevertheless, Figure 3.12 demonstrates the most promising findings that the ROM-H model successfully predicts the four-gyre pattern while the other two basic ROM approaches fail for Re $= 400$ and $R = 10$ combination. To support this finding, we again plot the time series evolution for both a_1 and a_{10}, as presented in Figures 3.13

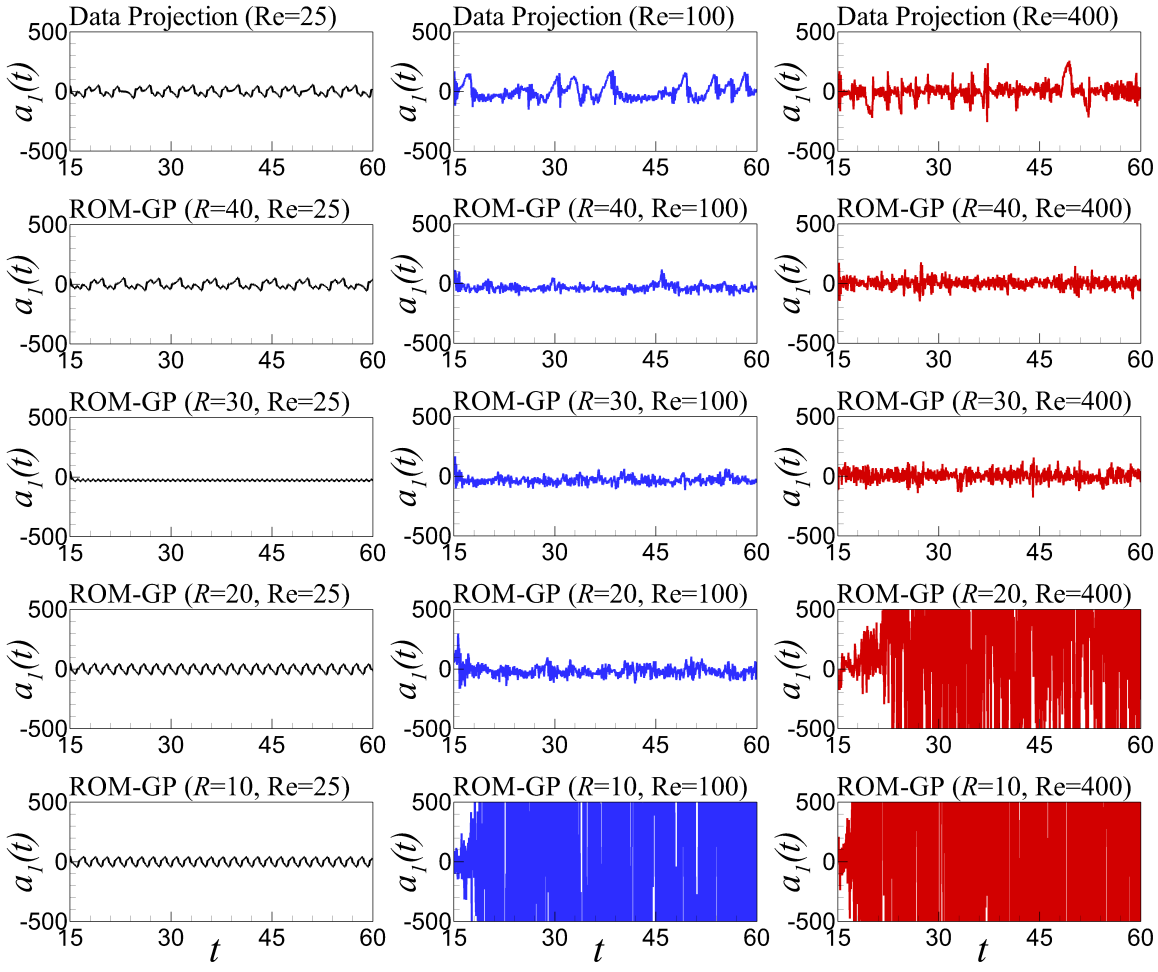


Figure 3.8: Time series evolution of the first modal coefficient, $a_1(t)$ between $t = 15$ and $t = 60$ for FOM projection and standard ROM-GP with varying Re ($\mathrm{Re} = 25, 100, 400$) and POD modes ($R = 10, 20, 30, 40$). $R = 20$ modes for $\mathrm{Re} = 100$, and $R = 30$ modes for $\mathrm{Re} = 400$.

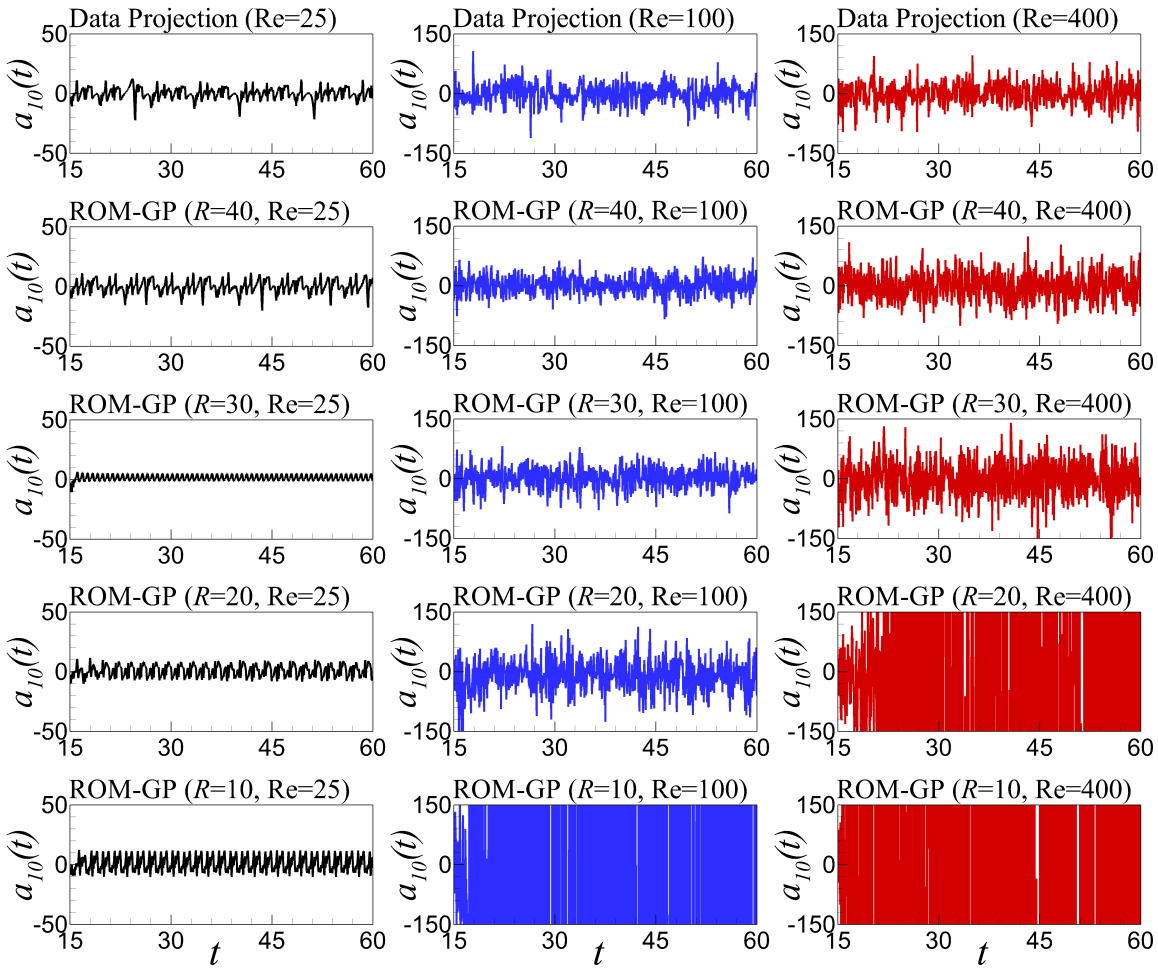


Figure 3.9: Time series evolution of the tenth modal coefficient, $a_{10}(t)$ between $t = 15$ and $t = 60$ for FOM projection and standard ROM-GP with varying Re (Re $= 25, 100, 400$) and POD modes ($R = 10, 20, 30, 40$).

and 3.14, to find that the ROM-H model shows an impressive prediction of the FOM projected time series for even higher Re. The prediction of ROM-ANN model is better than the ROM-GP model using same values for the parameters, but not as accurate as the ROM-H model for Re = 400. As we can see, the ROM-H model is capable of capturing the physics utilizing only 10 modes, which leads to an excellent development of a more efficient ROM than the standard ROM-GP, which requires higher modes to capture the physics for Re = 400. A summary of computational overhead is presented in Table 3.3.

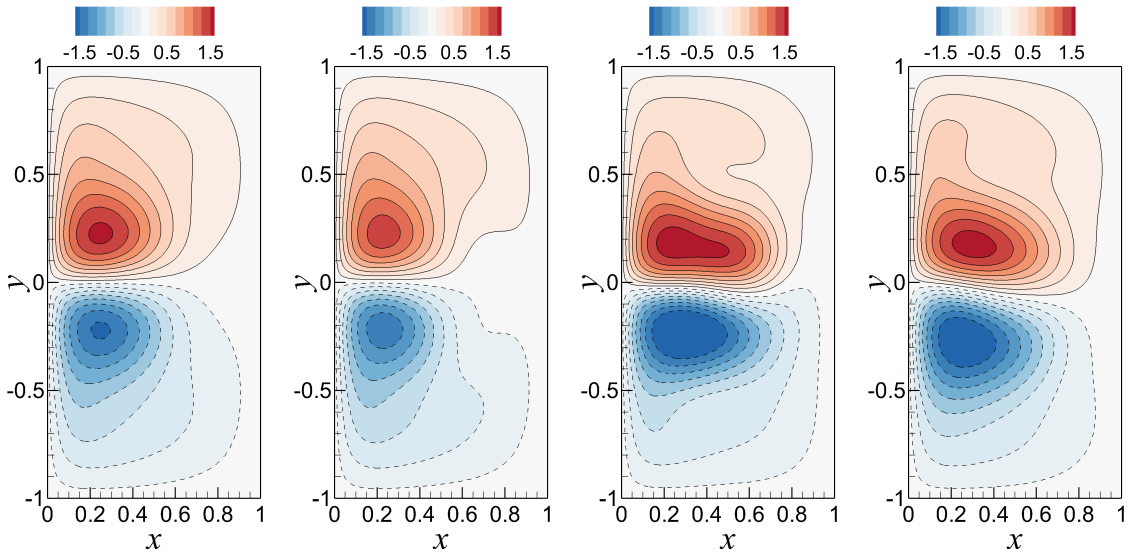


Figure 3.10: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and different ROM simulations for Re = 25 and $R = 10$ modes: (a) ψ_{FOM} at a resolution of 128×256; (b) $\psi_{\text{ROM-GP}}$; (c) $\psi_{\text{ROM-ANN}}$; and (d) $\psi_{\text{ROM-H}}$. This figure clearly illustrates that all models have captured the two-gyre circulation pattern successfully.

3.7.4 Sensitivity analysis with respect to ELM neurons

Based on the results in Section 3.7.3, we can state that the prediction performance of our proposed ROM-H model is better than the ROM-ANN model for this test case. However, sensitivity analysis of the ANN architecture should be performed before

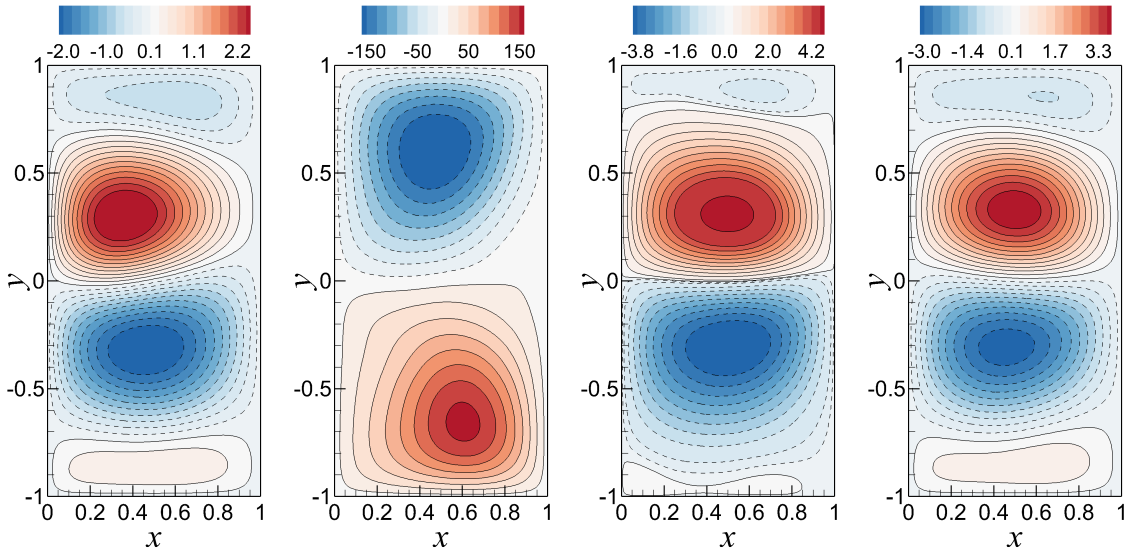


Figure 3.11: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and different ROM simulations for Re $= 100$ and $R = 10$ modes: (a) ψ_{FOM} at a resolution of 128×256; (b) $\psi_{\text{ROM-GP}}$; (c) $\psi_{\text{ROM-ANN}}$; and (d) $\psi_{\text{ROM-H}}$. The figure shows that ROM-ANN and ROM-H have captured the four-gyre circulation pattern similar to the FOM solution while ROM-GP has shown an unphysical two-gyre pattern.

Table 3.3: The computational CPU time in seconds required for ROM simulations between $t = 15$ and $t = 60$. Note that the FOM simulation is 4.82 h and offline POD basis generation takes about 23.38 min. The offline training time for ANN is less than one second due to the extremely fast ELM approach.

	$\boldsymbol{R = 10}$	$\boldsymbol{R = 20}$	$\boldsymbol{R = 30}$	$\boldsymbol{R = 40}$
Pre-computing time for the inner products	1.06	4.37	11.10	22.93
ROM-GP simulation time	5.56	41.51	153.68	380.54
ROM-ANN simulation time	16.41	96.73	209.16	428.51
ROM-H simulation time	21.29	92.67	273.70	496.32

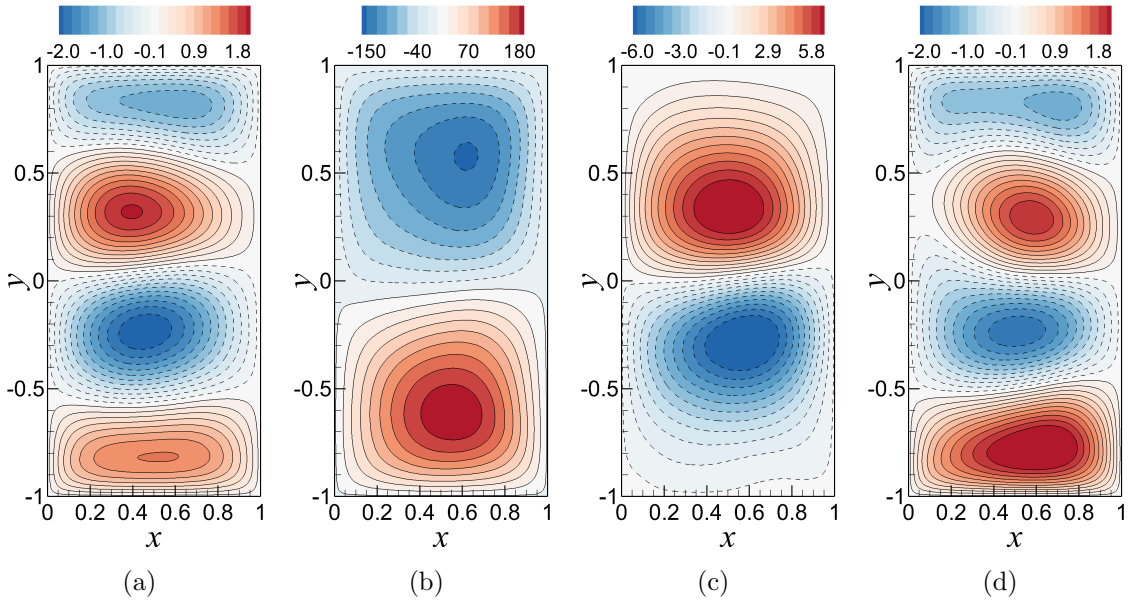


Figure 3.12: Mean stream function contour plots between $t = 15$ and $t = 60$ obtained by reference FOM and different ROM simulations for Re $= 400$ and $R = 10$ modes: (a) ψ_{FOM} at a resolution of 128×256; (b) $\psi_{\text{ROM-GP}}$; (c) $\psi_{\text{ROM-ANN}}$; and (d) $\psi_{\text{ROM-H}}$. Note that only the proposed ROM-H model has successfully captured the four-gyre circulation.

reaching any conclusion to verify whether the network output alters with the change in the total number of neurons in hidden layer, Q (Montano and Palmer, 2003). Because of the black box nature of ANN, we perform a sensitivity analysis with respect to $Q = 20, 40, 80$ neurons for both ROM-ANN and ROM-H models. It can be clearly seen in Figure 3.15 of mean stream function results that the ROM-ANN model is showing similar performance improvements for all different number of neurons for $R = 10$ and Re $= 100$ cases. Furthermore, in Figure 3.16, the ROM-H model is showing an excellent agreement with FOM solution for all different number of neurons. On the other hand, the time series of modal coefficient plots for ROM-ANN model in Figures 3.17 and 3.18 exhibit some noticeably unusual traits in a number of cases, for example, a_1 ($Q = 40$, Re $= 25$), a_1 ($Q = 20$, Re $= 100$), a_{10} ($Q = 40$, Re $= 25$). In addition, the ROM-H model displays satisfactory consistency in its predictive behavior with respect to the variation of Q, as shown in Figures 3.19 and 3.20. This suggests

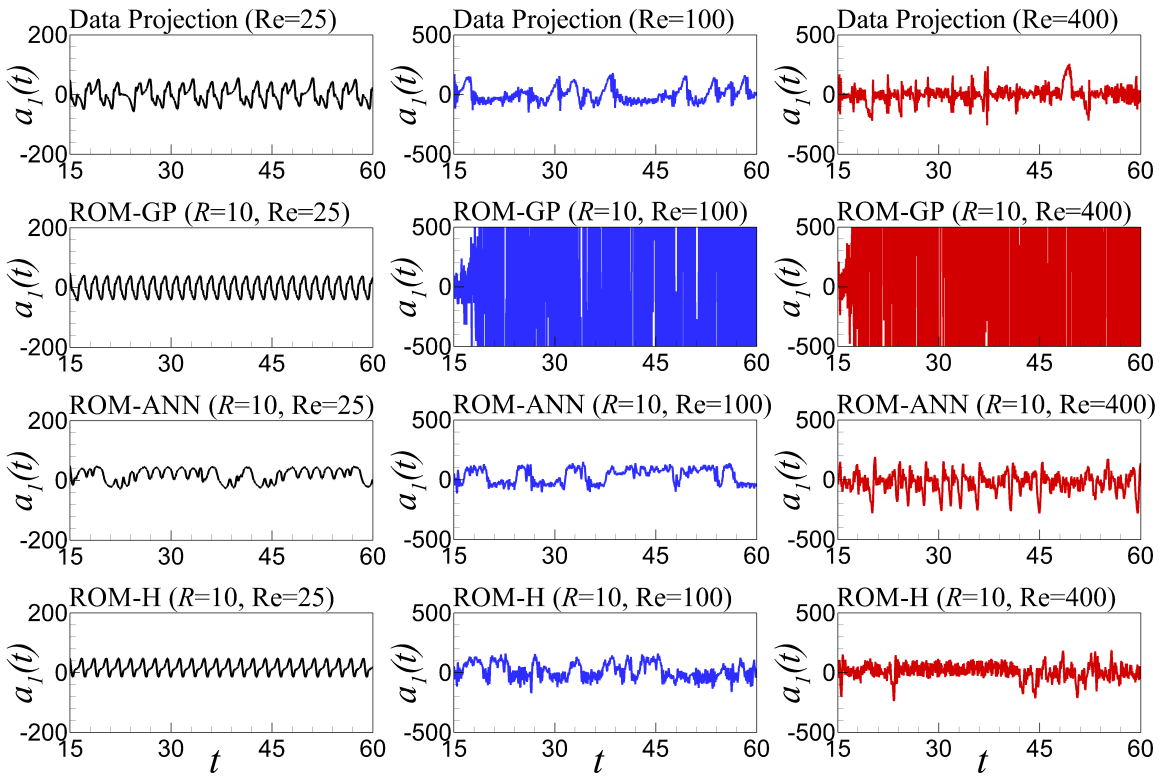


Figure 3.13: Time series evolution of the first modal coefficient, $a_1(t)$, between $t = 15$ and $t = 60$ for FOM projection and different ROM approaches (ROM-GP, ROM-ANN, and ROM-H) with varying Re ($\mathrm{Re} = 25, 100, 400$) and POD modes ($R = 10, 20, 30, 40$).

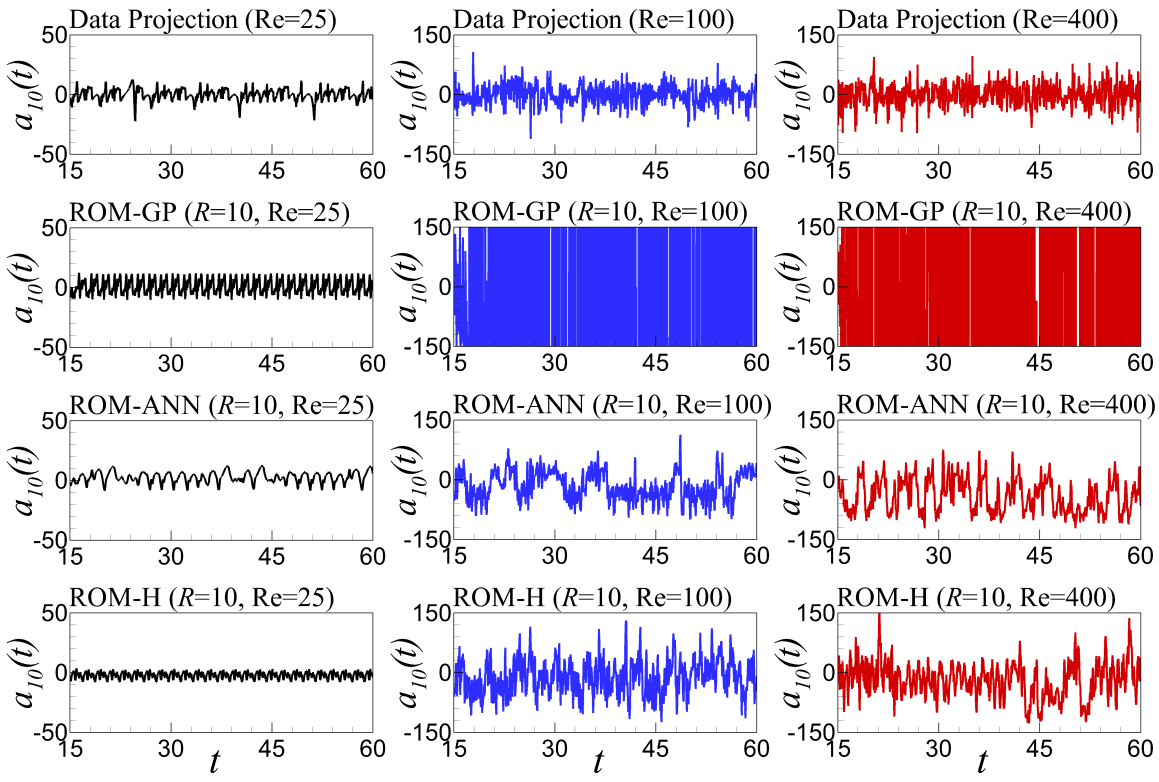


Figure 3.14: Time series evolution of the tenth modal coefficient, $a_{10}(t)$, between $t = 15$ and $t = 60$ for FOM projection and different ROM approaches (ROM-GP, ROM-ANN, and ROM-H) with varying Re ($\mathrm{Re} = 25, 100, 400$) and POD modes ($R = 10, 20, 30, 40$).

that the ROM-H model is robust with respect to the change in total number of neurons included in the hidden layer.

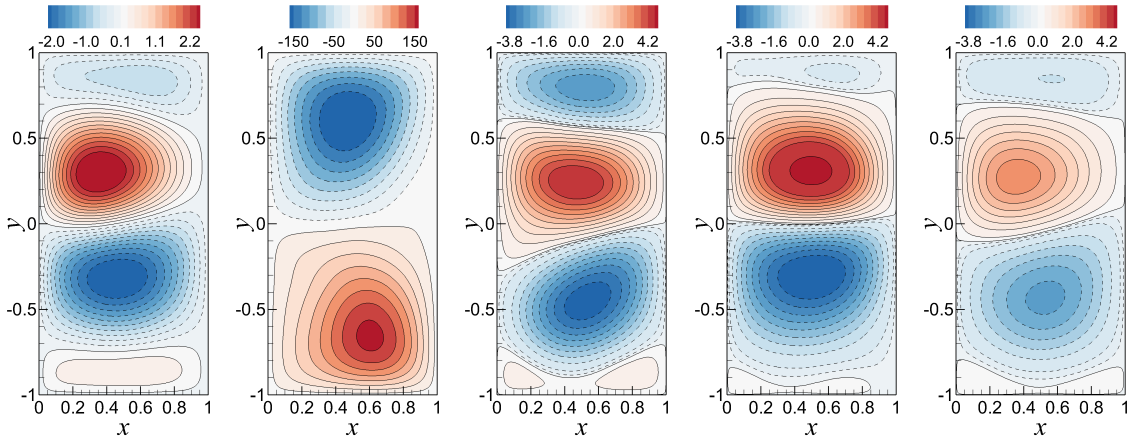


Figure 3.15: Sensitivity analysis with respect to the number of neurons in ELM of fully non-intrusive ROM-ANN approach using mean stream function contour plots ($R = 10$ and $\mathrm{Re} = 100$): (a) ψ_{FOM}; (b) $\psi_{\mathrm{ROM\text{-}GP}}$; (c) $\psi_{\mathrm{ROM\text{-}ANN}}$ with $Q = 20$ neurons; (d) $\psi_{\mathrm{ROM\text{-}ANN}}$ with $Q = 40$ neurons; and (e) $\psi_{\mathrm{ROM\text{-}ANN}}$ with $Q = 80$ neurons.

3.7.5 Time series evolution and out-of-sample forecasting

To further illustrate the robustness and enhanced potential of the proposed model, here we present a couple more analyses on the ROM-GP, ROM-ANN, and ROM-H approaches. In Figure 3.21, we present the temporal evolution of a_1 obtained by ROM-GP, ROM-ANN and ROM-H model along with FOM projection for $\mathrm{Re} = 25$. The figure displays the forecasting capability of ROM models for a long period of time up to $t = 180$ even though the data snapshots were taken between $t = 15$ and $t = 60$. For $\mathrm{Re} = 100$ and 400 in Figures 3.22 and 3.23, respectively, the ROM-ANN and ROM-H models still show a stable time series results, whereas the ROM-GP prediction becomes unphysical. The long time forecasting can be a very useful property of the ROM-H model to reduce the computational cost, and can be applied for practical applications related to forecasting. An out-of-sample a posteriori analysis is also performed where the snapshot data and training dataset are prepared at a particular physical condition to predict the performance

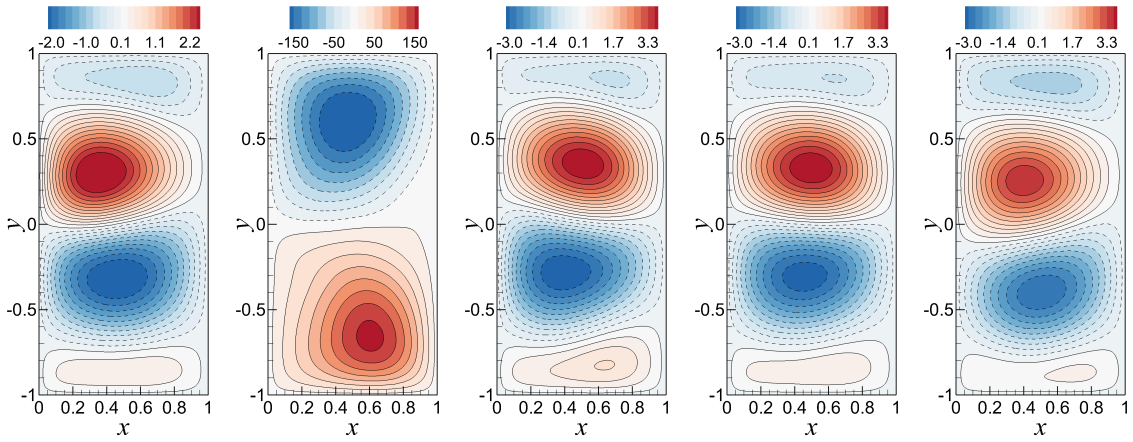


Figure 3.16: Sensitivity analysis with respect to the number of neurons in ELM of ROM-H approach using mean stream function contour plots ($R = 10$ and $\mathrm{Re} = 100$): (a) ψ_{FOM}; (b) $\psi_{\text{ROM-GP}}$; (c) $\psi_{\text{ROM-H}}$ with $Q = 20$ neurons; (d) $\psi_{\text{ROM-H}}$ with $Q = 40$ neurons; and (e) $\psi_{\text{ROM-H}}$ with $Q = 80$ neurons.

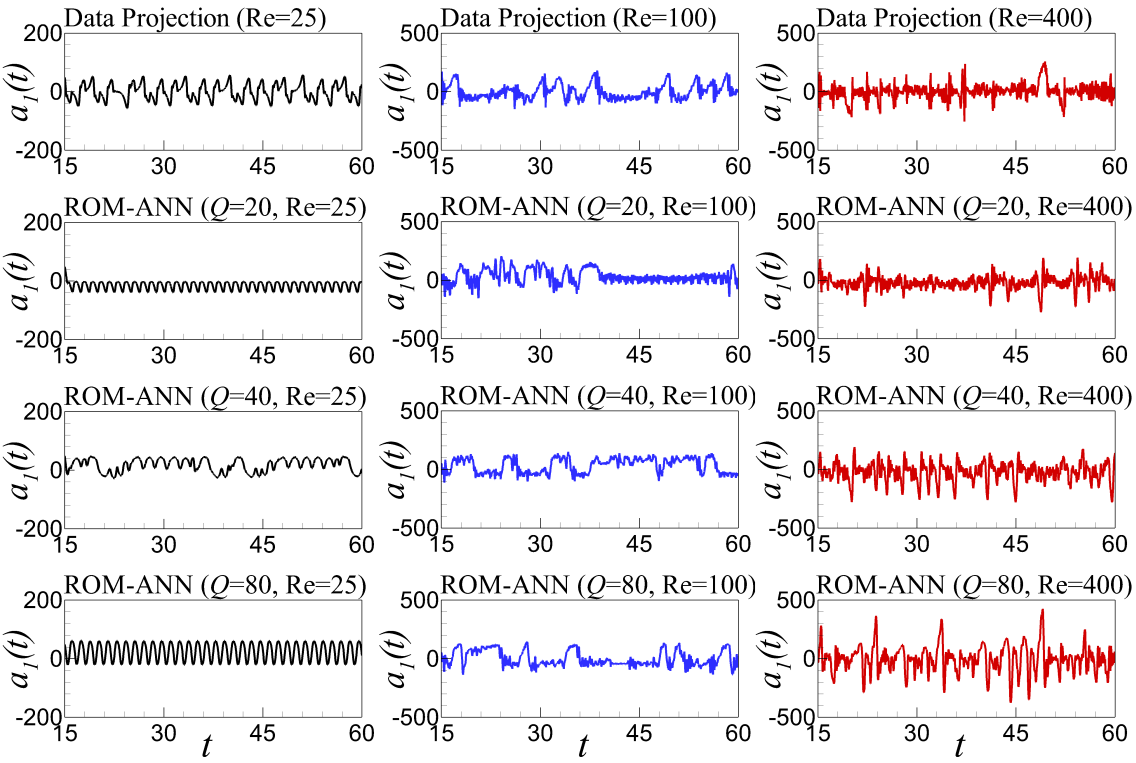


Figure 3.17: Sensitivity analysis with respect to the number of ELM neurons of fully non-intrusive ROM-ANN approach using time series of the first modal coefficient, $a_1(t)$ for $\mathrm{Re} = 25, 100, 400$ and $Q = 20, 40, 80$.

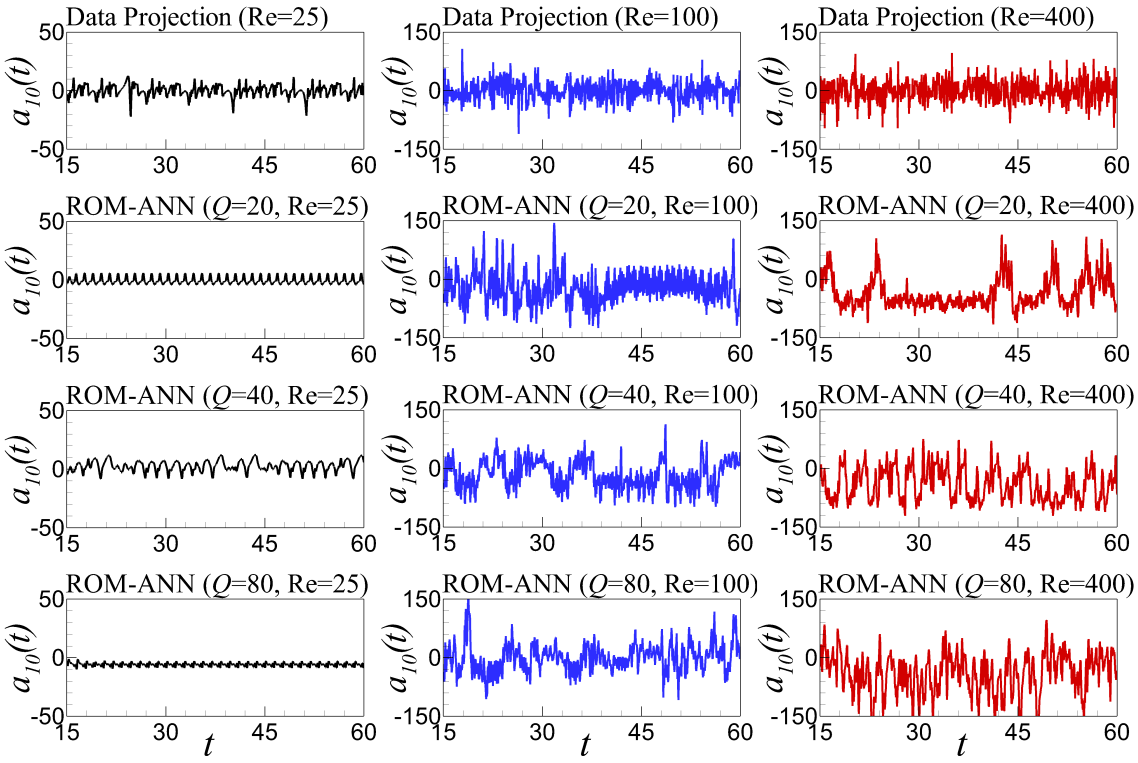


Figure 3.18: Sensitivity analysis with respect to the number of ELM neurons of fully non-intrusive ROM-ANN approach using time series of the tenth modal coefficient, $a_{10}(t)$ for Re $= 25, 100, 400$ and $Q = 20, 40, 80$.

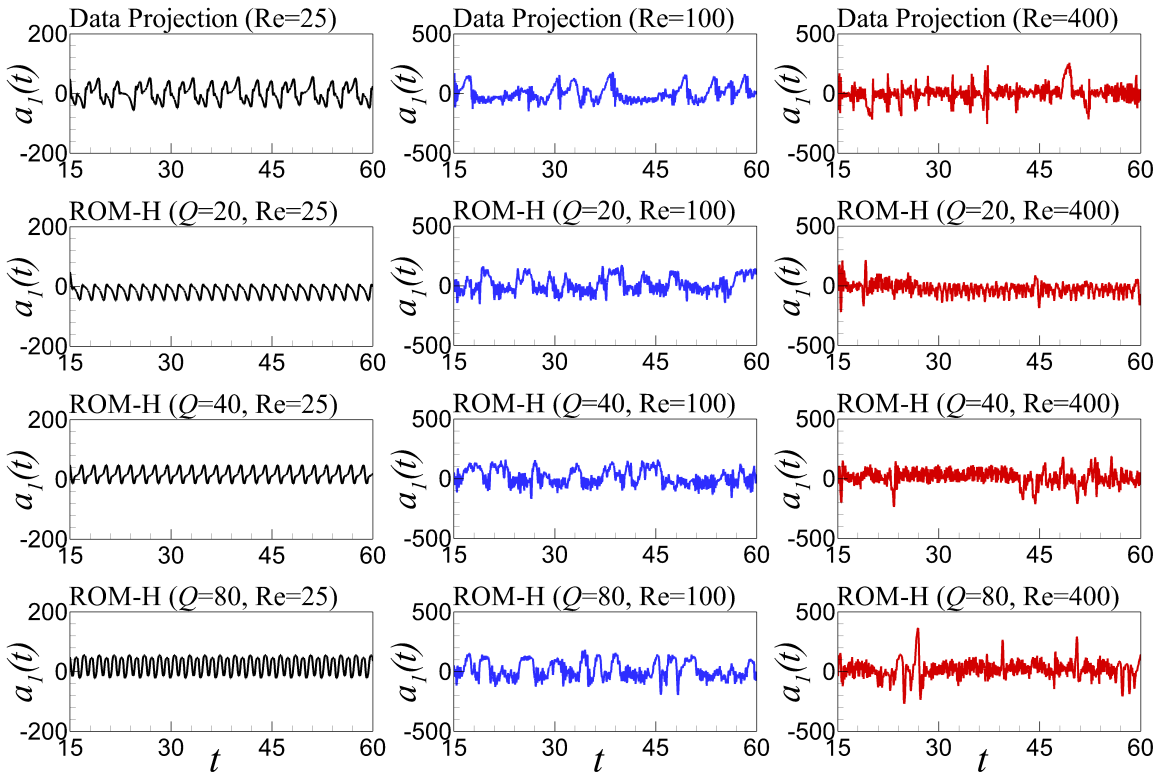


Figure 3.19: Sensitivity analysis with respect to the number of ELM neurons of ROM-H approach using time series of the first modal coefficient, $a_1(t)$ for $\mathrm{Re} = 25, 100, 400$ and $Q = 20, 40, 80$.

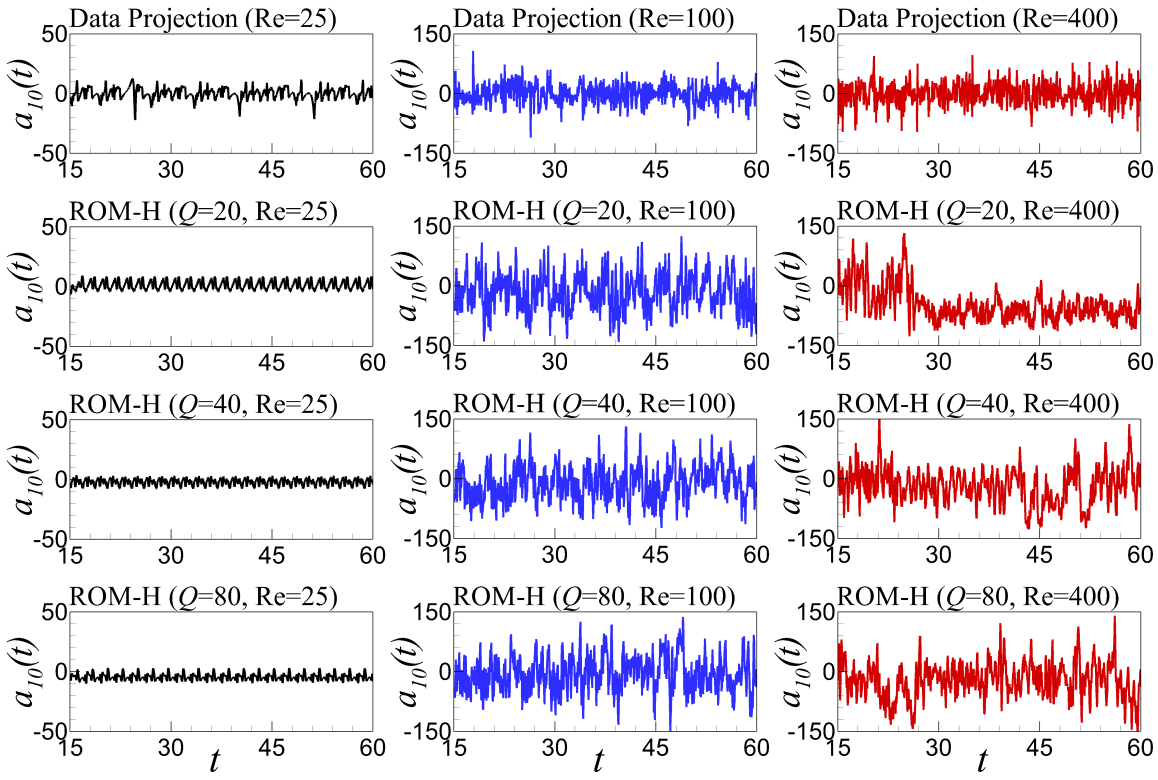


Figure 3.20: Sensitivity analysis with respect to the number of ELM neurons of ROM-H approach using time series of the tenth modal coefficient, $a_{10}(t)$ for $\mathrm{Re} = 25, 100, 400$ and $Q = 20, 40, 80$.

of the models at different physical condition. In this study, we conduct two out-of-sample analyses: in one case, we generate the snapshots and training data at Re = 100 (lower value than the target prediction state) to predict the performance at Re = 200 and, in the other case, we generate the snapshots and training data at Re = 400 (higher value than the target prediction state) to predict the performance at Re = 200. As displayed in Figure 3.24 for the time series of a_1, the ROM-GP again fails to capture the underlying physics for R = 10. The ROM-ANN prediction seems not so random but still does not represent the true statistics. However, the ROM-H prediction series matches the scales of the time series of FOM data projection. We can get a clear idea of this quantitative analysis by observing the visual representation of the mean stream function field in Figure 3.25. It is apparent that only the ROM-H prediction matches the FOM result since the underlying four-gyre dynamics is captured. Finally, we analyze the prediction performance at Re = 200 by using the data collected at Re = 400. The time series plot in Figure 3.26 reveals that neither ROM-GP nor ROM-ANN can capture the true scales while once again, the ROM-H model shows an impressive prediction of the true physics at Re = 200 using Re = 400 data. Figure 3.27 shows the mean field representation which indicates the ROM-H model is capable of out-of-sample forecasting using data from higher Re. These analyses of longer time forecasting and out-of-sample validation puts more strength on the conclusion drawn above that the proposed ROM-H method is more robust in predicting the underlying full order physics than the fully non-intrusive ROM-ANN and standard ROM-GP methods.

3.8 Summary

In this work, we propose a reliable and robust reduced-order modeling technique using a hybrid framework. The hybridization is done by combining the standard Galerkin projection based ROM and ELM neural network architecture. We consider an ocean circulation problem governed by a single-layer quasi-geostrophic model as

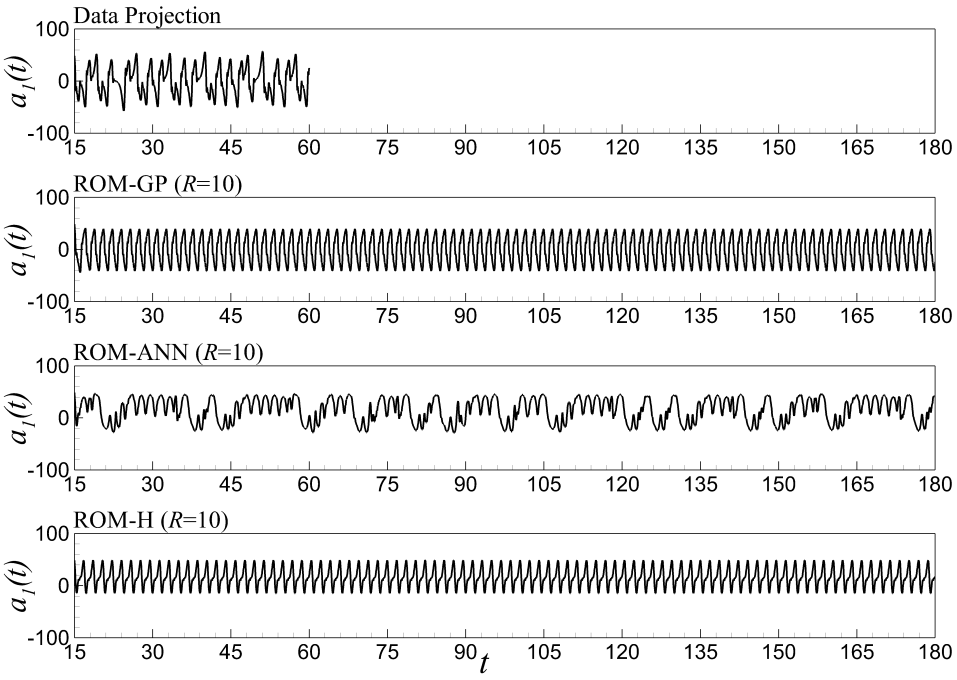


Figure 3.21: Forecasting of temporal mode evolution of $a_1(t)$ for Re = 25. Note that ROMs are trained by the data snapshots obtained between $t = 15$ and $t = 60$.

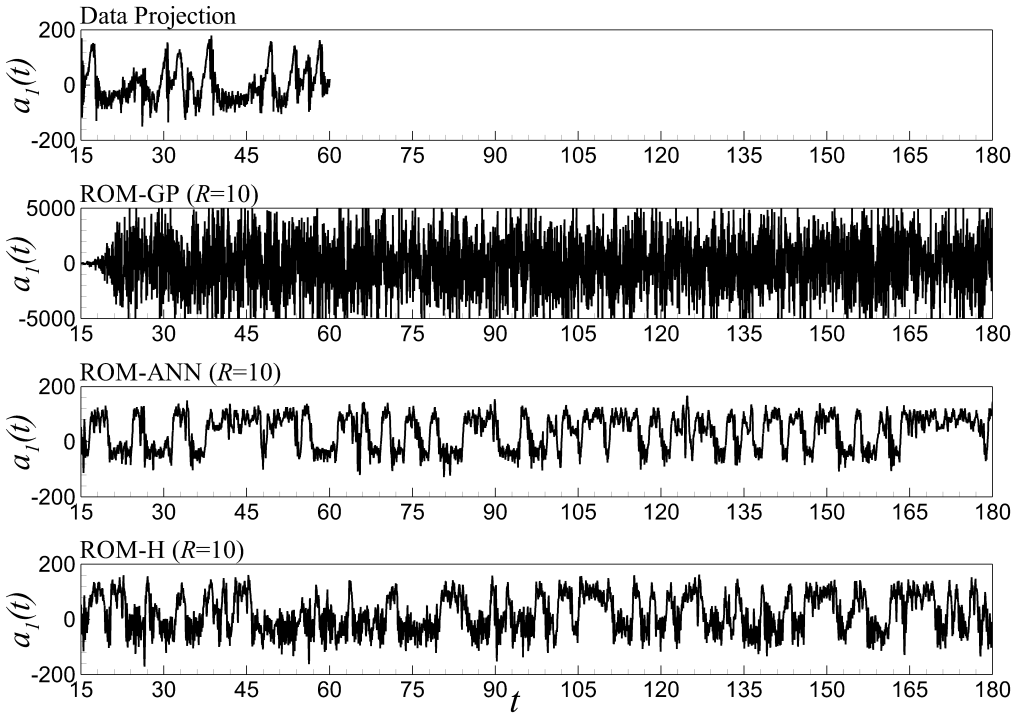


Figure 3.22: Forecasting of temporal mode evolution of $a_1(t)$ for Re = 100. Note that all ROMs are trained by the data snapshots obtained between $t = 15$ and $t = 60$. Although the ROM-GP approach provides reasonable (physical) results at the beginning (i.e., $t < 18$), the error then is amplified exponentially.

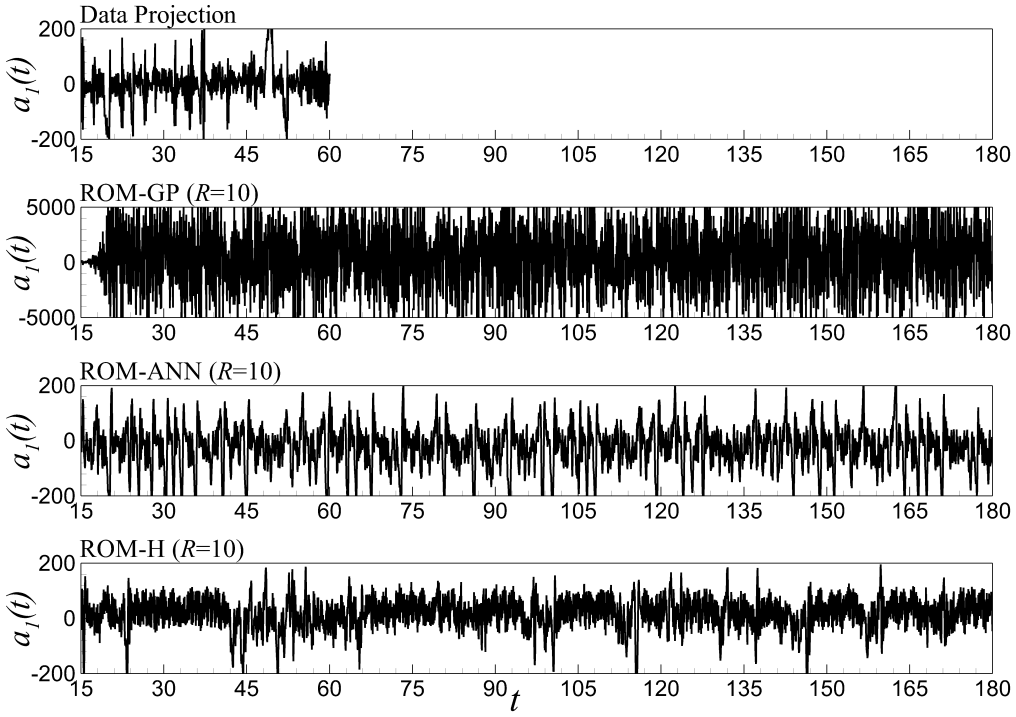


Figure 3.23: Forecasting of temporal mode evolution of $a_1(t)$ for Re = 400. Note that all ROMs are trained by the data snapshots obtained between $t = 15$ and $t = 60$. Although the ROM-GP approach provides reasonable (physical) results at the beginning (i.e., $t < 18$), the error is then amplified exponentially.

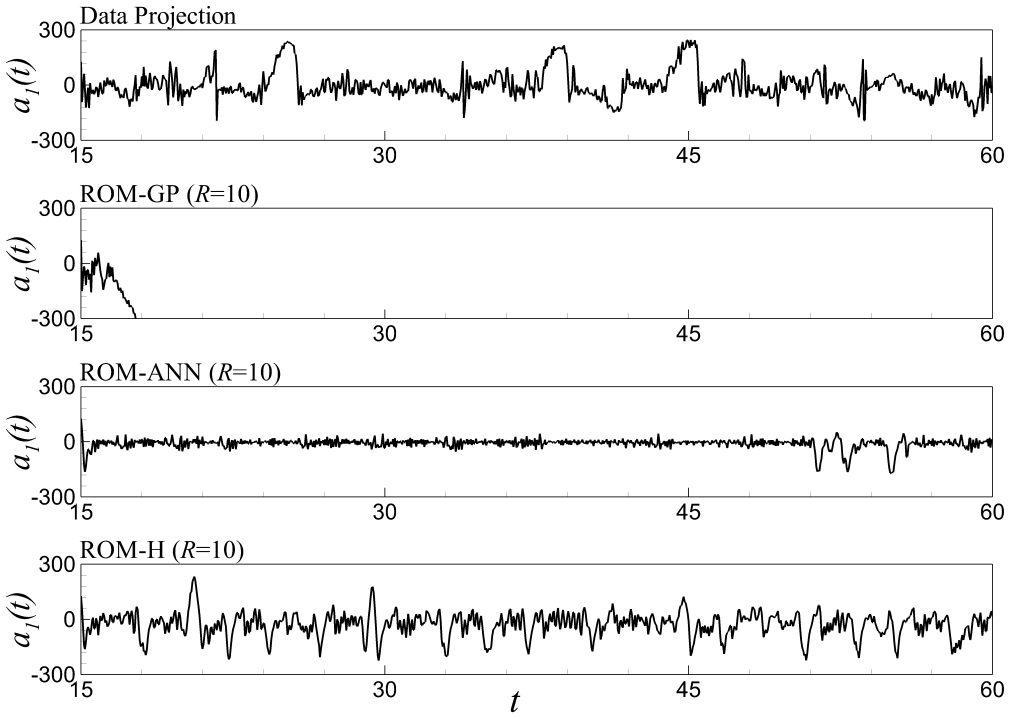


Figure 3.24: Time series evolution of the first temporal coefficient $a_1(t)$ for the out-of-sample forecast by FOM and different ROM approaches (ROM-GP, ROM-ANN, and ROM-H) using $R = 10$ POD modes. Predictive performance is shown for $\mathrm{Re} = 200$ while the training has been performed using the data generated at $\mathrm{Re} = 100$.

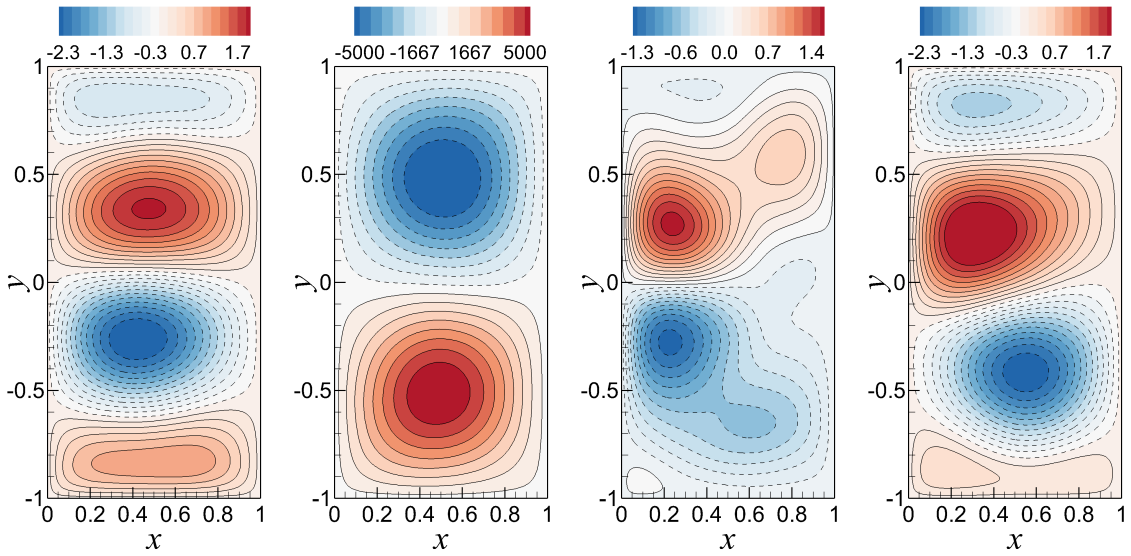


Figure 3.25: Out-of-sample forecast of mean stream function by: (a) FOM; (b) ROM-GP; (c) ROM-ANN; and (d) ROM-H using $R = 10$ POD modes. Predictive performance is shown for $\mathrm{Re} = 200$ while the training has been performed using the data generated at $\mathrm{Re} = 100$. This figure clearly illustrates that the ROM-H model has successfully predicted the FOM solution while the ROM-GP model prediction is unphysical and the ROM-ANN prediction is not as accurate as the ROM-H prediction.

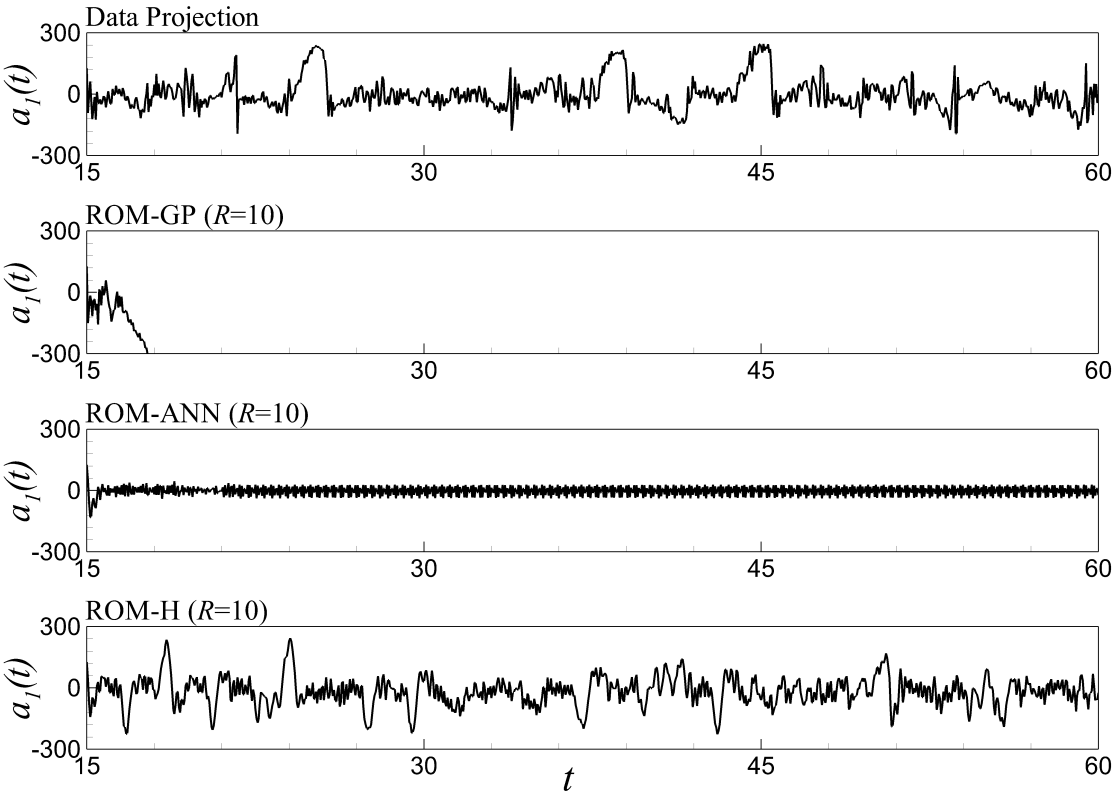


Figure 3.26: Time series evolution of the first temporal coefficient $a_1(t)$ for the out-of-sample forecast by FOM and different ROM approaches (ROM-GP, ROM-ANN, and ROM-H) using $R = 10$ POD modes. Predictive performance is shown for $\mathrm{Re} = 200$ while the training has been performed using the data generated at $\mathrm{Re} = 400$.

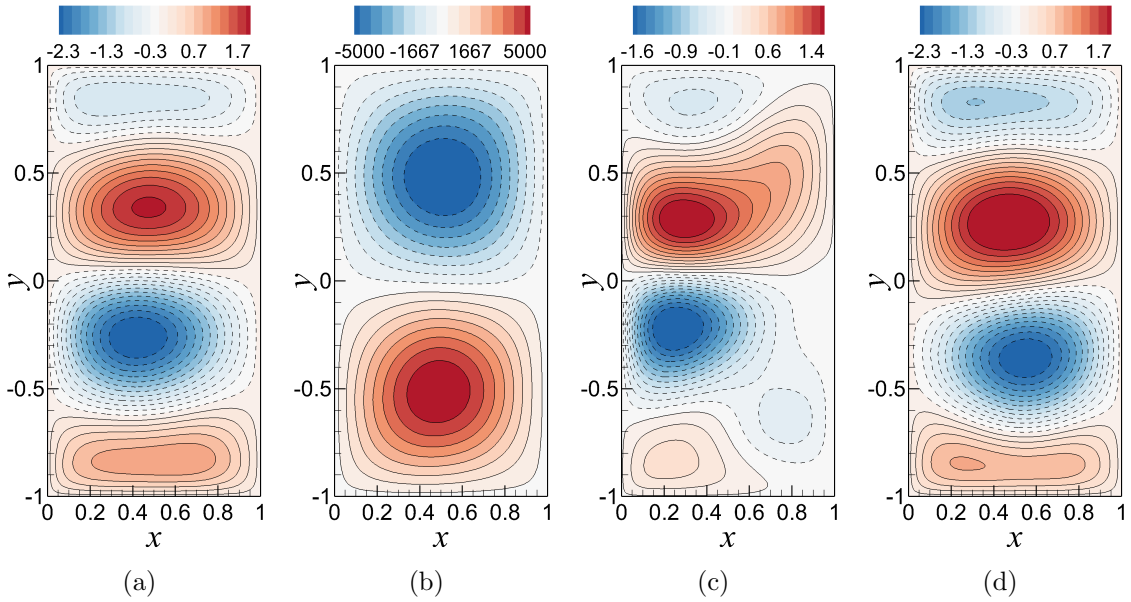


Figure 3.27: Out-of-sample forecast of mean stream function by: (a) FOM; (b) ROM-GP; (c) ROM-ANN; and (d) ROM-H using $R = 10$ POD modes. Predictive performance is shown for $\mathrm{Re} = 200$ while the training has been performed using the data generated at $\mathrm{Re} = 400$. Similar to the previous results, this figure also shows that the ROM-H prediction is better than the other two model predictions.

a test platform to analyze the prediction performance and robustness of the model compared to the physics-based and data-driven models. The training and snapshots data for POD basis construction are generated using a FOM simulation. The resolved ROM coefficients are utilized as inputs for the simple feedforward ELM network. We then introduce a weighting parameter to dynamically account for contributions from the physics-based and data-driven models. In other words, we propose a hybrid reduced-order model (ROM) comprising a first principle Galerkin model and a data driven ANN model for the dynamics based on POD mode amplitudes. Therefore, the key novelty of this hybrid ROM is a simple weighting between first-principle and data-driven model. Furthermore, we show an ad-hoc dynamical approach to estimate this weighting parameter which provides a freedom in the hybridization procedure to come up with a good possible combination of component model contributions.

To assess the performance of our proposed model, we perform a number of analyses

based on the mean stream function field variable and time series evolution statistics. Our study reveals that the proposed hybrid ROM model can successfully capture the underlying physics using a small number of modes which means the hybrid model is capable of performing a robust model order reduction process than the standard projection based ROM and data-driven ROM approaches. To validate the fidelity of the results obtained by the hybrid framework, we conduct a sensitivity analysis which reveals that the hybrid model is consistent with respect to the total number of hidden layer neurons of the ELM network. To show the robustness of the model, we also perform a long time forecasting and out-of-sample validation analysis. The hybrid model successfully captures the flow dynamics for these cases, while the other two component models fail to predict the true physics for higher Re and lower number of modes. Overall, our primary goal of leveraging the advantages of two basic reduced-order modeling techniques is fulfilled by the proposed hybrid ROM model in a sense that the model is data-driven and can be generalized for complex fluid flow problems, the model is robust and capable of reducing the total computational cost significantly, and the model neither requires any extra stabilization modification to account for numerical instabilities nor is it fully dependent on black box predictions.

CHAPTER IV

A Non-Intrusive Reduced Order Modeling Framework for Quasi-Geostrophic Turbulence[†]

4.1 Abstract

In this study, we present a non-intrusive reduced order modeling (ROM) framework for large-scale quasi-stationary systems. The framework proposed herein exploits the time series prediction capability of long short-term memory (LSTM) recurrent neural network architecture such that: (i) in the training phase, the LSTM model is trained on the modal coefficients extracted from the high-resolution data snapshots using proper orthogonal decomposition (POD) transform, and (ii) in the testing phase, the trained model predicts the modal coefficients for the total time recursively based on the initial time history. Hence, no prior information about the underlying governing equations is required to generate the ROM. To illustrate the predictive performance of the proposed framework, the mean flow fields and time series response of the field values are reconstructed from the predicted modal coefficients by using an inverse POD transform. As a representative benchmark test case, we consider a two-dimensional quasi-geostrophic (QG) ocean circulation model which, in general, displays an enormous range of fluctuating spatial and temporal scales. We first illustrate that the conventional Galerkin projection based reduced order modeling of such systems requires a high number of POD modes to obtain a stable flow physics. In addition, ROM-GP does not seem to capture the intermittent bursts appearing in the

[†]This chapter is adapted from Rahman et al., arXiv preprint arXiv:1906.11617 (2019), (under review for publication at Physical Review E).

dynamics of the first few most energetic modes. However, the proposed non-intrusive ROM framework based on LSTM (ROM-LSTM) yields a stable solution even for a small number of POD modes. We also observe that the ROM-LSTM model is able to capture quasi-periodic intermittent bursts accurately, and yields a stable and accurate mean flow dynamics using the time history of a few previous time states, denoted as the lookback time-window in this paper. Throughout the paper, we demonstrate our findings in terms of time series evolution of the field values and mean flow patterns, which suggest that the proposed fully non-intrusive ROM framework is robust and capable of predicting noisy nonlinear fluid flows in an extremely efficient way compared to the conventional projection based ROM framework.

4.2 Introduction

Large-scale turbulent flows, such as atmospheric and geophysical flows, are nonlinear dynamical systems which exhibit an enormous range of complex, coherent spatio-temporal scales. Over the past half century, computational approaches have made a significant contribution to characterize and understand the behavior of such flow phenomena. To resolve physical problems with high spatio-temporal variabilities through numerical simulation, one needs a high-fidelity modeling technique like direct numerical simulation (DNS). However, a huge amount of computational resources are required to capture the fine details of the flow dynamics which can become inefficient and unmanageable after some level of accuracy. Although there has been a continuous growth in computer power and performance following Moore's law during the past few decades (Mack, 2011), the progress has started to stagnate in the recent years (Powell, 2008; Waldrop, 2016; Theis and Wong, 2017; Kumar, 2018). As a result, one of the most active research areas in modeling of turbulent flow dynamics is the development of efficient and robust algorithms that aim at achieving the maximum attainable quality of numerical simulations with optimal computational costs. Indeed, computational

costs can be reduced by using low-fidelity models such as large eddy simulation (LES) (Sagaut, 2006) and Reynolds-averaged Navier-Stokes (RANS) (Tennekes et al., 1972) with additional approximations in the governing equations to neglect some of the physical aspects for closure modeling. Even so, these techniques require parameter calibration to approximate the true solution to any degree of confidence and may thus increase costs related to model validation, benchmark data generation, and efficient analysis of the generated data sets. As an alternative to the existing techniques, the reduced order modeling (ROM) approach has quickly become a promising approach to reduce the computational burden of high-fidelity simulations. In general, ROM works in such a way that the high-dimensional complex dynamical systems will be represented with much lower-dimensional (but dense) systems while keeping the solution quality within the acceptable range (Quarteroni et al., 2015; Taira et al., 2017). An introduction to ROM methodologies can be found in recent review articles (Benner et al., 2015; Taira et al., 2017; Rowley and Dawson, 2017).

There have been a significant number of strategies proposed over the years to obtain ROMs of nonlinear dynamical systems. These ROM techniques have been utilized for a wide variety of applications related to, e.g., flow control (Noack et al., 2011; Brunton and Noack, 2015; Ito and Ravindran, 1998), data assimilation in weather and climate modeling (Daescu and Navon, 2008; Cao et al., 2007), and uncertainty quantification (Ullmann and Lang, 2014; Chen and Quarteroni, 2015; Haasdonk et al., 2013). Among the different variants of ROM strategies, the Galerkin projection combined with proper orthogonal decomposition (POD) based ROMs (ROM-GP) have been utilized extensively in various areas (Carlberg et al., 2011; Lucia et al., 2004; Bistrian and Navon, 2015; Brenner et al., 2012; Burkardt et al., 2006; Iollo et al., 2000; Freno and Cizmas, 2014). POD, also known as principal component analysis (PCA), is a mathematical technique to extract the dominant statistical characteristics from turbulent flow fields by identifying the most energetic modes (Lumley, 1967;

Antoulas et al., 2001b; Chinesta et al., 2011; Benner et al., 2015; Rowley and Dawson, 2017). These few POD modes possess the fine-scale details of the system and have the capability of representing the true physics accurately. Over the years, considerable work has been done to improve the regular POD approaches (Hesthaven et al., 2015; Lassila et al., 2013; Quarteroni et al., 2015; Berkooz et al., 1993; Taira et al., 2017; Aubry et al., 1992; Benner et al., 2015; Rowley et al., 2004; Noack et al., 2011; Sieber et al., 2016).

In general, POD uses the data obtained from experiments or high-fidelity numerical simulations and generates an orthonormal set of spatial basis vectors describing the main directions (modes) by which the flow is represented optimally, in an L_2 sense (Berkooz et al., 1993). The most energetic modes are kept to generate the reduced order system while the other modes are truncated. However, it has been observed that the discarded modes often contribute to the evolving dynamics of large-scale complex turbulent flow systems, like the geophysical flows (Lassila et al., 2014), resulting in instabilities and modeling errors in the solution (Akhtar et al., 2009; Rempfer, 2000; Rowley and Williams, 2006; Iollo et al., 2000). Thus, several research efforts have been devoted to improve the stability of ROM-GP frameworks by addressing the truncated modes contributions (Aubry et al., 1988; Rempfer and Fasel, 1994b,a; Cazemier et al., 1998; Xie et al., 2017). Noack et al. (Noack et al., 2003) proposed an extra 'shift-mode' for accurate representation of the unstable steady solution. Several closure modeling ideas are devised to resolve the weak dissipation associated with POD modes by introducing eddy-viscosity terms (similar to LES eddy-viscosity models) (Sirisup and Karniadakis, 2004; Cordier et al., 2013; Östh et al., 2014). San and Iliescu (San and Iliescu, 2014) improved the ROM performance by finding an optimal value for eddy-viscosity parameter with the assumption that the amount of dissipation is not identical for all the POD modes. In our recent work, we proposed an automated approach to find the eddy-viscosity parameter dynamically to stabilize the ROM-GP

model (Rahman et al., 2019a). An alternative approach to find the eddy-viscosity parameter dynamically has been proposed by using an extreme learning machine architecture (San and Maulik, 2018a). With the growing interest in data-driven modeling of ROMs using machine learning (ML) architectures, there has been another dimension of research introduced to the community for the improvement in ROM performance, referred as hybrid ROM approach. Generally, the hybridization is done by combining an imperfect physics-based model with a data-driven technique to get a hybrid scheme, and it is observed that the hybrid model shows better predictive performance than the component models (Rahman et al., 2018; Wan et al., 2018; Xie et al., 2018a; Pathak et al., 2018).

In this paper, we develop a fully non-intrusive ROM approach as a potential alternative to already existing ROM methodologies. Indeed, physics-based (intrusive) ROM frameworks require an approximation of stabilization or regularization parameters and depend on underlying governing equation to get the solution. On the other hand, the hybrid approaches require computation of both intrusive and non-intrusive contributions, which can make the overall computation expensive. However, it is well-known that a non-intrusive approach can make the framework greatly efficient when it can be implemented successfully. With the abundance of massive amounts of data resources from high-fidelity simulations, field measurements, and experiments, the data-driven modeling approaches are currently considered some of the most promising methods across different scientific and research communities. In the past few years, artificial neural networks (ANNs) and other ML techniques have started a revolution in turbulence modeling community (Raissi et al., 2019; Maulik and San, 2017; Raissi et al., 2018; Lee et al., 1997; Maulik et al., 2018; Faller and Schreck, 1997; San and Maulik, 2018b; Wang et al., 2019; Moosavi et al., 2015; Kani and Elsheikh, 2017). Interested readers are directed to Refs. (Brunton et al., 2019; Kutz, 2017; Durbin, 2018; Duraisamy et al., 2018; Gamboa, 2017) for more on the influence of ML on fluid

mechanics, specifically turbulence modeling.

With a goal to develop an efficient and robust non-intrusive ROM framework for large-scale quasi-stationary systems like geophysical flows, we propose a methodology based on long short-term memory (LSTM) recurrent neural networks. Since reduced order modeling of such noisy large-scale systems is comparatively difficult due to instabilities, which results in using a very large number of POD modes to capture the true physics, our main motivation in this study is to utilize the time series prediction capability of LSTM (Gers et al., 2002; Wang et al., 2018; Yeo and Melnyk, 2019; Sak et al., 2014; Gamboa, 2017) to capture the flow physics with a very few POD modes. As detailed in Ref. (Yeo and Melnyk, 2019), LSTM is very robust in predicting a very noisy sequential time series. In general, for this type of random time series, LSTM does the prediction using its own internal dynamics, which is found stable and close to the true solution (Yeo and Melnyk, 2019; Mohan and Gaitonde, 2018). For this reason, we choose to utilize LSTM architecture based on our problem of interest, which is the large-scale quasi-stationary turbulence. However, we emphasize that this non-intrusive model can be developed by using other relevant neural network architectures as well. We also mention that the development of ROM using POD and LSTM has been used only in a few other works and proven to be successful in capturing the temporal dynamics of fluid flows. Wang et al. (Wang et al., 2018) proposed a non-intrusive ROM (NIROM) based on LSTM and used it to predict laminar flows. In another recent work, Vlachas et al. (Vlachas et al., 2018) proposed a data-driven method based on LSTM to predict the state derivative of chaotic systems using the short-term history of the reduced order states. The predicted derivatives are then used for one-step forward prediction of the high-dimensional dynamics. The authors further developed a hybrid framework combining mean stochastic model and LSTM for data-driven to extend the forecasting capability of the proposed approach. To do the dimensionality reduction, the authors utilized discrete Fourier transform, singular value

decomposition, and empirical orthogonal functions. Mohan and Gaitonde (Mohan and Gaitonde, 2018) developed a non-intrusive ROM using LSTM and POD for flow control applications through a detailed analysis on different ROM-LSTM training and testing hyper-parameter tuning parameters. Even though the authors' idea of developing non-intrusive ROM based LSTM by replacing Galerkin projection is similar to our present work, their work is mostly focused on exploring the capability of LSTM in modeling the flow in reduced order space for data sets with less randomness. Indeed, the data sets with less randomness have more "memory" in it, i.e., there are persistent or anti-persistent trends and thus, are more controllable through LSTM hyper-parameters. On the other hand, our present work is focused on exploring the capability of ROM-LSTM framework in resolving large-scale geophysical flow problem where the data sets mostly do not follow any particular trend. To this end, we develop a modular ROM-LSTM approach in chaotic and quasi-stationary systems to see whether it can overcome the instability issues associated with conventional ROMs for noisy dynamical systems. To assess our proposed framework, we consider the barotropic vorticity equation (BVE) representing the single-layer quasi-geostrophic (QG) model as an example of the quasi-stationary system. We observe a remarkably efficient predictive performance by the proposed framework based on LSTM (ROM-LSTM) through a number of numerical experiments and analyses.

The layout of the paper is as follows: Section 4.3 provides an overview of the barotropic vorticity equation describing a single-layer QG ocean model. In Section 4.4, dimension reduction through Galerkin-projection and proper orthogonal decomposition is illustrated briefly. Our proposed non-intrusive ROM-LSTM framework with a brief introduction to LSTM are presented in Section 4.5. In Section 4.6, we evaluate the predictive performance of the proposed ROM framework with respect to the standard ROM and full order model solutions. Finally, Section 4.7 provides a summary of this study and the conclusions drawn from it.

4.3 Single-Layer Quasi-Geostrophic (QG) Ocean Circulation Model

In the present study, we consider the simple single-layer QG ocean circulation model to develop and assess the performance of different ROM approaches. Following Refs. (Holm and Nadiga, 2003; San et al., 2011), we consider the single-layer QG problem as a benchmark for wind-driven, large-scale oceanic flow. Wind-driven flows of mid-latitude ocean basins have been studied frequently by modelers using idealized single- and double-gyre wind forcing, which helps in understanding various aspects of ocean dynamics, including the role of mesoscale eddies and their effect on mean circulation. However, modeling the vast range of spatio-temporal scales of the oceanic flows with all the relevant physics has always been challenging. As a result, the numerical simulation of oceanic and atmospheric flows still requires approximations and simplifications of the full model. The barotropic vorticity equation (BVE) describing the single-layer QG equation with dissipative and forcing terms is one of the most commonly used models for the double-gyre wind-driven geophysical flows (Majda and Wang, 2006).

The BVE model shares many features with the two-dimensional Euler and Navier-Stokes equations and has been extensively used over the years to describe various aspects of the largest scales of turbulent geophysical fluid dynamics (Holland and Rhines, 1980; Munk and Wunsch, 1982; Griffa and Salmon, 1989; Cummins, 1992; Greatbatch and Nadiga, 2000; Nadiga and Margolin, 2001). Using $\beta-$plane assumption reasonable for most oceanic flows, the dimensionless vorticity-streamfunction formulation of the forced-dissipative BVE can be written as (Rahman et al., 2018):

$$\frac{\partial \omega}{\partial t} + J(\omega, \psi) - \frac{1}{\mathrm{Ro}} \frac{\partial \psi}{\partial x} = \frac{1}{\mathrm{Re}} \nabla^2 \omega + \frac{1}{\mathrm{Ro}} \sin(\pi y), \tag{4.1}$$

where ∇^2 is the standard two-dimensional Laplacian operator. ω and ψ are the

kinematic vorticity and streamfunction, respectively, defined as:

$$\omega = \nabla \times \mathbf{u}, \tag{4.2}$$

$$\mathbf{u} = \nabla \times \psi \hat{k}, \tag{4.3}$$

where $\mathbf{u}$ is the two-dimensional velocity field and $\hat{k}$ refers to the unit vector perpendicular to the horizontal plane. The kinematic equation connecting the vorticity and streamfunction can be found by substituting the velocity components in terms of streamfunction in Equation (4.2), which yields the following Poisson equation:

$$\nabla^2 \psi = -\omega. \tag{4.4}$$

Equation (4.1) contains two dimensionless parameters, Reynolds number (Re) and Rossby number (Ro), which are related to the physical parameters and non-dimensional variables in the following way:

$$\mathrm{Re} = \frac{VL}{\nu}, \quad \mathrm{Ro} = \frac{V}{\beta L^2}, \tag{4.5}$$

where ν is the horizontal eddy viscosity of the BVE model and β is the gradient of the Coriolis parameter at the basin center ($y = 0$). L is the basin length scale and V is the velocity scale, also known as the Sverdrup velocity (Sverdrup, 1947), and is given by

$$V = \frac{\tau_0}{\rho H} \frac{\pi}{\beta L}, \tag{4.6}$$

where τ_0 is the maximum amplitude of the double-gyre wind stress, ρ is the mean fluid density, and H is the mean depth of the ocean basin. Despite not being explicitly represented in Equation (4.1), there are two important relevant physical parameters, the Rhines scale, δ_I, and the Munk scale, δ_M, which are the non-dimensional boundary

layer thicknesses for the inertial and viscous (Munk) layer of the basin geometry, respectively. As a physical interpretation of these parameters in BVE model, δ_I accounts for the strength of nonlinearity and δ_M is a measure of dissipation strength. δ_I and δ_M can be defined as

$$\frac{\delta_I}{L} = \left(\frac{V}{\beta L^2}\right)^{\frac{1}{2}}, \quad \frac{\delta_M}{L} = \left(\frac{\nu}{\beta L^3}\right)^{\frac{1}{3}} \tag{4.7}$$

and are related to Ro and Re by the following relations

$$\frac{\delta_I}{L} = (\mathrm{Ro})^{\frac{1}{2}}, \quad \frac{\delta_M}{L} = \left(\frac{\mathrm{Ro}}{\mathrm{Re}}\right)^{\frac{1}{3}}. \tag{4.8}$$

Finally, the nonlinear advection term in Equation (4.1) is given by the Jacobian

$$J(\omega, \psi) = \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y}. \tag{4.9}$$

Since ocean circulation models where the Munk and Rhines scales are close to each other, like the QG model, remain time dependent rather being converged to a steady state as time approaches to infinity (Medjo, 2000), numerical computations of these models are conducted in a statistically steady state, also known as the quasi-stationary state. Hence, in our study, we utilize numerical schemes suited for simulation of such type of ocean models and for long-time integration. Details of the relevant numerical discretization schemes, Poisson solver, and problem definitions for this study can be found in elsewhere (Rahman et al., 2018; San et al., 2011; San, 2016).

4.4 Intrusive ROM-GP Methodology

The intrusive ROM framework is developed based on a standard Galerkin projection methodology using the method of snapshots, an efficient method for computing the POD basis functions (Sirovich, 1987). In this section, we give a brief idea on the

ROM-GP framework utilized in our work. We obtain N number of snapshots for vorticity field, $\omega(x, y, t_n)$ for $n = 1, 2, ..., N$ at pseudo-time $t = t_n$ from full order model simulation (FOM). Algorithm 1 describes the POD basis construction procedure from the stored snapshots.

We can approximate the field variables, i.e., kinematic vorticity and streamfunction using the most energetic R POD modes, where $R << N$, such that these R largest energy containing modes correspond to the largest eigenvalues $(\lambda_1, ..., \lambda_R)$. The resulting full expression for the field variables can be written as:

$$\omega(x, y, t) = \bar{\omega}(x, y) + \sum_{k=1}^{R} a_k(t)\phi_k(x, y), \tag{4.10}$$

$$\psi(x, y, t) = \bar{\psi}(x, y) + \sum_{k=1}^{R} a_k(t)\theta_k(x, y), \tag{4.11}$$

where $a_k(t)$ accounts for both streamfunction and vorticity based on the kinematic relation given by Eq. (4.4). It should be mentioned that in our ROM formulations, we use the following angle-parenthesis definition for the inner product of two arbitrary functions f and g:

$$\int_{\Omega} f(x, y)g(x, y)dxdy = \langle f; g \rangle. \tag{4.12}$$

Algorithm 1 POD basis construction

1: Compute the time-invariant mean fields and the fluctuation fields (mean-subtracted snapshots) for the given number of snapshots of the 2D vorticity field as:

$$\bar{\omega}(x, y) = \frac{1}{N} \sum_{n=1}^{N} \omega(x, y, t_n), \tag{4.13}$$

$$\omega'(x, y, t_n) = \omega(x, y, t_n) - \bar{\omega}(x, y). \tag{4.14}$$

2: An $N \times N$ snapshot data matrix $\mathbf{A} = [a_{ij}]$ is computed from the inner product of mean-subtracted snapshots

$$a_{ij} = \langle \omega'(x, y, t_i); \omega'(x, y, t_j) \rangle, \tag{4.15}$$

where i and j refer to the snapshot indices.

3: Compute the optimal POD basis functions by performing an eigendecomposition of $\mathbf{A}$ as $\mathbf{AV} = \mathbf{V\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix whose entries are the eigenvalues λ_k of $\mathbf{A}$, and $\mathbf{V}$ is a matrix whose columns $\mathbf{v}_k$ are the corresponding eigenvectors. In our computations, we use the eigensystem solver based on the Jacobi transformations since $\mathbf{A}$ is a symmetric positive definite matrix (Press et al., 1992).

4: Using the eigenvalues stored in a descending order (i.e., $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$), for proper selection of the POD modes in $\mathbf{\Lambda}$, compute the orthogonal POD basis functions for the vorticity field ϕ_k as

$$\phi_k(x, y) = \frac{1}{\sqrt{\lambda_k}} \sum_{n=1}^{N} v_k^n \omega'(x, y, t_n), \tag{4.16}$$

where v_k^n is the n^{th} component of the eigenvector $\mathbf{v}_k$. The scaling factor, $1/\sqrt{\lambda_k}$, is to guarantee the orthonormality of POD modes, i.e., $\langle \phi_i; \phi_j \rangle = \delta_{ij}$, where δ_{ij} is the Kronecker delta.

5: Obtain the k^{th} mode for the streamfunction, $\theta_k(x, y)$ utilizing the linear dependence between streamfunction and vorticity given by Equation (4.4):

$$\nabla^2 \theta_k = -\phi_k, \tag{4.17}$$

for each $k = 1, 2, ..., R$ where R is the number of retained modes ($R << N$). To be able to use the same $a_k(t)$ coefficients for both streamfunction and vorticity fields, the following elliptic equation holds true for the mean variables:

$$\nabla^2 \bar{\psi} = -\bar{\omega}. \tag{4.18}$$

6: Construct k^{th} time-dependent modal coefficients $a_k(t_n)$ for N snapshots by using

POD modes and forward transformation:

$$a_k(t_n) = \langle \omega(x, y, t_n) - \bar{\omega}(x, y); \phi_k \rangle. \tag{4.19}$$

We refer to (Rahman et al., 2019a) for the details of the integration technique utilized in this study. In conventional projection based intrusive ROM framework, we apply Galerkin Projection to the governing equation, which yields R coupled ordinary differential equations (ODEs) for the time evolution of the temporal modes of the system while the spatial modes are kept constant (Kunisch and Volkwein, 2002; Rowley et al., 2004; Aubry et al., 1988). Any standard time integration technique can be utilized to solve the coupled ODE system, since the basis functions and corresponding modal coefficients will be precomputed in the offline computation stage. The Galerkin projection approach is summarized in Algorithm 2.

Algorithm 2 Galerkin projection to obtain ROM

1: Given an initial condition $\omega(x, y, t_0)$ at time t_0, compute the initial modal coefficients $a_k(t_0)$ using the relation below:

$$a_k(t_0) = \langle \omega(x, y, t_0) - \bar{\omega}(x, y); \phi_k \rangle. \tag{4.20}$$

2: Perform an orthogonal Galerkin projection by multiplying the governing equation with the POD basis functions and integrating over the domain (Rapún and Vega, 2010), which will yield the following dynamical system for a_k:

$$\frac{da_k}{dt} = \mathfrak{B}_k + \sum_{i=1}^{R} \mathfrak{L}_k^i a_i + \sum_{i=1}^{R} \sum_{j=1}^{R} \mathfrak{N}_k^{ij} a_i a_j, \tag{4.21}$$

where $k = 1, 2, ..., R$ and the predetermined model coefficients can be computed by the following numerical integration (offline computing):

$$
\begin{aligned}
\mathfrak{B}_k &= \big\langle -J(\bar{\omega},\bar{\psi}) + \frac{1}{\mathrm{Ro}}(\sin(\pi y) + \frac{\partial \bar{\psi}}{\partial x}) + \frac{1}{\mathrm{Re}}\nabla^2\bar{\omega}; \phi_k \big\rangle, \\
\mathfrak{L}^i_k &= \big\langle -J(\bar{\omega},\theta_i) - J(\phi_i,\bar{\psi}) + \frac{1}{\mathrm{Ro}}\frac{\partial \theta_i}{\partial x} + \frac{1}{\mathrm{Re}}\nabla^2\phi_i; \phi_k \big\rangle, \\
\mathfrak{N}^{ij}_k &= \big\langle -J(\phi_i,\theta_j); \phi_k \big\rangle. \qquad (4.22)
\end{aligned}
$$

4.5 Non-Intrusive ROM-LSTM Methodology

In this section, we discuss our proposed ROM-LSTM methodology. As outlined in Algorithm 1, we obtain the time-dependent modal coefficients a_k by performing a POD transform on stored snapshot data. The modal coefficients are a sequence of data points with respect to time, i.e., a time series representing the underlying dynamical system. In intrusive or physics-based ROM, we do Galerkin projection using governing equation to obtain a coupled system of ODEs for a_k, and then solve the ODE system on the given time interval. However, the limitations of projection based ROMs, such as susceptibility to instability for noisy data set, numerical constraints for solving ODE system, or inefficient reduced order modeling, encourage us to replace the physics-based Galerkin projection phase of ROM-GP methodology with a data-driven approach. Among the variety of ideas to resolve the issues associated with projection based ROM, a number of published works related to ROM based on POD and neural networks have shown signs of future success. The recurrent neural network (RNN) is one of the widely used neural network architectures in ROMs which is designed to operate on input information as well as the previously stored observations to predict the dependencies among the temporal data sequences (Jaeger and Haas, 2004; LeCun et al., 2015). LSTM is a special variant of RNN which is capable of tracking long-term dependencies among the input data sequences. Hence, we consider LSTM recurrent neural network to develop our non-intrusive ROM-LSTM framework. Before describing the ROM-LSTM procedure, we first briefly review the LSTM architecture.

As the name suggests, RNNs contain recurrent or cyclic connections that enable them to model complex time-varying data sequences with a wide range of temporal dependencies or correlations between them. In general, RNNs map a sequence of data to another sequence through time using cyclic connections, and constrain some of the connections to hold the same weights using back-propagation algorithm (Rumelhart et al., 1988). However, the standard RNN architecture suffers from vanishing gradient problem when the gradient of some weights starts to become too small or too large (Bengio et al., 1994). This leads to the development of improved RNN architectures which overcome the modeling issues of standard RNNs. One of the most successful forms of improved RNN architectures is the LSTM network, which solves the limitation of vanishing gradients (Hochreiter and Schmidhuber, 1997). In contrast to most of the ANN architectures, LSTM operates by cell states and gating mechanisms to actively control the dynamics of cyclic connections and thus, resolves the vanishing gradient issues. Similar to the standard RNNs, LSTM can learn and predict the temporal dependencies based on the input information and previously acquired information, i.e., the internal memory of LSTM allows the network to find the relationship between the current input and stored information to make a prediction.

The conventional LSTM architecture contains memory blocks in the recurrent hidden layers, which have memory cells to store the cell states and gates to control the flow of information. Each memory block has an input gate controlling the flow of input activations into the cell, a forget gate to adaptively forgetting and resetting the cell's memory (to prevent over-fitting by processing continuous inflow of input streams), and the output gate controlling the output flow of cell activations into the next cell. In our LSTM architecture, we consider an input sequential data matrix $\mathcal{X}_k = \left\{a_k^{(1)}, a_k^{(2)}, ..., a_k^{(N-\sigma)}\right\}$ for training, where σ is the lookback time-window. The lookback time-window, in our definition, means the time history over which the LSTM model does the training and prediction recursively. Indeed, increasing the value of

σ increases the quality of training the model, but makes the model dependent on an increased number of initial states during prediction. The LSTM model is trained to an output sequential data matrix $\mathcal{Y}_k = \left\{a_k^{(\sigma+1)}, a_k^{(\sigma+2)}, ..., a_k^{(N)}\right\}$ recursively (here, $k = 1, 2, ..., R$ and $\mathcal{X}, \mathcal{Y} \in a$). Considering input gate as $\mathcal{I}$, the forget gate as $\mathcal{F}$, the output gate as $\mathcal{O}$, the cell activation vectors as c, and the cell output as h, the LSTM model does the mapping from the input states to an output state by using the following equations at any instance n (Sak et al., 2014; Yeo and Melnyk, 2019):

$$\text{Input network:} \quad \mathcal{X}_k^{(n)} = \tanh\left(W_h h_k^{(n)} + W_{\mathcal{Y}} \mathcal{Y}_k^{(n)}\right), \tag{4.23}$$

$$\text{Gate functions:} \quad m_k^{(n)} = \zeta\left(W_m \mathcal{X}_k^{(n)} + b_m\right), \tag{4.24}$$

$$\text{for} \quad m \in (\mathcal{I}, \mathcal{F}, \mathcal{O})$$

$$\text{Internal cell state:} \quad c_k^{(n+1)} = \mathcal{F}_k^{(n)} \odot c_k^{(n)} + \mathcal{I}_k^{(n)} \odot \xi, \tag{4.25}$$

$$\text{where} \quad \xi = \tanh\left(W_c \mathcal{X}_k^{(n)} + b_c\right)$$

$$\text{Output state:} \quad h_k^{(n+1)} = \mathcal{O}_k^{(n)} \odot \tanh\left(c_k^{(n+1)}\right), \tag{4.26}$$

$$\text{Output network:} \quad \widehat{\mathcal{Y}_k}^{(n+1)} = W_{\mathcal{Y}2} \tanh\big(W_{\mathcal{Y}1} h_k^{(n+1)} + b_{\mathcal{Y}1}\big) + b_{\mathcal{Y}2}, \tag{4.27}$$

where W represents the weight matrices for each gates, $\widehat{\mathcal{Y}_k}^{(n+1)}$ is the LSTM prediction, b denotes the bias vectors for each gates, $\odot$ is the element-wise product of vectors, and ζ is the logistic sigmoid function.

Similar to the ROM-GP methodology, the workflow of the ROM-LSTM framework consists of two phases as displayed in Figure 4.1. In the offline training phase, we first obtain POD basis functions and modal coefficients using Algorithm 1. The known time series of modal coefficients from training snapshots are used to train the LSTM model. Based on the values of σ, the input of the LSTM model $\mathcal{M}$ will be the previous time states of the input modal coefficients for R retained modes and the output of

the model will be the next time state recursively for R modes. Training LSTM model is the computationally heavier part of the ROM-LSTM framework, but this is done offline. In online testing phase, we recursively predict the modal coefficients for the total time using the trained model $\mathcal{M}$. The input of the trained model $\mathcal{M}$ will be the initial states $\left\{a_1^{(1)}, \ldots, a_R^{(1)}; \ldots; a_1^{(\sigma)}, \ldots, a_R^{(\sigma)}\right\}$ based on the preselected value of σ and the output will be recursive prediction of corresponding future time states. Thus, we bypass the physics-based Galerkin projection part with completely data-driven neural network approach to predict the modal coefficients. Also, the computational cost of prediction through trained LSTM network is significantly lower than the physics-based approach. Finally, we reconstruct the mean vorticity and streamfunction fields using inverse transform to analyze the behavior of the quasi-stationary flow. The key steps of the ROM-LSTM framework are outlined below in Algorithm 3.

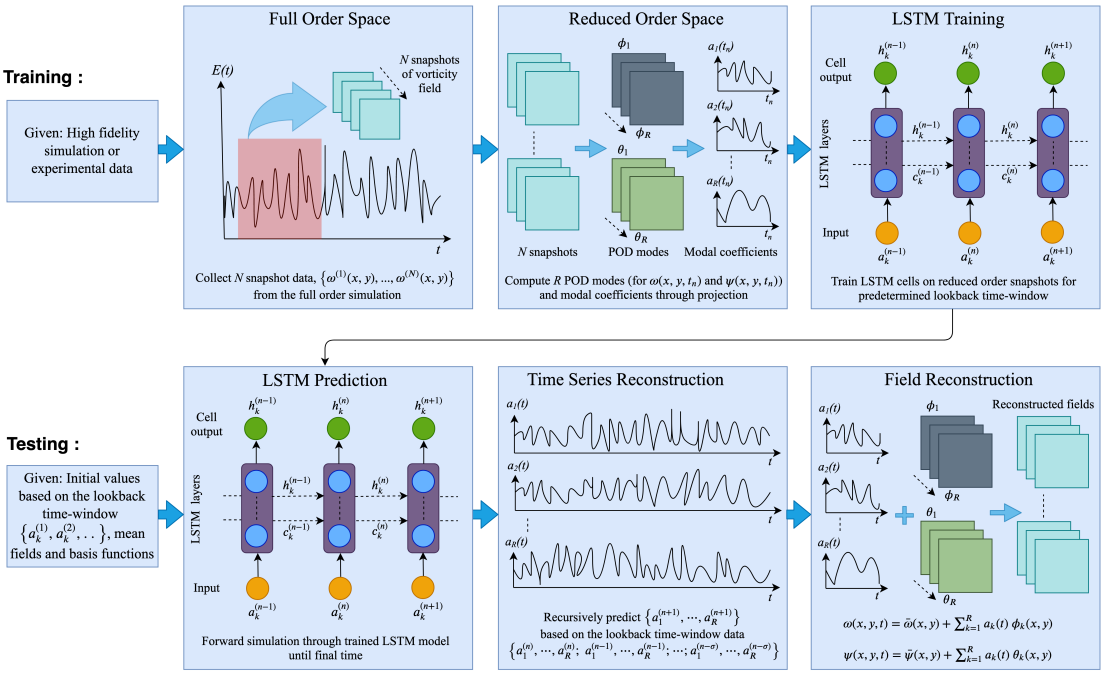


Figure 4.1: Workflow diagram of the ROM-LSTM framework. Note that the training phase (offline computation) is computationally heavier compared to the testing (online computation) phase.

To design our LSTM architecture for ROM-LSTM framework, we utilize Keras

(Chollet et al., 2015), a high level API designed for deep learning, combined with standard Python libraries. The FOM simulation for data snapshots generation and POD basis construction codes are written in FORTRAN programming language. We utilize a deep LSTM network stacking 6 LSTM layers with 40 neurons in each layer. The computational cost is found manageable in this deep architecture setup, which encourages us to perform all the numerical experiments with this same setup. The mean-squared error (MSE) is chosen as the loss function, and a variant of stochastic gradient descent method, called ADAM (Kingma and Ba, 2014), is used to optimize the mean-squared loss. The other relevant hyper-parameters utilized in our LSTM architecture are documented in Table. 4.1. The hyper-parameters are kept constant for all simulations to obtain a fair comparison between the results in different numerical experiment runs. It should be noted that the training data is normalized by the minimum and maximum of each time series to be in between the range $[-1, +1]$.

Algorithm 3 ROM-LSTM framework

Training (offline)

1: Collect N snapshot data for the vorticity field, $\omega(x, y, t_n)$ from the FOM simulation. Here, $\omega(x, y, t_n) = \left\{\omega^{(1)}(x, y), \omega^{(2)}(x, y), \ldots, \omega^{(N)}(x, y)\right\}$.

2: Compute R POD modes for kinematic vorticity, ϕ_k and streamfunction, θ_k using Equation (4.16) and Equation (4.18), respectively, for $k = 1, 2, ..., R$.

3: Construct modal coefficients by a forward transform through projection

$$a_k(t_n) = \langle \omega(x, y, t_n) - \bar{\omega}(x, y); \phi_k \rangle, \tag{4.28}$$

where $a_k(t_n) = \left\{a_k^{(1)}, a_k^{(2)}, \ldots, a_k^{(N)}\right\}$.

4: Train LSTM model on reduced order snapshots for selected lookback time-window σ:

$$\mathcal{M} : \left\{ a_1^{(n)}, \ldots, a_R^{(n)}; \ \ldots \ ; a_1^{(n-\sigma)}, \ldots, a_R^{(n-\sigma)} \right\}$$
$$\Rightarrow \left\{ a_1^{(n+1)}, \ldots, a_R^{(n+1)} \right\}. \tag{4.29}$$

Testing (Online)

5: Given initial values $\{a_k^{(1)}, a_k^{(2)}, \ldots, a_k^{(\sigma)}\}$ based on σ, precomputed mean values and basis functions.

6: Use the trained LSTM model $\mathcal{M}$ to recursively predict $a_k(t)$ until final time reached.

7: Reconstruct the mean fields by inverse transform using Equation (4.10) and Equation (4.11).

Table 4.1: A list of hyper-parameters utilized to train the LSTM network for all numerical experiments.

Parameters	Values
Number of hidden layers	6
Number of neurons in each hidden layer	40
Batch size	16
Epochs	500
Activation functions in the LSTM layers	tanh
Training-testing ratio	4:9
Validation data set	20%
Loss function	MSE
Optimizer	ADAM

4.6 Numerical Results

The predictive performance of the ROM-LSTM framework is thoroughly examined in this section in terms of time series evolution of the modal coefficients and mean flow fields. It is well documented in literature that the ROM-GP framework is incapable of capturing mean flow dynamics for quasi-stationary flows using lower number of POD modes, and susceptible to instability (San and Iliescu, 2014, 2015). There have

been a number of approaches proposed in previous literature to improve the ROM performance. One way to stabilize the ROM is by adding an empirical stabilization parameter based on the analogy between large eddy simulation and truncated modal projection (Aubry et al., 1988; Wang et al., 2012). Later, it is found that the ROM performance further improves taking the optimal value for the stabilization parameter rather than selecting it arbitrarily (Rempfer, 1991; San and Iliescu, 2014; Cazemier, 1997; San and Iliescu, 2015). In our previous work, we have shown that computing the stabilization parameter dynamically at each time step improve the ROM performance significantly (Rahman et al., 2019a).

However, the proposed ROM-LSTM methodology has several advantages over the physics-based approaches, such as, no dependence on the underlying governing dynamical system to obtain the solution, i.e., the process is free of numerical constraints, no burden of adding stabilization parameter to account for instability issues and so on.

To reach a conclusion about the performance of the ROM-LSTM framework, we compare ROM-LSTM predictions with the FOM simulation and the standard ROM-GP results. Moreover, we present the performance of the ROMs based on lookback time-window σ and LSTM training for different number of POD modes to show the robustness and capability of the proposed framework. Furthermore, we present the L_2-norm errors to perform a quantitative assessment on the accuracy of the ROM-LSTM solutions with respect to ROM-GP solutions.

We choose the single-layer QG problem as our test bed to evaluate the performance of ROMs. Because of the complex flow behavior with wide range of scales, QG problem has been utilized as test problem in many notable literature (Cushman-Roisin and Beckers, 2011; Holm and Nadiga, 2003; San et al., 2011; Cummins, 1992; Greatbatch and Nadiga, 2000). To make the analyses simple and easily understandable, we present simulation results only for Re= 450 and Ro= 3.6×10^{-3} flow condition, which can

be considered turbulent enough and suitable for reduced order modeling. The FOM simulation is done from $t = 0$ to $t = 100$ using a constant time step of $\Delta t = 2.5 \times 10^{-4}$ on a Munk layer resolving 256×512 grid resolution (i.e., consisting of about four grid points in the Munk scale, i.e., $\delta_M/L = 0.02$). To avoid the initial transient time interval, we store 400 data snapshots from $t = 10$ to $t = 50$ to generate the POD bases and modal coefficients to train the LSTM model. We refer to Ref. (Rahman et al., 2019a) to get an idea on the POD analysis as well as the instantaneous vorticity field plots for the same flow condition. The computational domain of our test problem is $(x, y) \in [0, 1] \times [-1, 1]$. To understand the nature of the QG data set, we compute the Hurst exponent, H, for the modal coefficients. The Hurst exponent is a statistical measure of the presence of long-term trends in a non-stationary time series (Hurst, 1951). Thus, the Hurst exponent can help in selecting the appropriate model for a given time series prediction. We also note that the Hurst exponent has been utilized in many research fields, e.g., hydrology, finance, and healthcare indsutry (Chandra et al., 2013; Koutsoyiannis, 2003; Lahmiri and Boukadoum, 2014; Bassler et al., 2006; Carbone et al., 2004). H can be statistically defined as (Qian and Rasheed, 2004):

$$\mathrm{E}\left[\frac{\mathrm{Range}(n)}{\mathrm{SD}(n)}\right] = kn^H, \quad \text{as } n \to \infty. \tag{4.30}$$

Here, E is the expected value of the ratio between the range of the first n cumulative deviations from the mean and their corresponding standard deviations (SD), n is the time span of the observations, and k is constant. The range of H is in between 0 and 1. $H \to 1$ means a persistent series (a strong trend in the time series at hand), $H \to 0$ means an anti-persistent series (a time series with long-term switching between high and low values) and $H \approx 0.5$ indicates a random series (fewer correlations between current and future observations). Interested readers are directed to Ref. (Mohan and Gaitonde, 2018) for a detailed description of suitability of LSTM as a predictive modeling approach for different time series data using the measurement of H. We

calculate the H for modal coefficients of QG data set for given flow conditions using the so called rescaled range (R/S) analysis, popularized by Mandelbrot and Wallis (Mandelbrot and Wallis, 1968, 1969). The details of (R/S) analysis can be found in Ref. (Qian and Rasheed, 2004). The Hurst exponents for the modal coefficients of QG case are tabulated in Table 4.2, where we can see that the values of H are around 0.5. This indicates the randomness of the QG problem, which can be a good representative of large-scale quasi-stationary geophysical turbulent flow systems.

Table 4.2: Hurst exponents of modal coefficients.

Modal coefficient	Hurst exponent
$a_1(t)$	0.52
$a_2(t)$	0.35
$a_3(t)$	0.63
$a_4(t)$	0.59
$a_5(t)$	0.49
$a_6(t)$	0.59
$a_7(t)$	0.59
$a_8(t)$	0.46
$a_9(t)$	0.58
$a_{10}(t)$	0.53

Figure 4.2 shows the mean streamfunction and vorticity field contours obtained by the ROM-GP model. To compare the predictive performance of the ROM-GP model with respect to the true solution, we include the mean contour plots of FOM simulation on the left column as well. We can see the full order solution displays a four-gyre circulation patterns for both mean streamfunction and vorticity fields. Since the instantaneous fields for the QG flow is always fluctuating in time, it becomes difficult to compare solutions of different models at the same time state. However, the mean fields always exhibit the four-gyre circulation for higher Re (highly turbulent regime, i.e., turbulence with weak dissipation) which implies a state of turbulent equilibrium between two inner gyres circulation representing the wind stress curl forcing and the outer gyres representing the eddy flux of potential vorticity (the northern and

southern gyres found in geostrophic turbulence) (Greatbatch and Nadiga, 2000). In our study, the time-averaged (mean) field data are obtained by averaging between $t = 10$ to $t = 100$. Another point to be noted in FOM field plots that the bright orange circulations in the four-gyres (top circulation of the inner gyres and bottom circulation of the outer gyres) indicate the circulation in counter clockwise or positive direction and the other two circulations signifies the circulation in clockwise direction. We can observe in Figure 4.2 that the ROM-GP simulations with $R = 10$ and $R = 20$ modes display a non-physical two-gyre circulation for streamfunction whereas the vorticity field does not capture almost any conclusive physical pattern. However, the results improve with increasing modes as we can see the streamfunction contour is showing clear four-gyre patterns even though the vorticity plots are very noisy compared to the true solution. These observations are supported by the time series evolution of first and tenth modal coefficient plots in Figure 4.3. It is apparent that the increasing modes stabilize the noise to reach a physical solution for both modal coefficients.

We note that the time scale in our formulation is normalized by L/V to obtain dimensionless time unit. Following (San and Staples, 2013a), typical oceanic values (e.g., $L = 2000$ km and $\beta = 1.75 \times 10^{-11}$ m^{-1}s^{-1}) yield approximately $L/V = 0.25$ year for Ro $= 0.0036$. Therefore, a numerical simulation over 100 computational time units refers to the evolution of flow dynamics over 25 years in physical time. Therefore, the intermittent bursts appeared in the true projection of the most energetic mode (i.e., $a_1(t)$ indicate the seasonal variations in QG dynamics. Although ROM-GP yields non-physical solution for $R = 10$ and $R = 20$ cases, $a_k(t)$ series reaches more meaningful levels for $R = 30$ and beyond. However, it is hard to claim from Figure 4.3 that the ROM-GP yields an accurate prediction of these seasonal bursts even for higher R values.

We present the field contours obtained by ROM-LSTM based on different σ values in Figure 4.4. It can be seen that $\sigma = 1$ and $\sigma = 2$ do not provide much accurate

results as the patterns get distorted in some extent even though they are being able to capture the four-gyre. However, both streamfunction and vorticity contours show a stable and accurate prediction of the true mean fields for $\sigma = 4$ and $\sigma = 5$. Though the vorticity field contour is not displaying as smooth contour lines as the true solution due to the reduction of dimension order, it is showing a better performance compared to the ROM-GP solutions. As shown in the recent work of Yeo (Yeo, 2019), the LSTM network trained on a noisy data learns to reduce the contributions of noisy input data by developing its own dynamics and thus, the prediction remains close to the truth rather than being unstable due to noisy input data. Hence, the LSTM prediction is expected to yield a stable and physical solution for a fluctuating quasi-stationary system. It should be noted that these results are obtained for LSTM training with $R = 10$ modes. The time series evolution plots for the modal coefficients based on $\sigma = 5$ and $R = 10$ modes in Figure 4.5 show that ROM-LSTM time series predictions are almost on top of the true projection of modal coefficients. Even though the model is trained for $t = 10$ to $t = 50$ only, the ROM-LSTM model is able to obtain a stable and accurate prediction up to the final time $t = 100$.

Another impressive observation on the predictive capability of the ROM-LSTM framework is presented in Figure 4.6 where we show the mean field plots based on the number of modes retained to train the LSTM model. We keep $\sigma = 5$ for this numerical experiment. As we can see the ROM-LSTM model is being able to capture the four-gyre circulation even with only two modes. Indeed, the first few modes contain most of the dynamics in the system and we can also see reduction of some smaller scales for lower mode predictions. Nevertheless, this finding indicates the prediction capability of the ROM-LSTM framework to produce a stable solution of a noisy system. However, we have seen the ROM-GP model becomes unstable to predict noisy data set with lower number of modes which makes it very inefficient. In contrast, the proposed non-intrusive framework can be very efficient to produce stable

solution with a very few modes. Since we observe promising predictive performance for training with 2 modes only, we present a couple of more analyses on results obtained by the ROM-LSTM framework retaining 2 modes for LSTM training. We can see in Figure 4.7 that lower σ values simulations are unable to capture the fluctuations along the mean and goes almost straight along the line after a few time states. The model starts to capture the fluctuating flow fields with the increase of σ values. The field plots in Figure 4.8 also displays the similar conclusions. Since the lower σ value solutions stay along the line around the mean (unlike rapid oscillations in ROM-GP solutions), the field plots still show the mean physics to some extent. It is obvious that the model with lower σ ignores most of the scales of the system. However, the prediction improves with higher σ as displayed in the Figure 4.8.

Finally, we include a comparison plot in Figure 4.9 where we present the first two modal coefficients prediction obtained by different ROM set up. The σ value is kept 5 for all the ROM-LSTM simulations. As expected, the ROM-GP solutions for 10 modes become totally non-physical and unstable. On the contrary, the ROM-LSTM predictions for $R = 2$, $R = 4$, $R = 8$ and $R = 10$ modes show a good match between the true solution and the prediction. For the quantitative assessment on the accuracy of both ROM-GP and ROM-LSTM frameworks, L_2-norm errors of the reduced order models (with respect to FOM) for the mean vorticity and streamfunction fields are tabulated in Table 4.3. The root mean-square error or Euclidean L_2-norm error is computed by:

$$\mathrm{L}_2 = ||e||^2 = \sqrt{\frac{1}{N_x N_y} \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} e_{i,j}^2} \tag{4.31}$$

where N_x and N_y are the grid resolutions in x and y directions. For the vorticity field,

the error i.e., the difference between the predicted mean and FOM solution mean is:

$$e_{i,j} = \left|\bar{\omega}_{i,j}^{\mathrm{ROM}} - \bar{\omega}_{i,j}^{\mathrm{FOM}}\right|. \tag{4.32}$$

For ROM-LSTM framework, the results are presented for $R = 10$ modes. We can observe that the prediction accuracy increases with the increase in lookback time-window σ and we can obtain a more accurate result than the ROM-GP simulation with $R = 80$ using only 10 modes in ROM-LSTM framework. We present the CPU time per time step (between $t = 10$ and $t = 100$) for ROM-LSTM framework simulations based on $R = 10$ modes and different σ in Table 4.4. We can observe a gradual reduction of computational time (for both training and testing) with lower values of σ. All the simulations of ROM-LSTM frameworks are done in Python programming environment and CPU time is computed as per time step. The computational time step is set to 1.00×10^{-1} for online testing. In our FOM simulation in FORTRAN, the CPU time is about 1.17×10^{-1} (between $t = 0$ and $t = 100$), where computational time step is set $\Delta t = 2.5 \times 10^{-5}$ due to the CFL restriction of numerical stability for our explicit forward model on the resolution of 256×512. We refer to Ref. (Rahman et al., 2019a) to get an idea about the computational overhead for ROM-GP frameworks using same flow conditions. It should be mentioned that the ROM-GP computations are done using FORTRAN programming platform. Even so, we observe our ROM-LSTM CPU times are in the same order of ROM-GP simulations with $R = 80$ modes. In Table 4.4, we show the computational cost for $R = 10$ modes only since the CPU time (both training and testing) for other ROM-LSTM runs using different modeling conditions are in same order.

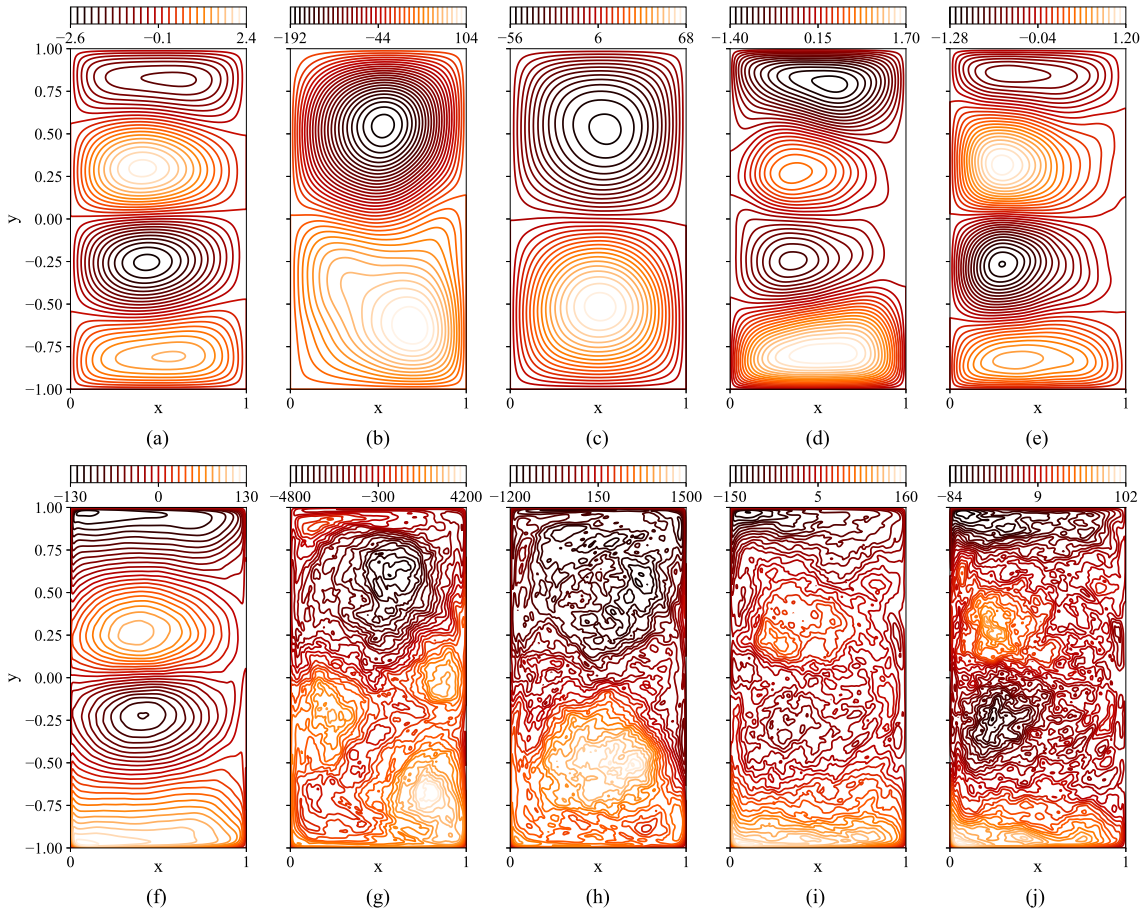


Figure 4.2: Mean streamfunction and vorticity fields obtained by the FOM simulation and the standard ROM-GP simulation at $\mathrm{Re} = 450$ and $\mathrm{Ro} = 3.6 \times 10^{-3}$ flow condition. (a) ψ_{FOM} at a resolution of 256×512, (b) $\psi_{\text{ROM-GP}}$ with $R = 10$ modes, (c) $\psi_{\text{ROM-GP}}$ with $R = 20$ modes, (d) $\psi_{\text{ROM-GP}}$ with $R = 40$ modes, (e) $\psi_{\text{ROM-GP}}$ with $R = 80$ modes, (f) ω_{FOM} at a resolution of 256×512, (g) $\omega_{\text{ROM-GP}}$ with $R = 10$ modes, (h) $\omega_{\text{ROM-GP}}$ with $R = 20$ modes, (i) $\omega_{\text{ROM-GP}}$ with $R = 40$ modes, (j) $\omega_{\text{ROM-GP}}$ with $R = 80$ modes.

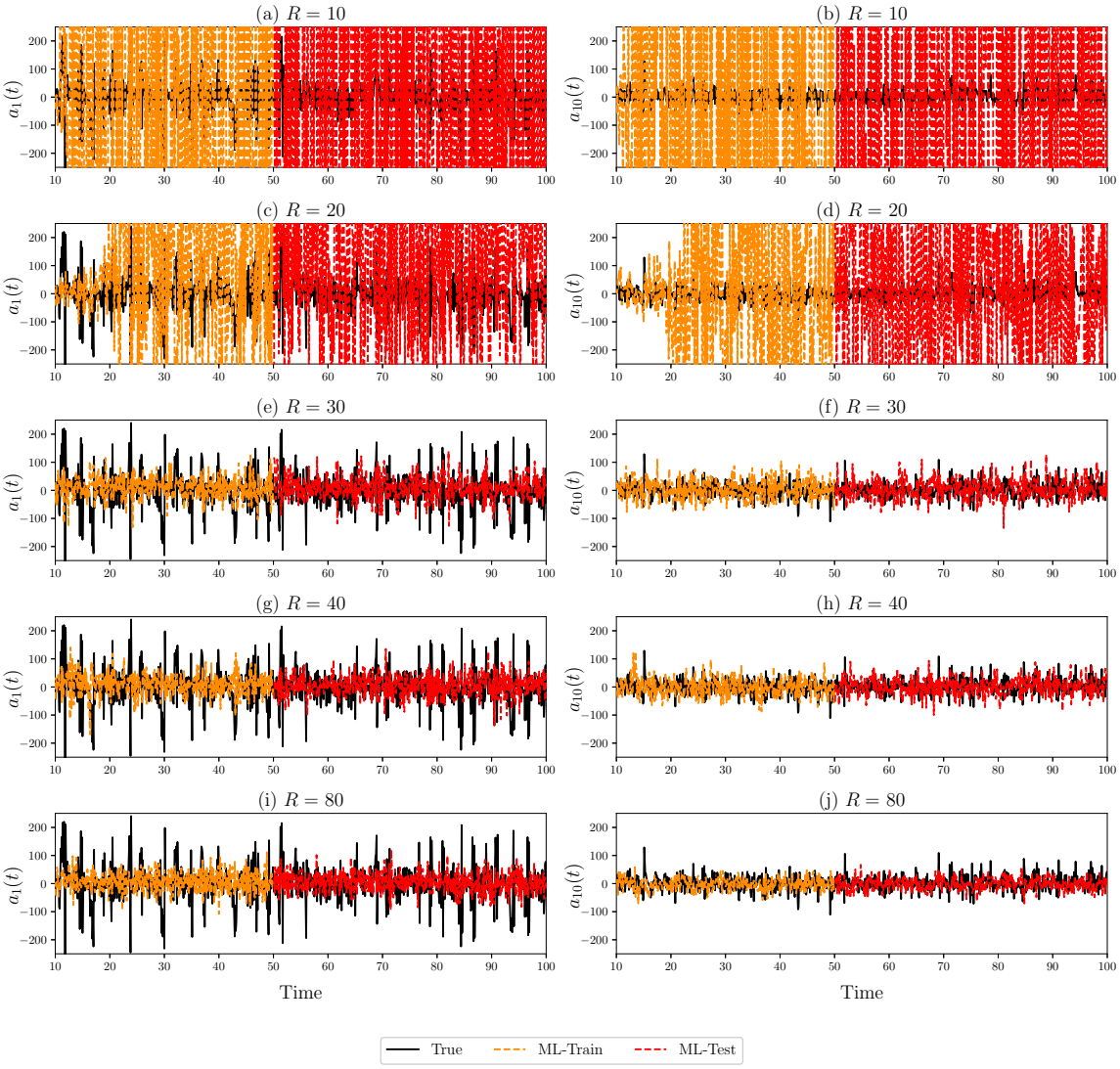


Figure 4.3: Time series evolution of the first and tenth modal coefficients, $a_1(t)$ and $a_{10}(t)$ respectively, between $t = 10$ to $t = 100$ for standard ROM-GP simulation at Re $= 450$ and Ro $= 3.6 \times 10^{-3}$. (a) $a_1(t)$ for ROM-GP with $R = 10$ modes, (b) $a_{10}(t)$ for ROM-GP with $R = 10$ modes, (c) $a_1(t)$ for ROM-GP with $R = 20$ modes, (d) $a_{10}(t)$ for ROM-GP with $R = 20$ modes, (e) $a_1(t)$ for ROM-GP with $R = 30$ modes, (f) $a_{10}(t)$ for ROM-GP with $R = 30$ modes, (g) $a_1(t)$ for ROM-GP with $R = 40$ modes, (h) $a_{10}(t)$ for ROM-GP with $R = 40$ modes, (i) $a_1(t)$ for ROM-GP with $R = 80$ modes, (j) $a_{10}(t)$ for ROM-GP with $R = 80$ modes. True projection series is underlined in each figure with black straight line. The training zone is shown with orange dashed line (from $t = 10$ to $t = 50$) and the out-of-sample testing zone is shown with red dashed line (from $t = 51$ to $t = 100$) in ROM-LSTM solution series in each figure.

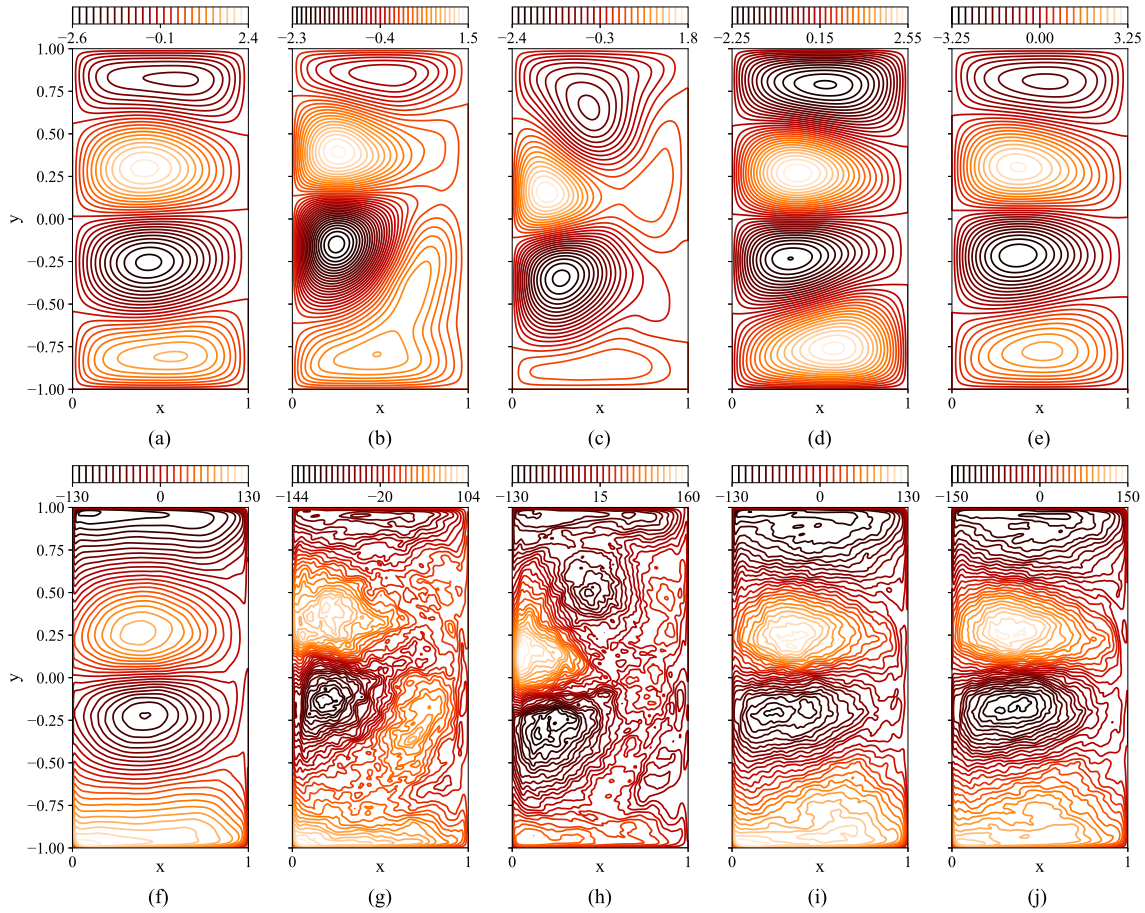


Figure 4.4: Mean streamfunction and vorticity fields obtained by the ROM-LSTM simulation based on different lookback time-window, σ at $\mathrm{Re} = 450$ and $\mathrm{Ro} = 3.6\times10^{-3}$ flow condition. (a) ψ_{FOM} at a resolution of 256×512, (b) $\psi_{\text{ROM-LSTM}}$ with $\sigma = 1$, (c) $\psi_{\text{ROM-LSTM}}$ with $\sigma = 2$, (d) $\psi_{\text{ROM-LSTM}}$ with $\sigma = 4$, (e) $\psi_{\text{ROM-LSTM}}$ with $\sigma = 5$, (f) ω_{FOM} at a resolution of 256×512, (g) $\omega_{\text{ROM-LSTM}}$ with $\sigma = 1$, (h) $\omega_{\text{ROM-LSTM}}$ with $\sigma = 2$, (i) $\omega_{\text{ROM-LSTM}}$ with $\sigma = 4$, (j) $\omega_{\text{ROM-LSTM}}$ with $\sigma = 5$. Note that the LSTM model is trained with $R = 10$ modes.

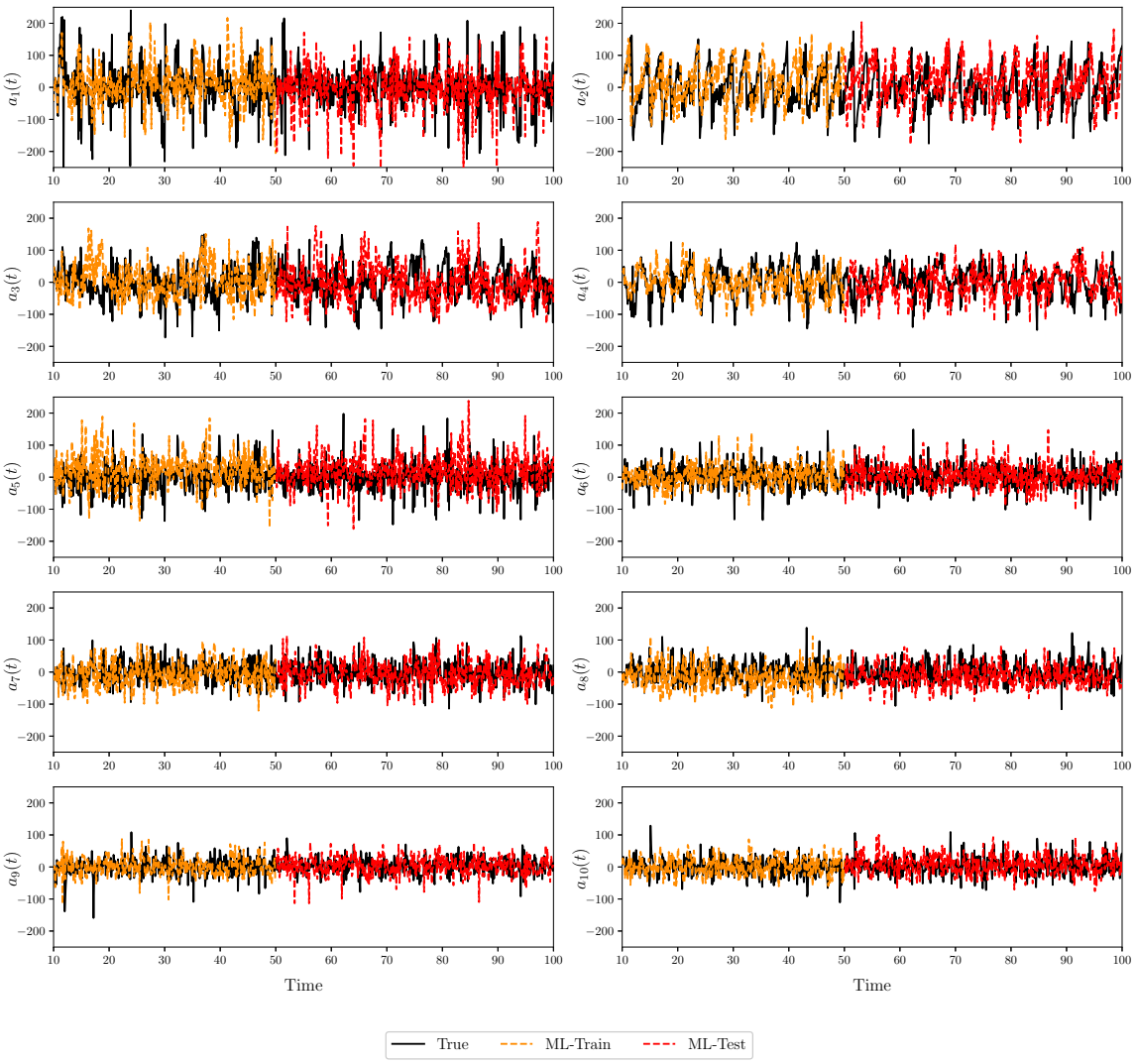


Figure 4.5: Time series evolution of the modal coefficients between $t = 10$ to $t = 100$ for ROM-LSTM simulation at Re $= 450$ and Ro $= 3.6 \times 10^{-3}$. Note that the LSTM model is trained with $R = 10$ modes and $\sigma = 5$. True projection series is underlined in each figure with black straight line. The training zone is shown with orange dashed line (from $t = 10$ to $t = 50$) and the out-of-sample testing zone is shown with red dashed line (from $t = 51$ to $t = 100$) in ROM-LSTM solution series in each figure.

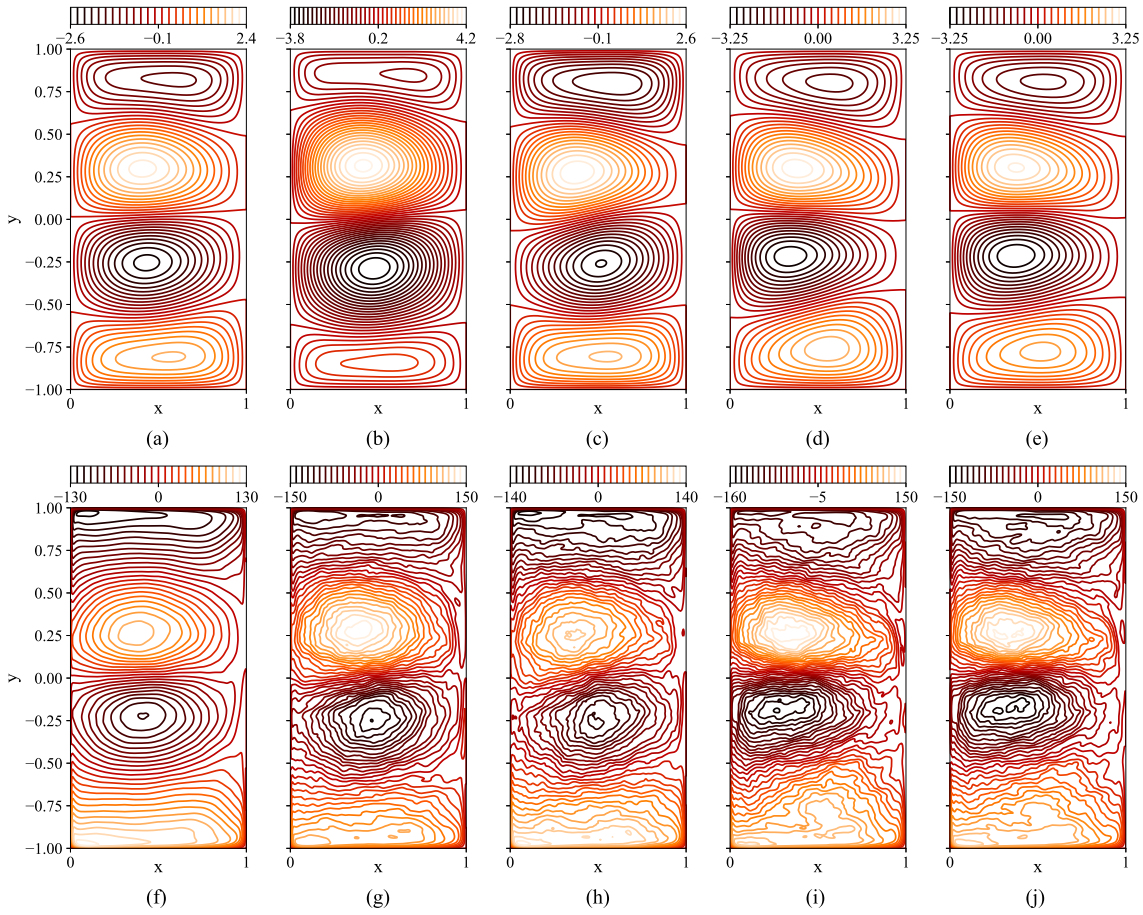


Figure 4.6: Mean streamfunction and vorticity fields obtained by the ROM-LSTM simulation based on the number of modes to train the LSTM model at Re $= 450$ and Ro $= 3.6 \times 10^{-3}$ flow condition. (a) ψ_{FOM} at a resolution of 256×512, (b) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $R = 2$ modes, (c) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $R = 4$ modes, (d) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $R = 8$ modes, (e) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $R = 10$ modes, (f) ω_{FOM} at a resolution of 256×512, (g) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $R = 2$ modes, (h) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $R = 4$ modes, (i) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $R = 8$ modes, (j) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $R = 10$ modes. Note that the LSTM model is trained with $\sigma = 5$.

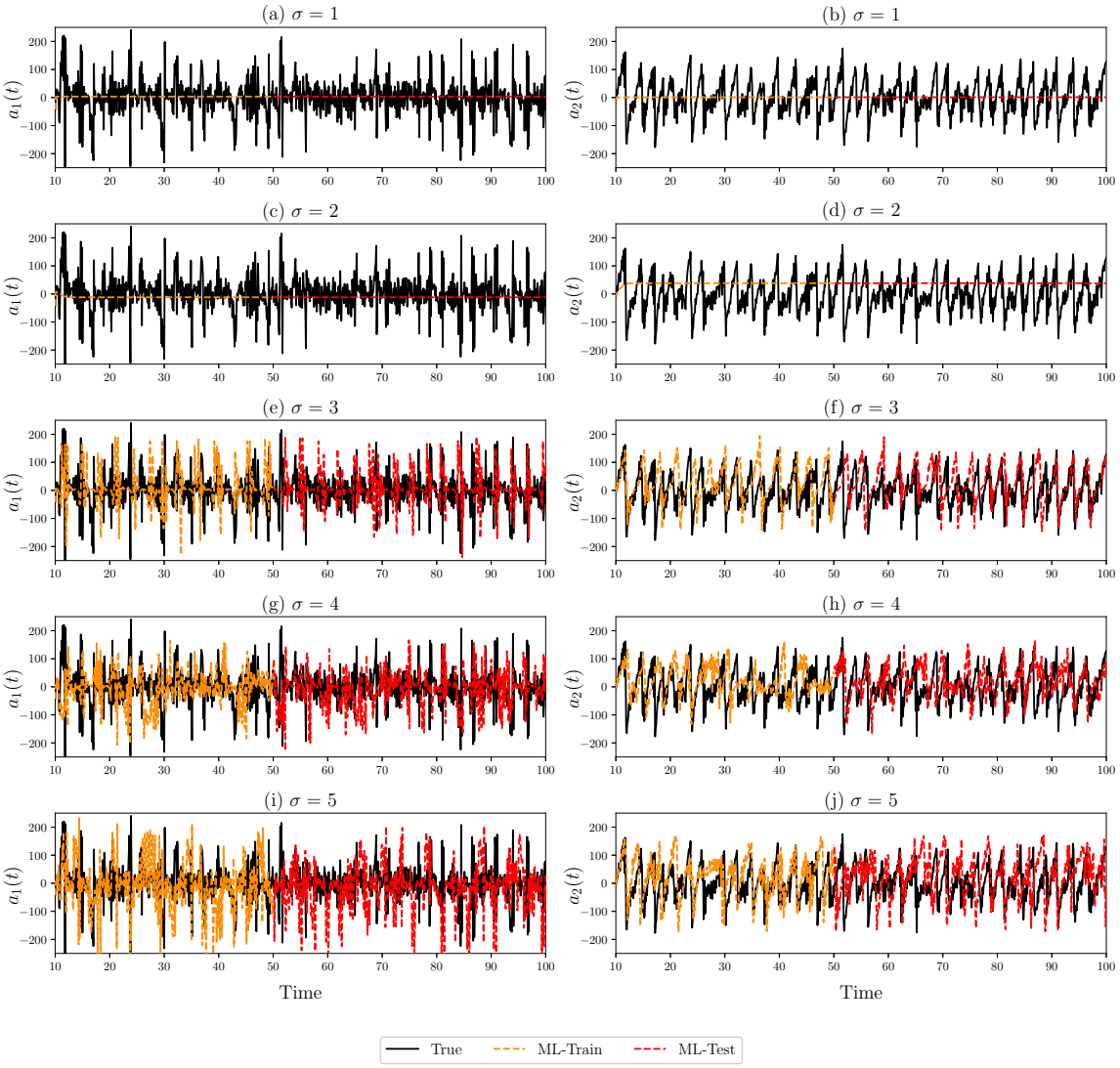


Figure 4.7: Time series evolution of the modal coefficients between $t = 10$ to $t = 100$ for ROM-LSTM simulation based on different lookback time-windows, σ and LSTM training with $R = 2$ modes at Re $= 450$ and Ro $= 3.6 \times 10^{-3}$. (a) $a_1(t)$ with $\sigma = 1$, (b) $a_2(t)$ with $\sigma = 1$, (c) $a_1(t)$ with $\sigma = 2$, (d) $a_2(t)$ with $\sigma = 2$, (e) $a_1(t)$ with $\sigma = 3$, (f) $a_2(t)$ with $\sigma = 3$, (g) $a_1(t)$ with $\sigma = 4$, (h) $a_2(t)$ with $\sigma = 4$, (i) $a_1(t)$ with $\sigma = 5$, (j) $a_2(t)$ with $\sigma = 5$. True projection series is underlined in each figure with black straight line. The training zone is shown with orange dashed line (from $t = 10$ to $t = 50$) and the out-of-sample testing zone is shown with red dashed line (from $t = 51$ to $t = 100$) in ROM-LSTM solution series in each figure.

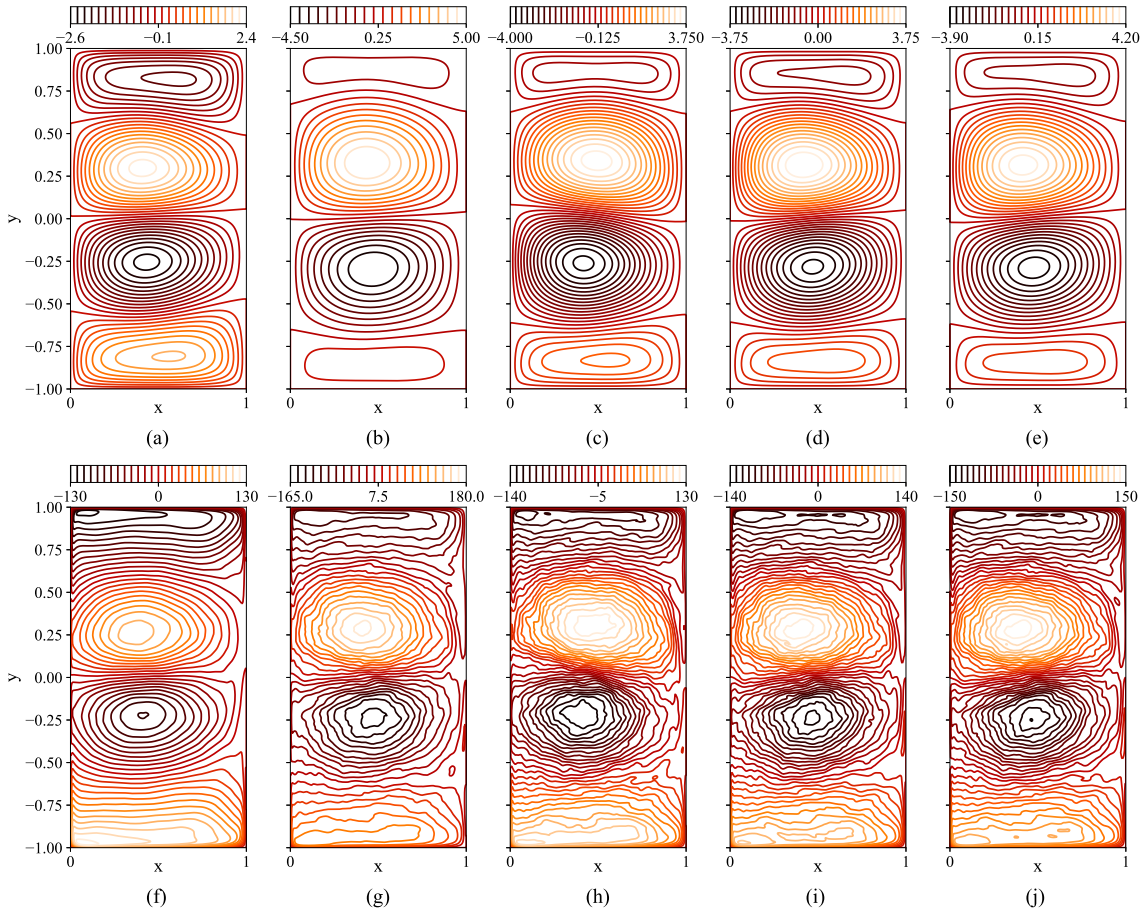


Figure 4.8: Mean streamfunction and vorticity fields obtained by the ROM-LSTM simulation based on different lookback time-windows, σ and LSTM training with $R = 2$ modes at $\mathrm{Re} = 450$ and $\mathrm{Ro} = 3.6 \times 10^{-3}$. (a) ψ_{FOM} at a resolution of 256×512, (b) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 2$, (c) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 3$, (d) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 4$, (e) $\psi_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 5$, (f) ω_{FOM} at a resolution of 256×512, (g) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 2$, (h) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 3$, (i) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 4$, (j) $\omega_{\text{ROM-LSTM}}$ for LSTM training with $\sigma = 5$.

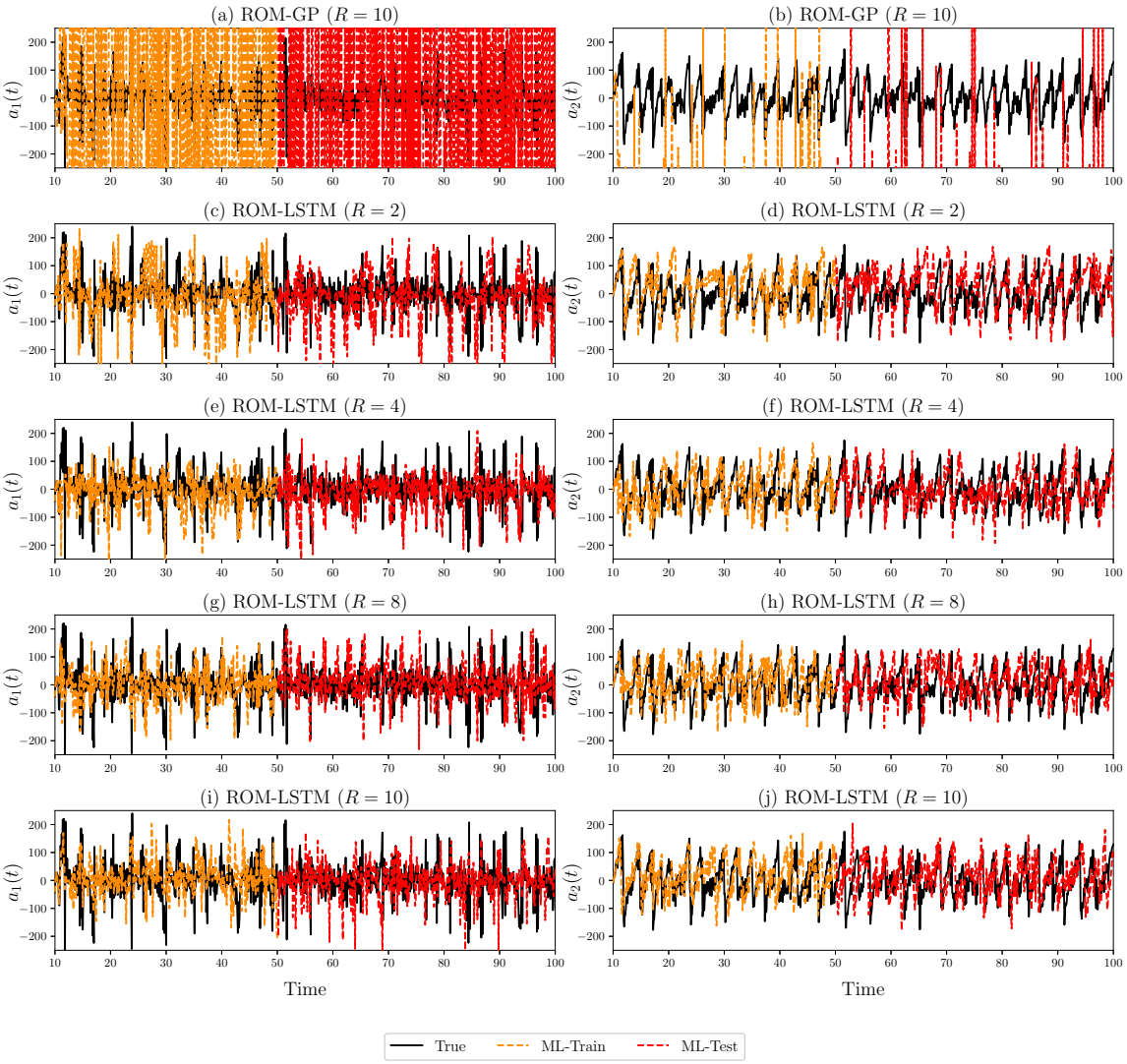


Figure 4.9: Time series evolution of first two modal coefficients, $a_1(t)$ and $a_2(t)$ respectively, between $t = 10$ to $t = 100$ for different ROMs at Re $= 450$ and Ro $= 3.6 \times 10^{-3}$. (a) $a_1(t)$ for ROM-GP with $R = 10$ modes, (b) $a_2(t)$ for ROM-GP with $R = 10$ modes, (c) $a_1(t)$ for ROM-LSTM trained with $R = 2$ modes, (d) $a_2(t)$ for ROM-LSTM trained with $R = 2$ modes, (e) $a_1(t)$ for ROM-LSTM trained with $R = 4$ modes, (f) $a_2(t)$ for ROM-LSTM trained with $R = 4$ modes, (g) $a_1(t)$ for ROM-LSTM trained with $R = 8$ modes, (h) $a_2(t)$ for ROM-LSTM trained with $R = 8$ modes, (i) $a_1(t)$ for ROM-LSTM trained with $R = 10$ modes, (j) $a_2(t)$ for ROM-LSTM trained with $R = 10$ modes. True projection series is underlined in each figure with black straight line. The training zone is shown with orange dashed line (from $t = 10$ to $t = 50$) and the out-of-sample testing zone is shown with red dashed line (from $t = 51$ to $t = 100$) in ROM-LSTM solution series in each figure.

Table 4.3: L_2-norm errors of the reduced order models (with respect to FOM) for the mean vorticity and streamfunction fields. Note that the ROM-LSTM model trained with $R = 10$ modes results are presented here.

	Vorticity	Streamfunction
Intrusive ROM		
ROM-GP ($R = 10$)	3.19×10^6	5.59×10^3
ROM-GP ($R = 20$)	4.46×10^5	9.87×10^2
ROM-GP ($R = 30$)	9.35×10^2	9.99×10^{-1}
ROM-GP ($R = 40$)	6.60×10^2	4.33×10^{-1}
ROM-GP ($R = 80$)	1.16×10^3	3.84×10^{-1}
Non-intrusive ROM		
ROM-LSTM ($\sigma = 1$)	1.90×10^3	6.12×10^{-1}
ROM-LSTM ($\sigma = 2$)	2.50×10^3	7.01×10^{-1}
ROM-LSTM ($\sigma = 3$)	8.63×10^2	3.65×10^{-1}
ROM-LSTM ($\sigma = 4$)	2.69×10^2	2.09×10^{-1}
ROM-LSTM ($\sigma = 5$)	4.70×10^2	1.29×10^{-1}

Table 4.4: Computational overhead for the ROM-LSTM model trained with $R = 10$ modes. For training, CPU time is presented as per epoch for 400 samples and for testing, CPU time is presented as per time step. Note that, the time step for testing is set 1.00×10^{-1} since the non-intrusive set up is free of numerical stability constraints.

ROM-LSTM	Training time (s)	Testing time (s)
$\sigma = 1$	8.10×10^{-2}	1.15×10^{-3}
$\sigma = 2$	1.07×10^{-1}	1.38×10^{-3}
$\sigma = 3$	1.30×10^{-1}	1.56×10^{-3}
$\sigma = 4$	1.59×10^{-1}	1.79×10^{-3}
$\sigma = 5$	1.80×10^{-1}	2.00×10^{-3}

4.7 Summary and Conclusions

In this paper, we propose an efficient and robust fully non-intrusive ROM framework to capture the large spatio-temporal scale of fluctuating quasi-stationary systems. Due to the robustness and stability of LSTM recurrent neural network in predicting noisy dynamical systems, we consider LSTM architecture to develop our data-driven ROM, denoted as ROM-LSTM in this paper. As an example of large-scale turbulent flows exhibiting a wide range of spatio-temporal scales, we investigate the reduced order modeling of a simple general ocean circulation model, single-layer QG turbulence, to assess the predictive performance of our proposed ROM-LSTM framework. It was previously observed that the conventional physics-based (or intrusive) ROM of QG model requires a large number of POD modes to yield stable and physical flow dynamics. However, the proposed ROM-LSTM framework shows a very promising improvement in reduced order modeling that only a few modes are able to capture a physical solution without any prior knowledge about the underlying governing equations. We first demonstrate that the conventional Galerkin projection ROM approach yields non-physical predictions when we use a small number of representative modes. Although ROM-GP converges to a more physical solution when increasing the number of modes, it does not seem to capture the intermittent bursts appearing in the dynamics of the first few most energetic modes. However, the proposed ROM-LSTM approach is able to capture these bursts and yields remarkably accurate results even when using a small number of modes.

The proposed methodology consists of two phases: offline training and online testing or prediction phase. Initially, we collect the high-fidelity simulation or experimental data snapshots for a certain flow condition. The data snapshots are collected up to a certain time of the full order model simulation for training. Then we do a mapping of the high-resolution instantaneous data snapshots into a reduced order, i.e., low-dimensional space through POD transform. In this process, we generate POD

basis functions of the field variables and time dependent modal coefficients for training the LSTM architecture. The LSTM architecture is trained for the modal coefficients based on a preselected lookback time-window, σ. In the online phase, the trained model is used to predict the modal coefficients recursively for the total time based on initial time history and σ. Finally, we reconstruct the mean fields for analyses using the predicted coefficients, precomputed basis functions, and mean field values.

We demonstrate the performance of the ROM-LSTM through time series evolution of modal coefficients and mean vorticity and streamfunction fields. To assess the performance of the proposed model, the ROM-LSTM predictions are compared with the high-dimensional solutions as well as with the conventional Galerkin projection based ROM (ROM-GP) solutions. We find that the ROM-LSTM predictions are stable and accurate even with only a couple of POD modes. On the other hand, the ROM-GP framework, as expected, requires a very large number of modes to obtain a physically stable solution, since the ROM-GP framework is susceptible to numerical instability in quasi-stationary flow fields. We further observe that the ROM-LSTM framework gives accurate and physical predictions based on a few time history data points. Indeed, if we increase the value of σ, the prediction accuracy will increase, but the computational cost of offline training and online prediction will also go up. To quantify the accuracy of the prediction of ROM-LSTM framework, we present the L_2-norm errors for ROM-GP and ROM-LSTM frameworks, which show that the proposed framework trained with 10 modes and $\sigma = 5$ gets a better accuracy than the ROM-GP predictions with 40 or 80 modes.

Based on our findings, we conclude that the ROM-LSTM framework is a better tool than the ROM-GP framework for large-scale quasi-stationary flows in terms of prediction and reduced order modeling. Since the ROM-LSTM framework is fully non-intrusive, it does not rely on the governing equations to obtain the solution, which means that there are no numerical constraints while predicting the solutions.

Additionally, it is computationally more efficient to predict the solution using a trained model rather than the physics-based approach of solving ODEs. Hence, the proposed ROM-LSTM framework can be considered a very promising approach in developing a robust and efficient ROMs for large-scale flows with noisy spatio-temporal behavior. In our future works, we will focus on testing the ROM-LSTM framework in more complex three-dimensional turbulent flow problems. We also plan to improve the existing framework based on our findings, and implement the updated framework in suitable ROM applications, such as, flow control, uncertainty quantification, and data assimilation.

CHAPTER V

Concluding Remarks

In this chapter, we summarize the major aspects of this work, draw some key conclusions based on the findings from different numerical experiments, and finally articulate several potential research avenues that may emerge from this thesis work.

5.1 Summary and Discussions

In this thesis, we propose three distinct ROM ideas to improve POD based ROM framework for quasi-stationary geophysical flows. As a representative test problem of geophysical flows, we consider the simple single-layer QG ocean circulation model to investigate the robustness of the proposed frameworks. In the current age of massive amount of data resources, researchers are more keen to build modeling algorithms based on existing data information. One benefit of building models based on a data set is to get rid of dependence on underlying equation of a system. Hence, the same model can be utilized for any kind of data set with similar trend irrespective to the governing equations. It has been accepted to the turbulence modeling community that ROM has great potential to reduce the computational expenses significantly which can make large-scale turbulent flow simulations manageable with acceptable range of accuracy. However, developing an efficient and robust ROM model has still been a subject of active research. The main motivation of conducting this study is to develop efficient and robust ROM frameworks for complex large-scale flow problems using physics-based, data-driven, or hybrid of both approaches, solution methodologies. In the physics-based solution approach, we put forth a dynamic closure modeling

framework for ROM (ROM-D) based on the idea of test truncation, analogous to the test filtering in dynamic eddy viscosity model in LES, in order to stabilize the POD and Galerkin projection based ROM. Since the solution is obtained using the underlying governing equation, this solution approach is considered as physics-based approach. In the data-driven solution approach, we propose a fully non-intrusive ROM framework (ROM-LSTM) using the time series prediction capability of LSTM recurrent neural network for noisy dynamical systems. Since the training and prediction is done by LSTM architecture, this framework is fully free of underlying governing equation to obtain the solution. For the hybrid ROM approach, we demonstrate one way of developing hybrid ROM model (ROM-H) by combining the standard Galerkin projection based ROM (physics-based) and ELM neural network architecture (data-driven) to obtain the solutions. It should be noted that the hybridization can be done many other different ways using different variants of physics-based and data-driven architectures. We also present a qualitative and quantitative assessments of the proposed models with respect to the full order model simulation and other types of ROMs using the mean flow field contours and time series evolution of the flow field. We also present the strengths and weaknesses of different solution strategies in chapter 3. To outline the conclusions, following we present the key sentences drawn from different numerical experiments in this study:

1. The physics-based ROM-D model with 10 modes predicts the true solution with a same order of accuracy as a standard Galerkin projection ROM solution with 80 modes. This implies considerable savings in terms of both storage needs and computing time. Since the ROM-D model includes dissipative contributions from truncated modes through the stabilization parameter, it is expected to capture a greater percentage of the energy in the system using lower R values in ROM-D model. As a result, we can see a huge reduction in overall computational overhead for equally accurate solution using ROM-D model.

2. For physics-based dynamic ROM approach, the higher modes combined with different values of test truncation exhibits more gain in accuracy of the predictions. The model displays a consistent prediction for 20 modes irrespective to different test truncation values which indicates the robustness of the ROM-D model.

3. The ROM-D model shows better extrapolatory predictive performance than the conventional projection based ROM for the same value of POD modes.

4. The ROM-H model proposed in chapter 3 is shown to have better predictive performance than the standard physics-based ROM and data-driven ROM. The ROM-H model performs better for long time forecasting and out of sample prediction as well.

5. Even though there is a compromise in simulation time for ROM-H model, it is observed that the ROM-H model is successful in leveraging the advantages of component ROM models to generate an accurate solution using a very lower number of POD modes.

6. The non-intrusive ROM-LSTM framework shows a stable and accurate prediction even with only a couple of POD modes and without any prior knowledge about the underlying governing equation. The prediction accuracy of ROM-LSTM model with 10 modes is equivalent to the conventional ROM predictions with 80 modes.

7. The ROM-LSTM framework gives a consistent prediction based on a very few time history data for LSTM model training. The prediction is stable and accurate for the out of sample data which indicates the long time forecasting capability of the ROM-LSTM model for noisy dynamical systems.

5.2 Future Work

In future, many engineered systems will have plenty of sensor data and will require fast, real-time simulation and monitoring of complex systems for on the spot analysis, control and decision-making in the context of Digital Twin a possibility (Damjanovic-Behrendt and Behrendt, 2019; Kalashnikova and Barone, 2010; Tao et al., 2018). A Digital Twin may be defined as: the virtual representation of a physical object or system across its life cycle, using real-time data and other sources to enable understanding, learning, reasoning, and dynamic re-calibration for improved decision-making (Grieves, 2011). Some examples of such systems are self-aware aerospace vehicles, smart manufacturing, real-time adaptive emergency response, quantification of the compressible captive-carry environment, and so on. In Ref. (Tao et al., 2018) and the references therein, the concept and development of Digital Twin and their future prospect and current challenges in the industry applications are reviewed in details. To develop Digital Twin in complex environments, the system will require adaptive and computationally efficient data-enabled modeling frameworks and algorithms which can leverage the advantages of existing data. Since ROMs are by construction small in terms of computational memory and fast, they can be a good fit for these purposes. Using an offline/online decomposition of tasks, ROMs can enable rapid prediction, inversion, control, and uncertainty quantification of large-scale scientific and engineering systems (Aguado et al., 2017; Belkadi et al., 2019; Borzacchiello et al., 2019, 2017). Indeed, there exists a number of reduced order modeling techniques and we yet need to address several shortcomings of conventional ROM frameworks to reach fully application level. Hence, in this thesis work, we concentrate on improving the performance of POD based ROM by utilizing several modeling strategies and the proposed models are observed to improve the performance significantly. As a continuation of this work, the following studies can be devised:

1. In this study, we implement the proposed models in a static fashion, i.e., we

collect a set of data, then develop a new reduced model from scratch for that data set only for prediction. However, to do real-time analysis, we will encounter a rapid change in input data set and it will become costly to recourse to the high-dimensional system every time. Hence, we can implement the proposed ROMs in a dynamical sense where the reduced models will be adapted during online phase with respect to dynamically collected data from simulations (Peherstorfer and Willcox, 2015). This will enable us to do online control, optimization, uncertainty quantification type of applications using ROMs.

2. Using unsupervised machine learning strategy like k-means clustering, cluster set of snapshot data can be generated to construct separate local ROM for each cluster. During the online prediction phase, the state of the input data stream can be classified and the local ROM can be selected accordingly (Peherstorfer et al., 2014).

3. The proposed ROMs can be tested for parametric industrial applications in CFD as a future work. Rozza et al. presented an overview of the challenges faced and possible solutions in using ROMs for industrial CFD problems (Rozza et al., 2018).

4. ROMs can be utilized to reduce the computational cost associated with the optimization loops in data assimilation. Data assimilation is a powerful technique which provides an optimal state estimate of an evolving system , widely used in operational weather and ocean forecasting, satellite observations and so on. An efficient ROM framework, as proposed in this study, can enhance the performance efficiency of data assimilation as well.

5. There have been a number of published literature on the flow control applications using ROMs which can also be a future research direction of testing the proposed ROM frameworks.

6. The development of Digital Twin for the offshore wind power generation can be a promising future prospect of developing efficient ROM frameworks for wind driven ocean circulation model. To harvest offshore wind energy, the wind farms are constructed in the bodies of water. Based on the collection of sparse data from different wind turbines of the wind farms through sensors, the direction of the wind turbine blades can be manipulated using real-time simulation to generate more energy or to reduce the damage on the interior and exterior of the wind turbines due to terrain induced turbulence.

References

Aguado, J. V., Borzacchiello, D., Ghnatios, C., Lebel, F., Upadhyay, R., Binetruy, C., and Chinesta, F. (2017). A Simulation App based on reduced order modeling for manufacturing optimization of composite outlet guide vanes. *Advanced Modeling and Simulation in Engineering Sciences*, 4(1):1.

Akhtar, I., Nayfeh, A. H., and Ribbens, C. J. (2009). On the stability and extension of reduced-order Galerkin models in incompressible flows. *Theoretical and Computational Fluid Dynamics*, 23(3):213–237.

Akhtar, I., Wang, Z., Borggaard, J., and Iliescu, T. (2012). A new closure strategy for proper orthogonal decomposition reduced-order models. *Journal of Computational and Nonlinear Dynamics*, 7(3):034503.

Amsallem, D. and Farhat, C. (2012). Stabilization of projection-based reduced-order models. *International Journal for Numerical Methods in Engineering*, 91(4):358–377.

Antoulas, A. C., Sorensen, D. C., and Gugercin, S. (2001a). A survey of model reduction methods for large-scale systems. *Contemporary Mathematics*, 280:193–220.

Antoulas, A. C., Sorensen, D. C., and Gugercin, S. (2001b). A survey of model reduction methods for large-scale systems. *Contemporary Mathematics*, 280:193–220.

Arakawa, A. (1966). Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. Part I. *Journal of Computational Physics*, 1(1):119–143.

Aubry, N. (1991). On the hidden beauty of the proper orthogonal decomposition. *Theoretical and Computational Fluid Dynamics*, 2(5-6):339–352.

Aubry, N., Guyonnet, R., and Lima, R. (1991). Spatiotemporal analysis of complex signals: theory and applications. *Journal of Statistical Physics*, 64(3-4):683–739.

Aubry, N., Guyonnet, R., and Lima, R. (1992). Spatio-temporal symmetries and bifurcations via bi-orthogonal decompositions. *Journal of Nonlinear Science*, 2(2):183–215.

Aubry, N., Holmes, P., Lumley, J. L., and Stone, E. (1988). The dynamics of coherent structures in the wall region of a turbulent boundary layer. *Journal of Fluid Mechanics*, 192:115–173.

Baiges, J., Codina, R., and Idelsohn, S. (2015). Reduced-order subscales for POD models. *Computer Methods in Applied Mechanics and Engineering*, 291:173–196.

Balajewicz, M. and Dowell, E. H. (2012). Stabilization of projection-based reduced order models of the Navier–Stokes. *Nonlinear Dynamics*, 70(2):1619–1632.

Balajewicz, M. J., Dowell, E. H., and Noack, B. R. (2013). Low-dimensional modelling of high-Reynolds-number shear flows incorporating constraints from the Navier–Stokes equation. *Journal of Fluid Mechanics*, 729:285–308.

Bassler, K. E., Gunaratne, G. H., and McCauley, J. L. (2006). Markov processes, Hurst exponents, and nonlinear diffusion equations: With application to finance. *Physica A: Statistical Mechanics and its Applications*, 369(2):343–353.

Behzad, F., Helenbrook, B. T., and Ahmadi, G. (2015). On the sensitivity and accuracy of proper-orthogonal-decomposition-based reduced order models for Burgers equation. *Computers & Fluids*, 106:19–32.

Behzad, F., Helenbrook, B. T., and Ahmadi, G. (2018). Multilevel Algorithm for Obtaining the Proper Orthogonal Decomposition. *AIAA Journal*, pages 1–14.

Belkadi, F., Dhuieb, M. A., Aguado, J. V., Laroche, F., Bernard, A., and Chinesta, F. (2019). Intelligent Assistant System as a context-aware decision-making support for the workers of the future. *Computers & Industrial Engineering*.

Bengio, Y., Simard, P., Frasconi, P., et al. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166.

Benner, P., Gugercin, S., and Willcox, K. (2015). A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531.

Bennett, A. F. (2005). *Inverse modeling of the ocean and atmosphere*. Cambridge University Press, New York.

Bergmann, M., Bruneau, C.-H., and Iollo, A. (2009). Enablers for robust POD models. *Journal of Computational Physics*, 228(2):516–538.

Berkooz, G., Holmes, P., and Lumley, J. L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575.

Berloff, P. (2015). Dynamically consistent parameterization of mesoscale eddies. Part I: Simple model. *Ocean Modelling*, 87:1–19.

Berloff, P. (2016). Dynamically Consistent Parameterization of Mesoscale EddiesPart II: Eddy Fluxes and Diffusivity from Transient Impulses. *Fluids*, 1(3):22.

Bistrian, D. and Navon, I. (2015). An improved algorithm for the shallow water equations model reduction: Dynamic Mode Decomposition vs POD. *International Journal for Numerical Methods in Fluids*, 78(9):552–580.

Bistrian, D. A. and Navon, I. M. (2017). Randomized dynamic mode decomposition for nonintrusive reduced order modelling. *International Journal for Numerical Methods in Engineering*, 112(1):3–25.

Borggaard, J., Iliescu, T., and Wang, Z. (2011). Artificial viscosity proper orthogonal decomposition. *Mathematical and Computer Modelling*, 53(1-2):269–279.

Borzacchiello, D., Aguado, J. V., and Chinesta, F. (2017). Reduced order modelling for efficient numerical optimisation of a hot-wall chemical vapour deposition reactor. *International Journal of Numerical Methods for Heat & Fluid Flow*, 27(7):1602–1622.

Borzacchiello, D., Aguado, J. V., and Chinesta, F. (2019). Non-intrusive sparse subspace learning for parametrized problems. *Archives of Computational Methods in Engineering*, 26(2):303–326.

Bouchet, F. and Simonnet, E. (2009). Random changes of flow topology in two-dimensional and geophysical turbulence. *Physical Review Letters*, 102(9):094504.

Brenner, T. A., Fontenot, R. L., Cizmas, P. G., OBrien, T. J., and Breault, R. W. (2012). A reduced-order model for heat transfer in multiphase flow and practical aspects of the proper orthogonal decomposition. *Computers & Chemical Engineering*, 43:68–80.

Brunton, S., Noack, B., and Koumoutsakos, P. (2019). Machine Learning for Fluid Mechanics. *arXiv preprint arXiv:1905.11075*.

Brunton, S. L. and Noack, B. R. (2015). Closed-loop turbulence control: Progress and challenges. *Applied Mechanics Reviews*, 67(5):050801.

Buljak, V. (2011). *Inverse analyses with model reduction: proper orthogonal decomposition in structural mechanics*. Springer Science & Business Media, New York.

Burkardt, J., Gunzburger, M., and Lee, H.-C. (2006). POD and CVT-based reduced-order modeling of Navier–Stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 196(1-3):337–355.

Byrne, D., Münnich, M., Frenger, I., and Gruber, N. (2016). Mesoscale atmosphere ocean coupling enhances the transfer of wind energy into the ocean. *Nature Communications*, 7:ncomms11867.

Cancelliere, R., Deluca, R., Gai, M., Gallinari, P., and Rubini, L. (2017). An analysis of numerical issues in neural training by pseudoinversion. *Computational and Applied Mathematics*, 36(1):599–609.

Cao, Y., Zhu, J., Navon, I. M., and Luo, Z. (2007). A reduced-order approach to four-dimensional variational data assimilation using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 53(10):1571–1583.

Carbone, A., Castelli, G., and Stanley, H. E. (2004). Time-dependent Hurst exponent in financial time series. *Physica A: Statistical Mechanics and its Applications*, 344(1-2):267–271.

Carlberg, K., Bou-Mosleh, C., and Farhat, C. (2011). Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181.

Cazemier, W. (1997). *Proper orthogonal decomposition and low dimensional models for turbulent flows*. PhD thesis, Rijksuniversiteit Groningen.

Cazemier, W., Verstappen, R., and Veldman, A. (1998). Proper orthogonal decomposition and low-dimensional models for driven cavity flows. *Physics of Fluids*, 10(7):1685–1699.

Chandra, B. S., Sastry, C. S., and Jana, S. (2013). Telecardiology: Hurst exponent based anomaly detection in compressively sampled ECG signals. In *2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013)*, pages 350–354. IEEE.

Charney, J. G. (1949). On a physical basis for numerical prediction of large-scale motions in the atmosphere. *Journal of Meteorology*, 6(6):372–385.

Charney, J. G., Fjörtoft, R., and Neumann, J. v. (1950). Numerical integration of the barotropic vorticity equation. *Tellus*, 2(4):237–254.

Chen, K. K., Tu, J. H., and Rowley, C. W. (2012). Variants of dynamic mode decomposition: boundary condition, Koopman, and Fourier analyses. *Journal of Nonlinear Science*, 22(6):887–915.

Chen, P. and Quarteroni, A. (2015). A new algorithm for high-dimensional uncertainty quantification based on dimension-adaptive sparse grid approximation and reduced basis methods. *Journal of Computational Physics*, 298:176–193.

Chinesta, F., Ladeveze, P., and Cueto, E. (2011). A short review on model order reduction based on proper generalized decomposition. *Archives of Computational Methods in Engineering*, 18(4):395.

Chollet, F. et al. (2015). Keras. `https://keras.io`.

Cizmas, P. G., Richardson, B. R., Brenner, T. A., OBrien, T. J., and Breault, R. W. (2008). Acceleration techniques for reduced-order models based on proper orthogonal decomposition. *Journal of Computational Physics*, 227(16):7791–7812.

Cordier, L., El Majd, B. A., and Favier, J. (2010). Calibration of POD reduced-order models using Tikhonov regularization. *International Journal for Numerical Methods in Fluids*, 63(2):269–296.

Cordier, L., Noack, B. R., Tissot, G., Lehnasch, G., Delville, J., Balajewicz, M., Daviller, G., and Niven, R. K. (2013). Identification strategies for model-based control. *Experiments in Fluids*, 54(8):1580.

Couplet, M., Sagaut, P., and Basdevant, C. (2003). Intermodal energy transfers in a proper orthogonal decomposition–Galerkin representation of a turbulent separated flow. *Journal of Fluid Mechanics*, 491:275–284.

Cummins, P. F. (1992). Inertial gyres in decaying and forced geostrophic turbulence. *Journal of Marine Research*, 50(4):545–566.

Cushman-Roisin, B. and Beckers, J.-M. (2011). *Introduction to geophysical fluid dynamics: physical and numerical aspects*. Academic Press, Waltham, MA.

Cushman-Roisin, B. and Manga, M. (1995). Introduction to Geophysical Fluid Dynamics. *Pure and Applied Geophysics*, 144(1):177–178.

Daescu, D. and Navon, I. (2008). A dual-weighted approach to order reduction in 4DVAR data assimilation. *Monthly Weather Review*, 136(3):1026–1041.

Daescu, D. N. and Navon, I. M. (2007). Efficiency of a POD-based reduced second-order adjoint model in 4D-Var data assimilation. *International Journal for Numerical Methods in Fluids*, 53(6):985–1004.

Damjanovic-Behrendt, V. and Behrendt, W. (2019). An open source approach to the design and implementation of Digital Twins for Smart Manufacturing. *International Journal of Computer Integrated Manufacturing*, pages 1–19.

Duraisamy, K., Iaccarino, G., and Xiao, H. (2018). Turbulence Modeling in the Age of Data. *arXiv preprint arXiv:1804.00183*.

Duraisamy, K., Zhang, Z. J., and Singh, A. P. (2015). New approaches in turbulence

and transition modeling using data-driven techniques. In *53rd AIAA Aerospace Sciences Meeting*, page 1284.

Durbin, P. A. (2018). Some recent developments in turbulence closure modeling. *Annual Review of Fluid Mechanics*, 50:77–103.

Esteban, M. and Leary, D. (2012). Current developments and future prospects of offshore wind and ocean energy. *Applied Energy*, 90(1):128–136.

Evensen, G. (2009). *Data assimilation: the ensemble Kalman filter*. Springer Science & Business Media, New York.

Faller, W. E. and Schreck, S. J. (1997). Unsteady fluid mechanics applications of neural networks. *Journal of Aircraft*, 34(1):48–55.

Freno, B. A. and Cizmas, P. G. (2014). A proper orthogonal decomposition method for nonlinear flows with deforming meshes. *International Journal of Heat and Fluid Flow*, 50:145–159.

Gamahara, M. and Hattori, Y. (2017). Searching for turbulence models by artificial neural network. *Physical Review Fluids*, 2(5):054604.

Gamboa, J. C. B. (2017). Deep learning for time-series analysis. *arXiv preprint arXiv:1701.01887*.

Gautier, N., Aider, J.-L., Duriez, T., Noack, B., Segond, M., and Abel, M. (2015). Closed-loop separation control using machine learning. *Journal of Fluid Mechanics*, 770:442–457.

Germano, M., Piomelli, U., Moin, P., and Cabot, W. H. (1991). A dynamic subgrid-scale eddy viscosity model. *Physics of Fluids A: Fluid Dynamics*, 3(7):1760–1765.

Gers, F. A., Eck, D., and Schmidhuber, J. (2002). Applying LSTM to time series predictable through time-window approaches. In *Neural Nets WIRN Vietri-01*, pages 193–200. Springer, New York, United States.

Ghil, M., Chekroun, M. D., and Simonnet, E. (2008). Climate dynamics and fluid mechanics: Natural variability and related uncertainties. *Physica D: Nonlinear Phenomena*, 237(14-17):2111–2126.

Giere, S., Iliescu, T., John, V., and Wells, D. (2015). SUPG reduced order models for convection-dominated convection–diffusion–reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 289:454–474.

Gottlieb, S. and Shu, C.-W. (1998). Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, 67(221):73–85.

Greatbatch, R. J. and Nadiga, B. T. (2000). Four-gyre circulation in a barotropic model with double-gyre wind forcing. *Journal of Physical Oceanography*, 30(6):1461–1471.

Grieves, M. (2011). *Virtually perfect: Driving innovative and lean products through product lifecycle management*. Space Coast Press.

Griffa, A. and Salmon, R. (1989). Wind-driven ocean circulation and equilibrium statistical mechanics. *Journal of Marine Research*, 47(3):457–492.

Gunzburger, M. D. (2012). *Flow control*. Springer Science & Business Media, Berlin.

Haasdonk, B., Urban, K., and Wieland, B. (2013). Reduced basis methods for parameterized partial differential equations with stochastic influences using the Karhunen–Loeve expansion. *SIAM/ASA Journal on Uncertainty Quantification*, 1(1):79–105.

Hagan, M. T., Demuth, H. B., Beale, M. H., and De Jesús, O. (1996). *Neural network design*, volume 20. Pws Pub. Boston.

He, J., Sarma, P., and Durlofsky, L. J. (2011). Use of reduced-order models for improved data assimilation within an EnKF context. In *SPE Reservoir Simulation Symposium, 21-23 February, The Woodlands, Texas, USA*. Society of Petroleum Engineers.

Hesthaven, J. S., Rozza, G., and Stamm, B. (2015). *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer, New York, United States.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hoffman, J. D. and Frankel, S. (2001). *Numerical methods for engineers and scientists*. CRC Press, New York.

Hogg, A. M. C., Dewar, W. K., Berloff, P., Kravtsov, S., and Hutchinson, D. K. (2009). The effects of mesoscale ocean–atmosphere coupling on the large-scale ocean circulation. *Journal of Climate*, 22(15):4066–4082.

Holland, W. R. (1978). The role of mesoscale eddies in the general circulation of the oceanNumerical experiments using a wind-driven quasi-geostrophic model. *Journal of Physical Oceanography*, 8(3):363–392.

Holland, W. R. and Rhines, P. B. (1980). An example of eddy-induced ocean circulation. *Journal of Physical Oceanography*, 10(7):1010–1031.

Holm, D. D. and Nadiga, B. T. (2003). Modeling mesoscale turbulence in the barotropic double-gyre circulation. *Journal of Physical Oceanography*, 33(11):2355–2365.

Holmes, P., Lumley, J. L., Berkooz, G., and Rowley, C. W. (2012). *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge University Press, New York.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Houtekamer, P. L. and Mitchell, H. L. (1998). Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, 126(3):796–811.

Houtekamer, P. L. and Mitchell, H. L. (2001). A sequential ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 129(1):123–137.

Hua, B. and Haidvogel, D. (1986). Numerical simulations of the vertical structure of quasi-geostrophic turbulence. *Journal of the Atmospheric Sciences*, 43(23):2923–2936.

Huang, G.-B. and Babri, H. A. (1998). Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. *IEEE Transactions on Neural Networks*, 9(1):224–229.

Huang, G.-B. and Chen, L. (2007). Convex incremental extreme learning machine. *Neurocomputing*, 70(16-18):3056–3062.

Huang, G.-B., Wang, D. H., and Lan, Y. (2011). Extreme learning machines: a survey. *International Journal of Machine Learning and Cybernetics*, 2(2):107–122.

Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. (2006). Extreme learning machine: theory and applications. *Neurocomputing*, 70(1):489–501.

Hurst, H. E. (1951). Long-term storage capacity of reservoirs. *Transactions of the American Society of Civil Engineers*, 116:770–799.

Iollo, A., Lanteri, S., and Désidéri, J.-A. (2000). Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396.

Ito, K. and Ravindran, S. (1998). A reduced-order method for simulation and control of fluid flows. *Journal of Computational Physics*, 143(2):403–425.

Jaeger, H. and Haas, H. (2004). Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication. *Science*, 304(5667):78–80.

Jang, J.-S. R., Sun, C.-T., and Mizutani, E. (1997). Neuro-fuzzy and soft computing; a computational approach to learning and machine intelligence. *IEEE Transactions on Automatic Control*, 42(10):1482–1484.

Kalashnikova, I. and Barone, M. (2010). On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment. *International Journal for Numerical Methods in Engineering*, 83(10):1345–1375.

Kalb, V. L. and Deane, A. E. (2007). An intrinsic stabilization scheme for proper orthogonal decomposition based low-dimensional models. *Physics of Fluids*, 19(5):054106.

Kalnay, E. (2003). *Atmospheric modeling, data assimilation and predictability*. Cambridge University Press, New York.

Kani, J. N. and Elsheikh, A. H. (2017). DR-RNN: A deep residual recurrent neural network for model reduction. *arXiv preprint arXiv:1709.00939*.

Kasun, L. L. C., Zhou, H., Huang, G.-B., and Vong, C. M. (2013). Representational learning with extreme learning machine for big data. *IEEE Intelligent Systems*, 28(6):31–34.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kondrashov, D., Chekroun, M. D., and Berloff, P. (2018). Multiscale Stuart-Landau Emulators: Application to Wind-Driven Ocean Gyres. *Fluids*, 3(1):21.

Kosambi, D. D. (2016). Statistics in function space. In Ramaswamy, R., editor, *D D Kosambi*, pages 115–123. Springer, New Delhi, India.

Koutsoyiannis, D. (2003). Climate change, the Hurst phenomenon, and hydrological statistics. *Hydrological Sciences Journal*, 48(1):3–24.

Kumar, S. (2018). The end of Moores law and reinventing computing. In Choi, J. H., editor, *High-Speed and Lower Power Technologies: Electronics and Photonics*. CRC Press, Boca Raton.

Kunisch, K. and Volkwein, S. (2002). Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical Analysis*, 40(2):492–515.

Kutz, J. N. (2017). Deep learning in fluid dynamics. *Journal of Fluid Mechanics*, 814:1–4.

Lahmiri, S. and Boukadoum, M. (2014). New approach for automatic classification of Alzheimer's disease, mild cognitive impairment and healthy brain magnetic resonance images. *Healthcare Technology Letters*, 1(1):32–36.

Lapedes, A. S. and Farber, R. M. (1988). How neural nets work. In *Neural Information Processing Systems*, pages 442–456.

Lassila, T., Manzoni, A., Quarteroni, A., and Rozza, G. (2013). Model order reduction in fluid dynamics: challenges and perspectives. In Quarteroni, A. and Rozza, G., editors, *Reduced Order Methods for Modeling and Computational Reduction*. Springer, Milano.

Lassila, T., Manzoni, A., Quarteroni, A., and Rozza, G. (2014). Model order reduction in fluid dynamics: challenges and perspectives. In *Reduced Order Methods for Modeling and Computational Reduction*, pages 235–273. Springer.

Laurie, J. and Bouchet, F. (2015). Computation of rare transitions in the barotropic quasi-geostrophic equations. *New Journal of Physics*, 17(1):015009.

Law, K. J. and Stuart, A. M. (2012). Evaluating data assimilation algorithms. *Monthly Weather Review*, 140(11):3757–3782.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436.

Lee, C., Kim, J., Babcock, D., and Goodman, R. (1997). Application of neural networks to turbulence control for drag reduction. *Phys. Fluids*, 9(6):1740–1747.

Lilly, D. K. (1992). A proposed modification of the Germano subgrid-scale closure method. *Physics of Fluids*, 4(3):633–635.

Lin, C.-J., Weng, R. C., and Keerthi, S. S. (2008). Trust region newton method for logistic regression. *Journal of Machine Learning Research*, 9(Apr):627–650.

Loiseau, J.-C., Noack, B. R., and Brunton, S. L. (2018). Sparse reduced-order modelling: sensor-based dynamics to full-state estimation. *Journal of Fluid Mechanics*, 844:459–490.

Lucia, D. J., Beran, P. S., and Silva, W. A. (2004). Reduced-order modeling: new approaches for computational physics. *Progress in Aerospace Sciences*, 40(1-2):51–117.

Lumley, J. L. (1967). The structure of inhomogeneous turbulent flows. *Atmospheric Turbulence and Radio Wave Propagation*, pages 166–178.

Lumley, J. L. (1970). *Stochastic tools in turbulence*. Academic Press, New York.

Lynch, P. (2008). The origins of computer weather prediction and climate modeling. *Journal of Computational Physics*, 227(7):3431–3444.

Mack, C. A. (2011). Fifty years of Moore's law. *IEEE Transactions on Semiconductor Manufacturing*, 24(2):202–207.

Majda, A. and Wang, X. (2006). *Nonlinear dynamics and statistical theories for basic geophysical flows*. Cambridge University Press, New York.

Mandelbrot, B. B. and Wallis, J. R. (1968). Noah, Joseph, and operational hydrology. *Water Resources Research*, 4(5):909–918.

Mandelbrot, B. B. and Wallis, J. R. (1969). Robustness of the rescaled range R/S in the measurement of noncyclic long run statistical dependence. *Water Resources Research*, 5(5):967–988.

Maulik, R. and San, O. (2017). A neural network approach for the blind deconvolution of turbulent flows. *Journal of Fluid Mechanics*, 831:151–181.

Maulik, R., San, O., Rasheed, A., and Vedula, P. (2018). Data-driven deconvolution for large eddy simulations of Kraichnan turbulence. *Physics of Fluids*, 30(12):125109.

McWilliams, J. C. (2006). *Fundamentals of geophysical fluid dynamics*. Cambridge University Press, New York.

Medjo, T. T. (2000). Numerical simulations of a two-layer quasi-geostrophic equation of the ocean. *SIAM Journal on Numerical Analysis*, 37(6):2005–2022.

Milk, R., Rave, S., and Schindler, F. (2016). pyMOR–generic algorithms and interfaces for model order reduction. *SIAM Journal on Scientific Computing*, 38(5):S194–S216.

Miller, R. N. (2007). *Numerical modeling of ocean circulation*. Cambridge University Press.

Mohan, A. T. and Gaitonde, D. V. (2018). A deep learning based approach to reduced order modeling for turbulent flow control using LSTM neural networks. *arXiv preprint arXiv:1804.09269*.

Mohebujjaman, M., Rebholz, L., and Iliescu, T. (2018). Physically-constrained data-driven, filtered reduced order modeling of fluid flows. *arXiv:1806.00350*.

Mohebujjaman, M., Rebholz, L. G., and Iliescu, T. (2019). Physically constrained data-driven correction for reduced-order modeling of fluid flows. *International Journal for Numerical Methods in Fluids*, 89(3):103–122.

Moin, P. (2010). *Fundamentals of engineering numerical analysis*. Cambridge University Press, New York.

Montano, J. and Palmer, A. (2003). Numeric sensitivity analysis applied to feedforward neural networks. *Neural Computing & Applications*, 12(2):119–125.

Moore, G. E. (1965). Cramming more components onto integrated circuits. *Electronics*, 38:1–4.

Moore, G. E. (1975). Progress in digital integrated electronics. In *Electron Devices Meeting*, volume 21, pages 11–13.

Moosavi, A., Stefanescu, R., and Sandu, A. (2015). Efficient Construction of Local Parametric Reduced Order Models Using Machine Learning Techniques. *arXiv preprint arXiv:1511.02909*.

Munk, W. H. (1950). On the wind-driven ocean circulation. *Journal of Meteorology*, 7(2):80–93.

Munk, W. H. and Wunsch, C. I. (1982). Observing the ocean in the 1990s. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 307(1499):439–464.

Nadiga, B. T. and Margolin, L. G. (2001). Dispersive–dissipative eddy parameterization in a barotropic model. *Journal of Physical Oceanography*, 31(8):2525–2531.

Navon, I. M. (2009). Data assimilation for numerical weather prediction: a review. In Park, S. K. and Xu, L., editors, *Data assimilation for atmospheric, oceanic and hydrologic applications*, pages 21–65. Springer, New York.

Navon, I. M., Zou, X., Derber, J., and Sela, J. (1992). Variational data assimilation with an adiabatic version of the NMC spectral model. *Monthly Weather Review*, 120(7):1433–1446.

Noack, B. R., Afanasiev, K., Morzynski, M., Tadmor, G., and Thiele, F. (2003). A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497(1):335–363.

Noack, B. R., Fernex, D., and Semaan, R. (2017). xROM–a Toolkit for Reduced Order Modelling of Fluid Flows (Version 1.0). Technical report, Reference Manual, LIMSI-CNRS and TU Braunschweig.

Noack, B. R., Morzynski, M., and Tadmor, G. (2011). *Reduced-order modelling for flow control*, volume 528. Springer Science & Business Media, Berlin, Germany.

Östh, J., Noack, B. R., Krajnović, S., Barros, D., and Borée, J. (2014). On the need for a nonlinear subscale turbulence term in POD models as exemplified for a high-Reynolds-number flow over an Ahmed body. *Journal of Fluid Mechanics*, 747:518–544.

Özgökmen, T. M. and Chassignet, E. P. (1998). Emergence of inertial gyres in a two-layer quasigeostrophic ocean model. *Journal of Physical Oceanography*, 28(3):461–484.

Pathak, J., Wikner, A., Fussell, R., Chandra, S., Hunt, B. R., Girvan, M., and Ott, E. (2018). Hybrid forecasting of chaotic processes: using machine learning in conjunction with a knowledge-based model. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 28(4):041101.

Peherstorfer, B., Butnaru, D., Willcox, K., and Bungartz, H.-J. (2014). Localized discrete empirical interpolation method. *SIAM Journal on Scientific Computing*, 36(1):A168–A192.

Peherstorfer, B. and Willcox, K. (2015). Online adaptive model reduction for nonlinear systems via low-rank updates. *SIAM Journal on Scientific Computing*, 37(4):A2123–A2150.

Pelc, R. and Fujita, R. M. (2002). Renewable energy from the ocean. *Marine Policy*, 26(6):471–479.

Pinnau, R. (2008). Model reduction via proper orthogonal decomposition. In Schilders, W. H. A., van der Vorst, A. H., and Rommes, J., editors, *Model Order Reduction: Theory, Research Aspects and Applications*, pages 95–109. Springer.

Powell, J. R. (2008). The quantum limit to Moore's law. *Proceedings of the IEEE*, 96(8):1247–1248.

Powers, J. G., Klemp, J. B., Skamarock, W. C., Davis, C. A., Dudhia, J., Gill, D. O., Coen, J. L., Gochis, D. J., Ahmadov, R., and Peckham, S. E. (2017). The weather research and forecasting model: Overview, system efforts, and future directions. *Bulletin of the American Meteorological Society*, 98(8):1717–1737.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1992). *Numerical Recipes in FORTRAN*. Cambridge University Press, New York.

Protas, B., Noack, B. R., and Östh, J. (2015). Optimal nonlinear eddy viscosity in Galerkin models of turbulent flows. *Journal of Fluid Mechanics*, 766:337–367.

Puzyrev, V., Ghommem, M., and Meka, S. (2019). pyROM: A computational framework for reduced order modeling. *Journal of Computational Science*, 30:157–173.

Qian, B. and Rasheed, K. (2004). Hurst exponent and financial market predictability. In *IASTED conference on Financial Engineering and Applications*, pages 203–209.

Quarteroni, A., Manzoni, A., and Negri, F. (2015). *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, New York.

Rahman, S., Ahmed, S. E., San, O., et al. (2019a). A dynamic closure modeling framework for model order reduction of geophysical flows. *Physics of Fluids*, 31(04):046602.

Rahman, S., San, O., and Rasheed, A. (2018). A hybrid approach for model order reduction of barotropic quasi-geostrophic turbulence. *Fluids*, 3(4):86.

Rahman, S. M., Pawar, S., San, O., Rasheed, A., and Iliescu, T. (2019b). A non-intrusive reduced order modeling framework for quasi-geostrophic turbulence.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2018). Multistep neural networks for data-driven discovery of nonlinear dynamical systems. *arXiv preprint arXiv:1801.01236*.

Raissi, M., Perdikaris, P., and Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707.

Rapún, M.-L. and Vega, J. M. (2010). Reduced order models based on local POD plus Galerkin projection. *Journal of Computational Physics*, 229(8):3046–3063.

Rempfer, D. (1991). *Koherente strukturen und chaos beim laminar-turbulenten grenzschichtumschlag*. PhD thesis, University of Stuttgart.

Rempfer, D. (1994). On the structure of dynamical systems describing the evolution

of coherent structures in a convective boundary layer. *Physics of Fluids*, 6(3):1402–1404.

Rempfer, D. (2000). On low-dimensional Galerkin models for fluid flow. *Theoretical and Computational Fluid Dynamics*, 14(2):75–88.

Rempfer, D. and Fasel, H. F. (1994a). Dynamics of three-dimensional coherent structures in a flat-plate boundary layer. *Journal of Fluid Mechanics*, 275:257–283.

Rempfer, D. and Fasel, H. F. (1994b). Evolution of three-dimensional coherent structures in a flat-plate boundary layer. *Journal of Fluid Mechanics*, 260:351–375.

Rowley, C. W., Colonius, T., and Murray, R. M. (2004). Model reduction for compressible flows using POD and Galerkin projection. *Physica D: Nonlinear Phenomena*, 189(1-2):115–129.

Rowley, C. W. and Dawson, S. T. (2017). Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49:387–417.

Rowley, C. W. and Williams, D. R. (2006). Dynamics and control of high-Reynolds-number flow over open cavities. *Annual Review of Fluid Mechanics*, 38:251–276.

Rozza, G., Malik, H., Demo, N., Tezzele, M., Girfoglio, M., Stabile, G., and Mola, A. (2018). Advances in Reduced Order Methods for Parametric Industrial Problems in Computational Fluid Dynamics. *arXiv preprint arXiv:1811.08319*.

Rumelhart, D. E., Hinton, G. E., Williams, R. J., et al. (1988). Learning representations by back-propagating errors. *Cognitive Modeling*, 5(3):1.

Sachs, E. W. and Volkwein, S. (2010). POD-Galerkin approximations in PDE-constrained optimization. *GAMM-Mitteilungen*, 33(2):194–208.

Sagaut, P. (2006). *Large eddy simulation for incompressible flows: an introduction*. Springer Science & Business Media, New York.

Sak, H., Senior, A., and Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth annual conference of the international speech communication association*.

San, O. (2016). Numerical assessments of ocean energy extraction from western boundary currents using a quasi-geostrophic ocean circulation model. *International Journal of Marine Energy*, 16:12–29.

San, O. and Iliescu, T. (2014). Proper orthogonal decomposition closure models for fluid flows: Burgers equation. *International Journal of Numerical Analysis & Modeling, Series B*, 5:217–237.

San, O. and Iliescu, T. (2015). A stabilized proper orthogonal decomposition reduced-order model for large scale quasigeostrophic ocean circulation. *Advances in Computational Mathematics*, 41(5):1289–1319.

San, O. and Maulik, R. (2018a). Extreme learning machine for reduced order modeling of turbulent geophysical flows. *Physical Review E*, 97(4):042322.

San, O. and Maulik, R. (2018b). Neural network closures for nonlinear model order reduction. *Advances in Computational Mathematics*, 44:1717–1750.

San, O. and Staples, A. E. (2013a). An efficient coarse grid projection method for quasigeostrophic models of large-scale ocean circulation. *International Journal for Multiscale Computational Engineering*, 11(5).

San, O. and Staples, A. E. (2013b). A coarse-grid projection method for accelerating incompressible flow computations. *Journal of Computational Physics*, 233:480–508.

San, O., Staples, A. E., Wang, Z., and Iliescu, T. (2011). Approximate deconvolution large eddy simulation of a barotropic ocean circulation model. *Ocean Modelling*, 40(2):120–132.

Sieber, M., Paschereit, C. O., and Oberleithner, K. (2016). Spectral proper orthogonal decomposition. *Journal of Fluid Mechanics*, 792:798–828.

Siedler, G., Griffies, S. M., Gould, J., and Church, J. A. (2013). *Ocean circulation and climate: a 21st century perspective*, volume 103. Academic Press.

Sirisup, S. and Karniadakis, G. E. (2004). A spectral viscosity method for correcting the long-term behavior of POD models. *Journal of Computational Physics*, 194(1):92–116.

Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571.

Sun, Z.-L., Choi, T.-M., Au, K.-F., and Yu, Y. (2008). Sales forecasting using extreme learning machine with applications in fashion retailing. *Decision Support Systems*, 46(1):411–419.

Sverdrup, H. U. (1947). Wind-driven currents in a baroclinic ocean; with application to the equatorial currents of the eastern pacific. *Proceedings of the National Academy of Sciences of the United States of America*, 33(11):318.

Taira, K., Brunton, S. L., Dawson, S. T., Rowley, C. W., Colonius, T., McKeon, B. J., Schmidt, O. T., Gordeyev, S., Theofilis, V., and Ukeiley, L. S. (2017). Modal analysis of fluid flows: An overview. *AIAA Journal*, pages 4013–4041.

Tao, F., Zhang, H., Liu, A., and Nee, A. Y. (2018). Digital twin in industry: state-of-the-art. *IEEE Transactions on Industrial Informatics*, 15(4):2405–2415.

Tennekes, H., Lumley, J. L., Lumley, J., et al. (1972). *A first course in turbulence*. MIT press.

Theis, T. N. and Wong, H.-S. P. (2017). The end of Moore's law: a new beginning for information technology. *Computing in Science & Engineering*, 19(2):41.

Ullmann, S. and Lang, J. (2014). POD-Galerkin modeling and sparse-grid collocation for a natural convection problem with stochastic boundary conditions. In *Sparse Grids and Applications-Munich 2012*, pages 295–315. Springer.

Vallis, G. K. (2006). *Atmospheric and Oceanic Fluid Dynamics: Fundamentals and Large-scale Circulation*. Cambridge University Press, New York.

Vallis, G. K. (2017). *Atmospheric and oceanic fluid dynamics*. Cambridge University Press, New York, United States.

Vlachas, P. R., Byeon, W., Wan, Z. Y., Sapsis, T. P., and Koumoutsakos, P. (2018). Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2213):20170844.

Vreman, A. (2004). An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications. *Physics of Fluids*, 16(10):3670–3681.

Waldrop, M. M. (2016). The chips are down for Moore's law. *Nature News*, 530(7589):144.

Wan, Z. Y., Vlachas, P., Koumoutsakos, P., and Sapsis, T. (2018). Data-assisted reduced-order modeling of extreme events in complex dynamical systems. *PloS One*, 13(5):e0197704.

Wang, Q., Hesthaven, J. S., and Ray, D. (2019). Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem. *Journal of Computational Physics*, 384:289–307.

Wang, Z., Akhtar, I., Borggaard, J., and Iliescu, T. (2011). Two-level discretizations of nonlinear closure models for proper orthogonal decomposition. *Journal of Computational Physics*, 230(1):126–146.

Wang, Z., Akhtar, I., Borggaard, J., and Iliescu, T. (2012). Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Computer Methods in Applied Mechanics and Engineering*, 237:10–26.

Wang, Z., Xiao, D., Fang, F., Govindan, R., Pain, C. C., and Guo, Y. (2018). Model identification of reduced order fluid dynamics systems using deep learning. *International Journal for Numerical Methods in Fluids*, 86(4):255–268.

Wedi, N. P. (2014). Increasing horizontal resolution in numerical weather prediction and climate simulations: illusion or panacea? *Philosophical Transactions of the Royal Society A*, 372(2018):20130289.

Wells, D., Wang, Z., Xie, X., and Iliescu, T. (2017). An evolve-then-filter regularized reduced order model for convection-dominated flows. *International Journal for Numerical Methods in Fluids*, 84(10):598–615.

Wilcox, D. C. (1998). *Turbulence modeling for CFD*. DCW Industries, La Canada Flintridge, CA.

Williams, M. O., Schmid, P. J., and Kutz, J. N. (2013). Hybrid reduced-order integration with proper orthogonal decomposition and dynamic mode decomposition. *Multiscale Modeling & Simulation*, 11(2):522–544.

Wright, S. and Nocedal, J. (1999). Numerical optimization. *Springer Science*, 35(67-68):7.

Wu, J.-L., Xiao, H., and Paterson, E. (2018). Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids*, 3(7):074602.

Xiang, J., Westerlund, M., Sovilj, D., and Pulkkis, G. (2014). Using extreme learning machine for intrusion detection in a big data environment. In *Proceedings of the 2014 Workshop on Artificial Intelligent and Security Workshop*, pages 73–82. ACM.

Xie, X., Mohebujjaman, M., Rebholz, L. G., and Iliescu, T. (2018a). Data-driven filtered reduced order modeling of fluid flows. *SIAM Journal on Scientific Computing*, 40(3):B834–B857.

Xie, X., Wells, D., Wang, Z., and Iliescu, T. (2017). Approximate deconvolution reduced order modeling. *Computer Methods in Applied Mechanics and Engineering*, 313:512–534.

Xie, X., Wells, D., Wang, Z., and Iliescu, T. (2018b). Numerical analysis of the Leray reduced order model. *Journal of Computational and Applied Mathematics*, 328:12–29.

Yeo, K. (2019). Short note on the behavior of recurrent neural network for noisy dynamical system. *arXiv preprint arXiv:1904.05158*.

Yeo, K. and Melnyk, I. (2019). Deep learning algorithm for data-driven simulation of noisy dynamical system. *Journal of Computational Physics*, 376:1212–1231.

Zhou, H., Soh, Y. C., Jiang, C., and Wu, X. (2015). Compressed representation learning for fluid field reconstruction from sparse sensor observations. In *2015 International Joint Conference on Neural Networks (IJCNN)*.

VITA

Sk. Mashfiqur Rahman

Candidate for the Degree of

Master of Science

Thesis: REDUCED ORDER MODELING OF GEOPHYSICAL FLOWS USING PHYSICS-BASED AND DATA-DRIVEN MODELING TECHNIQUES

Major Field: Mechanical & Aerospace Engineering

Biographical:

Education:

Completed the requirements for the degree of Master of Science with a major in Mechanical Engineering at Oklahoma State University in July 2019.

Received a Bachelors of Science in Mechanical Engineering at Bangladesh University of Engineering and Technology, Dhaka, Bangladesh in March 2016.

Experience:

Graduate Research Assistant, Computational Fluid Dynamics Laboratory, Oklahoma State University.

Professional Affiliations:

Student member in American Physical Society (APS) and Society for Industrial and Applied Mathematics (SIAM).