REDUCED-ORDER MODELLING OF SHALLOW WATER EQUATIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF APPLIED MATHEMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SÜLEYMAN YILDIZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
SCIENTIFIC COMPUTING

JULY 2021

Approval of the thesis:

REDUCED-ORDER MODELLING OF SHALLOW WATER EQUATIONS

submitted by **SÜLEYMAN YILDIZ** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Scientific Computing Department, Middle East Technical University** by,

Prof. Dr. Sevtap Kestel
Director, Graduate School of **Applied Mathematics** ————————

Assoc. Prof. Dr. Hamdullah Yücel
Head of Department, **Scientific Computing** ————————

Prof. Dr. Bülent Karasözen
Supervisor, **Department of Mathematics &**
Institute of Applied Mathematics, METU ————————

Examining Committee Members:

Prof. Dr. Ömür Uğur
Institute of Applied Mathematics, METU ————————

Prof. Dr. Bülent Karasözen
Department of Mathematics &
Institute of Applied Mathematics, METU ————————

Assoc. Prof. Dr. Hamdullah Yücel
Institute of Applied Mathematics, METU ————————

Assoc. Prof. Dr. Murat Uzunca
Department of Mathematics, Sinop University ————————

Prof. Dr. Ayhan Aydın
Department of Mathematics, Atılım University ————————

Date: ————————

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: SÜLEYMAN YILDIZ

Signature   :

ABSTRACT

REDUCED-ORDER MODELLING OF SHALLOW WATER EQUATIONS

Yıldız, Süleyman

Ph.D., Department of Scientific Computing

Supervisor   : Prof. Dr. Bülent Karasözen

July 2021, 128 pages

The shallow water equations (SWEs) consist of a set of two-dimensional partial differential equations (PDEs) describing a thin inviscid fluid layer flowing over the topography in a frame rotating about an arbitrary axis. SWEs are widely used in modeling large-scale atmosphere/ocean dynamics and numerical weather prediction. High-resolution simulations of the SWEs require long time horizons over global scales when combined with accurate resolution in time and space makes simulations very time-consuming. While high-resolution ocean-modeling simulations are still feasible on large HPC machines, performing many query applications, such as repeated evaluations of the model over a range of parameter values, at these resolutions, is not feasible. Reduced-order modeling enables fast simulation of the PDEs using high-fidelity solutions. In this thesis, reduced-order models (ROMs) are investigated for the rotating SWE, with constant (RSWE) and non-traditional SWE with full Coriolis force (NTSWE), and for rotating thermal SWE (RTSWE) while preserving their non-canonical Hamiltonian-structure, the energy, and Casimir's, i.e. mass, enstrophy, vorticity, and buoyancy. Two different approaches are followed for constructing ROMs; the traditional intrusive model order reduction with Galerkin projection and the data-driven, non-intrusive ROMs. The full order models (FOM) of the SWE, which needed to construct the ROMs are obtained by discretizing the SWE in space by finite differences by preserving the skew-symmetric structure of the Poisson matrix. Applying intrusive proper orthogonal decomposition (POD) with the Galerkin

projection, energy preserving ROMs are constructed for the NTSWE and RTSWE in skew gradient form. Due to the nonlinear terms, the dimension of the reduced-order system scales with the dimension of the FOM. The nonlinearities in the ROM are computed by applying the discrete empirical interpolation (DEIM) method to reduce the computational cost. The computation of the reduced-order solutions is accelerated further by the use of tensor techniques. For the RSWE in linear-quadratic form, the dimension of the reduced solutions is obtained using tensor algebra without necessitating hyper-reduction techniques like the DEIM. Applying POD in a tensorial framework by exploiting matricizations of tensors, the computational cost is further reduced for the rotating SWE in linear-quadratic as well in skew-gradient form. In the data-driven, non-intrusive ROMs are learned only from the snapshots by solving an appropriate least-squares optimization problem in a low-dimensional subspace. Data-driven ROMs are constructed for the NTSWE and RTSWE with the operator inference (OpInf). Computational challenges such as ill-conditioning and regularization are discussed. The non-intrusive model order reduction framework is extended to a parametric case, whereas we make use of the parameter dependency at the level of the PDE without interpolating between the reduced operators.

The intrusive and non-intrusive ROMs in linear-quadratic and skew-gradient form yield a clear separation of the offline and online computational costs. Both ROMs behave similarly and can accurately predict in the test and training data and capture system behavior in the prediction phase. The preservation of physical quantities in the ROMs of the SWEs such as energy (Hamiltonian), and other conserved quantities, i.e., mass, buoyancy, and total vorticity, enables that the models fit better to data and stable solutions are obtained in the long-term predictions which are robust to parameter changes while exhibiting several orders of magnitude computational speed-up over the FOMs.

Keywords: Hamiltonian systems, energy, conserved quantities, proper orthogonal decomposition, discrete empirical interpolation, tensors, operator inference, least-squares

ÖZ

SIĞ SULARDA DALGA DENKLEMLERİ İÇİN MODEL İNDİRGEME YÖNTEMLERİ

Yıldız, Süleyman

Doktora, Bilimsel Hesaplama Bölümü

Tez Yöneticisi   : Prof. Dr. Bülent Karasözen

Temmuz 2021, 128 sayfa

Sığ su denklemleri, rastgele bir eksen etrafında dönen bir çerçeve içinde topografya üzerinden akan ince bir viskoz olmayan sıvı tabakasını tanımlayan bir dizi iki boyutlu kısmi diferansiyel denklemden oluşur. Bu denklemler, büyük ölçekli atmosfer / okyanus dinamiklerinin modellenmesinde ve sayısal hava tahmininde yaygın olarak kullanılmaktadır. Sığ su denklemlerinin yüksek çözünürlüklü simülasyonları, zaman ve mekanda doğru çözünürlükle birleştirildiğinde, küresel ölçekler üzerinde uzun zaman ufku gerektirir, simülasyonları çok zaman alıcı hale getirir. Yüksek çözünürlüklü okyanus modelleme simülasyonları yüksek performanslı makinelerinde hala uygulanabilir olsa da, bu çözünürlüklerde modelin bir dizi parametre değeri üzerinden tekrarlanan değerlendirmeleri gibi birçok tekrarın gerçekleştirilmesi mümkün değildir. Düşük mertebeden modelleme, yüksek kaliteli çözümler kullanarak kısmi diferansiyel denklemlerin hızlı simülasyonunu sağlar. Bu tezde, rotasyonel sığ su denklemleri, sabit ve tam Coriolis kuvvetine sahip geleneksel olmayan sığ su denklemleri ve termal sığ su denklemleri için kanonik olmayan Hamilton yapısını, enerjiyi ve Casimirleri, yani kütleyi, entropiyi, vortisite ve kaldırma kuvveti özelliklerini koruyan indirgenmiş mertebeden modeller incelenmiştir. İndirgenmiş modelleri oluşturmak için iki farklı yaklaşım izlenmiştir; Galerkin projeksiyonu ile geleneksel müdahaleci model indirgeme ve veriye dayalı müdahaleci olmayan model indirgeme yöntemleri. İndirgenmiş modelleri oluşturmak için gerekli olan sığ su denklemlerinin tam

mertebeli modelleri, Poisson matrisinin ters-simetrik yapısı korunarak uzaydaki sığ su denklemlerinin sonlu farklarla ayrıştırılmasıyla elde edilmiştir. Galerkin projeksiyonu ile müdahaleci uygun ortogonal ayrıştırma (POD) uygulayarak, tam Coriolis kuvvetine sahip geleneksel olmayan ve termal sığ su denklemi için çarpık gradyan formunda enerji koruyan indirgenmiş modeller oluşturulmuştur. Doğrusal olmayan terimler nedeniyle, indirgenmiş modelin boyutu, tam modelin boyutuyla ölçeklenir. İndirgenmiş modeldeki doğrusal olmayan terimler, hesaplama maliyetini düşürmek için ayrık deneysel interpolasyon (DEIM) yöntemi uygulanarak hesaplanmıştır. İndirgenmiş modellerin çözümlerin hesaplanması, tensör tekniklerinin kullanılmasıyla daha da hızlandırılır. Lineer-kuadratik formdaki rotasyonel sığ su denklemleri için, indirgenmiş çözümlerin boyutu, DEIM gibi hiper indirgeme tekniklerini gerektirmeden tensör cebiri kullanılarak elde edilir. Tensörlerin matrisizasyonlarından yararlanarak tensörel bir çerçevede POD uygulandığında, doğrusal-kuadratik ve çarpık-gradyan formunda rotasyonel sığ su denklemleri için hesaplama maliyeti daha da azaltılmıştır. Veriye dayalı, müdahaleci olmayan indirgenmiş modeller, düşük boyutlu bir alt uzayda uygun bir en küçük kareler optimizasyon problemini çözerek yalnızca anlık görüntülerden öğrenilir. Veriye dayalı indirgenmiş modeller, tam Coriolis kuvvetine sahip geleneksel olmayan ve termal sığ su denklemi için operatör çıkarımı (OpInf) metodu ile oluşturulur. Kötü koşullandırma ve iyileştirme gibi hesaplama zorlukları tartışılmaktadır. Müdahaleci olmayan model indirgeme çerçevesi parametrik bir duruma genişletilirken, indirgenmiş operatörler arasında enterpolasyon yapmadan kısmi diferansiyel denklemler düzeyinde parametre bağımlılığını kullanıyoruz.

Doğrusal-kuadratik ve çarpık gradyan biçimindeki müdahaleci ve müdahaleci olmayan düşük mertebeden modeller, çevrimdışı ve çevrimiçi hesaplama maliyetlerinin net bir şekilde ayrılmasını sağlar. Her iki düşük mertebeden modelde benzer şekilde davranmıştır ve test ve eğitim verilerinde doğru bir şekilde öngörüde bulunabilmiştir ve tahmin aşamasında sistem davranışını yakalayabilmiştir. Enerji (Hamiltonian) gibi sığ su denklemlerinin indirgenmiş modellerinde fiziksel niceliklerin korunması ve diğer korunan nicelikler, yani kütle, kaldırma kuvveti ve toplam vortisite, modellerin verilere daha iyi uymasını sağlar ve kararlı çözümler tam modeller üzerinden birkaç büyüklük derecesinde hesaplama hızı sergilerken parametre değişikliklerine karşı dayanıklı olan uzun vadeli tahminlerde elde edilir.

Anahtar Kelimeler: Hamilton sistemleri, enerji, korunan nicelikler, uygun ortogonal ayrıştırma, ayrık ampirik interpolasyon, tensörler, operatör çıkarımı, en küçük kareler

To My Wife

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

1.1 Shallow Water Equations

In this thesis, reduced-order models (ROMs) are constructed for the shallow water equations (SWEs) frequently used to model large-scale atmosphere/ocean dynamics and numerical weather prediction. They contain a collection of two-dimensional partial differential equations (PDEs) illustrating a thin inviscid fluid layer circulating over a bottom topography in a frame rotating about an arbitrary axis. Preserving physics in reduced models is the central theme of the thesis. In particular, this work aims to construct a framework for ROMs that retains the Hamiltonian structure. Many geophysical flows can be written in Hamiltonian form [85]. The non-canonical Hamiltonian/Poisson form of the SWE in the rotational frame with constant Coriolis force was introduced first in [103]. Later on, the Nambu formulation of the rotating shallow water equation (RSWE) [104], RSWE with complete Coriolis force, the so-called non-traditional shallow water equation (NTSWE) [41, 113], and multi-layer RSWE [113] are developed. The discrete energy conservation follows from the anti-symmetry of the discrete Poisson bracket. Additional preserved quantities (Casimir) are potential enstrophy, vorticity, and mass. The energy cascades to large scales whilst enstrophy cascades to small scales [8, 113]. The RSWE [103] is a broadly used theoretic model in planetary and geophysical fluid dynamics for investigating rotating inviscid fluids with one or multilayers. The horizontal velocity is supposed to be depth-independent within each layer, so the fluid moves in columns. Yet, the RSWE model does not permit the gradients of the mean density and/or temperature widely in the oceans and atmosphere. The rotating thermal shallow water equation

(RTSWE) [40, 43, 99, 122] represents an extension of the RSWE, which include the horizontal density/temperature gradients in addition. The RTSWE is a simplification of the three-dimensional primitive equations and the Euler-Boussinesq model, which are commonly used for computationally simulating large-scale ocean and atmosphere circulation dynamics, such as the general circulation models [130], planetary flows [123], and modeling atmospheric and oceanic temperature fronts [42, 127], thermal instabilities [53]. Consequently, the energy and the enstrophy preserving numerical schemes of the SWEs lead to stable solutions in the long term integration [4]. High-resolution simulations of the SWEs require long time horizons over global scales. Simulations in time and space are very time-consuming for many query applications, such as repeated evaluations of the model over a range of parameter values. In this thesis, ROMs are constructed for fast simulation of the above mentions three types of SWEs.

1.2 Model Order Reduction

Realistic simulations of SWEs require a fine discretization on space-time grids to obtain reliable high-fidelity solutions. Therefore, real-time simulations need a large amount of computation time and computer memory. Moreover, the computational cost linked with fully resolved simulations remains a barrier in many applications. Model order reduction (MOR) techniques allow the construction of low-dimensional ROMs for the high dimensional full-order models (FOMs), generated by the discretization of PDEs with the finite-difference, finite-element, spectral elements, finite-volume, and discontinuous Galerkin methods. The ROMs are computationally efficient and accurate and are worthy when a FOM needs to be simulated multiple times for different parameter settings or in multi-query scenarios such as in optimization, and for predicting the model for a long time horizon.

ROMs consist of two classes, the data-driven non-intrusive and the intrusive ROMs. In intrusive ROMs, the high-fidelity FOMs are projected on a low dimensional reduced space, usually applying the proper orthogonal decomposition (POD) [16, 108]. There are other MOR techniques like the dynamic mode decomposition (DMD) [100, 107] and balanced truncation [92] which are frequently used. POD is a commonly

2

used ROM technique, which extracts dominant POD modes from the snapshots of the FOM solutions. An essential feature of the ROMs is the offline-online decomposition. The offline stage consists of the computation of the FOM and the construction of the reduced basis, whereas the online stage consists of the solution of the projected problem onto the low-dimensional reduced space. Typically, POD basis are determined by applying the singular value decomposition (SVD) to the snapshot matrix. After obtaining the POD basis, ROM is constructed by employing Galerkin projection. For PDEs with polynomial nonlinearities like the SWEs, the projection-based reduced model admits an efficient low-dimensional numerical representation independent of the dimension of the FOM without necessitating hyper-reduction techniques like empirical/discrete empirical interpolation method (DEIM) [7, 29]. We refer to [15, 96, 106] for an overview of the available MOR techniques.

Projection-based MOR methods are intrusive since they require access to the discrete operators of the FOM. Traditional intrusive MOR techniques thus possess limited scope in some applications. For instance, the traditional intrusive MOR methods are not suitable when proprietary software is used to solve the PDEs, where the governing equations are known, but full-order operators are not available. Another challenging case for intrusive MOR methods could be when the FOM solution is obtained through experiments, where there is no model nor the discretized operators. These bottlenecks of the intrusive MOR methods give rise to the other class of ROMs, so-called data-driven or non-intrusive ROMs. The non-intrusive ROMs are fundamentally different from the intrusive ROMs. Unlike the intrusive projection-based MOR methods, in the non-intrusive MOR techniques, reduced models are learnt from snapshots, i.e., either by numerical approximations or measurements of the states of the dynamical systems, when the operators of the discretized systems are not available. There are several software packages that are able to simulate SWEs for a given parameter set and an initial condition [48, 38]. Machine learning techniques are an essential tool for constructing models and analyzing the underlying process of the dynamics from the data. Recent advances in machine learning methods such as neural networks offer new occasions to construct more efficient and accurate ROMs. They learn the model from the training data, where they neither require direct access to the high-fidelity model operators nor any extra information about the process. Nevertheless, to learn

the model accurately, they require a large amount of necessary data, which imposes a burden in the context of large-scale PDE simulations [116]. Incorporating existing knowledge about the physics of the models requires less training data. We review the most relevant literature about learning dynamical-system models from data related to this work. Recently, data-fit non-intrusive ROMs with artificial neural networks (ANNs) are also used for time-dependent dynamical systems [64, 121].

Another class of non-intrusive ROM modeling is the DMD, which determines the reduced nonlinear dynamical systems, by fitting the snapshots to the linear operators with respect to the L_2 norm [100, 107]. Using the Koopman operator based methods, the methodology of DMD has been extended nonlinear dynamical systems [124]. In [20, 77, 105], the ROMs are constructed in the high-dimensional systems by exploiting the sparsity of the governing equations. Non-intrusive and intrusive MOR methods, DMD and POD, have been intensively studied for the SWEs; see, e.g., [17, 18, 44, 79, 78, 3, 109]. The DMD is also extended to quadratic bilinear systems [52].

Recently, constructing non-intrusive ROMs by the operator inference (OpInf) method has gained significant attention. The OpInf method first studied for polynomial nonlinearities in [90], where the operators defining the governing equations of ROM learned by solving an optimization problem, i.e., least-squares problem. The OpInf method is a non-intrusive ROM in a sense; it does not need the high-dimensional semi-discretized operators of the PDEs, but only some information at the PDE level. Later, the OpInf method extended to the nonlinear systems that can be written as a polynomial or quadratic-bilinear system by using lifting variable transformation, called Lift & Learn method [72, 95]. In [95], the OpInf method is applied to the nonlinear models in which the structure of the nonlinear terms are conserved while learning ROMs from data. Data-driven non-intrusive ROMs via regularized OpInf are applied to combustion problems [83, 115]. Data-driven ROMs of fluid dynamics are studied by employing the OpInf method, such as incompressible flows [13] and SWEs [128]. In [93] the OpInf method is generalized to the PDE setting through lifting variables. For the linear FOMs, the OpInf method and DMD equivalent, i.e., the inferred operators, are the same.

The Markovian systems are memory free systems, where the future states of systems just depend on the current state and not on the previous ones. In contrast to the Markovian systems, the non-Markovian systems are considered to have a memory, where the memory indicates that the future states depend on the current and previous states. The OpInf [90] with the projected trajectories leads to non-Markovian dynamics in the low-dimensional subspaces even though the FOMs are Markovian, as known from, e.g., the Mori–Zwanzig formalism [32, 49]. On the other hand, intrusive ROMs are Markovian systems. Thus, the reduced-order operators learned by the OpInf method with projected trajectories are not exactly the projection-based intrusive reduced-order operators. In [89], the Markovian dynamics in the non-intrusive reduced-order system are recovered by a data sampling scheme that iterates between time stepping the high-dimensional FOM and the low-dimensional ROM after projection. Hence, the Markovian dynamics are maintained in low-dimensional subspaces. In [89], it is shown that under particular conditions, the operator inference with re-projected trajectories results in the intrusive reduced-order operators for dynamical systems with nonlinear polynomial terms. Recently in [118], probabilistic a posteriori error estimators are derived for the OpInf with re-projection for linear parabolic PDEs. The non-Markovian OpInf is investigated with partial information in [119]. A deep learning version of the OpInf is introduced in [54].

1.3 Outline of the Thesis

In this thesis, intrusive and non-intrusive ROMs are constructed for the RSWE, NTSWE, and RTSWE. In Figure 1.1, an overview about the thesis is given.

In Chapter 2, the Hamiltonian structure of the RSWE, NTSWE, and RTSWE is presented with the full spatial and temporal discretization. All three SWEs are discretized with the centered second-order finite differences in space that preserve the skew-symmetric structure in the semi-discrete form, i.e., a skew-gradient system is obtained. We consider two different formulations of all the SWEs; non-canonical Hamiltonian/Poisson PDE and PDE with linear-quadratic terms as primitive equations. Both versions of the SWEs are discretized in space using finite differences while preserving the skew-symmetric structure of the Poisson matrix. The result-
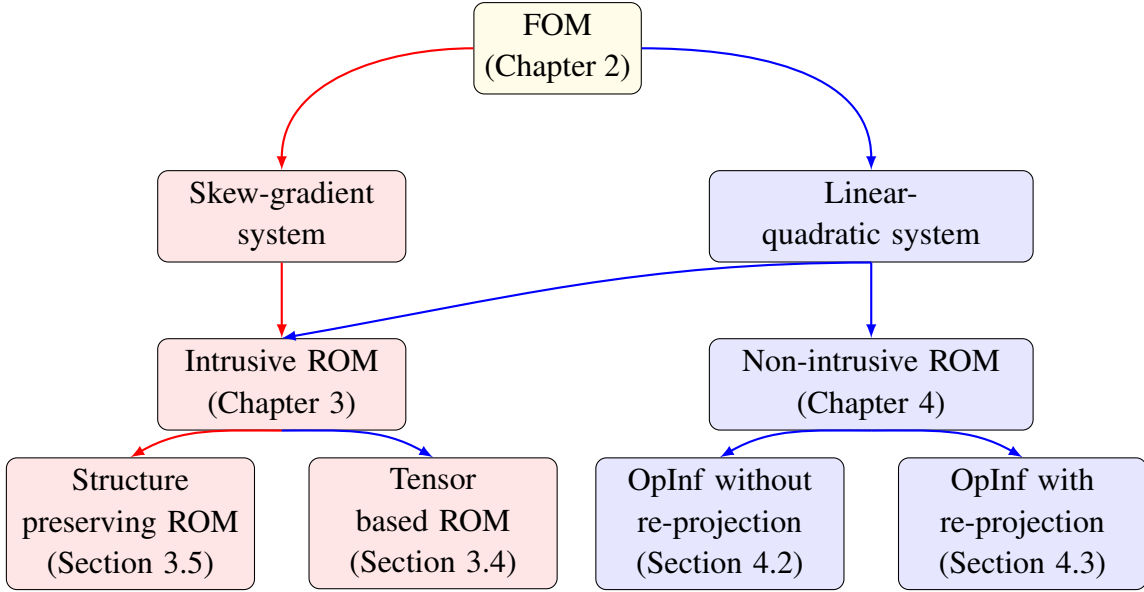
Figure 1.1: Outline of the thesis.

ing semi-discrete systems are skew-gradient and linear quadratic ordinary differential equation systems with multiple conserved quantities. The skew-gradient system is integrated in time with the energy preserving implicit average vector field (AVF) method [24, 33]. On the other hand, the linear-quadratic ODE system in time has solved by the linearly implicit Kahan's method [67, 26]. Both AVF and Kahan's method are second-order convergent in time. Due to its implicit nature, the AVF method needs iterative solvers like Newton's method at each time step for solving the nonlinear systems arising from time discretization. However, the linearly implicit Kahan's method requires solving only one linear system of equations in each time step. Both methods preserve well the conserved quantities of the SWEs, like the energy, enstrophy, mass, and vorticity in long-time integration [104, 43], and they build the basis for the ROM in the subsequent Chapters.

In Chapter 3, DEIM and POD based intrusive ROMs of SWEs are constructed. We start by briefly introducing the POD and DEIM methods in Sections 3.1 and 3.2. In Section 3.3, we introduce tensors and useful tensor techniques. Next, we employ POD Galerkin projection to the RSWE in the form of a linear-quadratic system, in Section 3.4 of the Chapter 3. Nevertheless, the dimension of the ROM arising from the RSWE still scales with the dimension of the FOM. For PDEs with the nonlinear polynomial terms like SWEs, the efficiency of ROMs can be obtained without approximating the

nonlinear terms through sampling, where the reduced operators can be precomputed in the "offline" stage. Thus, ROMs permit a clear separation of the offline-online stages. Projection of FOM corresponding to the RSWE onto the reduced space results in low-dimensional matrix operators that conserve the linear-quadratic structure of the FOM. The advantage of the offline-online separation is the nonlinearities can be evaluated without approximating in contrast to the hyper-reduction methods. In recent years, ROMs for PDEs with nonlinear polynomial terms have been broadly studied, where the computationally efficient ROMs are constructed by using some tools from tensor theory and by matricizations of tensors [10, 12, 11, 56, 72]. With the tensorial framework combined by POD, i.e., by exploiting matricizations of tensors, the online computational cost is reduced significantly. Hence, the ROM for the linear-quadratic system gains an efficient offline-online decomposition and does not require an additional step of hyper-reduction, like DEIM. Here to deal with tensors, we consider MULTIPROD [76] to accelerate the tensor calculations. Numerical results show that the energy (Hamiltonian) and other conserved quantities of the RSWE like the mass, the total vorticity, and the enstrophy are preserved in the long-time integration.

Typically, the preservation of nonlinear invariants like energy is not guaranteed with traditional MOR techniques. The violation of such invariants in ROMs generally yields an unstable or inaccurate reduced-order system, even when the high-fidelity system is stable. Thus, preservation of the physical laws of the ROMs by the reduced system plays a vital role to obtain in physically meaningful reduced systems for fluid dynamics problems, such as incompressible and compressible Euler equation [2]. The stability of ROMs over the long-time integration has been studied in the context of Lagrangian systems [23], and for port-Hamiltonian systems [28].

In this work, we study structure-preserving ROMs for NTSWE and RTSWE that utilize the skew-symmetry of the centered discretization schemes to recover conservation of the energy at the level of the reduced system. There is no numerical integrator method that conserves both the symplectic/Poisson structure and the energy of a Hamiltonian system [57]. We remark that some integrators exist only for specific Poisson systems, which preserve the Poisson structure unlike the symplectic integrators for all canonical Hamiltonian systems [62]. Since the reduced-order NTSWE

and RTSWE are skew-gradient systems with a first integral, i.e., Hamiltonian, and Casimirs, energy-preserving integrators can be easily applied. In recent years, ROMs are constructed that conserve the reduced skew-symmetic constant Poisson matrix for the Korteweg de Vries (KdV) equation with the energy in [51, 84, 70]. The AVF method is applied for the reduced-order modeling of Hamiltonian systems like the Korteweg-de Vries (KdV) equation [51, 62, 84] and the nonlinear Schrödinger equation [70]. Constructing on the approach in [51, 84] in the Sections 3.5.1 and 3.5.3 , ROMs are constructed that preserve the state-dependent skew-gradient structure of the semi-discretized NTSWE and RTSWE with energy preserving time integrator AVF. The straightforward application of a structure-preserving hyper-reduction technique does not allow separation of online and offline computation of the nonlinear terms of the non-canonical Hamiltonian systems like the RTSWE, and consequently, the online computational cost is not reduced. Approximation of the Poisson matrix and the gradient of the Hamiltonian of the RTSWE with the DEIM results in a skew-gradient reduced system. In Section 3.5.1, the DEIM has applied only the gradient of the Hamiltonian of the NTSWE and not to the nonlinear terms of the skew-symmetric Poisson matrix. In Section 3.5.3, the POD-DEIM ROM for the RTSWE is constructed using tensor techniques that preserves also the skew-symmetric structure of the Poisson matrix.

POD-DEIM ROM results show that the reduced discrete energy (Hamiltonian) and other conserved quantities like enstrophy, mass, and vorticity are well-preserved in the long-term using POD and DEIM. The ROMs with the POD and DEIM provide accurate and stable approximate solutions of the RTSWE while exhibiting several orders of magnitude computational speed-ups over the FOMs.

In Chapter 4, we discuss a non-intrusive MOR method that learns low-dimensional dynamical models for a parametrized shallow water equation from data. First, we describe the general framework of the OpInf method for linear-quadratic PDEs in Section 4.1. In Section 4.2, we consider NTSWE as a FOM and focus on recovering ROM in a non-intrusive way. The method in Section 4.2 is non-intrusive in the sense that it does need access to the semi-discretized operators of the NTSWE in any form. Instead, we assume the snapshots are obtained using a black-box solver. Therefore, we aim at recovering the ROMs only from the snapshots. The non-intrusive ROM

8

of NTSWE is obtained by solving a least-squares problem in a low-dimensional subspace. Hence, we also discuss computational difficulties that arise from the least-squares problem being ill-conditioned. Moreover, we extend the non-intrusive MOR framework to a parametric case, where we utilize the parameter dependency at the level of the PDEs. We demonstrate the efficiency of the non-intrusive MOR technique by constructing ROM for NTSWE and compare it with an intrusive POD method. Besides, we examine the predictive skills of both ROMs outside the range of the training data.

In Section 4.3, we use the OpInf with re-projection [89] to learn the dynamics of RTSWE by preserving the low-dimensional Markovian dynamics. To obtain an efficient intrusive ROM, we apply the POD by utilizing the matricization of tensors [12, 11, 69], and the sparse matrix technique MULTIPROD [76]. The numerical results show that learned non-intrusive ROM from the OpInf method with re-projection [89] are very close to the intrusive ROM obtained by POD. Moreover, the energy and other physical quantities are preserved with the OpInf and POD without any drift over time. Moreover, we compare OpInf with the POD method in the parametric case, where the Coriolis parameter is assumed to have known parametric dependency at the PDE level, whereas in [90, 89] the parametric ROMs are obtained at each training parameter by interpolation. We demonstrate that under particular assumptions on the time discretization [90, 89] the intrusive POD model converges to the learned non-intrusive OpInf model. Numerical results indicate that both the OpInf and the POD methods can predict the dynamics of the parametric FOM with high accuracy. Moreover, we demonstrate that the ROMs show high precision in the training set and tolerable accuracy in the prediction phase. The data matrices arising from the OpInf have large condition numbers, resulting in an ill-conditioned optimization problem. Therefore in the optimization problem, a proper regularization method like Tikhonov regularization or the truncated SVD should be used. Because the decay of the singular values in the data matrices does not give any information about the choice of tolerance for the regularization parameter, we determine the regularization tolerances by the L-curve. Speed-up factors of order two over the FOM are achieved for both ROMs, whereas the OpInf is more costly than the POD due to the re-projection.

The simulations on Sections 3.4.1, 3.5.2, 3.5.4 and 4.3.3 are performed on a machine

9

with Intel® Core^{TM} i7 2.5 GHz 64 bit CPU, 8 GB RAM, Windows 10, using 64 bit MatLab R2019a.

CHAPTER 2

FULL-ORDER MODELS FOR SHALLOW WATER EQUATIONS

This Chapter presents the FOM for two-dimensional SWEs, which consist of a set of hyperbolic PDEs illustrating a sheer inviscid fluid layer circulating over a bottom topography in a rotating frame. SWEs are extensively used in a conceptual way, such as investigating large-scale atmosphere/ocean dynamics and numerical weather prediction. SWEs are also used to study geophysical wave phenomena, e.g., the Rossby and Kelvin waves in the oceans and the atmosphere, large-scale geophysical flow prediction [9, 34], investigation of planetary flows [123], and baroclinic instability [19, 120]. Energy and enstrophy are the most important physical laws of the SWEs, whereas the energy cascades to large scales whilst enstrophy cascades to small scales [8, 113]. Typically, the ocean and atmosphere models include only the contribution to the Coriolis force from the component of the planetary rotation vector that is locally normal to geopotential surfaces when the vertical length scales are much smaller than the horizontal length scales, known as a traditional approximation. Nevertheless, most of the atmospheric and oceanographic phenomena are extensively affected by the non-traditional component of the Coriolis force [112], such as deep convection [81], Ekman spirals [75], and internal waves [47]. The NTSWE [41, 110, 113] has a similar structural form as the traditional SWE [103]. The key difference occurs in distinguishing between particle velocities and canonical velocities. A drawback of modeling of SWEs with multilayers is that it does not admit horizontal temperature and density gradients. Within each layer, the horizontal velocity is assumed to be depth-independent, so the fluid moves in columns. The RTSWE represents an extension of the RSWE, to include horizontal density/temperature gradients, also known

as Ripa equation [99]. The RTSWE can be derived similarly to the RSWE by the mean-field approximation with changing the hypothesis of uniform density/temperature [53, 129].

Enstrophy cascades to small scales and energy cascades to large scales in atmospheric and ocean flows [8, 113]. Hence their preservation is essential to obtain stable solutions in the long term integration [4]. Most of the geophysical flows can be represented in Hamiltonian form [85]. The non-canonical Hamiltonian form of the SWE in the rotational frame with constant Coriolis force was in by Salmon in[103]. Later on, SWE with complete Coriolis force [41, 113], the Nambu formulation [104], and multi-layer SWE [113] have been formulated. The preservation of the discrete energy comes from the antisymmetry of the discrete Poisson bracket. Other important preserved quantities are potential enstrophy, mass, and vorticity. In this thesis, we have considered two different forms of the SWEs; the non-canonical Hamiltonian form and a linear-quadratic PDE form in the f-plane, where the region is assumed to be small enough that the latitudinal variation in the Coriolis parameter can be ignored. Both the non-canonical Hamiltonian form and a linear-quadratic PDE form of the SWEs are discretized in space using finite differences by conserving the skew-symmetry in the Poisson matrix. The semi-discretized system of the ordinary differential equations (ODEs) is also in Poisson form and has linear-quadratic terms. For the time integration of the SWE in the non-canonical Hamiltonian form, fully implicit energy preserving AVF method [24, 33] is considered. Moreover, the linear-quadratic system of SWE is integrated in time by the linearly implicit Kahan's method [67, 26]. Both AVF and Kahan's methods are second-order convergent in time. The AVF method needs iterative solvers like Newton's method to solve the nonlinear systems arising from time discretization due to its implicit nature. In contrast to the AVF method, the linearly implicit Kahan's method needs solving only one linear system of equations in each time step. Both AVF and Kahan's methods conserve the energy and Casimir of the SWEs, like the energy, vorticity, enstrophy and mass in long-time integration [33, 26].

We remark that both RTSWE and RSWE as a hyperbolic system with nonlinear conservation laws were discretized using cell-centred finite volume methods [45] and the well-balanced schemes central upwind and finite volume method [31, 74, 73].

In Section 2.1 we introduce the three types of the SWEs; the RSWE with constant Coriolis force, the NTSWE with full Coriolis force, and the RTSWE. Spatial and temporal discretizations for each SWE are described in Sections 2.2, and 2.3, respectively.

2.1 Rotating Shallow Water Equation

In this section, we introduce the RSWE, NTSWE and RTSWE in Hamiltonian and linear-quadratic form.

2.1.1 Rotating Shallow Water Equation in Hamiltonian form

The two-dimensional RSWE is given on a rectangular domain $\Omega = [a, b] \times [c, d] \subset \mathbb{R}^2$ $(a, b, c, d \in \mathbb{R})$, as [103]

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= qvh - \Phi_x, \\
\frac{\partial v}{\partial t} &= -quh - \Phi_y, \\
\frac{\partial h}{\partial t} &= -(uh)_x - (vh)_y,
\end{aligned}
\tag{2.1}
$$

where $q(\boldsymbol{x}, t) = (v_x(\boldsymbol{x}, t) - u_y(\boldsymbol{x}, t) + f(\boldsymbol{x}, t))/h(\boldsymbol{x}, t)$ is the potential vorticity with the Coriolis force $f(\boldsymbol{x}, t)$, $h(\boldsymbol{x}, t)$ is the fluid depth, $u(\boldsymbol{x}, t)$ is the particle velocity in x-direction, and $v(\boldsymbol{x}, t)$ is the particle velocity in y-direction. The partial derivatives with respect to x and y components are denoted with subscripts x and y, respectively. Moreover, the Bernoulli potential is given that $\Phi(\boldsymbol{x}, t) = (1/2)u(\boldsymbol{x}, t)^2 + (1/2)v(\boldsymbol{x}, t)^2 + gh(\boldsymbol{x}, t)$, for gravity constant g. We consider the RSWE (2.1) under periodic boundary conditions

$$
z(a, y, t) = z(b, y, t), \quad z(x, c, t) = z(x, d, t), \quad z \in \{u, v, h\},
$$

and with the initial conditions

$$
u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}), \quad v(\boldsymbol{x}, 0) = v_0(\boldsymbol{x}), \quad h(\boldsymbol{x}, 0) = h_0(\boldsymbol{x}).
$$

13

The RWSE (2.1) is a non-canonical Hamiltonian PDE with the Poisson structure

$$\frac{\partial z}{\partial t} = \mathcal{J}(z)\frac{\delta \mathcal{H}}{\delta z} = \begin{pmatrix} 0 & q & -\partial_x \\ -q & 0 & -\partial_y \\ -\partial_x & -\partial_y & 0 \end{pmatrix}\begin{pmatrix} uh \\ vh \\ \Phi \end{pmatrix}, \tag{2.2}$$

where $z(\boldsymbol{x},t) = (u(\boldsymbol{x},t), v(\boldsymbol{x},t), h(\boldsymbol{x},t))^T$ is the solution vector, $\boldsymbol{v}(\boldsymbol{x},t) = (u(\boldsymbol{x},t), v(\boldsymbol{x},t))^T$ is the velocity field, and $\mathcal{H}(z)$ is the Hamiltonian

$$\mathcal{H}(z) = \frac{1}{2}\iint h(\boldsymbol{v}\cdot\boldsymbol{v} + gh)d\boldsymbol{x}. \tag{2.3}$$

The skew-symmetric Poisson bracket is defined for any two functionals $\mathcal{A}$ and $\mathcal{B}$ [80, 103] as

$$\{\mathcal{A},\mathcal{B}\} = \iint\left(q\frac{\delta((\mathcal{A},\mathcal{B})}{\delta(u,v)} - \frac{\delta\mathcal{A}}{\delta\boldsymbol{v}}\cdot\nabla\frac{\delta\mathcal{B}}{\delta h} + \frac{\delta\mathcal{B}}{\delta\boldsymbol{v}}\cdot\nabla\frac{\delta\mathcal{A}}{\delta h}\right)d\boldsymbol{x}, \tag{2.4}$$

where $\nabla = (\partial x, \partial y)^T$, and $\delta\mathcal{A}/\delta\boldsymbol{v}$ is the functional derivative of $\mathcal{A}$ with respect to $\boldsymbol{v}$. The functional Jacobian is given by

$$\frac{\delta(\mathcal{A},\mathcal{B})}{\delta(u,v)} = \frac{\delta\mathcal{A}}{\delta u}\frac{\delta\mathcal{B}}{\delta v} - \frac{\delta\mathcal{B}}{\delta u}\frac{\delta\mathcal{A}}{\delta v}.$$

Although the matrix $\mathcal{J}$ in (2.2) is not skew-symmetric, the skew-symmetry of the Poisson bracket is shown up after integration by parts [80], and the Poisson bracket satisfies the Jacobi identity

$$\{\mathcal{A},\{\mathcal{B},\mathcal{D}\}\} + \{\mathcal{B},\{\mathcal{D},\mathcal{A}\}\} + \{\mathcal{D},\{\mathcal{A},\mathcal{B}\}\} = 0,$$

for any three functionals $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{D}$. Conservation of the Hamiltonian (2.3) follows from the antisymmetry of the Poisson bracket (2.4)

$$\frac{d\mathcal{H}}{dt} = \{\mathcal{H},\mathcal{H}\} = 0.$$

Other conserved quantities of the RSWE [103] are the Casimirs of the form

$$\mathcal{C} = \iint hG(q)d\boldsymbol{x},$$

where G is an arbitrary function of the potential vorticity q. The Casimirs are additional constants of motion which commute with any functional $\mathcal{A}$, i.e., the Poisson bracket vanishes

$$\{\mathcal{A},\mathcal{C}\} = 0, \quad \forall\mathcal{A}(\boldsymbol{z}) \quad \text{or} \quad \mathcal{J}^{ij}\frac{\partial\mathcal{C}}{\partial\boldsymbol{z}^j} = 0.$$

14

Important special cases of the Casimirs are potential enstrophy $\mathcal{Z}$, vorticity $\mathcal{V}$, and mass $\mathcal{M}$, given by

$$
\begin{aligned}
\mathcal{Z} &= \frac{1}{2} \int \int hq^2 d\boldsymbol{x} = \frac{1}{2} \int \int \frac{1}{h} \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} + f \right)^2 d\boldsymbol{x}, \\
\mathcal{V} &= \int\int hq d\boldsymbol{x}, \ \mathcal{M} = \int\int h d\boldsymbol{x}.
\end{aligned}
\tag{2.5}
$$

2.1.2 The f-plane and β-plane Approximations

A common technique for studying geophysical flows in the atmosphere is the "tangent plane" approximation, which is obtained by projecting the RSWE from the surface of Earth, onto local tangent plane. The tangent plane approximation is valid when horizontal scales small, e.g., mesoscale phenomena. The Cartesian coordinates x, y, and z in the tangent plane point North, East, and outward from the centre of the Earth, respectively.

The RSWE (2.1) describes inviscid fluid rotating with angular velocity vector $\boldsymbol{\Omega} = (\Omega^x, \Omega^y, \Omega^z)$. Both Ω^x and Ω^y depend on x and y axes but not on z. The angular velocity vector can be written as [111]:

$$
\Omega^x = \Omega \cos \theta \sin \phi, \quad \Omega^y = \Omega \cos \theta \cos \phi, \quad \text{and} \quad \Omega^z = \Omega \sin \theta, \tag{2.6}
$$

where θ is the angle corresponding to the latitude, and ϕ is the angle determining the orientation between the x-axis and the eastward direction. In this study, we set the x-axis of the rotation vector to zero, implying that it is aligned to the East. In Figure2.1, the components of the angular velocity vector are shown for the latitude θ and $\phi = 0$. Hence, the components of the Coriolis force can be written as

$$
2\boldsymbol{\Omega} \times \mathbf{u} \approx (-2\Omega v \sin \theta, 2\Omega u \sin \theta, -2\Omega u \cos \theta) = (-fv, fu, -2\Omega u \cos \theta), \tag{2.7}
$$

where the Coriolis parameter $f = 2\Omega \sin \theta$. The vertical component of (2.7) is usually neglected due to dominant terms in the vertical equation of motion.

The f-plane approximation assumes that the Coriolis force is constant over the tangent plane so that f can be approximated as, $f_0 = 2\Omega \sin \theta_0$, where θ_0 is the central latitude of the region. The f-plane model is more useful when the horizontal scales

small. A more realistic model on the approximation of the Coriolis parameter over a larger horizontal scales is called β-plane approximation, which takes the variation of the Coriolis parameter into account. The β-plane approximation can be obtained by expanding the Coriolis parameter in a Taylor series about the central latitude θ_0 as

$$f \approx f_0 + y \left(\frac{\partial f}{\partial y} \right)_{\theta_0} = f_0 + \beta y. \tag{2.8}$$
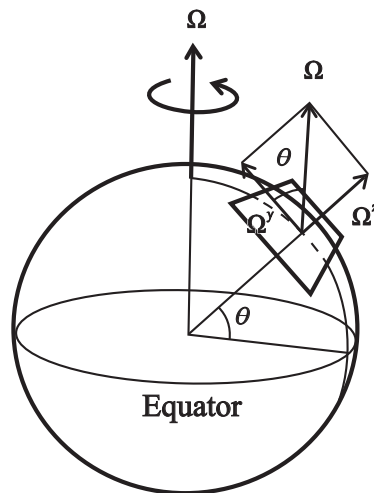
Figure 2.1: Outline of the components of the angular velocity vector Ω with respect to the latitude angle θ as $\Omega^x = 0$, (reproduced from [61]).

2.1.3 Rotating Shallow Water Equation in the f-plane

Generally, in the ocean and atmosphere models, computing the Coriolis force is done by considering just the component of the planetary rotation vector that is locally normal to geopotential surfaces, which is known as "traditional approximation" [113]. On the other hand, the so-called β-plane approximation is applicable when the earth's surface is considered locally flat with the latitudinal variation of the Coriolis force, where the Coriolis force approximated as $f \approx f_0 \beta y$. The β-plane approximation is also used frequently in the literature (see for example [17, 35]). In some models, the latitudinal variation in the Coriolis parameter is neglected for small regions, and the Coriolis force f is assumed to be a constant value, which corresponds to the f-plane approximation of the RSWE.

16

Inserting the potential vorticity q in (2.1), the RSWE can be written in the f-plane as

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= -uu_x - vu_y - gh_x + fv, \\
\frac{\partial v}{\partial t} &= -uv_x - vv_y - gh_y - fu, \\
\frac{\partial h}{\partial t} &= -(uh)_x - (vh)_y.
\end{aligned}
\tag{2.9}
$$

2.1.4 Non-Traditional Rotating Shallow Water Equation

Most of the atmosphere and ocean models consider the traditional approximation of the Coriolis force. Nevertheless, many oceanographic and atmospheric phenomena are primarily influenced by the non-traditional component of the Coriolis force [112], such as Ekman spirals [75], deep convection [81], and internal waves [47]. The nondimensional NTSWE [41, 110, 113] has the same structural form as the traditional SWE [103] by distinguishing between the canonical velocities $\tilde{u}(x, y, t)$ and $\tilde{v}(x, y, t)$, and particle velocities $u(x, y, t)$ and $v(x, y, t)$

$$
\begin{aligned}
\frac{\partial \tilde{u}}{\partial t} &= hqv - \partial\Phi_x, \\
\frac{\partial \tilde{v}}{\partial t} &= -hqu + \Phi_y, \\
\frac{\partial h}{\partial t} &= -\frac{\partial}{\partial x}(hu) - \frac{\partial}{\partial y}(hv),
\end{aligned}
\tag{2.10}
$$

where $h(x, y, t)$ denote the height field, and x and y denote horizontal distances within a constant geopotential surface. The one-layer NTSWE (2.10) describes an inviscid fluid flowing over bottom topography at $z = h_b(x, y)$ in a frame rotating with angular velocity vector $\mathbf{\Omega} = (\Omega^{(x)}, \Omega^{(y)}, \Omega^{(z)})$. The orientation of the x and y axes are considered arbitrary with respect to North. In traditional rotating and non-rotating SWEs, only the particle velocity components appear. The canonical velocity components are related to the canonical momentum per mass or to the depth average of particle velocities as

$$
\tilde{u} = u + 2\Omega^{(y)}\left(h_b + \frac{1}{2}h\right), \quad \tilde{v} = v - 2\Omega^{(x)}\left(h_b + \frac{1}{2}h\right).
\tag{2.11}
$$

The Bernoulli potential Φ and potential vorticity q are given by

$$
\begin{aligned}
\Phi &= \frac{1}{2}(u^2 + v^2) + g(h_b + h) + h(\Omega^{(x)}v - \Omega^{(y)}u), \\
q &= \frac{1}{h}(2\Omega^{(z)} + \tilde{v}_x - \tilde{u}_y).
\end{aligned}
$$

17

The non-traditional parameter is given as $\sigma = H/R_d$, where H represents the layer thickness scale and R_d is Rossby deformation radius, and g denotes the gravitational acceleration [41, 110].

The traditional and non-traditional RSWE differ only by a function of the space alone, so their time derivatives are identical. The non-rotating, traditional SWE [102] and NTSWE (2.10) have the same Hamiltonian structure and Poisson bracket [41, 110, 113]

$$\frac{\partial \tilde{z}}{\partial t} = \mathcal{J}(\tilde{z})\frac{\delta \mathcal{H}}{\delta \tilde{z}} = \begin{pmatrix} 0 & q & -\partial_x \\ -q & 0 & -\partial_y \\ -\partial_x & -\partial_y & 0 \end{pmatrix} \begin{pmatrix} hu \\ hv \\ \Phi \end{pmatrix}, \tag{2.12}$$

where $\tilde{z} = (\tilde{u}, \tilde{v}, h)$. The Hamiltonian or the energy of (2.10) is given in terms of particle velocity components by

$$\mathcal{H}(z) = \iint \left\{ \frac{1}{2}h(u^2 + v^2) + gh\left(h_b + \frac{1}{2}h\right) \right\}d\boldsymbol{x}, \tag{2.13}$$

over a periodic domain. We remark that the Hamiltonian (2.12) is treated as a function of the canonical velocity components $\tilde{u}$ and $\tilde{v}$ and the layer thickness using the relations (2.11).

The Casimirs are the potential enstrophy

$$\mathcal{Z} = \frac{1}{2}\int\int hq^2 d\boldsymbol{x} = \frac{1}{2}\int\int \frac{1}{h}\left(\Omega^{(z)} + \frac{\partial \tilde{v}}{\partial x} - \frac{\partial \tilde{u}}{\partial y}\right)^2 d\boldsymbol{x}, \tag{2.14}$$

the mass $\mathcal{M} = \iint hd\boldsymbol{x}$, and the vorticity $\mathcal{V} = \iint hqd\boldsymbol{x}$.

2.1.5 Rotating Thermal Shallow Water Equation

RSWE is widely used in planetary and geophysical fluid dynamics as an conceptual model for the behaviour of rotating inviscid fluids with one or more layers. The horizontal velocity is considered to be depth-independent in RSWE, so the fluid moves in columns in each layer. A drawback of this model is that it does not admit horizontal temperature and density gradients. The RTSWE takes into account the horizontal density/temperature gradients compared with RSWE. It is also known in the literature as Ripa equation [99]. The RTSWE can be obtained along the same lines as the

18

RSWE, by vertical averaging of the primitive equations in the Boussinesq approximation, and using the hypothesis of columnar motion (mean-field approximation), but relaxing the hypothesis of uniform density/temperature [53, 129]. The RTSWE provides a depth-averaged description of motions in a fluid layer that permits horizontal variations in material properties. When the layer depth is supposed to be small compared with a typical horizontal length scale, the vertical fluid acceleration in the layer may be neglected, then the RTSWE [39, 122, 43] has the form

$$
\begin{aligned}
\frac{\partial h}{\partial t} &= -(hu)_x - (hv)_y, \\
\frac{\partial u}{\partial t} &= hqv - \left(\frac{u^2 + v^2}{2}\right)_x - \frac{h}{2}s_x - s(h + b)_x, \\
\frac{\partial v}{\partial t} &= -hqu - \left(\frac{u^2 + v^2}{2}\right)_y - \frac{h}{2}s_y - s(h + b)_y, \\
\frac{\partial s}{\partial t} &= -us_x - vs_y,
\end{aligned}
\tag{2.15}
$$

where $u(x, y, t)$ and $v(x, y, t)$ are the depth-averaged relative velocities, $h(x, y, t)$ is the fluid height, $b(x, y)$ represents the bottom topography, g is the gravity constant, $s = g\frac{\rho}{\rho_0}$ is the buoyancy, where the $\rho = \rho(y)$ horizontally varying density, ρ_0 is the reference density used in the Boussinesq approximation. Density variations in the ocean are proportional to temperature variations, in the atmospheric applications, ρ and ρ_0 should be replaced with the potential temperature θ and θ_0. An f-plane approximation is not suitable when considering RSWE over large length scales, whereas in the β-plane approximation, f varies with the latitude, $f = f_0 + \beta y$, which is applicable for larger length scales. The RTSWE (2.15) is considered as the RSWE on doubly periodic domain $\Omega \in \mathrm{R}^d$, $d = 1, 2$ and in time interval $[0, T]$ with the initial conditions

$$
u(\boldsymbol{x}, 0) = u_0(\boldsymbol{x}), \ v(\boldsymbol{x}, 0) = v_0(\boldsymbol{x}), \ h(\boldsymbol{x}, 0) = h_0(\boldsymbol{x}), \ s(\boldsymbol{x}, 0) = s_0(\boldsymbol{x}).
$$

The RTSWE has as similar non-canonical Hamiltonian/Poisson structure as the RSWE [39, 122, 43]

$$
\frac{\partial z}{\partial t} = -\mathcal{J}(z)\frac{\delta \mathcal{H}}{\delta z} = -
\begin{pmatrix}
0 & \partial_x & \partial_y & 0 \\
\partial_x & 0 & -q & -h^{-1}s_x \\
\partial_y & q & 0 & -h^{-1}s_y \\
0 & h^{-1}s_x & h^{-1}s_y & 0
\end{pmatrix}
\begin{pmatrix}
\frac{u^2+v^2}{2} + sh + sb \\
hu \\
hv \\
\frac{h^2}{2} + hb
\end{pmatrix},
\tag{2.16}
$$

19

with the conserved Hamiltonian or the energy

$$\mathcal{E}(z) = \int_\Omega \left(\frac{h^2 s}{2} + hsb + h\frac{u^2 + v^2}{2} \right) d\Omega, \tag{2.17}$$

where $z = (h, u, v, s)^T$. Other conserved quantities are the total mass, the total potential vorticity, and the total buoyancy

$$\mathcal{M} = \int h \, d\Omega, \quad \mathcal{Q} = \int q \, d\Omega, \quad \mathcal{B} = \int hs \, d\Omega, \tag{2.18}$$

respectively. Unlike the RSWE, in case RTSWE, the potential enstrophy is not conserved [43].

2.2 Spatial Discretization

For the space discretization of the RSWE (2.1), the spatial domain $\Omega = (a, b) \times (c, d)$ is discretized on the uniform grid nodes $\boldsymbol{x}_{ij} = (x_i, y_j)^T$, where the nodes are $x_i = a + (i-1)\Delta x$ and $y_j = c + (j-1)\Delta y$, $i = 1, \ldots, N_x + 1$, $j = 1, \ldots, N_y + 1$ so that the spatial domain is divided into N_x and N_y equidistant intervals in x and y-directions, respectively. Hence, the mesh sizes becomes $\Delta y = (d-c)/N_y$ and $\Delta x = (b-a)/N_x$. After the spatial discretization, the semi-discrete time-dependent solution are defined as follows

$$\begin{aligned}
\boldsymbol{u}(t) &= (u_{11}(t), \ldots, u_{1N_y}(t), u_{21}(t), \ldots, u_{2N_y}(t), \ldots, u_{N_xN_y}(t))^T, \\
\boldsymbol{v}(t) &= (v_{11}(t), \ldots, v_{1N_y}(t), v_{21}(t), \ldots, v_{2N_y}(t), \ldots, v_{N_xN_y}(t))^T, \\
\boldsymbol{h}(t) &= (h_{11}(t), \ldots, h_{1N_y}(t), h_{21}(t), \ldots, h_{2N_y}(t), \ldots, h_{N_xN_y}(t))^T,
\end{aligned} \tag{2.19}$$

where each semi-discrete particle velocities $u_{ij}(t)$, $v_{ij}(t)$ and the fluid depth $h_{ij}(t)$ denotes the approximate solutions of $u(\boldsymbol{x}, t)$, $v(\boldsymbol{x}, t)$, and $h(\boldsymbol{x}, t)$, respectively at the grid nodes $\boldsymbol{x}_{ij}$ and time t, $i = 1, \ldots, N_x$, $j = 1, \ldots, N_y$. We remark that each semi-discrete solution vector in (2.19) has $N := N_x N_y$ elements because we ignore the solutions on the topmost and rightmost boundary due to the periodic boundary conditions. In the remainder of this Chapter, we ease the notation by omitting the time dependency of the semi-discrete solutions, i.e., by simply writing $\boldsymbol{u}$, $\boldsymbol{v}$, and $\boldsymbol{h}$, the semi-discrete vector for the solution vector z is defined by $\boldsymbol{z} = (\boldsymbol{u}; \boldsymbol{v}; \boldsymbol{h}) \in \mathbb{R}^{3N}$. Similarly, the semi-discrete vector for the potential vorticity q is defined by $\boldsymbol{q}$.

The first-order partial derivatives are approximated by one-dimensional central finite differences, and then one-dimensional operators are extended to two dimensions by utilizing the Kronecker product. For a positive integer s, let $\widetilde{D}_s$ denotes the matrix of central finite-differences to the first-order ordinary differential operator under periodic boundary conditions

$$
\widetilde{D}_s = \begin{pmatrix} 0 & 1 & & & -1 \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 0 & 1 \\ 1 & & & -1 & 0 \end{pmatrix} \in \mathbb{R}^{s \times s}.
$$

Then, on the two-dimensional mesh, the central finite-difference matrices corresponding to the first-order partial derivative operators ∂_x and ∂_y are given, respectively, by

$$
D_x = \frac{1}{2\Delta x} \widetilde{D}_{N_x} \otimes I_{N_y} \in \mathbb{R}^{N \times N} , \quad D_y = \frac{1}{2\Delta y} I_{N_x} \otimes \widetilde{D}_{N_y} \in \mathbb{R}^{N \times N},
$$

where $\otimes$ denotes the Kronecker product, and I_{N_x} and I_{N_y} are the identity matrices of size N_x and N_y, respectively.

The semi-discrete formulation of the RSWE (2.2) is given as the following $3N$-dimensional system of Hamiltonian ODEs

$$
\frac{d\boldsymbol{z}}{dt} = J(\boldsymbol{z})\nabla H(\boldsymbol{z}) = \begin{pmatrix} 0 & \boldsymbol{q}^d & -D_x \\ -\boldsymbol{q}^d & 0 & -D_y \\ -D_x & -D_y & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{u} \circ \boldsymbol{h} \\ \boldsymbol{v} \circ \boldsymbol{h} \\ \frac{1}{2}(\boldsymbol{u} \circ \boldsymbol{u} + \boldsymbol{v} \circ \boldsymbol{v}) + g\boldsymbol{h} \end{pmatrix}, \quad (2.20)
$$

where $\circ$ denotes element-wise or Hadamard product. In (2.20), the matrix $\boldsymbol{q}^d \in \mathbb{R}^{N \times N}$ corresponds to the diagonal matrix with the diagonal elements $\boldsymbol{q}^d_{ii} = \boldsymbol{q}_i$, where $\boldsymbol{q} \in \mathbb{R}^N$ denotes the semi-discrete potential vorticity vector, $i = 1, \dots, N$.

The fully discrete conserved quantities of RSWE are defined as:

- energy

$$
H^k = H(\boldsymbol{z}^k) = \frac{1}{2} \sum_{i=1}^{N} ((\boldsymbol{u}_i^k)^2 + (\boldsymbol{v}_i^k)^2 + g\boldsymbol{h}_i^k)\boldsymbol{h}_i^k \Delta x \Delta y, \quad (2.21)
$$

- potential enstrophy

$$
Z^k = Z(\boldsymbol{z}^k) = \frac{1}{2} \sum_{i=1}^{N} \frac{((D_x \boldsymbol{v}^k)_i - (D_y \boldsymbol{u}^k)_i + f)^2}{\boldsymbol{h}_i^k} \Delta x \Delta y, \quad (2.22)
$$

- vorticity

$$V^k = V(\boldsymbol{z}^k) = \sum_{i=1}^{N} ((D_x \boldsymbol{v}^k)_i - (D_y \boldsymbol{u}^k)_i + f) \Delta x \Delta y, \qquad (2.23)$$

- and mass

$$M^k = M(\boldsymbol{z}^k) = \sum_{i=1}^{N} \boldsymbol{h}_i^k \Delta x \Delta y. \qquad (2.24)$$

The RTSWE (2.15) is discretized similarly as the RSWE. The time-dependent approximate semi-discrete solution vectors are given as

$$
\begin{aligned}
\boldsymbol{u}(t) &= (u_{11}(t), \ldots, u_{1N_x}(t), u_{21}(t), \ldots, u_{2N_x}(t), \ldots, u_{N_x N_y}(t))^T, \\
\boldsymbol{v}(t) &= (v_{11}(t), \ldots, v_{1N_x}(t), v_{21}(t), \ldots, v_{2N_x}(t), \ldots, v_{N_x N_y}(t))^T, \\
\boldsymbol{h}(t) &= (h_{11}(t), \ldots, h_{1N_x}(t), h_{21}(t), \ldots, h_{2N_x}(t), \ldots, h_{N_x N_y}(t))^T, \\
\boldsymbol{s}(t) &= (s_{11}(t), \ldots, s_{1N_x}(t), h_{21}(t), \ldots, s_{2N_x}(t), \ldots, s_{N_x N_y}(t))^T.
\end{aligned}
\qquad (2.25)
$$

The semi-discrete form of the RTSWE (2.2) leads to a $4N$ dimensional system of Hamiltonian ODEs in skew-gradient form

$$
\begin{aligned}
\frac{d\tilde{\boldsymbol{z}}}{dt} &= -J(\tilde{\boldsymbol{z}}) \nabla_{\boldsymbol{z}} H(\boldsymbol{z}) \\
&= - \begin{pmatrix} 0 & D_x & D_y & 0 \\ D_x & 0 & -\boldsymbol{q} & -\boldsymbol{h}^{-1} D_x \cdot \boldsymbol{s} \\ D_y & \boldsymbol{q} & 0 & -\boldsymbol{h}^{-1} D_y \cdot \boldsymbol{s} \\ 0 & \boldsymbol{h}^{-1} D_x \cdot \boldsymbol{s} & \boldsymbol{h}^{-1} D_y \cdot \boldsymbol{s} & 0 \end{pmatrix} \begin{pmatrix} \frac{u^2+v^2}{2} + \boldsymbol{s} \circ \boldsymbol{h} + sb \\ \boldsymbol{h} \circ \boldsymbol{u} \\ \boldsymbol{h} \circ \boldsymbol{v} \\ \frac{h^2}{2} + \boldsymbol{h}b \end{pmatrix},
\end{aligned}
\qquad (2.26)
$$

where the matrix $\boldsymbol{q}^d \in \mathbb{R}^{N \times N}$, again denotes the diagonal matrix containing the semi-discrete vorticity $\boldsymbol{q} \in \mathbb{R}^N$ in the principal diagonal, as in (2.2).

The fully discrete form of the energy, total vorticity, mass, and buoyancy are given at the time instances t_k as

$$H^k = \sum_{i=1}^{n} \left(\frac{1}{2} (\boldsymbol{h}_k^i)^2 \boldsymbol{s}_i^k + \boldsymbol{h}_k^i \boldsymbol{s}_i^k b + \boldsymbol{h}_k^i \frac{(\boldsymbol{u}_k^i)^2 + (\boldsymbol{v}_k^i)^2}{2} \right) \Delta x \Delta y, \qquad (2.27a)$$

$$M^k = \sum_{i=1}^{n} (\boldsymbol{h}_k^i) \Delta x \Delta y, \qquad (2.27b)$$

$$Q^k = \sum_{i=1}^{n} \left((D_x \boldsymbol{v}^k)_i - (D_y \boldsymbol{u}^k)_i + f \right) \Delta x \Delta y, \qquad (2.27c)$$

$$B^k = \sum_{i=1}^{n} (\boldsymbol{h}_k^i \boldsymbol{s}_k^i) \Delta x \Delta y, \qquad (2.27d)$$

respectively.

2.3 Temporal Discretization

In the following we present two different time dscretization of the SWEs. The first one is the average vector field (AVF) method for the time integration of the ODE system (2.20) in Poisson form and the second one is for the Kahan's method for the semi-discretized SWEs as linear-quadratic sysyems of ODEs.

2.3.1 Average Vector Field Method

The time domain $[0, T]$ is divided into K uniform intervals with the time-step size $\Delta t = T/K$ as $0 = t_0 < t_1 < \ldots < t_K = T$, and $t_k = k\Delta t$, $k = 0, 1, \ldots, K$. Then, we denote by $\boldsymbol{u}^k = \boldsymbol{u}(t_k)$, $\boldsymbol{v}^k = \boldsymbol{v}(t_k)$, and $\boldsymbol{h}^k = \boldsymbol{h}(t_k)$ the full discrete solution vectors at time t_k.

Time integration of the ODE system (2.20) in Poisson form by the AVF method [33] yields

$$\boldsymbol{z}^{k+1} = \boldsymbol{z}^k + \Delta t J\left(\frac{\boldsymbol{z}^{k+1} + \boldsymbol{z}^k}{2}\right) \int_0^1 \nabla H(\xi(\boldsymbol{z}^{k+1} - \boldsymbol{z}^k) + \boldsymbol{z}^k)d\xi. \qquad (2.28)$$

The main advantage of the AVF method [33] is the conservation of the higher-order polynomial Hamiltonians, including the cubic Hamiltonian H of the SWE (2.1). Quadratic physical quantities like circulation and mass are also preserved exactly by the AVF method. Nevertheless, the higher-order polynomial physical quantities like enstrophy (cubic) are not conserved. Implementing the AVF method, the integral on the right-hand side of (2.28) need to be computed. Because the discrete Hamiltonian H and the Casimirs, like potential enstrophy, circulation, and mass are polynomial, they can be exactly integrated with a Gaussian quadrature rule of the appropriate order. The AVF method is studied with finite element discretization of the RSWE [8, 125] and for RTSWE [43] in Poisson form. Although the desired properties, such as conservation of energy, are obtained, the AVF method is computationally expensive due to its implicit form. In [43], the semi-implicit form of the AVF method is studied with a quasi-Newton solver and simplified Jacobian for the RTSWE. In [113], the NTSWE in the Hamiltonian form is discretized in space following [103] by the Arakawa-Lamb discretization [4] in space, where it is shown

that by using the non-structure preserving time integrators like the Runge-Kutta and Adams-Bashforth methods, causes drifts in the energy and enstrophy [113]. On the other hand, preservation of multiple integrals like the enstrophy at the same time [37] by geometric integrators like the AVF method is not possible. In [8, 125], the AVF method employed to the SWEs in a different Hamiltonian form, where the SWEs are discretized in space by compatible finite elements. Central difference approximation of the first order differential operators ensures that the discretized ODE is in skew-symmetric form, which is necessary for the preservation of the Hamiltonian and other conserved quantities. The skew-gradient structure of RTSWE is also preserved using finite element methods in [43], the RSWE and linear SWE with discontinuous Galerkin method [46, 126], and finite volume method [97].

The AVF method is used frequently energy preserving and dissipating integrator for canonical Hamiltonian and dissipative systems [24]), and multysymplecic PDEs [50]. There are also higher-order versions of the AVF method, but they are more computational costly

2.3.2 Kahan's Method

The spatial discretization of the RSWE (2.9) results in the following system of ODEs:

$$
\begin{aligned}
\frac{d\boldsymbol{u}}{dt} &= -\boldsymbol{u} \circ (D_x \boldsymbol{u}) - \boldsymbol{v} \circ (D_y \boldsymbol{u}) - g D_x \boldsymbol{h} + f\boldsymbol{v}, \\
\frac{d\boldsymbol{v}}{dt} &= -\boldsymbol{u} \circ (D_x \boldsymbol{v}) - \boldsymbol{v} \circ (D_y \boldsymbol{v}) - g D_y \boldsymbol{h} - f\boldsymbol{u}, \\
\frac{d\boldsymbol{h}}{dt} &= -D_x(\boldsymbol{u} \circ \boldsymbol{h}) - D_y(\boldsymbol{v} \circ \boldsymbol{h}).
\end{aligned}
\tag{2.29}
$$

The semi-discrete RSWE (2.29) consists of linear and quadratic terms, which can be written as

$$
\frac{d\boldsymbol{z}}{dt} = F(\boldsymbol{z}) = R_1(\boldsymbol{z}) + R_2(\boldsymbol{z}) + L(\boldsymbol{z}),
\tag{2.30}
$$

where the $L(\boldsymbol{z})$ includes the linear terms, while the quadratic vector $R_1(\boldsymbol{z})$ and $R_2(\boldsymbol{z})$ includes the quadratic terms, given by

$$
R_1(\boldsymbol{z}) = \begin{pmatrix} -\boldsymbol{u} \circ (D_x \boldsymbol{u}) \\ -\boldsymbol{u} \circ (D_x \boldsymbol{v}) \\ -D_x(\boldsymbol{u} \circ \boldsymbol{h}) \end{pmatrix}, \quad
R_2(\boldsymbol{z}) = \begin{pmatrix} -\boldsymbol{v} \circ (D_y \boldsymbol{u}) \\ -\boldsymbol{v} \circ (D_y \boldsymbol{v}) \\ -D_y(\boldsymbol{v} \circ \boldsymbol{h}) \end{pmatrix}, \quad
L(\boldsymbol{z}) = \begin{pmatrix} -g D_x \boldsymbol{h} + f\boldsymbol{v} \\ -g D_y \boldsymbol{h} - f\boldsymbol{u} \\ 0 \end{pmatrix}.
$$

For the linear-quadratic autonomous ODE systems like (2.29), Kahan introduced an "unconventional" discretization [67]

$$\frac{\boldsymbol{z}^{k+1} - \boldsymbol{z}^k}{\Delta t} = R_1^f(\boldsymbol{z}^k, \boldsymbol{z}^{k+1}) + R_2^f(\boldsymbol{z}^k, \boldsymbol{z}^{k+1}) + \frac{1}{2}L(\boldsymbol{z}^k + \boldsymbol{z}^{k+1}),$$

where the symmetric bilinear forms $R_1^f(\cdot, \cdot)$ and $R_2^f(\cdot, \cdot)$ are computed by the polarization [25] of the quadratic vector fields $R_1(\cdot)$ and $R_2(\cdot)$, respectively, defined by

$$R_i^f(\boldsymbol{z}^k, \boldsymbol{z}^{k+1}) = \frac{1}{2}\left(R_i(\boldsymbol{z}^k + \boldsymbol{z}^{k+1}) - R_i(\boldsymbol{z}^k) - R_i(\boldsymbol{z}^{k+1})\right), \quad i = 1, 2.$$

Kahan's method is time-reversal, second-order, and linearly implicit method, i.e., it need just one Newton iteration per each time step [68]. The fully discrete RSWE equations (2.30) with Kahan's method is given as

$$\left(I_{3N} - \frac{\Delta t}{2}F'(\boldsymbol{z}^k)\right)\frac{\boldsymbol{z}^{k+1} - \boldsymbol{z}^k}{\Delta t} = F(\boldsymbol{z}^k), \tag{2.31}$$

where $I_{3N} \in \mathbb{R}^{3N \times 3N}$ denotes the identity matrix and $F'(\boldsymbol{z}^k)$ denotes for the Jacobian matrix of $F(\boldsymbol{z})$ evaluated at $\boldsymbol{z}^k$. Analytical form of the Jacobian matrix $F'(\boldsymbol{z})$ is given as

$$F'(\boldsymbol{z}) = -\begin{pmatrix} \boldsymbol{u}^d D_x + (D_x\boldsymbol{u})^d + \boldsymbol{v}^d D_y & (D_y\boldsymbol{u})^d - f & gD_x \\ (D_x\boldsymbol{v})^d + f & \boldsymbol{u}^d D_x + \boldsymbol{v}^d D_y + (D_y\boldsymbol{v})^d & gD_y \\ D_x\boldsymbol{h}^d & D_y\boldsymbol{h}^d & D_x\boldsymbol{u}^d + D_y\boldsymbol{v}^d \end{pmatrix},$$

where the superscript d denotes the diagonal matrix possessing the a vector in principal diagonal, i.e., for any vector $\boldsymbol{a} \in \mathbb{R}^N$, the matrix $\boldsymbol{a}^d \in \mathbb{R}^{N \times N}$ is the diagonal matrix with the diagonal elements $\boldsymbol{a}_{ii}^d = \boldsymbol{a}_i$.

Spatial discretization of the RTSWE (2.15) results in the following linear-quadratic ODE system

$$\begin{aligned} \frac{d\boldsymbol{h}}{dt} &= -D_x(\boldsymbol{u} \circ \boldsymbol{h}) - D_y(\boldsymbol{v} \circ \boldsymbol{h}), \\ \frac{d\boldsymbol{u}}{dt} &= -\boldsymbol{u} \circ (D_x\boldsymbol{u}) - \boldsymbol{v} \circ (D_y\boldsymbol{u}) - \frac{\boldsymbol{h}}{2} \circ D_x\boldsymbol{s} - \boldsymbol{s} \circ D_x\boldsymbol{h} - \boldsymbol{s} \circ D_x\boldsymbol{b} + f\boldsymbol{v}, \\ \frac{d\boldsymbol{v}}{dt} &= -\boldsymbol{u} \circ (D_x\boldsymbol{v}) - \boldsymbol{v} \circ (D_y\boldsymbol{v}) - \frac{\boldsymbol{h}}{2} \circ D_y\boldsymbol{s} - \boldsymbol{s} \circ D_y\boldsymbol{h} - \boldsymbol{s} \circ D_y\boldsymbol{b} - f\boldsymbol{u}, \\ \frac{d\boldsymbol{s}}{dt} &= -\boldsymbol{u} \circ D_x\boldsymbol{s} - \boldsymbol{v} \circ D_y\boldsymbol{s}. \end{aligned} \tag{2.32}$$

which can be written in matrix-tensor form as

$$\frac{d\boldsymbol{z}}{dt} = \boldsymbol{F}(\boldsymbol{z}) = \mathbf{A}\boldsymbol{z} + \mathbf{H}(\boldsymbol{z} \otimes \boldsymbol{z}), \tag{2.33}$$

where $\mathbf{A} \in \mathbb{R}^{4N \times 4N}$ is a linear matrix operator containing the linear terms

$$\mathbf{A}\boldsymbol{z} = \begin{pmatrix} 0 \\ -\boldsymbol{s} \circ D_x \boldsymbol{b} + f\boldsymbol{v} \\ -\boldsymbol{s} \circ D_y \boldsymbol{b} - f\boldsymbol{u} \\ 0 \end{pmatrix},$$

and $\mathbf{H} \in \mathbb{R}^{4N \times (4N)^2}$ is a matricized tensor operator containing the quadratic terms

$$\mathbf{H}(\boldsymbol{z} \otimes \boldsymbol{z}) = \begin{pmatrix} -D_x(\boldsymbol{u} \circ \boldsymbol{h}) - D_y(\boldsymbol{v} \circ \boldsymbol{h}), \\ -\boldsymbol{u} \circ (D_x \boldsymbol{u}) - \boldsymbol{v} \circ (D_y \boldsymbol{u}) - \frac{\boldsymbol{h}}{2} \circ D_x \boldsymbol{s} - \boldsymbol{s} \circ D_x \boldsymbol{h}, \\ -\boldsymbol{u} \circ (D_x \boldsymbol{v}) - \boldsymbol{v} \circ (D_y \boldsymbol{v}) - \frac{\boldsymbol{h}}{2} \circ D_y \boldsymbol{s} - \boldsymbol{s} \circ D_y \boldsymbol{h}, \\ -\boldsymbol{u} \circ D_x \boldsymbol{s} - \boldsymbol{v} \circ D_y \boldsymbol{s} \end{pmatrix}.$$

CHAPTER 3

PROJECTION BASED INTRUSIVE REDUCED-ORDER MODELLING

Realistic numerical simulations of the SWEs are computationally costly because they require a large amount of computer memory and computation time. ROMs have been developed as an effective strategy to decrease the computational cost of simulating high-fidelity PDEs by constructing a low-dimensional linear subspace (reduced space) that approximately expresses the solution to the PDEs with a significantly low computational cost. The solutions (trajectories) of the high-fidelity FOM, obtained by space-time discretization of PDEs and then projected onto reduced space using the POD [16, 108]. The POD is a computationally efficient reduced-order modeling technique that been extensively applied in large-scale simulations of nonlinear PDEs. The dominant modes of the PDEs are recovered from the trajectories of the FOM. The computation of the FOM trajectories is performed in the "offline" stage. On the other hand, the "online" stage indicates the solution of the reduced system in the low-dimensional subspace. Various ROMs are constructed for the SWEs, in conservative form [79, 78], in the f-plane [44], in the β-plane [35, 36], with POD and in the β-plane [17, 18] with DMD. MOR methods for SWEs have been intensively investigated in the literature, see, e.g., [3, 17, 18, 44, 69, 78, 79, 109].

The main obstacle in constructing the efficient ROMs from the high-dimensional FOM is the efficient evaluation of the nonlinear terms on the POD basis. The computational cost is decreased by interpolating the nonlinear terms and evaluating them in the sampling points, known as hyper-reduction techniques [7, 29, 5, 131, 87, 22]. In this thesis, as a hyper-reduction technique, we consider DEIM [29], which is one

of the most popular hyper-reduction methods. The number of sampling points used in DEIM scales with the reduced dimension. For PDEs possessing nonlinear polynomial terms like SWEs, ROMs do not need approximating the nonlinearities through hyper-reduction techniques. The reduced operators can be precomputed in the offline stage. After projecting the full-order operators onto the reduced space, the low-dimensional ROM preserves the polynomial structure of the FOM. In recent years, for PDEs with nonlinear polynomial terms, the computationally efficient ROMs are developed by using some properties from tensor theory and by matricizations of tensors [10, 12, 11, 56, 72].

In the traditional projection-based intrusive MOR techniques, the conservation of physical quantities like energy is not, in general, guaranteed. When the physical quantities violated, ROMs frequently results in an unstable or qualitatively wrong reduced system, even when the high-fidelity system is stable. Thus, preservation of the physical quantities of the FOMs by the reduced system yields in physically meaningful ROMs for fluid dynamics problems, such as compressible and incompressible Euler equation [2]. The stability of ROMs over the long-time integration has been studied in the context of Lagrangian systems [23], and port-Hamiltonian systems [28]. In [1, 91], symplectic Galerkin projection are constructed for linear and nonlinear Hamiltonian systems, such as the linear wave equation, nonlinear Schrödinger equation, and Sine-Gordon equation that capture the symplectic structure of Hamiltonian systems to enforce long term stability to the ROMs.

We study structure preserving reduced-order modeling for the RTSWE that exploits skew-symmetry of the centered discretization schemes to recover conservation of the energy at the level of the ROMs. In Sections 3.1, 3.2 and 3.3, the POD and DEIM are introduced and the tensor algebra is briefly described, which are needed for constructing ROMs.

In Section 3.4, ROM are constructed by exploiting the linear-quadratic-structure of the RSWE after the semi-discretization in space without necessitating hyper reduction techniques like the DEIM. Utilizing tensor techniques [12, 11, 69], the computation of the ROM is further accelerated while preserving the skew-symmetric structure. For the time integration of the ROM, we have considered the linearly implicit Kahan's

28

method. The cubic Hamiltonian (energy) of the RSWE and the linear and quadratic Casimirs, i.e., the mass, the buoyancy, and the total vorticity, are well preserved and the speed-ups of order two are achieved over the FOM.

In [51, 63, 84], the skew-gradient structure of the Hamiltonian system of ODEs, such as the Korteweg-de Vries (KdV) equation with the constant Poisson matrix, are preserved in the ROMs using the modified POD-Galerkin projection. However, RTSWE possesses an additional difficulty in constructing structure-preserving ROM with the state-dependent skew-symmetric Poisson matrix. Moreover, due to the rational nonlinear term in the Poisson matrix, constructing efficient ROMs by offline-online decomposition becomes challenging in the case of all SWEs. Thus, in the case of the state-dependent skew-symmetric Poisson structure, preserving the skew-gradient structure in the ROMs becomes more challenging than with the constant Poisson structure. Another bottleneck is the lack of the numerical integration technique that preserving both the Poisson structure and the energy of a Hamiltonian system [57]. When the ROMs of RTSWE maintain the skew-gradient structure and thus have the energy-preservation law, energy-preserving integrators can be easily applied. We consider the AVF method [33] to integrate the semi-discrete skew-gradient RTSWE, which is conjugated to a Poisson integrator. Recently, the AVF method is used to construct ROMs to Hamiltonian systems like nonlinear Schrödinger equation [70] and the Korteweg-de Vries equation [51, 63, 84]. To separate the high-dimensional variables in the computation of the nonlinearities in the reduced form, we consider DEIM. Nevertheless, the basic implementation of a proposed structure-preserving DEIM does not permit the separation of online and offline phases in the nonlinear terms of the non-canonical Hamiltonian systems like the RTSWE. To accelerate the online computation of the ROMs, we approximated the Poisson matrix and the gradient of the Hamiltonian of the RTSWE with the DEIM, which yields a skew-gradient ROM with linear and quadratic terms only.

In Section 3.5.1, ROMs are constructed by using the POD-Galerkin projection which preserves the skew-gradient structure of the semi-discrete non-traditional rotating SWE (NTSWE) with full Coriolis force. The DEIM is applied only for the gradient of the Hamiltonian, but not for the skew-symmetric Poisson matrix.

In Section 3.5.3 a POD-DEIM ROM is constructed for the RTSWE that preserving both the skew-symmetric Poisson matrix and structure of the skew-gradient reduced system. Numerical simulations for the double vortex test case from [43] show that by preserving the structure of FOM of RTSWE, i.e., preservation of the skew-gradient structure, the long term integration of the ROM results in a physically meaningful solution. Furthermore, the numerical results indicate that ROMs of the RTSWE with the DEIM and POD yield a stable approximation of the solutions while exhibiting several orders of magnitude computational speed-ups over the FOMs.

3.1   Proper Orthogonal Decomposition (POD)

POD method seeks to find optimal basis for low-dimensional approximation of subsets in the Hilbert space. The basis vectors obtained by POD is extracted from the spatio-temporal observation matrix called "snapshot" matrix. Let us define the snapshot matrix as follows

$$S = \left(\mathbf{z}^1, \mathbf{z}^2, \ldots, \mathbf{z}^K\right) \tag{3.1}$$

where $\mathbf{z}^k = \mathbf{z}(t_k)$ called the snapshot at time instances t_k for $k = 1, \ldots, K$. The snapshots are obtained from the solution of the FOM. The POD basis vectors ψ_i for $i = 1, \ldots, r$ satisfy the following optimization problem

$$\underset{\{\psi_i\}_{i=1}^r}{\arg\min} \sum_{j=1}^K \|\mathbf{z}^j - \sum_{j=1}^r ((\mathbf{z}^j)^T \psi_i)\psi_i\|_2^2 \tag{3.2}$$

$$\text{subject to } \psi_i^T \psi_i = \delta_{ij}, \text{ for } 1 \leq i, j \leq r,$$

where δ_{ij} is the Kronecker delta. Now, assume that dimension of the FOM is greater than the number of snapshots: $N > K$. Then, the solution of the minimization problem (3.2) can be expressed in terms of left singular vectors of the snapshot matrix S. Let us define the SVD of snapshot matrix as

$$S = V\Sigma W^T,$$

where $\Sigma = diag(\sigma_1, \ldots, \sigma_K)$ is a diagonal matrix, containing singular values and $V = [v_1, \ldots, v_K]$ is the matrix containing left singular vectors. The POD modes can be expressed via first r left singular vectors. Let us denote $V_r = [v_1, \cdots, v_r]$ as the

30

POD basis matrix. Using r POD modes the error in the optimization problem (3.2) becomes

$$\min_{V_r \in \mathbb{R}^{N \times r}} ||S - V_r V_r^T S||_F^2 = \sum_{j=r+1}^{K} \sigma_j^2 \,,$$

where $|| \cdot ||_F$ denotes the Frobenius norm. The optimization error gives motivation to determine the order of the approximation r via following relative "cumulative energy" criteria

$$\frac{\sum_{j=1}^{n} \sigma_{i,j}^2}{\sum_{j=1}^{K} \sigma_{i,j}^2} > 1 - \kappa, \tag{3.3}$$

where κ is a user-specified tolerance.

Using POD modes the state can be approximated as $\mathbf{z} = \widehat{\mathbf{z}} \approx V_r \mathbf{z}_r$. After substituting the approximation of the state in (2.30) the approximate model takes form of

$$V_r \frac{d\mathbf{z}_r}{dt} - F(\widehat{\mathbf{z}}) = R(\widehat{\mathbf{z}}). \tag{3.4}$$

Finally, employing Galerkin projection on (3.4), the ROM becomes

$$\frac{d\mathbf{z}_r}{dt} - V_r^T F(\widehat{\mathbf{z}}) = V_r^T R(\widehat{\mathbf{z}}) = 0. \tag{3.5}$$

The deterministic SVD for the snapshot matrices of size $N \times K$ has complexity of $\mathcal{O}(\min(NK^2, N^K))$. In this study, we use the randomized SVD (rSVD) [58, 6] which is faster than the deterministic SVD for big matrices and has complexity of $\mathcal{O}(NKr)$.

3.2   Discrete Empirical Interpolation Method (DEIM)

In the nonlinear PDEs like RSWE, the right-hand side $F(\mathbf{z})$ contains nonlinear terms, which posses burden in Jacobian and online computation of the ROM. For instance, let the right-hand side contains $F(\mathbf{z}) = A\mathbf{z} + N(\mathbf{z})$, where A corresponds linear operator and $N(\mathbf{z})$ is nonlinear function. Using POD-Galerkin ROM, right-hand becomes $V_r^T F(V_r \mathbf{z}_r) = V_r^T A V_r \mathbf{z}_r + V_r^T N(V_r \mathbf{z}_r)$, where $V_r^T A V_r$ can be precomputed, but $V_r^T N(V_r \mathbf{z}_r)$ still needs to evaluated in FOM dimension. One way to deal with inefficiency due to nonlinearities in the FOM is using discrete empirical interpolation

31

method (DEIM). The DEIM approximates the nonlinearities using interpolation on a subset which is independent of FOM dimension.

Let us consider a nonlinear vector $\boldsymbol{f}(\widehat{\mathbf{z}}) \in \mathbb{R}^N$. The DEIM procedure in [29] introduced to approximate the nonlinear vectors by interpolating them onto an empirical basis, that is,

$$\boldsymbol{f}(\widehat{\mathbf{z}}) \approx \Phi \mathsf{c}(t),$$

where $\Phi = [\phi_1, \ldots, \phi_p] \in \mathbb{R}^{N \times p}$ is the basis matrix and $\mathsf{c}(t) : [0, T] \mapsto \mathbb{R}^p$ is the reduced coefficient vector. Let us define $\mathsf{P} = [e_{\rho_1}, \ldots, e_{\rho_p}] \in \mathbb{R}^{N \times p}$ as the subset of columns of the identity matrix. Let $(\mathsf{P})^\top \Phi$ is invertible, then, the coefficient vector $\mathsf{c}(t)$ can be uniquely obtained by solving $(\mathsf{P})^\top \Phi \mathsf{c}(t) = (\mathsf{P})^\top \boldsymbol{f}(\widehat{\boldsymbol{z}}(t))$. Hence, the nonlinear terms can be approximated as follows

$$\boldsymbol{f}(\widehat{\boldsymbol{z}}(t)) \approx \Phi \mathsf{c}(t) = \Phi((\mathsf{P})^\top \Phi)^{-1}(\mathsf{P})^\top \boldsymbol{f}(\widehat{\boldsymbol{z}}(t)), \tag{3.6}$$

where $\Phi((\mathsf{P})^\top \Phi)^{-1}$ can be precomputed.

Moreover, the reduced approximated nonlinear vectors using DEIM are defined as $\boldsymbol{f}_r(\widehat{\boldsymbol{z}}(t)) := (\mathsf{P})^\top \boldsymbol{f}(\widehat{\boldsymbol{z}}(t)) \in \mathbb{R}^p$, by which online computation of the nonlinear term $\boldsymbol{f}(\widehat{\boldsymbol{z}}(t)) \in \mathbb{R}^N$ no longer scales with the FOM dimension N and instead just $p \ll N$ dimensional DEIM reduced nonlinear vectors $\boldsymbol{f}_r(\widehat{\boldsymbol{z}}(t))$ need to be computed. The entries of reduced nonlinear terms $\boldsymbol{f}_r(\widehat{\boldsymbol{z}}(t))$ consist of the p selected entries of the nonlinear vectors $\boldsymbol{f}(\widehat{\boldsymbol{z}}(t))$ among N entries.

The accuracy of DEIM related on the selection of the basis, and not much by the choice of the selection matrix P. Generally, the interpolation basis $\{\phi_1, \ldots, \phi_p\}$ is obtained from the snapshots matrices of the nonlinear vectors, e.g.,

$$\mathsf{S}_{\boldsymbol{f}} = [\boldsymbol{f}^1, \boldsymbol{f}^2, \cdots, \boldsymbol{f}^K] \in \mathbb{R}^{N \times K}, \tag{3.7}$$

where $\boldsymbol{f}^k = \boldsymbol{f}(\widehat{\boldsymbol{z}}^k)$ denotes the nonlinear vector $\boldsymbol{f}(\widehat{\boldsymbol{z}}(t))$ at time t_k, computed by using the solution vectors $\widehat{\boldsymbol{z}}^k = \widehat{\boldsymbol{z}}(t_k)$, $k = 1, \ldots, K$. The columns of the matrices $\Phi = [\phi_1, \ldots, \phi_p]$ are determined as the first $p \ll N$ dominant left singular vectors in the SVD of $\mathsf{S}_{\boldsymbol{f}}$. The selection matrix P for DEIM is determined by a greedy algorithm based on the system residual; see [29, Algorithm 3.1].

A sophisticated way of determining the sampling points is the Q-DEIM [132] which depends on QR decomposition with column pivoting. It is showed that Q-DEIM

32

yields better stability and accuracy properties of the determined selection matrix $\mathsf{P}^{(j)}$ with the pivoted QR-factorization of $(\Phi^{(j)})^\top$. In this section, we employ Q-DEIM for obtaining the selection matrices $\mathsf{P}^{(j)}$, see Algorithm 1.

Algorithm 1 Q-DEIM

1: **Input:** Basis matrix $\Phi \in \mathbb{R}^{N \times p}$

2: **Output:** Selection matrix P

3: Perform pivoted QR factorization of $\Phi^\top$ so that $\Phi^\top \Pi = QR$

4: Set $\mathsf{P} = \Pi(:, 1:p)$

3.3 Tensor Algebra

Typically, tensors are defined as multidimensional arrays which storing the numbers, e.g. real numbers. The number of a tensor dimension called the order or mode of a tensor. For instance, scalars can be represented as zeroth-order tensors, vectors can be represented as first-order tensors, and matrices can be represented as second-order tensors. In this study, we will focus on tensors of order three. Figure 3.1 shows tensors of order zero to up to order three.
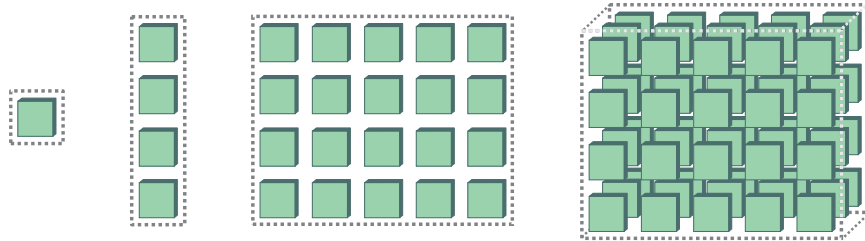
Figure 3.1: $x \in \mathbb{R}$, $\boldsymbol{x} \in \mathbb{R}^4$, $\boldsymbol{X} \in \mathbb{R}^{4 \times 5}$, $\mathscr{X} \in \mathbb{R}^{4 \times 5 \times 3}$.

In this section, we use a similar notation with [71]. The subarray obtained by fixing all indices except one index is called fibers, e.g., for order-three tensor $\boldsymbol{x}_{:jk} = \boldsymbol{x}_{jk}$. Fixing all except two indices of tensors called the slices of the tensor, e.g., $\boldsymbol{X}_{::k} = \boldsymbol{X}_k$. Fibers obtained by fixing the first index, the second index, and the third index are called column, row, and tube fibers, respectively. Frontal, lateral, and horizontal slices are obtained by fixing first and second, first and third, second and third indices, respectively. Fibers and slices for a order-three tensor are shown in Figure 3.2 and 3.3.
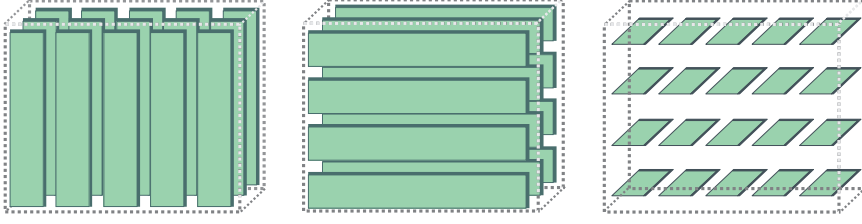
Figure 3.2: (left): Column, (middle): row, (right): tube fibers of a mode-3 tensor.
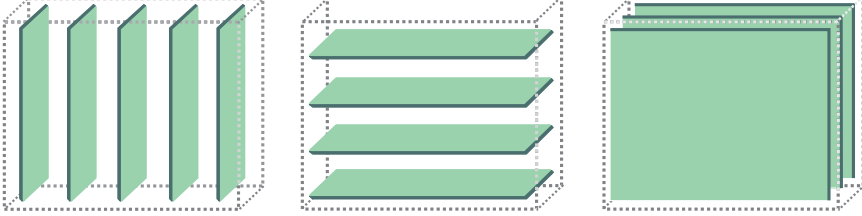
Figure 3.3: (left): Lateral, (middle): horizontal, (right): frontal slices of a mode-3 tensor.

For a given matrix $\boldsymbol{X} \in \mathbb{R}^{I \times J}$ vertically stacking the columns of $\boldsymbol{X}$ into a vector called vectorization of the matrix.

$$
\mathrm{vec}(\boldsymbol{X}) = \begin{bmatrix} \boldsymbol{x}_{:1} \\ \boldsymbol{x}_{:2} \\ \vdots \\ \boldsymbol{x}_{:J} \end{bmatrix} . \tag{3.8}
$$

Similiar to vectorization by reordering the tensor into a matrix called matricization. In this study, we focus on the mode-n matricization (unfolding). For a given tensor $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$, mode-$n$ matricization of the tensor into matrix is denoted as $\boldsymbol{X}_{(n)} \in \mathbb{R}^{I_n \times (I_1 \cdot \ldots \cdot I_{n-1} \cdot I_{n+1} \cdot \ldots \cdot I_N)}$.

Let us consider $m \in \boldsymbol{M}$ as the element of the matricized tensor and $x \in \mathscr{X}$ as the element of a tensor. The mode-n unfolding is defined as following:

$$
x_{i_1, i_2, \cdots, i_N} \mapsto m_{i_n, j} \text{ with } j = 1 + \sum_{\substack{k=1 \\ k \neq n}}^{N} \left((i_k - 1) \prod_{\substack{m=1 \\ m \neq n}}^{k-1} I_m \right) \tag{3.9}
$$

.

For instance, let $\mathscr{X} \in \mathbb{R}^{3 \times 3 \times 2}$ be a tensor with the following frontal slices:

$$
\boldsymbol{X}_1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, \qquad \boldsymbol{X}_2 = \begin{bmatrix} 10 & 11 & 12 \\ 13 & 14 & 15 \\ 16 & 17 & 18 \end{bmatrix} .
$$

Then, the three mode-n matricizations are:

$$
\boldsymbol{X}_{(1)} =
\begin{bmatrix}
1 & 2 & 3 & 10 & 11 & 12 \\
4 & 5 & 6 & 13 & 14 & 15 \\
7 & 8 & 9 & 16 & 17 & 18
\end{bmatrix},
$$

$$
\boldsymbol{X}_{(2)} =
\begin{bmatrix}
1 & 4 & 7 & 10 & 13 & 16 \\
2 & 5 & 8 & 11 & 14 & 17 \\
3 & 6 & 9 & 12 & 15 & 18
\end{bmatrix},
$$

$$
\boldsymbol{X}_{(3)} =
\begin{bmatrix}
1 & 4 & 7 & 2 & 5 & 8 & 3 & 6 & 9 \\
10 & 13 & 16 & 11 & 14 & 17 & 12 & 15 & 18
\end{bmatrix}.
$$

3.3.1 Important Tensor Products

Definition 1. *The Kronecker product between any two matrices $\boldsymbol{A} \in \mathbb{R}^{I \times J}$ and $\boldsymbol{B} \in \mathbb{R}^{K \times L}$ is defined as follows*

$$
\begin{aligned}
\boldsymbol{A} \otimes \boldsymbol{B} &:=
\begin{bmatrix}
a_{11}\boldsymbol{B} & a_{12}\boldsymbol{B} & \cdots & a_{1J}\boldsymbol{B} \\
a_{21}\boldsymbol{B} & a_{22}\boldsymbol{B} & \cdots & a_{2J}\boldsymbol{B} \\
\vdots & \vdots & \ddots & \vdots \\
a_{I1}\boldsymbol{B} & a_{I2}\boldsymbol{B} & \cdots & a_{IJ}\boldsymbol{B}
\end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{a}_1 \otimes \boldsymbol{b}_1 & \boldsymbol{a}_1 \otimes \boldsymbol{b}_2 & \cdots & \boldsymbol{a}_J \otimes \boldsymbol{b}_{L-1} & \boldsymbol{a}_J \otimes \boldsymbol{b}_L \end{bmatrix} \in \mathbb{R}^{(IK) \times (JL)},
\end{aligned}
\tag{3.10}
$$

where $\otimes$ denotes the Kronecker product.

Definition 2. *Let $\boldsymbol{A} \in \mathbb{R}^{I \times K}$ and $\boldsymbol{B} \in \mathbb{R}^{J \times K}$ be two matrices, the Khatri-Rao (column-wise Kronecker) product between two matrices is defined as:*

$$
\boldsymbol{A} \odot \boldsymbol{B} := \begin{bmatrix} \boldsymbol{a}_1 \otimes \boldsymbol{b}_1 & \boldsymbol{a}_2 \otimes \boldsymbol{b}_2 & \cdots & \boldsymbol{a}_K \otimes \boldsymbol{b}_K \end{bmatrix} \in \mathbb{R}^{(IJ) \times K},
\tag{3.11}
$$

where $\odot$ denote Khatri-Rao product.

Definition 3. *Let $A \in \mathbb{R}^{I \times J}$ and $B \in \mathbb{R}^{I \times J}$ matrices, the Hadamard (element-wise) product between two matrices is defined as follows*

$$A \circ B := \begin{bmatrix} a_{11}b_{11} & a_{12}b_{12} & \cdots & a_{1J}b_{1J} \\ a_{21}b_{21} & a_{22}b_{22} & \cdots & a_{2J}b_{2J} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I1}b_{I1} & a_{I2}b_{I2} & \cdots & a_{IJ}b_{IJ} \end{bmatrix} \in \mathbb{R}^{I \times J}, \tag{3.12}$$

where $\circ$ denotes Hadamard product.

Definition 4. *Let the tensor $\mathscr{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ and the matrix $M \in \mathbb{R}^{J \times I_n}$ are given, the n-mode product between these tensor and matrix is defined as*

$$(\mathscr{X} \times_n M)_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 \ldots i_N} m_{j i_n}, \tag{3.13}$$

where $\times_n$ denotes the n-mode product.

An equivalent expression of Definition (4) can be obtained using unfolded tensors as $Y_{(n)} = M X_{(n)}$.

3.4 ROM for RSWE in Tensor Framework

In this subsection, we will illustrate the usage of tensorial techniques to obtain ROM with online cost independent of FOM dimension.

We construct the ROM by approximating the solutions of FOM (2.29) onto a low-dimensional subspace spanned by the POD basis vectors. The POD basis vectors obtained from the snapshot matrix, whose columns consist of fully discrete solution vectors at each time instances.

Typically, all the state variables are stacked in one vector to determine the POD basis vectors, then the common reduced subspace is determined by taking the SVD of the snapshot data. When the governing PDEs like the RSWE are coupled, computing POD basis by the stacked states causes unstable ROMs [95, 98] due to the unpreserved coupling structure of the FOM. To conserve the coupling structure of RSWE in the ROMs, the POD basis vectors are determined separately for each the state vector h, u, and v.

The ROM arising from the fluid problems poses fluctuations, which can be avoided by the mean centred snapshots are obtained at time instances t_k, $k = 1, \ldots, K$, in the snapshot matrices S_u, S_v, and S_h

$$S_u = \left(\mathbf{u}^1 - \overline{\mathbf{u}}, \mathbf{u}^2 - \overline{\mathbf{u}}, \cdots, \mathbf{u}^K - \overline{\mathbf{u}}\right) \in \mathbb{R}^{N \times K},$$

$$S_v = \left(\mathbf{v}^1 - \overline{\mathbf{v}}, \mathbf{v}^2 - \overline{\mathbf{v}}, \cdots, \mathbf{v}^K - \overline{\mathbf{v}}\right) \in \mathbb{R}^{N \times K},$$

$$S_h = \left(\mathbf{h}^1 - \overline{\mathbf{h}}, \mathbf{h}^2 - \overline{\mathbf{h}}, \cdots, \mathbf{h}^K - \overline{\mathbf{h}}\right) \in \mathbb{R}^{N \times K},$$

where $\overline{\mathbf{u}}, \overline{\mathbf{v}}, \overline{\mathbf{h}} \in \mathbb{R}^N$ denote the mean of the snapshots defined by

$$\overline{\mathbf{u}} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{u}^k, \quad \overline{\mathbf{v}} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{v}^k, \quad \overline{\mathbf{h}} = \frac{1}{K} \sum_{k=1}^{K} \mathbf{h}^k.$$

The mean-subtracted ROMs is used commonly in fluid dynamics to stabilize the ROMs, and it guarantees that reduced model solutions would satisfy the same boundary conditions for the FOM [17].

The objective of POD method is to form the POD basis matrices $V_u, V_v, V_h \in \mathbb{R}^{N \times r}$ which span approximately the column space of the snapshot matrices S_u, S_v and S_h, for a positive integer $r \ll K$, respectively. The POD basis matrices are obtained by the singular value decomposition (SVD) to the snapshot matrices

$$S_u = V_{r,u} \Sigma_{r,u} W_{r,u}^T, \quad S_v = V_{r,v} \Sigma_{r,v} W_{r,v}^T, \quad S_h = V_{r,h} \Sigma_{r,h} W_{r,h}^T,$$

where the columns of the orthonormal matrices $V_{r,i} \in \mathbb{R}^{N \times K}$ and $W_{r,i} \in \mathbb{R}^{K \times K}$ are the left and right singular vectors of the snapshot matrix $S_{r,i}$, respectively, and the diagonal matrix $\Sigma_{r,i} \in \mathbb{R}^{K \times K}$ with the diagonal elements $(\Sigma_{r,i})_{jj} = \sigma_{i,j}$, $j = 1, \ldots, K$, contains the singular values of $S_{r,i}$, $i \in \{u, v, h\}$. The POD basis matrix V_i constructed by the first r left singular vectors from $V_{r,i}$, $i \in \{u, v, h\}$. The left singular vectors of V_i, $i \in \{u, v, h\}$ are called POD modes. Using the POD basis following optimization problem is satisfied

$$\min_{V_i \in \mathbb{R}^{N \times n}} ||S_i - V_i V_i^T S_i||_F^2 = \sum_{j=n+1}^{K} \sigma_{i,j}^2, \quad i \in \{u, v, h\},$$

where $\| \cdot \|_F$ denotes the Frobenius norm. Hence, the projection error in the snapshot matrices is obtained from the sum of the squared singular values greater than the dimension of ROM. The projection error creates guidance on the accuracy of ROM.

After computing the POD basis matrices V_u, V_v, and V_h, we approximate $\widehat{\mathbf{u}}$, $\widehat{\mathbf{v}}$, and $\widehat{\mathbf{h}}$ to the FOM solutions $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{h}$ as follows

$$\mathbf{u} \approx \widehat{\mathbf{u}} = \overline{\mathbf{u}} + V_u \mathbf{u}_r , \quad \mathbf{v} \approx \widehat{\mathbf{v}} = \overline{\mathbf{v}} + V_v \mathbf{v}_r , \quad \mathbf{h} \approx \widehat{\mathbf{h}} = \overline{\mathbf{h}} + V_h \mathbf{h}_r ,$$

where $\mathbf{u}_r$, $\mathbf{v}_r$, $\mathbf{h}_r \in \mathbb{R}^r$ are the solution vectors of the reduced system of dimension $r \ll N$. For convenience, we also define the following vectors and matrix

$$\overline{\mathbf{z}} = \begin{pmatrix} \overline{\mathbf{u}} \\ \overline{\mathbf{v}} \\ \overline{\mathbf{h}} \end{pmatrix} \in \mathbb{R}^{3N} , \quad \mathbf{z}_r = \begin{pmatrix} \mathbf{u}_r \\ \mathbf{v}_r \\ \mathbf{h}_r \end{pmatrix} \in \mathbb{R}^{3r} , \quad V_z = \begin{pmatrix} V_u & & \\ & V_v & \\ & & V_h \end{pmatrix} \in \mathbb{R}^{3N \times 3r},$$

and then we get the approximation $\mathbf{z} \approx \widehat{\mathbf{z}} = \overline{\mathbf{z}} + V_z \mathbf{z}_r$. We also note that since the columns of the r-POD matrices are orthonormal, the approximation identity satisfies $\mathbf{z}_r = V_z^T (\widehat{\mathbf{z}} - \overline{\mathbf{z}})$.

Constructing ODEs and PDEs possessing the nonlinear polynomial terms, ROMs do not need to approximate the nonlinear term $F(\cdot)$ by sampling; hence the reduced-order operators arising from projection can be constructed in the offline phase. Separation of the offline-online computation as the hyper-reduction methods increase the computational efficiency of the ROM. Previously, the quadratic nonlinearities of the FOMs are utilized to construct ROMs for the Navier-Stokes equations [55, 65]. This approach avoids the approximation of the nonlinearities by hyper-reduction. Moreover, it enables the separation of offline and online phases of FOM and ROM, and the ROM conserve the linear-quadratic structure of the high-fidelity FOM.

We rewrite the FOM in terms of Kronecker product and tensors to avoid approximating the nonlinearities in the SWE (2.29) by the hyper-reduction methods, such as DEIM. The matricizations of the tensors allow us to perform mathematical operations with tensors easily. A popular matricization of a tensor $\mathcal{Q}$ is called n-mode matricization Q_n [10]. The quadratic systems like (2.29) possess three different unfoldings, depending on the n-mode used for the unfolding. We refer to [71] for the basic concepts and details of tensors. The matricizations is also advantageous in terms of matrix-matrix products so that tensor-matrix multiplications can be performed. The matriciziced tensor Q corresponds to the Hessian of the right-hand side of (2.29). In this study, we have exploited the structure of the matriciziced tensor Q to construct the

reduced Hessian efficiently [10]. The matricizied tensor Q corresponds to the unfolding of a 3-tensor $\mathcal{Q} \in \mathbb{R}^{3N \times 3N \times 3N}$. Due to the structure of well-known discretization methods such as finite differences and finite elements, the matrix Q is sparse. For the SWE with quadratic nonlinearities, the cross-terms $\mathbf{z}_i \cdot \mathbf{z}_j$ vanish for $|i - j| > 3$ in the semi-discretized ODE (2.29). Hence the number of nonzero terms of $Q_{(j)}$ is only $2N$.

The semi-discrete system (2.29) can be rewritten using the Kronecker product as the following linear-quadratic system of ODEs

$$\frac{d\mathbf{z}}{dt} = F(\mathbf{z}) = \widetilde{R}_1(\mathbf{z}) + \widetilde{R}_2(\mathbf{z}) + L(\mathbf{z}), \tag{3.14}$$

with the following quadratic terms

$$\widetilde{R}_1(\mathbf{z}) = -A^x Q \left(\begin{pmatrix} \mathbf{u} \\ \mathbf{u} \\ \mathbf{u} \end{pmatrix} \otimes (B^x \mathbf{z}) \right), \quad \widetilde{R}_2(\mathbf{z}) = -A^y Q \left(\begin{pmatrix} \mathbf{v} \\ \mathbf{v} \\ \mathbf{v} \end{pmatrix} \otimes (B^y \mathbf{z}) \right), \tag{3.15}$$

where $\otimes$ denotes the Kronecker product, and the matrices $A^x, A^y, B^x, B^y \in \mathbb{R}^{3N \times 3N}$ are given by

$$A^x = \begin{pmatrix} I_N & & \\ & I_N & \\ & & D_x \end{pmatrix}, \quad A^y = \begin{pmatrix} I_N & & \\ & I_N & \\ & & D_y \end{pmatrix},$$

$$B^x = \begin{pmatrix} D_x & & \\ & D_x & \\ & & I_N \end{pmatrix}, \quad B^y = \begin{pmatrix} D_y & & \\ & D_y & \\ & & I_N \end{pmatrix},$$

where I_N represent the identity matrix of size $N \times N$. In (3.15), the matrix $Q \in \mathbb{R}^{3N \times (3N)^2}$ represents the matricized 3-tensor such that $Q(\mathbf{z} \otimes \mathbf{z}) = \mathbf{z} \circ \mathbf{z}$ is satisfied. By substituting the approximation $\mathbf{z} \approx \widehat{\mathbf{z}} = \overline{\mathbf{z}} + V_z \mathbf{z}_r$ into (3.14), and employing Galerkin projection onto V_z, the following reduced linear-quadratic system is obtained:

$$\frac{d\mathbf{z}_r}{dt} = F_r(\widehat{\mathbf{z}}) = F_r^u(\widehat{\mathbf{z}}) + F_r^v(\widehat{\mathbf{z}}) + L_r(\widehat{\mathbf{z}}), \tag{3.16}$$

where we set

$$F_r^u(\widehat{\mathbf{z}}) = -V_z^T A^x Q \left(\begin{pmatrix} \widehat{\mathbf{u}} \\ \widehat{\mathbf{u}} \\ \widehat{\mathbf{u}} \end{pmatrix} \otimes (B^x \widehat{\mathbf{z}}) \right),$$

$$F_r^v(\widehat{\mathbf{z}}) = -V_z^T A^y Q \left(\begin{pmatrix} \widehat{\mathbf{v}} \\ \widehat{\mathbf{v}} \\ \widehat{\mathbf{v}} \end{pmatrix} \otimes (B^y \widehat{\mathbf{z}}) \right),$$

$$L_r(\widehat{\mathbf{z}}) = V_z^T L(\widehat{\mathbf{z}}).$$

Computation of the reduced quadratic terms $F_r^u(\widehat{\mathbf{z}})$ and $F_r^v(\widehat{\mathbf{z}})$ in the reduced model (3.16) is tricky. Next, we discuss the efficient computation of the reduced quadratic terms $F_r^u(\widehat{\mathbf{z}})$. Computation of the second reduced quadratic term $F_r^v(\widehat{\mathbf{z}})$ can be done in a similar way. Substituting the mean-centered approximation $\widehat{\mathbf{z}} = \overline{\mathbf{z}} + V_z \mathbf{z}_r$ into the quadratic term $F_r^u(\widehat{\mathbf{z}})$, and using the properties of the Kronecker product operation, we obtain

$$
\begin{aligned}
F_r^u(\widehat{\mathbf{z}}) &= F_r^u(\overline{\mathbf{z}} + V_z \mathbf{z}_r) \\
&= -V_z^T A^x Q \left(\begin{pmatrix} \overline{\mathbf{u}} + V_u \mathbf{u}_r \\ \overline{\mathbf{u}} + V_u \mathbf{u}_r \\ \overline{\mathbf{u}} + V_u \mathbf{u}_r \end{pmatrix} \otimes (B^x (\overline{\mathbf{z}} + V_z \mathbf{z}_r)) \right), \\
&= -V_z^T A^x Q \left(\begin{pmatrix} \overline{\mathbf{u}} \\ \overline{\mathbf{u}} \\ \overline{\mathbf{u}} \end{pmatrix} \otimes (B^x \overline{\mathbf{z}}) + \begin{pmatrix} \overline{\mathbf{u}} \\ \overline{\mathbf{u}} \\ \overline{\mathbf{u}} \end{pmatrix} \otimes (B^x V_z \mathbf{z}_r) \right) \\
&\quad - V_z^T A^x Q \left(\begin{pmatrix} V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \end{pmatrix} \otimes (B^x \overline{\mathbf{z}}) + \begin{pmatrix} V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \end{pmatrix} \otimes (B^x V_z \mathbf{z}_r) \right).
\end{aligned}
\tag{3.17}
$$

All the term are at most linear in (3.17), only last terms includes quadratic term. Using the properties of the Kronecker product high-dimensional variables can be de-

composed as follows

$$
-V_z^T A^x Q \left(\begin{pmatrix} V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \\ V_u \mathbf{u}_r \end{pmatrix} \otimes (B^x V_z \mathbf{z}_r) \right) = -V_z^T A^x Q \left(V_u^* \otimes (B^x V_z) \right) \left(\begin{pmatrix} \mathbf{u}_r \\ \mathbf{u}_r \\ \mathbf{u}_r \end{pmatrix} \otimes (\mathbf{z}_r) \right)
$$
$$
= Q_{u,r} \left(\begin{pmatrix} \mathbf{u}_r \\ \mathbf{u}_r \\ \mathbf{u}_r \end{pmatrix} \otimes (\mathbf{z}_r) \right), \tag{3.18}
$$

where the matrix $Q_{u,r} = -V_z^T A^x Q \left(V_u^* \otimes (B^x V_z) \right) \in \mathbb{R}^{3r \times (3r)^2}$ can be computed in the offline stage, and

$$
V_u^* = \begin{pmatrix} V_u & & \\ & V_u & \\ & & V_u \end{pmatrix} \in \mathbb{R}^{3N \times 3r}.
$$

The most significant computational issue in constructing $Q_{u,r}$ is the computation of the Kronecker product $V_u^* \otimes (B^x V_z) \in \mathbb{R}^{(3N)^2 \times (3r)^2}$, which has a cost of $\mathcal{O}(r^2 N^2)$ for quadratic terms. Due to the dense structure of POD basis matrices, the construction of $Q_{u,r}$ needs extra attention. In [10], the reduced matricized tensor $Q_{u,r}$ is constructed by avoiding the computation of the Kronecker product $V_u^* \otimes (B^x V_z)$, having a complexity of order $\mathcal{O}(r N^2)$. Therein, utilising the μ-mode (matrix) product, the reduced matricized tensor $Q_{u,r}$ efficiently constructed as following

- Compute $\mathcal{Y}^{3r \times 3N \times 3N}$ by $Y^{(1)} = -V_z^T A^x Q$,

- Compute $\mathcal{Z}^{3r \times 3r \times 3N}$ by $Z^{(2)} = V_z^T (B^x)^T Y^{(2)}$,

- Compute $\mathcal{Q}_{u,r}^{3r \times 3r \times 3r}$ by $Q_{u,r}^{(3)} = (V_u^*)^T Z^{(3)}$.

Although using the μ-mode (matrix) product to compute $Q_{u,r}$ decreases the complexity, still the matrix Q must constructed for each different polynomial nonlinear terms. Recently, in [12, 11] two new algorithms are used for efficient construction of the reduced matricized tensor $Q_{u,r}$ by taking advantage of the special structure of Kronecker product. The compact construction of the reduced matricized tensor $Q_{u,r}$ in

41

MatLab notation can be done as follows [12, 11]

$$
\begin{aligned}
Q_{u,r} &= -V_z^T A^x Q \left(V_u^* \otimes (B^x V_z) \right) \\
&= -V_z^T A^x Q (V_u^* \otimes G) \\
&= -V_z^T A^x \begin{pmatrix} V_u^*(1,:) \otimes G(1,:) \\ \vdots \\ V_u^*(3N,:) \otimes G(3N,:) \end{pmatrix},
\end{aligned} \tag{3.19}
$$

where $G = B^x V_z \in \mathbb{R}^{3N \times 3r}$. Hence, the reduced matricized tensor $Q_{u,r}$ can be formed without explicitly constructing the matrix Q. The complexity of (3.19) is $\mathcal{O}(Nr^3)$. The computation of (3.19) is carried out in the offline phase over a loop in the FOM dimension. Nevertheless, for large-scale problems, this operation is time-consuming. Next, we show the efficient construction of (3.19) for large-scale problems. The transpose of the Kronecker products of any given two vectors $\mathbf{a}$ and $\mathbf{b}$ can be represented as follows

$$
\begin{aligned}
(\text{vec}(\mathbf{b}\mathbf{a}^\top))^\top &= (\mathbf{a} \otimes \mathbf{b})^\top \\
&= \mathbf{a}^\top \otimes \mathbf{b}^\top,
\end{aligned} \tag{3.20}
$$

where vec $(\cdot)$ denotes vectorization of a matrix. Using (3.20), the matrix $N :=$ $Q(V_u^* \otimes G) \in \mathbb{R}^{3N \times (3r)^2}$ in (3.19) can be constructed as follows

$$
N(i,:) = \left(\text{vec} \left(G(i,:)^T V_u^*(i,:) \right) \right)^T, \quad i = 1, 2, \ldots, 3N.
$$

We use "MULTIPROD" [76] routine to increase the efficiency in the offline computation. MULTIPROD uses virtual array expansion to perform multiple matrix products. When the matrix $V_u^* \in \mathbb{R}^{3N \times 3r}$ is reshaped as $V_u^* \in \mathbb{R}^{3N \times 1 \times 3r}$, then MULTIPROD is applied to G and V_u^* in 2 and 3 dimensions. MULTIPROD assigns virtually a singleton to the third dimension of G, and we get the 3-dimensional array (tensor) $\mathcal{N} := \text{MULTIPROD}(G, V_u^*) \in \mathbb{R}^{3N \times 3r \times 3r}$. Thus, we can represent (3.19) as $Q_{u,r} = -V_z^T A^x N^{(1)}$, where $N^{(1)} \in \mathbb{R}^{3N \times (3r)^2}$ is the matricization of $\mathcal{N}$. In Section 3.4.1, we compare the computational efficiency of computing the reduced matricized tensor $Q_{u,r}$ by the algorithm in [10] with the algorithms in [12, 11] improved by the use of MULTIPROD.

3.4.1 Numerical Results

Our test example for the RSWE (2.9) on the spatial domain $\Omega = [0, 1]^2$ is given, with $g = 1$, $f = 0$, and with the initial conditions [114]

$$h(\mathbf{x}, 0) = 1 + \frac{1}{2} \exp\left[-25\left(x - \frac{1}{2}\right)^2 - 25\left(y - \frac{1}{2}\right)^2\right],$$

$$u(\mathbf{x}, 0) = -\frac{1}{2\pi} \sin(\pi x) \sin(2\pi y),$$

$$v(\mathbf{x}, 0) = \frac{1}{2\pi} \sin(2\pi x) \sin(\pi y).$$

The periodic boundary conditions are satisfied by the initial conditions. We set the final time $T = 50$ for simulation, and spatial and temporal domains are discretized with the mesh sizes $\Delta x = 0.01$ and $\Delta t = 4\Delta x$, respectively. This discretization leads to a FOM dimension of size $N = 10000$, and $K = 1250$ time intervals, therefore each snapshot matrix S_u, S_v, and S_h has size 10000×1250. The FOM of the SWE (2.9) is considered in the f-plane, and integrated with Kahan's time integrator (SWE-Kahan). To demonstrate the efficiency of the ROM, we consider (3.16) without tensorial framework (POD-Kahan) and with tensorial framework (TPOD-Kahan).

In Figure 3.4, the FOM solution of SWE-Kahan for the height field $\mathbf{h}$ and the velocity fields $\mathbf{u}$, $\mathbf{v}$ are given. We plot the normalized singular values of the snapshot matrices S_u, S_v, and S_h related to the velocity fields $\mathbf{u}$, $\mathbf{v}$, and the height field $\mathbf{h}$, respectively, in Figure 3.5. The normalized singular values exhibiting a slow decay for each snapshot matrix, which is a typical behavior for the problems with wave phenomena in fluid dynamics [88]. Slow decay in the singular values poses a big difficulty for ROMs, their dynamical behaviour cannot be captured accurately by the linear combination of a few POD modes.

To examine the conservation of the discrete enstrophy (2.22), discrete energy (2.21), and discrete vorticity (2.23) for a FOM solution vector $\mathbf{z}$ (or ROM approximation vector $\widehat{\mathbf{z}}$), we use the time-averaged absolute errors $\| \cdot \|_H$, $\| \cdot \|_Z$ and $\| \cdot \|_V$ defined as follows

$$\|\mathbf{z}\|_E = \frac{1}{K} \sum_{k=1}^{K} |E(\mathbf{z}^k) - E(\mathbf{z}^0)|, \quad E \equiv H, Z, V. \tag{3.21}$$

43
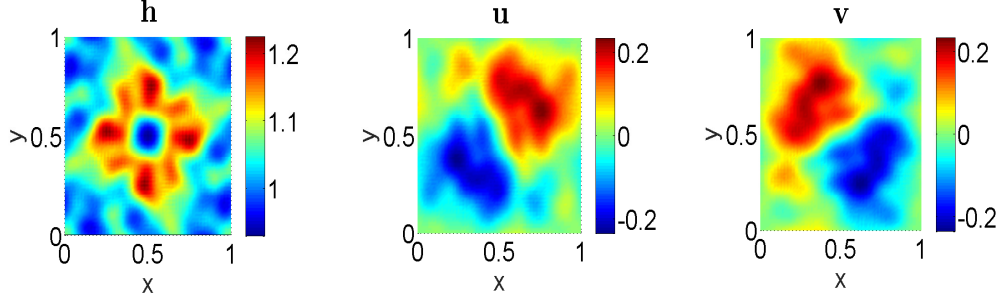
Figure 3.4: FOM solutions at final time $T = 50$.
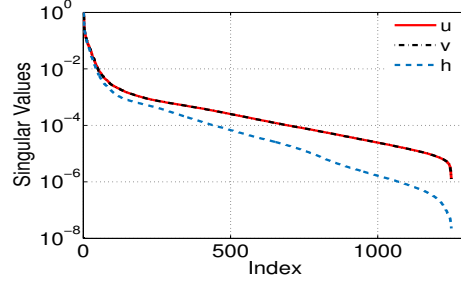
Figure 3.5: Singular values of the snapshot matrix.

Furthermore, to measure the global error between the fully discrete ROM approximation (FOM-ROM error) and the fully discrete FOM solution, we define the following time averaged relative L_2 errors for the each state variables $\mathbf{w} \equiv \mathbf{u}, \mathbf{v}, \mathbf{h}$

$$\|\mathbf{w} - \widehat{\mathbf{w}}\|_{Rel} = \frac{1}{K} \sum_{k=1}^{K} \frac{\|\mathbf{w}^k - \widehat{\mathbf{w}}^k\|_{L^2}}{\|\mathbf{w}^k\|_{L^2}}, \quad \|\mathbf{w}^k\|_{L^2} = \sum_{i=1}^{N} \mathbf{w}_i^k \Delta x \Delta y. \qquad (3.22)$$

To show the FOM-ROM error at a discrete time instance, we consider the node-wise difference between the FOM solution and ROM approximation.

The slow decay in the normalized singular values also cause a slow decay in FOM-ROM errors for all states with varying number of POD modes in Figure 3.6. Hence, the number of POD modes is set to a relatively big value, $r = 50$, according to the relative energy criteria (3.3) with $\kappa = 10^{-4}$.

In Figure 3.7, the FOM-ROM errors for height field and velocity fields are demonstrated, which indicates that height field error is slightly less accurate than the velocities error as in Figure 3.6.

In Figure 3.8, the discrete enstrophy error $|Z^k - Z^0|$, the discrete energy error $|H^k - H^0|$, and the discrete vorticity error $|V^k - V^0|$ are shown, $k = 1, \ldots, K$. The discrete

44
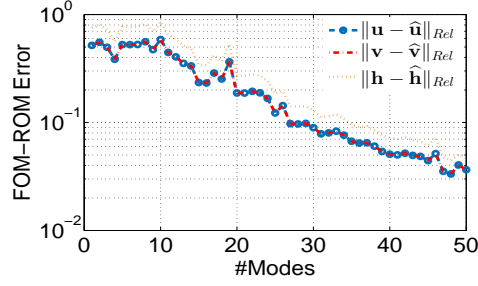
Figure 3.6: Time averaged relative L_2-errors vs. number of POD modes.
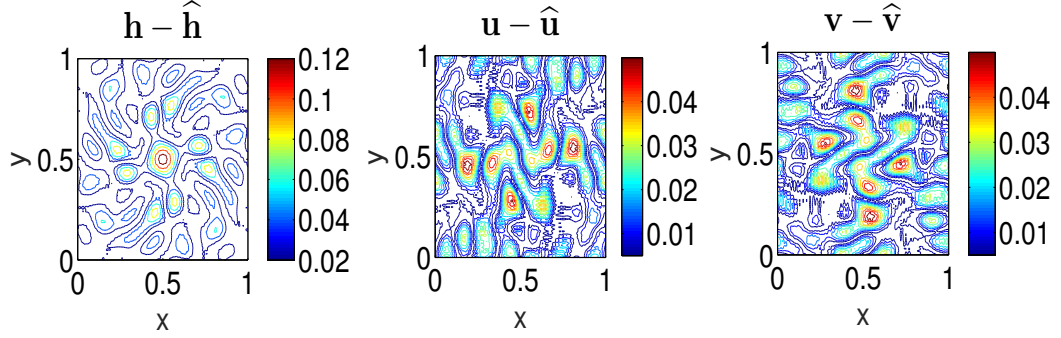
Figure 3.7: FOM-ROM errors at the final time $T = 50$.

enstrophy, energy, and vorticity are well preserved over the time interval $[0, 50]$. Since the vorticity is a quadratic quantity, it is preserved by both methods in Figure 3.8. The mass is conserved up to machine precision because it is a linear conserved quantity, and it is not shown here.

Table 3.1: Time averaged relative L^2-errors.

		$\|\mathbf{u} - \widehat{\mathbf{u}}\|_{Rel}$	$\|\mathbf{v} - \widehat{\mathbf{v}}\|_{Rel}$	$\|\mathbf{h} - \widehat{\mathbf{h}}\|_{Rel}$
Kahan	30 POD modes	1.265e-01	1.265e-01	1.567e-02

Table 3.2: Time averaged absolute errors for the conserved quantities.

		Energy	**Enstrophy**	**Vorticity**
Kahan	30 POD modes	2.901e-05	3.108e-03	3.454e-05
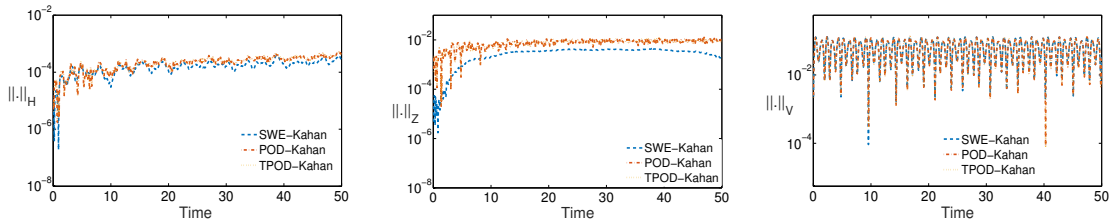
Figure 3.8: Errors in the conserved quatities. (left): energy error, (middle): enstrophy error and (right): vorticity error.

Table 3.3: Wall-clock time (in seconds) and speed-up factors.

		wall-clock time	speed-up
FOM		788.89	
POD	basis computation	31.02	
	online computation	289.01	
	total	321.03	2.45
TPOD	tensor computation [10](MP)	20.78 (9.43)	
	online computation	6.42	
	total [10](MP)	27.20 (15.89)	29.00 (49.64)

The time-averaged relative L_2-errors between full and reduced-order solutions are given in Table 3.1, which indicates the dynamics of the height fields is more accurately captured than for the velocity field components. The conserved quantities are accurately preserved by the TPOD-Kahan in Table 3.2.

The wall-clock times and corresponding speed-up factors in Table 3.3 for $r = 50$ POD, shows the computational efficiency obtained by the separation of the high-dimensional variables of the TPOD-Kahan. Table 3.3 presents the basis computation time, which is consist of SVD computation, and online computation time, which consists of the time required for projection and solution of ROM. The computational performance is further increased by utilizing the sparse matrix structure of the discretized RSWE using MULTIPROD (MP) in the algorithm of [12, 11] over [10] as shown in Table 3.3.

The computational time of the reduced matricized tensor $Q_{u,r} = -V_z^T A^x Q \left(V_u^* \otimes (B^x V_z)\right)$ is examined for the method in [10] (TS) is compared with the algorithms in [12, 11] utilising the MULTIPROD. In Figure 3.9, left, we present the required computational time versus the number of POD modes by fixing the number of grid points $N = 10000$. In contrast, the required computational time for the varying number of grid points using a fixed number of POD mode $r = 50$ is given in Figure 3.9, right. Both figures expose the computational efficiency using MULTIPROD by increasing the size of the FOM and ROMs.
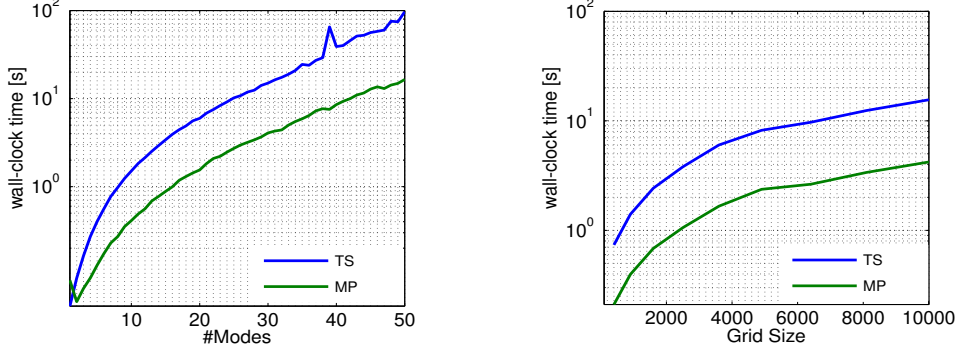
Figure 3.9: Tensor calculations by [10] (TS) and MULTIPROD (MP): (left) FOM is fixed with $N = 10000$, (right) ROM is fixed with $r = 50$.

3.5 Structure Preserving Reduced-Order Modelling

In this section, we derive ROMs that preserves the skew-gradient structure of NTSWE and RTSWE.

3.5.1 Nontraditional Rotating Shallow Water Equation

In this subsection, we construct ROM that maintaining the skew-gradient structure of the NTSWE (2.12) and, as a result, the Hamiltonian (2.13) of NTSWE. The NTSWE is a non-canonical Hamiltonian PDE with a state-dependent Poisson structure. Naive employment of the POD will not conserve the skew-gradient form of the NTSWE (2.12) in the reduced model. Energy preserving POD reduced models are formed for Hamiltonian systems with constant skew-symmetric matrices like the Korteweg de Vries equation [51, 84], and nonlinear Schrödinger equation (NLSE) [70]. It is possible to apply the approach in [51] to skew-gradient systems with state-dependent skew-symmetric structure as the NTSWE (2.12). We show the efficient evaluation of the state-dependent skew-gradient structure of (2.12) in the online stage, which is independent of the full dimension N.

We compute the POD basis from the mean subtracted snapshot matrices $S_{\tilde{u}}$, $S_{\tilde{v}}$, and S_h as done in Section 3.4. The snapshot matrices $S_{\tilde{u}}$, $S_{\tilde{v}}$, and S_h consist of the solutions of the fully discrete FOM (2.12) for each states $\tilde{u}$, $\tilde{v}$, and h, respectively.

After obtaining the POD basis V_i for the states $i = \tilde{u}, \tilde{v}, h$, the reduced-order approx-

47

imations becomes as follows

$$\tilde{\mathbf{u}} \approx \widehat{\tilde{\mathbf{u}}} = \overline{\tilde{\mathbf{u}}} + V_{\tilde{u}}\tilde{\mathbf{u}}_r, \quad \tilde{\mathbf{v}} \approx \widehat{\tilde{\mathbf{v}}} = \overline{\tilde{\mathbf{v}}} + V_{\tilde{v}}\tilde{\mathbf{u}}_r, \quad \mathbf{h} \approx \widehat{\mathbf{h}} = \overline{\mathbf{h}} + V_h\mathbf{h}_r, \tag{3.23}$$

where the reduced (coefficient) vectors $\tilde{\mathbf{u}}_r, \tilde{\mathbf{u}}_r, \mathbf{h}_r \in \mathbb{R}^r$ are the solutions of the following ROM of (2.12)

$$\frac{d}{dt}\tilde{\mathbf{z}}_r = V_z^T J(\widehat{\mathbf{z}})\nabla_{\mathbf{z}}H(\widehat{\mathbf{z}}), \tag{3.24}$$

where $\tilde{\mathbf{z}}_r = (\tilde{\mathbf{u}}_r, \tilde{\mathbf{v}}_r, \mathbf{h}_r)^T \in \mathbb{R}^{3r}$ consists of the reduced coefficients, $\overline{\tilde{\mathbf{u}}}, \overline{\tilde{\mathbf{v}}}, \overline{\mathbf{h}} \in \mathbb{R}^N$ denote the time averaged mean of the solutions and the components of the vector $\widehat{\mathbf{z}} = (\widehat{\tilde{\mathbf{u}}}, \widehat{\tilde{\mathbf{v}}}, \widehat{\mathbf{h}})$ are given as in (3.23). The block diagonal matrix V_z contains the matrix of POD modes for each state solution given by

$$V_z = \begin{pmatrix} V_{\tilde{u}} & & \\ & V_{\tilde{v}} & \\ & & V_h \end{pmatrix} \in \mathbb{R}^{3N \times 3r}.$$

Recently, symplectic MOR methods [1, 91] have been designed for canonical Hamiltonian PDEs with the constant skew-symmetric matrix

$$J_{2N} = \begin{pmatrix} 0_N & I_N \\ -I_N & 0_N \end{pmatrix} \in \mathbb{R}^{2N \times 2N},$$

where I_N and 0_N denote the identity and zero matrices, respectively. The skew-gradient structure of the FOM of NTSWE is not preserved in the ROM (3.24). However the approach in [1, 91] is not applicable to reduced system of NTSWE as demonstrated in [51], the reduced system (3.24) is not a skew-gradient system. A reduced-order skew-gradient system is constructed by adding $V_z V_z^T$ between $J(\widehat{\mathbf{z}})$ and $\nabla_{\mathbf{z}}H(\widehat{\mathbf{z}})$ [51], leading to the ROM

$$\frac{d}{dt}\tilde{\mathbf{z}}_r = J_r(\widehat{\mathbf{z}})\nabla_{\mathbf{z}_r}H(\widehat{\mathbf{z}}), \tag{3.25}$$

where $J_r(\widehat{\mathbf{z}}) = V_z^T J(\widehat{\mathbf{z}})V_z$ and $\nabla_{\mathbf{z}_r}H(\widehat{\mathbf{z}}) = V_z^T\nabla_{\mathbf{z}}H(\widehat{\mathbf{z}})$. The reduced-order NTSWE (3.25) is also solved by the AVF.

The reduced-order skew-gradient system (3.25) is not the same with the ROM (3.24) since $V_z V_z^T \neq I$ in general. Nevertheless, by increasing the reduced dimension, the vector fields of (3.24) and (3.25) become similar, and the solution to (3.24) well

approximates to solutions of the reduced skew-gradient system (3.25). An error analysis is given in [51]. Furthermore, the reduced model (3.24) differs from the reduced skew-gradient model (3.25) by conservation laws, i.e., for the reduced-order approximation $\widehat{z} = V_z z_r$ obtained by the reduced skew-gradient model (3.25), the Hamiltonian $H(\widehat{z})$ is constant as follows

$$
\begin{aligned}
\frac{d}{dt} H(V_z \boldsymbol{z}_r) &= [\nabla_{\mathbf{z}_r} H(V_z \boldsymbol{z}_r)]^T \frac{d}{dt} \boldsymbol{z}_r \\
&= -[\nabla_{\mathbf{z}_r} H(V_z \boldsymbol{z}_r)]^T \widetilde{J}_r(\widehat{\boldsymbol{z}}) \nabla_{\mathbf{z}_r} H(V_z \boldsymbol{z}_r) \\
&= -[V_z^T \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)]^T \widetilde{J}_r(\widehat{\boldsymbol{z}}) [V_z^T \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)] = 0,
\end{aligned}
$$

where we utilize the skew-symmetry of the reduced matrix $\widetilde{J}_r(\widehat{\boldsymbol{z}})$. Non-canonical Hamiltonian models with constant skew-symmetric matrices, like the KDV equations, are solved with the POD and DEIM methods following the approach in [51, 84], where the ROMs are integrated in time by the mid-point rule and AVF method.

The reduced NTSWE (3.25) can be written explicitly as

$$
\frac{d}{dt} \tilde{\mathbf{z}}_r = \begin{pmatrix} 0 & V_{\tilde{u}}^T \mathbf{q}^d V_{\tilde{v}} & -V_{\tilde{u}}^T D_x V_h \\ -V_{\tilde{v}}^T \mathbf{q}^d V_{\tilde{u}} & 0 & -V_{\tilde{v}}^T D_y V_h \\ -V_h^T D_x V_{\tilde{u}} & -V_h^T D_y V_{\tilde{v}} & 0 \end{pmatrix} V_z^T \nabla_{\mathbf{z}} H(\widehat{\mathbf{z}}). \tag{3.26}
$$

The reduced-order system (3.26) has constant matrices which can be precomputed in offline stage whereas the matrices $V_{\tilde{u}}^T \mathbf{q}^d V_{\tilde{v}}$ and $V_{\tilde{v}}^T \mathbf{q}^d V_{\tilde{u}}$ should be computed in online stage depending on the full order system. Utilizing the diagonal structure of $\mathbf{q}^d$ the computational complexity of evaluating the state dependent skew-symmetric matrix in (3.26) can be decreased similar to the skew-gradient systems with constant skew-symmetric matrices as discussed in [84]. Let vec$(\cdot)$ denotes vectorization of a matrix. For any $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{n \times p}$

$$
\text{vec}(AB) = (I_p \otimes A)\text{vec}(\mathbf{B}) = (B^\top \otimes I_m)\text{vec}(\mathbf{A}).
$$

Thus, for a diagonal matrix $D \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{n \times r}$

$$
\begin{aligned}
\text{vec}(V^\top D V) &= (I_r \otimes V^\top)\text{vec}(DV) \\
&= (I_r \otimes V^\top)(V^\top \otimes I_n)\text{vec}(D) \\
&= (V \otimes V)^\top \text{vec}(D) \\
&= (V \otimes V)^\top M^\top \tilde{D} \\
&= \begin{pmatrix} V(1,:) \otimes V(1,:) \\ \vdots \\ V(n,:) \otimes V(n,:) \end{pmatrix}^\top \tilde{D},
\end{aligned}
$$

where $M \in \mathbb{R}^{n \times n^2}$ is a matrix satisfying $M(a \otimes b) = a \circ b$ for any vector $a, b \in \mathbb{R}^n$, and $\tilde{D} = [D_{11}, D_{22}, \ldots, D_{nn}]^T \in \mathbb{R}^n$. Using the above result, the computational complexity of the matrix products $V_{\tilde{u}}^T \mathbf{q}^d V_{\tilde{v}}$ and $V_{\tilde{v}}^T \mathbf{q}^d V_{\tilde{u}}$ is reduced from $\mathcal{O}(r \cdot N(r + N))$ to $\mathcal{O}(r^2 \cdot N)$.

The nonlinear terms cause the computation of the reduced-order system to still scales with the dimension N of the FOM. We deal with this issue by employing the hyper-reduction technique DEIM given in Section (3.2). The ROM (3.25) can be rewritten as a nonlinear ODE system of the form

$$
\frac{d}{dt}\tilde{\mathbf{z}}_r = V_z^T \boldsymbol{f}(\tilde{\mathbf{z}}) = \begin{pmatrix} V_{\tilde{u}}^T \boldsymbol{f}_1(\tilde{\mathbf{z}}) \\ V_{\tilde{v}}^T \boldsymbol{f}_2(\tilde{\mathbf{z}}) \\ V_h^T \boldsymbol{f}_3(\tilde{\mathbf{z}}) \end{pmatrix}. \tag{3.27}
$$

In order to apply the DEIM basis, we construct the snapshot matrices defined by

$$
G_i = (\boldsymbol{f}_i^1, \boldsymbol{f}_i^2, \cdots, \boldsymbol{f}_i^K) \in \mathbb{R}^{N \times K}, \quad i = 1, 2, 3,
$$

where $\boldsymbol{f}_i^k = \boldsymbol{f}_i(\tilde{\mathbf{z}}^k)$ denotes the i-th component of the nonlinear term $\boldsymbol{f}(\tilde{\mathbf{z}})$ in (3.27) at time t_k computed by using the approximation of FOM solution vector $\tilde{\mathbf{z}}$, $k = 1, \ldots, K$. Hence, the nonlinear term $\boldsymbol{f}_i(\tilde{\mathbf{z}})$ can be approximated in the column space of the snapshot matrices G_i. First, we employ SVD to the nonlinear snapshot matrices G_i and find the basis matrices $\Phi_{\boldsymbol{f}_i, p} \in \mathbb{R}^{N \times p}$ whose columns are the basis vectors spanning the column space of the nonlinear snapshot matrices G_i. Next, we use the DEIM algorithm [29] to find a projection matrix $P_i \in \mathbb{R}^{N \times p}$

$$
\boldsymbol{f}_i(\tilde{\mathbf{z}}) \approx \Phi_{\boldsymbol{f}_i, p}(P_i^T \Phi_{\boldsymbol{f}_i, p})^{-1} P_i^T \boldsymbol{f}_i(\tilde{\mathbf{z}}),
$$

and then we use the DEIM approximation to the reduced nonlinear terms in (3.27) as

$$V_{\tilde{u}}^T \boldsymbol{f}_1(\tilde{\mathbf{z}}) \approx \mathcal{V}_{u,1}(P_1^T \boldsymbol{f}_1(\tilde{\mathbf{z}})), \quad V_{\tilde{v}}^T \boldsymbol{f}_2(\tilde{\mathbf{z}}) \approx \mathcal{V}_{v,2}(P_2^T \boldsymbol{f}_2(\tilde{\mathbf{z}})), \quad V_h^T \boldsymbol{f}_3(\tilde{\mathbf{z}}) \approx \mathcal{V}_{h,3}(P_3^T \boldsymbol{f}_3(\tilde{\mathbf{z}})),$$

where

$$\mathcal{V}_{u,1} = V_{\tilde{u}}^T V_{\boldsymbol{f}_1,p}(P_1^T V_{\boldsymbol{f}_1,p})^{-1}, \; \mathcal{V}_{v,2} = V_{\tilde{v}}^T V_{\boldsymbol{f}_2,p}(P_2^T V_{\boldsymbol{f}_2,p})^{-1}, \; \mathcal{V}_{h,3} = V_{h,n}^T V_{\boldsymbol{f}_3,p}(P_3^T V_{\boldsymbol{f}_3,p})^{-1}$$

are all the matrices of size $N \times p$, and they are precomputed in the offline stage. Using the DEIM approximations, the ROM (3.27) becomes

$$\frac{d}{dt}\tilde{\mathbf{z}}_r = \begin{pmatrix} \mathcal{V}_{u,1}\boldsymbol{f}_{r,1}(\tilde{\mathbf{z}}) \\ \mathcal{V}_{v,2}\boldsymbol{f}_{r,2}(\tilde{\mathbf{z}}) \\ \mathcal{V}_{h,3}\boldsymbol{f}_{r,3}(\tilde{\mathbf{z}}) \end{pmatrix}, \tag{3.28}$$

where the reduced nonlinearities $\boldsymbol{f}_{r,i}(\tilde{\mathbf{z}}) = P_i^T \boldsymbol{f}_i(\tilde{\mathbf{z}})$ are computed by considering just $p \ll N$ entries of the nonlinearities $\boldsymbol{f}_i(\tilde{\mathbf{z}})$ among N entries, $i = 1, 2, 3$. In addition, being an approximation to the right hand side of the ROM (3.27), the ROM (3.28) with DEIM approximately preserves the skew-gradient structure, but exactly at the interpolation points.

3.5.2 Numerical Results

In this subsection, we demonstrate two numerical experiments to show the efficiency of the ROMs. The first example covers the propagation of the inertia-gravity waves by Coriolis force, known as geostrophic adjustment [113]. The second example is based on the shear instability in the form of the roll-up of an unstable shear layer, known as barotropic instability [113]. For numerical experiments, we consider the non-dimensional structure of the NTSWE (2.12) with the setting

$$x = R_d \hat{x}, \quad y = R_d \hat{y}, \quad u = c\hat{u}, \quad v = c\hat{v}, \quad h = H\hat{h}, \quad h_b = H\hat{h}_b,$$

$$\left(\Omega^{(x)}, \Omega^{(y)}, \Omega^{(z)} \right) = \Omega \left(\hat{\Omega}^{(x)}, \hat{\Omega}^{(y)}, \hat{\Omega}^{(z)} \right),$$

where a variable with a hat indicates a dimensionless component, and Ω is planetary rotation rate to construct the gravity wave speed c

$$c = \sqrt{gH}, \quad R_d = \frac{c}{2\Omega}, \quad \sigma = \frac{H}{R_d} = \frac{2\Omega H}{c}.$$

The non-traditional parameter is given as $\sigma = H/R_d$, where R_d denotes Rossby deformation radius, H is the layer thickness scale, and g denotes the gravitational acceleration [41, 110]. The parameters are taken following [113] as $H = 1000$ m, $\Omega \approx 7.3 \times 10^{-5}$ rad s $^{-1}$, $g = 10^{-3}$ms^{-2}. The dimensionless variable of the rotation vector at latitude ϕ are taken as

$$\hat{\Omega}^{(x)} = 0, \quad \hat{\Omega}^{(y)} = \cos(\phi), \quad \hat{\Omega}^{(z)} = \sin(\phi),$$

where we set $\phi = \pi/4$ in the numerical experiments. In all the experiments, the spatial and temporal mesh sizes are taken as $\Delta x = 0.1$ and $\Delta t = 0.1$, respectively.

To determine the dimensions r and p of the POD and DEIM, respectively, we used the relative cumulative energy criteria (3.3). In the simulations, we fix $\kappa = 10^{-3}$ and $\kappa = 10^{-5}$ to catch at least 99.9% and 99.999% of the dynamics for the ROMs obtained with POD and DEIM, respectively. We consider the same number of modes for each state variable.

The error between a full solution and a reduced-order solution (FOM-ROM error) are obtained for the states $\mathbf{w} = \tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \mathbf{h}$ using the time averaged relative errors (3.22) in L^2-norm.

Example 1: *Single-layer geostrophic adjustment*

The NTSWE is considered on the periodic spatial domain $[-5, 5]^2$ and time interval $[0, 150]$ [113]. The initial conditions are prescribed in form of a motionless layer with an upward bulge of the height field

$$h(x, y, 0) = 1 + \frac{1}{2} \exp \left[- \left(\frac{4x}{5} \right)^2 - \left(\frac{4y}{5} \right)^2 \right],$$
$$u(x, y, 0) = 0,$$
$$v(x, y, 0) = 0.$$

The inertia-gravity waves diffuse after the collapse of the initial symmetric peak concerning axes. Nonlinear interactions create smaller waves diffusing around the domain, and increasingly more complicated patterns are occurred.

The snapshot matrix $S_{\tilde{u}}$, $S_{\tilde{v}}$, and S_h and the nonlinear snapshots has size 10000×1500. Using the energy criteria (3.3), we consider $r = 40$ POD modes and $m = 240$ DEIM

modes. In Fig. 3.10, the normalized singular values decay slowly, which is a key problem for PDEs with wave phenomena in fluid dynamics [88]. The slow decay of the normalized singular values also causes slow decay with small oscillations in FOM-ROM errors for all states with the increasing number of POD modes in Fig. 3.10.
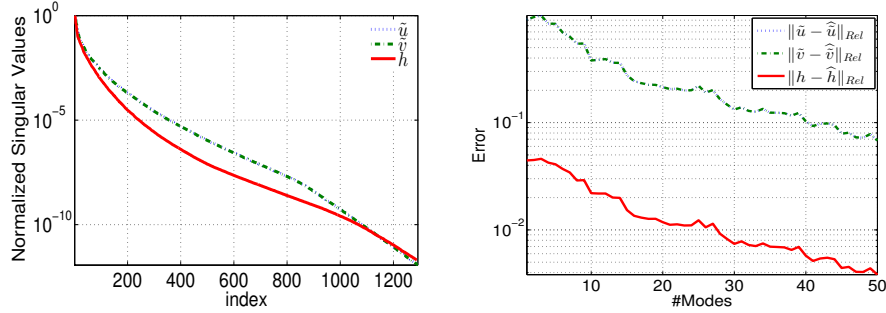
Figure 3.10: Normalized singular values for solution snapshots (left) and relative FOM-ROM errors (right).

The energy and enstrophy errors in Fig. 3.11 exhibit small drifts with bounded oscillations in time, i.e., they have conserved approximately at the same level of accuracy. The height **h** and the potential vorticity **q** are plotted at the final time, in Figs. 3.12-3.13. In [30], it is shown that a priori error bounds are proportional to the sums of the singular values corresponding to neglected POD basis vectors in the reduced-order system and the DEIM approximation of the nonlinear terms. A relatively large number of DEIM points are needed for convergence of the Newton method for solving the nonlinear fully discrete form of the reduced-order system. Figs. 3.12-3.13 and Tables 3.4-3.5 show that POD, POD-DEIM reduced solutions, approximately have the same level accuracy. The speed-up factors in Table 3.6 indicates that the ROM of NTSWE with DEIM increases the computational efficiency further.
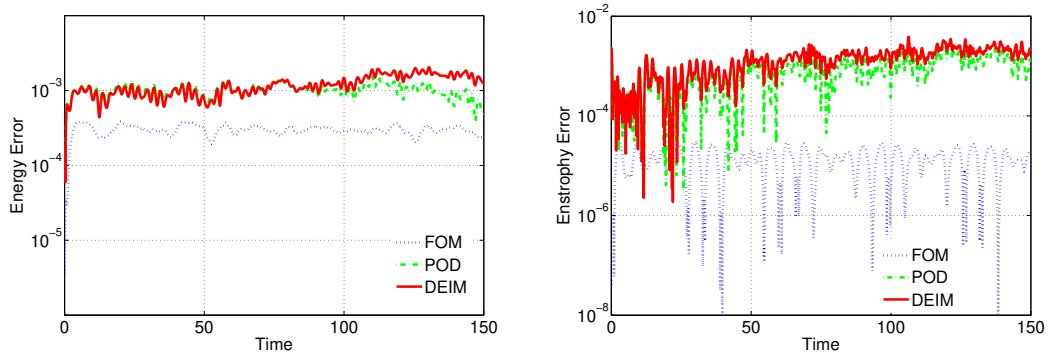
Figure 3.11: Energy error $|H^k - H^0|$ (left) and enstrophy error $|Z^k - Z^0|$ (right).
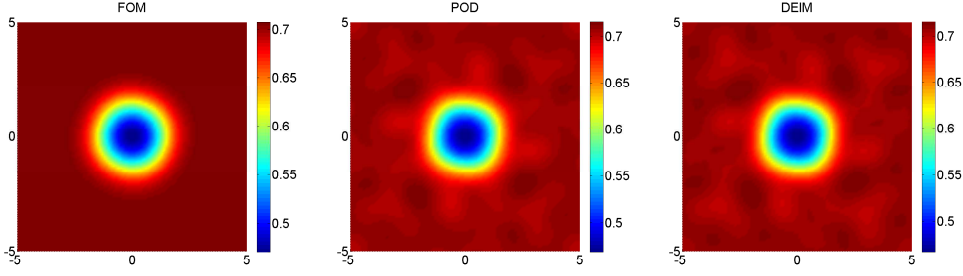
Figure 3.13: Full and reduced solutions for the potential vorticity **q** at the final time.
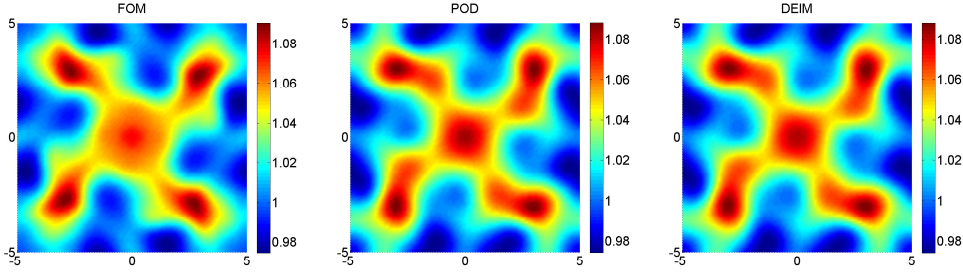
Figure 3.12: Full and reduced solutions for the height **h** at the final time.

Example 2: *Single-layer shear instability*

In the second example, the NTSWE is considered on the periodic spatial domain $[0, 10]^2$ and on the time domain $[0, 150]$ [113]. The initial conditions are as follows

$$h(x, y, 0) = 1 + \Delta_h \sin \left\{ \frac{2\pi}{L} \left[y - \Delta_y \sin \left(\frac{2\pi x}{L} \right) \right] \right\},$$

$$u(x, y, 0) = -\frac{2\pi \Delta_h}{\Omega^z L} \cos \left\{ \frac{2\pi}{L} \left[y - \Delta_y \sin \left(\frac{2\pi x}{L} \right) \right] \right\},$$

$$v(x, y, 0) = -\frac{4\pi^2 \Delta_h \Delta_y}{\Omega^z L^2} \cos \left\{ \frac{2\pi}{L} \left[y - \Delta_y \sin \left(\frac{2\pi x}{L} \right) \right] \right\} \cos \left(\frac{2\pi x}{L} \right),$$

where $\Delta_h = 0.2$, $\Delta_y = 0.5$ and the dimensionless spatial domain length $L = 10$, as the case in the first test example.

The singular values decay and corresponding FOM-ROM errors in Fig. 3.14 are similar to the first test example in Fig. 3.10.
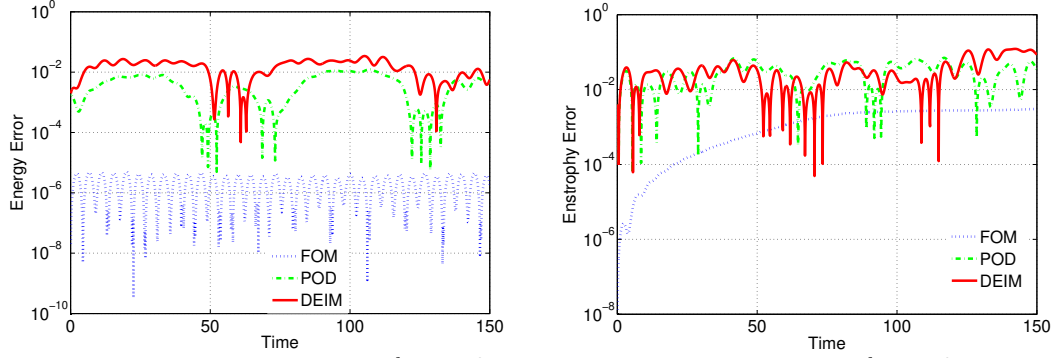
Figure 3.15: Energy error $|H^k - H^0|$ (left) and enstrophy error $|Z^k - Z^0|$ (right).
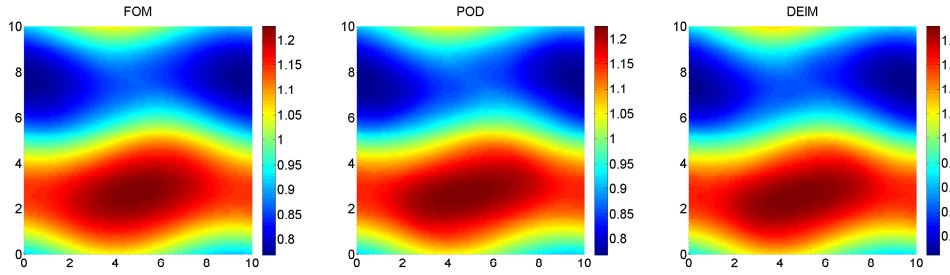
Figure 3.16: Full and reduced solutions for the height **h** at the final time.
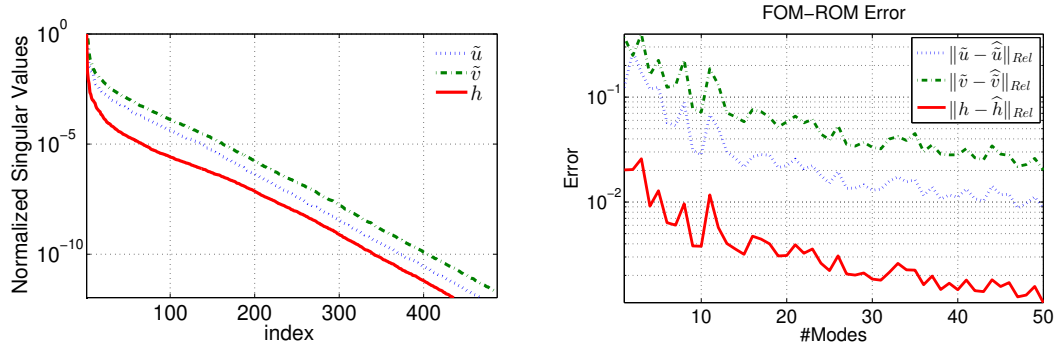
Figure 3.14: Normalized singular values for solution snapshots (left) and relative FOM-ROM errors (right).

As in previous example, each snapshot matrix has size 10000×1500. We set the number of POD and DEIM modes as $r = 10$ and $p = 345$, respectively, by using the energy criteria (3.3).

Fig. 3.15 shows that the energy and entropy errors are bounded over time with small oscillations, similar to the first example. Furthermore, the height field **h** and the potential vorticity field **q** are approximated properly by the ROMs at the final time, in Figs. 3.16-3.17. In Tables 3.4-3.5 and Table 3.6, the computational efficiency and accuracy of the ROMs are shown.
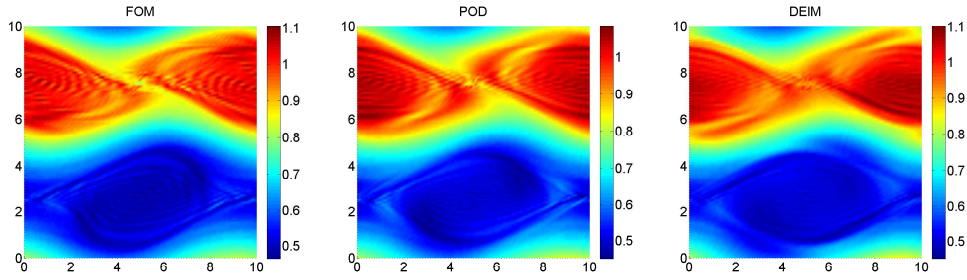
Figure 3.17: Full and reduced solutions for the potential vorticity **q** at the final time.

Table 3.4: Time averaged relative L^2-errors.

		$\|\tilde{\mathbf{u}} - \widehat{\tilde{\mathbf{u}}}\|_{Rel}$	$\|\tilde{\mathbf{v}} - \widehat{\tilde{\mathbf{v}}}\|_{Rel}$	$\|\mathbf{h} - \widehat{\mathbf{h}}\|_{Rel}$
Example 1	40 POD modes	1.151e-01	1.151e-01	6.172e-03
	240 DEIM modes	1.156e-01	1.156e-01	6.180e-03
Example 2	10 POD modes	3.946e-02	9.088e-02	4.224e-03
	345 DEIM modes	4.859e-02	1.011e-01	5.464e-03

Table 3.6: Wall-clock time (in seconds) and speed-up factors.

		Example 1		**Example 2**	
		wall-clock time	speed-up	wall-clock time	speed-up
FOM		1051.0		1038.1	
POD	basis computation	61.6		23.2	
	online computation	412.7	2.55	167.4	6.2
DEIM	basis computation	67.4		31.1	
	online computation	87.1	12.1	47.3	22.0

Table 3.5: Mean absolute FOM-ROM errors of the conserved quantities.

		Energy	**Enstrophy**
Example 1	40 POD modes	7.094e-04	9.067e-04
	240 DEIM modes	8.837e-04	1.370e-03
Example 2	10 POD modes	4.450e-03	3.068e-02
	345 DEIM modes	1.529e-02	3.589e-02

56

3.5.3 Rotating Thermal Shallow Water Equation

In this section, the skew-gradient structure of the semi-discrete RTSWE (2.26) is preserved in ROMs of RTSWE. Thus, the discrete quantities (2.27) are also preserved in the ROMs. The RTSWE is a non-canonical Hamiltonian PDE with a state-dependent Poisson structure. Direct employment of the POD will not conserve the skew-symmetric Poisson matrix in the reduced model so that the skew-gradient structure of the RTSWE (2.26) in the reduced form will not be preserved. The structure of the FOM is maintained by utilising the approach in [51] to the RTSWE (2.26). We demonstrate that the resulted ROMs conserves the skew-gradient structure of the FOM of RTSWE. The ROMs in time are integrated with the AVF method as for the FOM. Due to the approximation of the nonlinear terms in using DEIM, the discrete energy (Hamiltonian) of the FOM is preserved approximately. We have derived an upper bound for the conservation of the discrete energy by POD-DEIM.

The POD basis are constructed as in Section 3.4, where we again consider the mean subtracted snapshot matrices S_u, S_v, S_h, and S_s, which consist of the solutions of the fully discrete RTSWE (2.26) for the states $\mathbf{u}$, $\mathbf{v}$, $\mathbf{h}$, and $\mathbf{s}$, respectively.

Furthermore, we have the reduced-order approximations are as follows

$$\boldsymbol{u} \approx \widehat{\boldsymbol{u}} = \overline{\boldsymbol{u}} + V_u \boldsymbol{u}_r, \quad \boldsymbol{v} \approx \widehat{\boldsymbol{v}} = \overline{\boldsymbol{v}} + V_v \boldsymbol{v}_r, \quad \boldsymbol{h} \approx \widehat{\boldsymbol{h}} = \overline{\boldsymbol{h}} + V_h \boldsymbol{h}_r, \quad \boldsymbol{s} \approx \widehat{\boldsymbol{s}} = \overline{\boldsymbol{s}} + V_s \boldsymbol{s}_r, \quad (3.29)$$

where $\overline{\boldsymbol{u}}, \overline{\boldsymbol{v}}, \overline{\boldsymbol{h}}, \overline{\boldsymbol{s}}$ are time-averaged snapshots, and the vectors $\boldsymbol{u}_r, \boldsymbol{v}_r, \boldsymbol{h}_r$, and $\boldsymbol{s}_r$ are the ROM solutions which are the coefficient vectors for the reduced approximations $\widehat{\boldsymbol{u}}, \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{h}}$, and $\widehat{\boldsymbol{s}}$. The ROM is determined by the Galerkin projection onto the reduced space

$$\frac{d}{dt}\boldsymbol{z}_r = -V_z^T J(\widehat{\boldsymbol{z}}) \nabla_{\boldsymbol{z}} H(\widehat{\boldsymbol{z}}), \qquad (3.30)$$

where the ROM solution is $\boldsymbol{z}_r = (\boldsymbol{h}_r, \boldsymbol{u}_r, \boldsymbol{v}_r, \boldsymbol{s}_r) : [0, T] \mapsto \mathbb{R}^{r_h + r_u + r_v + r_s}$ and the related reduced approximation is $\widehat{\boldsymbol{z}} = (\widehat{\boldsymbol{h}}, \widehat{\boldsymbol{u}}, \widehat{\boldsymbol{v}}, \widehat{\boldsymbol{s}})$. The block diagonal matrix V_z contains the matrix of POD modes for each state variable

$$V_z = \begin{pmatrix} V_h & & & \\ & V_u & & \\ & & V_v & \\ & & & V_s \end{pmatrix} \in \mathbb{R}^{4N \times (r_h + r_u + r_v + r_s)}.$$

Although the formulae in this section are generally applicable for any number of modes r_h, r_u, r_v, and r_s, we use equal number of POD modes in the present study $r := r_h = r_u = r_v = r_s$.

Skew-gradient POD reduced system

We preserve the skew-gradient system in ROM of RTSWE by adding $V_z V_z^T \in \mathbb{R}^{r \times r}$ between $J(\widehat{z})$ and $\nabla_z H(\widehat{z})$ as in structure- preserving ROM of NTSWE 3.25, leading to

$$\frac{d}{dt}z_r = -J_r(\widehat{z})\nabla_{\mathbf{z}_r}H(\widehat{z}), \tag{3.31}$$

where $J_r(\widehat{z}) = V_z^T J(\widehat{z})V_z \in \mathbb{R}^{4r \times 4r}$ is the reduced skew-symmetric matrix and $\nabla_{\mathbf{z}_r}H(\widehat{z}) = V_z^T\nabla_{\mathbf{z}}H(\widehat{z}) \in \mathbb{R}^{4r}$ is the reduced discrete gradient of the Hamiltonian.

The main obstacle in constructing ROMs of RTSWE is the nonlinear state dependency in the skew-symmetric matrix of the semi-discretized RTSWE (2.26), which does not allow the offline-online separation in the ROMs. The skew-symmetric matrix $J_r(\widehat{z})$ in the reduced model of RTSWE (3.31) can be written as follows

$$J_r(\widehat{z}) = \begin{pmatrix} 0 & V_h^T D_x V_u & V_h^T D_y V_v & 0 \\ V_u^T D_x V_h & 0 & -V_u^T F_1(\widehat{z})V_v & -V_u^T F_2(\widehat{z})V_s \\ V_v^T D_y V_h & V_v^T F_1(\widehat{z})V_u & 0 & -V_v^T F_3(\widehat{z})V_s \\ 0 & V_s^T F_2(\widehat{z})V_u & V_s^T F_3(\widehat{z})V_v & 0, \end{pmatrix}, \tag{3.32}$$

where for the nonlinear vectors

$$\boldsymbol{f}_1(\widehat{z}) = \widehat{\boldsymbol{q}}, \quad \boldsymbol{f}_2(\widehat{z}) = \widehat{\boldsymbol{h}}^{-1} \circ (D_x \widehat{\boldsymbol{s}}), \quad \boldsymbol{f}_3(\widehat{z}) = \widehat{\boldsymbol{h}}^{-1} \circ (D_y \widehat{\boldsymbol{s}}),$$

the matrices $F_j(\widehat{z})$ refer to the diagonal matrices with the diagonal elements from the vectors $\boldsymbol{f}_j(\widehat{z})$, i.e., $(F_j(\widehat{z}))_{ii} = (\boldsymbol{f}_j(\widehat{z}))_i$, $j = 1, 2, 3$, $i = 1, \ldots, N$.

The semi-discrete equations of structure-preserving reduced model (3.31) can be written explicitly as

$$\begin{aligned}
\frac{d}{dt}\boldsymbol{h}_r &= -V_h^T D_x V_u V_u^T \boldsymbol{f}_5(\widehat{z}) - V_h^T D_y V_v V_v^T \boldsymbol{f}_6(\widehat{z}), \\
\frac{d}{dt}\boldsymbol{u}_r &= -V_u^T D_x V_h V_h^T \boldsymbol{f}_4(\widehat{z}) + V_u^T(\boldsymbol{f}_1(\widehat{z}) \circ V_v V_v^T \boldsymbol{f}_6(\widehat{z})) + V_u^T(\boldsymbol{f}_2(\widehat{z}) \circ V_s V_s^T \boldsymbol{f}_7(\widehat{z})), \\
\frac{d}{dt}\boldsymbol{v}_r &= -V_v^T D_y V_h V_h^T \boldsymbol{f}_4(\widehat{z}) - V_v^T(\boldsymbol{f}_1(\widehat{z}) \circ V_u V_u^T \boldsymbol{f}_5(\widehat{z})) + V_v^T(\boldsymbol{f}_3(\widehat{z}) \circ V_s V_s^T \boldsymbol{f}_7(\widehat{z})), \\
\frac{d}{dt}\boldsymbol{s}_r &= -V_s^T(\boldsymbol{f}_2(\widehat{z}) \circ V_u V_u^T \boldsymbol{f}_5(\widehat{z})) - V_s^T(\boldsymbol{f}_3(\widehat{z}) \circ V_v V_v^T \boldsymbol{f}_6(\widehat{z})),
\end{aligned} \tag{3.33}$$

where the nonlinear vectors $\boldsymbol{f}_j(\widehat{\boldsymbol{z}})$, $j = 4, 5, 6, 7$ are the components of the discrete gradient $\nabla_{\mathbf{z}} H(\widehat{\boldsymbol{z}})$ of Hamiltonian, which are given by

$$\boldsymbol{f}_4(\widehat{\boldsymbol{z}}) = \frac{\widehat{\boldsymbol{u}}^2 + \widehat{\boldsymbol{v}}^2}{2} + \widehat{\boldsymbol{s}} \circ \widehat{\boldsymbol{h}} + b\widehat{\boldsymbol{s}}, \ \ \boldsymbol{f}_5(\widehat{\boldsymbol{z}}) = \widehat{\boldsymbol{h}} \circ \widehat{\boldsymbol{u}}, \ \ \boldsymbol{f}_6(\widehat{\boldsymbol{z}}) = \widehat{\boldsymbol{h}} \circ \widehat{\boldsymbol{v}}, \ \ \boldsymbol{f}_7(\widehat{\boldsymbol{z}}) = \frac{\widehat{\boldsymbol{h}}^2}{2} + b\widehat{\boldsymbol{h}}.$$

Skew-gradient POD-DEIM reduced system

Due to the nonlinearities in the skew-symmetric Possion matrix, the computation of the POD-ROM (3.33) still scales with the FOM dimension N. Precisely, the nonlinear vectors $\boldsymbol{f}_j(\widehat{\boldsymbol{z}})$, $j = 1, 2, 3$, in (3.32) do not possess polynomial nonlinearities. Thus, they should be computed in the online stage depending on the FOM dimension. To conserve the structure in (3.31) in a efficient way, we approximate the nonlinearities individually inside the skew-symmetric reduced Poisson matrix $J_r(\widehat{\boldsymbol{z}})$ and the discrete reduced gradient $\nabla_{\mathbf{z}_r} H(\widehat{\boldsymbol{z}})$ of the Hamiltonian with DEIM. Using DEIM, the nonlinear vectors $\boldsymbol{f}_j(\widehat{\boldsymbol{z}})$ in (3.33) are approximated by

$$\boldsymbol{f}_j(\widehat{\boldsymbol{z}}(t)) \approx \Psi_j \boldsymbol{f}_{r,j}(\widehat{\boldsymbol{z}}(t)), \quad j = 1, \ldots, 7, \tag{3.34}$$

where $\Psi_j := \Phi^{(j)}((\mathsf{P}^{(j)})^\top \Phi^{(j)})^{-1} \in \mathbb{R}^{N \times p_j}$ are the precomputed matrices, $\mathsf{P}^{(j)}$ are the selection matrices, and $\Phi^{(j)}$ are the DEIM modes.

To determine the POD modes, we use following snapshot matrices of the nonlinear vectors given by

$$\mathsf{S}_{f_j} = [\boldsymbol{f}_j^1, \boldsymbol{f}_j^2, \ldots, \boldsymbol{f}_j^K] \in \mathbb{R}^{N \times K}, \quad j = 1, \ldots, 7, \tag{3.35}$$

where $\boldsymbol{f}_j^k = \boldsymbol{f}_j(\widehat{\boldsymbol{z}}^k)$ denote the nonlinear vectors $\boldsymbol{f}_j(\widehat{\boldsymbol{z}}(t))$ at time t_k, computed by using the solution vectors $\widehat{\boldsymbol{z}}^k = \widehat{\boldsymbol{z}}(t_k)$, $k = 1, \ldots, K$. Similar to the POD, we take the same number of DEIM modes for the nonlinear terms, i.e., $p := p_j, j = 1, \ldots 7$.

Applying the DEIM/Q-DEIM approximation (3.34), the POD reduced system (3.31) takes the form

$$\frac{d}{dt} \boldsymbol{z}_r = -\widetilde{J}_r(\widehat{\boldsymbol{z}}) \nabla_{\mathbf{z}_r} \widetilde{H}(\widehat{\boldsymbol{z}}), \tag{3.36}$$

where the skew-gradient structure of the reduced system is preserved. In the POD-DEIM reduced system (3.36), the terms $\nabla_{\mathbf{z}_r} \widetilde{H}(\widehat{\boldsymbol{z}})$ and $\widetilde{J}_r(\widehat{\boldsymbol{z}})$ are the DEIM reduced

discrete gradient of Hamiltonian and the DEIM reduced skew-symmetric Poisson matrix, respectively, given by

$$
\nabla_{\mathbf{z}_r}\widetilde{H}(\widehat{\mathbf{z}}) = V_z^T\nabla_{\mathbf{z}}\widetilde{H}(\widehat{\mathbf{z}}) =
\begin{pmatrix}
V_h^T & & & \\
& V_u^T & & \\
& & V_v^T & \\
& & & V_s^T
\end{pmatrix}
\begin{pmatrix}
\Psi_4\boldsymbol{f}_{r,4}(\widehat{\mathbf{z}}) \\
\Psi_5\boldsymbol{f}_{r,5}(\widehat{\mathbf{z}}) \\
\Psi_6\boldsymbol{f}_{r,6}(\widehat{\mathbf{z}}) \\
\Psi_7\boldsymbol{f}_{r,7}(\widehat{\mathbf{z}})
\end{pmatrix},
$$

$$
\widetilde{J}_r(\widehat{\mathbf{z}}) =
\begin{pmatrix}
0 & V_h^T D_x V_u & V_h^T D_y V_v & 0 \\
V_u^T D_x V_h & 0 & -V_u^T F_{r,1}(\widehat{\mathbf{z}})V_v & -V_u^T F_{r,2}(\widehat{\mathbf{z}})V_s \\
V_v^T D_y V_h & V_v^T F_{r,1}(\widehat{\mathbf{z}})V_u & 0 & -V_v^T F_{r,3}(\widehat{\mathbf{z}})V_s \\
0 & V_s^T F_{r,2}(\widehat{\mathbf{z}})V_u & V_s^T F_{r,3}(\widehat{\mathbf{z}})V_v & 0
\end{pmatrix},
$$

where $F_{r,j}(\widehat{\mathbf{z}})$ are the diagonal matrices of the related DEIM approximated nonlinear vectors, i.e., $(F_{r,j}(\widehat{\mathbf{z}}))_{ii} = (\Psi_j\boldsymbol{f}_{r,j}(\widehat{\mathbf{z}}))_i$, $j = 1, 2, 3$, $i = 1, \ldots, N$.

The skew-gradient structure of the FOM is maintained by both the POD-DEIM reduced model (3.36) and the POD reduced model (3.31). Nevertheless, the discrete Hamiltonian $H(\widehat{\mathbf{z}})$ is not preserved for the reduced-order approximation $\widehat{\mathbf{z}} = V_z\mathbf{z}_r$ obtained by the POD-DEIM reduced model (3.36) since the discrete gradient $\nabla_{\mathbf{z}}H(\widehat{\mathbf{z}})$ of the Hamiltonian in the reduced discrete gradient $\nabla_{\mathbf{z}_r}H(\widehat{\mathbf{z}}) = V_z^T\nabla_{\mathbf{z}}H(\widehat{\mathbf{z}})$ is replaced by the DEIM approximation $\nabla_{\mathbf{z}}\widetilde{H}(\widehat{\mathbf{z}})$. To see this, for the reduced approximation $\widehat{\mathbf{z}} = V_z\mathbf{z}_r$ obtained by the POD-DEIM reduced system (3.36), we have for the discrete Hamiltonian $H(\widehat{\mathbf{z}}) = H(V_z\mathbf{z}_r)$

$$
\begin{aligned}
\frac{d}{dt}H(V_z\mathbf{z}_r) &= \left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \frac{d}{dt}\mathbf{z}_r \\
&= -\left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \widetilde{J}_r(\widehat{\mathbf{z}})\nabla_{\mathbf{z}_r}\widetilde{H}(V_z\mathbf{z}_r) \\
&= -\left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \widetilde{J}_r(\widehat{\mathbf{z}})\left[V_z^T\nabla_{\mathbf{z}}\widetilde{H}(V_z\mathbf{z}_r)\right] \neq 0.
\end{aligned}
\tag{3.37}
$$

The Hamiltonian of the ROM is preserved approximately due to the nonlinear terms in the gradient of the Hamiltonian of ROM is approximated by the DEIM. An upper bound for the preservation of the discrete Hamiltonian by POD-DEIM method can be derived as follows: starting from the last row of (3.37), we can write

$$
\begin{aligned}
\frac{d}{dt}H(V_z\mathbf{z}_r) &= -\left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \widetilde{J}_r(\widehat{\mathbf{z}})\left[V_z^T\nabla_{\mathbf{z}}\widetilde{H}(V_z\mathbf{z}_r)\right] \\
&= -\left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \widetilde{J}_r(\widehat{\mathbf{z}})\left[V_z^T\nabla_{\mathbf{z}}\widetilde{H}(V_z\mathbf{z}_r) - V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right] \\
&= -\left[V_z^T\nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right]^T \widetilde{J}_r(\widehat{\mathbf{z}})V_z^T\left[\nabla_{\mathbf{z}}\widetilde{H}(V_z\mathbf{z}_r) - \nabla_{\mathbf{z}}H(V_z\mathbf{z}_r)\right],
\end{aligned}
\tag{3.38}
$$

60

where we use the fact that the reduced matrix $\widetilde{J}_r(\widehat{\boldsymbol{z}})$ is skew-symmetric and adding the term $-V_z^T \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)$ in the second row does not change the equality since we have that $\left[V_z^T \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)\right]^T \widetilde{J}_r(\widehat{\boldsymbol{z}}) \left[V_z^T \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)\right] = 0$ for the skew-symmetric matrix $\widetilde{J}_r(\widehat{\boldsymbol{z}})$. Using the equality in (3.38), we can bound the time derivative of the discrete Hamiltonian. Taking the norm of the last row in the equality (3.38), we can easily obtain the inequality

$$\left\|\frac{d}{dt} H(V_z \boldsymbol{z}_r)\right\| \leq \|\nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)\| \|V_z\|^2 \|\widetilde{J}_r(\widehat{\boldsymbol{z}})\| \|\nabla_{\mathbf{z}} \widetilde{H}(V_z \boldsymbol{z}_r) - \nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)\|, \quad (3.39)$$

where $\| \cdot \| := \| \cdot \|_2$ denotes the 2-norm. In terms of the nonlinear components $\boldsymbol{f}_j(V_z \boldsymbol{z}_r)$, $j = 4, 5, 6, 7$, and corresponding DEIM approximated nonlinear components in (3.34), of the discrete gradients $\nabla_{\mathbf{z}} H(V_z \boldsymbol{z}_r)$ and $\nabla_{\mathbf{z}} \widetilde{H}(V_z \boldsymbol{z}_r)$, respectively, the inequality (3.39) becomes

$$\left\|\frac{d}{dt} H(V_z \boldsymbol{z}_r)\right\| \leq \|V_z\|^2 \|\widetilde{J}_r(\widehat{\boldsymbol{z}})\| \sum_{j=4}^{7} \|\boldsymbol{f}_j(V_z \boldsymbol{z}_r)\| \|\boldsymbol{f}_j(V_z \boldsymbol{z}_r) - \Psi_j \boldsymbol{f}_{r,j}(V_z \boldsymbol{z}_r)\|. \quad (3.40)$$

Using the DEIM approximation error [29, Lemma 3.2] on the last term, we have for $j = 4, 5, 6, 7$

$$\|\boldsymbol{f}_j(V_z \boldsymbol{z}_r) - \Psi_j \boldsymbol{f}_{r,j}(V_z \boldsymbol{z}_r)\| \leq \|((\mathsf{P}^{(j)})^\top \Phi^{(j)})^{-1}\| \|(I_N - \Phi^{(j)}(\Phi^{(j)})^\top)\boldsymbol{f}_j(V_z \boldsymbol{z}_r)\|. \quad (3.41)$$

Finally, imposing the bounds (3.41) in the inequality (3.40), we obtain the following bound on the time derivative of the discrete Hamiltonian

$$\left\|\frac{d}{dt} H(V_z \boldsymbol{z}_r)\right\| \leq \|V_z\|^2 \|\widetilde{J}_r(\widehat{\boldsymbol{z}})\| \sum_{j=4}^{7} \|\boldsymbol{f}_j(V_z \boldsymbol{z}_r)\| \|((\mathsf{P}^{(j)})^\top \Phi^{(j)})^{-1}\| \|(I_N - \Phi^{(j)}(\Phi^{(j)})^\top)\boldsymbol{f}_j(V_z \boldsymbol{z}_r)\|.$$

We remark that the reduced model obtained by the POD (3.31) and the POD-DEIM (3.36) has a skew-gradient structure with the reduced-order skew-symmetric matrix. Here we use the energy preserving AVF integrator to the skew-gradient FOM (2.26) and as well as to the POD (3.31) and POD-DEIM (3.36) reduced skew-gradient systems, which is co-adjoint to a Poisson integrator and preserves the linear and quadratic Casimirs [33].

Fast solution of the reduced system with tensor techniques

To accelerate the online computations of the ROM (3.36) further, we employ the tensor techniques. The POD-DEIM reduced model (3.36) can be written explicitly as

the following system of ODEs in tensor form

$$
\begin{aligned}
\frac{d}{dt}\boldsymbol{h}_r &= -V_h^T D_x V_u V_u^T \Psi_5 \boldsymbol{f}_{r,5}(\widehat{\boldsymbol{z}}) - V_h^T D_y V_v V_v^T \Psi_6 \boldsymbol{f}_{r,6}(\widehat{\boldsymbol{z}}), \\
\frac{d}{dt}\boldsymbol{u}_r &= -V_u^T D_x V_h V_h^T \Psi_4 \boldsymbol{f}_{r,4}(\widehat{\boldsymbol{z}}) + G_1(\boldsymbol{f}_{r,1}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,6}(\widehat{\boldsymbol{z}})) + G_2(\boldsymbol{f}_{r,2}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,7}(\widehat{\boldsymbol{z}})), \\
\frac{d}{dt}\boldsymbol{v}_r &= -V_v^T D_y V_h V_h^T \Psi_4 \boldsymbol{f}_{r,4}(\widehat{\boldsymbol{z}}) - G_3(\boldsymbol{f}_{r,1}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,5}(\widehat{\boldsymbol{z}})) + G_4(\boldsymbol{f}_{r,3}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,7}(\widehat{\boldsymbol{z}})), \\
\frac{d}{dt}\boldsymbol{s}_r &= -G_5(\boldsymbol{f}_{r,2}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,5}(\widehat{\boldsymbol{z}})) - G_6(\boldsymbol{f}_{r,3}(\widehat{\boldsymbol{z}}) \otimes \boldsymbol{f}_{r,6}(\widehat{\boldsymbol{z}})),
\end{aligned}
\tag{3.42}
$$

where only the reduced nonlinear vectors $\boldsymbol{f}_{r,j}(\widehat{\boldsymbol{z}})$ are non-constant, $j = 1, \ldots, 7$. In (3.42), the small matrices $G_j \in \mathbb{R}^{r \times p^2}$, called reduced matricized tensors, are defined as

$$
\begin{aligned}
G_1 &= V_u^T G(\Psi_1 \otimes V_v V_v^T \Psi_6), \quad G_2 = V_u^T G(\Psi_2 \otimes V_s V_s^T \Psi_7), \\
G_3 &= V_v^T G(\Psi_1 \otimes V_u V_u^T \Psi_5), \quad G_4 = V_v^T G(\Psi_3 \otimes V_s V_s^T \Psi_7), \\
G_5 &= V_s^T G(\Psi_2 \otimes V_u V_u^T \Psi_5), \quad G_6 = V_s^T G(\Psi_3 \otimes V_v V_v^T \Psi_6),
\end{aligned}
$$

where the matrices $G \in \mathbb{R}^{N \times N^2}$ satisfy the identity $G(\boldsymbol{a} \otimes \boldsymbol{b}) = \boldsymbol{a} \circ \boldsymbol{b}$ for any vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^N$. Typically, the matrix G is called matricized tensor since the computation of the matrix product of G by Kronecker products like $G(\Psi_1 \otimes V_v V_v^T \Psi_6)$ are handled in the literature through converting the matrix G into a 3-tensor and using tensor algebra on it [10, 11, 12, 69].

All the reduced matricized tensors G_j can be precomputed in the offline stage, and the DEIM reduced nonlinear terms $\boldsymbol{f}_{r,j}(\widehat{\boldsymbol{z}}) = (\mathsf{P}^{(j)})^T \boldsymbol{f}_j(\widehat{\boldsymbol{z}})$ are computed by considering just $p \ll N$ entries of the nonlinear terms $\boldsymbol{f}_j(\widehat{\boldsymbol{z}})$ among N entries, $j = 1, \ldots, 7$. Thus, in the online stage, the cost of ROM (3.42) scales with $\mathcal{O}(rp^2)$, i.e., only depends on the reduced dimensions. We remark that it is also possible to obtain a structure-preserving ROM by using DEIM applied only to the nonlinear terms in the reduced Poisson matrix $J_r(\widehat{\boldsymbol{z}})$ and not to the reduced gradient of the Hamiltonian. Nevertheless, in this case, the online computational cost increases rapidly due to the cubic nonlinear terms.

Beside the computational efficiency in the online phase, a straightforward computation of the reduced matricized tensors G_j in the POD-DEIM reduced system (3.42) creates a big burden on the offline cost. Thus, we use the same procedure computing the matricized tensors G_j as in (3.19).

We remark that Poisson structure or energy preserving reduced models are constructed

for non-canonical Hamiltonian systems, such as for the modified KDV equation [84]. But tensor techniques are not exploited with the POD and DEIM. In [84], the Poisson matrix is approximated by DEIM but not the gradient of the Hamiltonian, which is linear in the state variable for the modified KdV equation. Our approach approximating both the Poisson matrix and the gradient of the Hamiltonian with the DEIM, can be applied to any non-canonical Hamiltonian system, that leads to a skew-gradient reduced system. Application of the tensor techniques to the reduced systems, speeds up the computation of the ROMs further.

3.5.4 Numerical Results

In this subsection, we illustrate the efficiency and accuracy of the ROMs of RTSWE for the double vortex test case [43]. The spatial domain is set to $\Omega = [0, L]^2$ with doubly periodic boundary conditions and no bottom topography ($b = 0$). The initial conditions are given by

$$h = H_0 - \Delta h \left[e^{-0.5((x_1')^2+(y_1')^2)} + e^{-0.5((x_2')^2+(y_2')^2)} - \frac{4\pi\sigma_x\sigma_y}{L^2} \right],$$

$$u = \frac{-g\Delta h}{f\sigma_y} \left[y_1'' e^{-0.5((x_1')^2+(y_1')^2)} + y_2'' e^{-0.5((x_2')^2+(y_2')^2)} \right],$$

$$v = \frac{g\Delta h}{f\sigma_x} \left[x_1'' e^{-0.5((x_1')^2+(y_1')^2)} + x_2'' e^{-0.5((x_2')^2+(y_2')^2)} \right],$$

$$s = g \left(1 + 0.05 sin \left[\frac{2\pi}{L}(x - xc) \right] \right),$$

where $xc = 0.5L$ and

$$x_1' = \frac{L}{\pi\sigma_x} \sin \left[\frac{\pi}{L}(x - xc_1) \right], \qquad x_2' = \frac{L}{\pi\sigma_x} \sin \left[\frac{\pi}{L}(x - xc_2) \right],$$

$$y_1' = \frac{L}{\pi\sigma_y} \sin \left[\frac{\pi}{L}(y - yc_1) \right], \qquad y_2' = \frac{L}{\pi\sigma_y} \sin \left[\frac{\pi}{L}(y - yc_2) \right],$$

$$x_1'' = \frac{L}{2\pi\sigma_x} \sin \left[\frac{2\pi}{L}(x - xc_1) \right], \qquad x_2'' = \frac{L}{2\pi\sigma_x} \sin \left[\frac{2\pi}{L}(x - xc_2) \right],$$

$$y_1'' = \frac{L}{2\pi\sigma_y} \sin \left[\frac{2\pi}{L}(y - yc_1) \right], \qquad y_2'' = \frac{L}{2\pi\sigma_y} \sin \left[\frac{2\pi}{L}(y - yc_2) \right].$$

The center of the two vortices are given by

$$xc_1 = (0.5 - ox)L, \ xc_2 = (0.5 + ox)L, \ yc_1 = (0.5 - oy)L, \ yc_2 = (0.5 + oy)L.$$

The parameters are $f = 0.00006147\text{s}^{-1}$, $L = 5000\text{km}$, $H_0 = 750\text{m}$, $\Delta h = 75\text{m}$, $g = 9.80616\text{ms}^{-2}$, $\sigma_x = \sigma_y = \frac{3}{40}L$, and $ox = oy = 0.1$. The FOM and ROMs simulations are performed on the spatial grid $\Omega = [0, L]^2$ with the spatial mesh sizes $\Delta x = \Delta y = 50\text{km}$. The number of time steps is set to $K = 250$ with the time step-size $\Delta t = 486\text{s}$, which leads to the final time $T = 33\text{h } 45\text{min}$. For each state variable, the size of the snapshot matrices are 10000×250.

The DEIM and POD basis are determined using the following relative cumulative energy criterion (3.3). The accuracy of conserved quantities is affected by DEIM approximation of the nonlinear terms. Consequently, the captured accuracy by the DEIM must be higher than the POD in relative cumulative energy. We set the POD and DEIM tolerances to $\kappa = 10^{-3}$ and $\kappa = 10^{-5}$, which capture at least 99.99% and 99.9999% of relative cumulative energy for POD and DEIM, respectively.

We use the time averaged relative L_2-errors (3.22) to measure the accuracy between FOM and ROM solutions for each state variable $w = u, v, h, s$.

Preservation of the discrete conserved quantities (2.27): the energy, buoyancy, total vorticity, and the mass of the FOM and ROM solutions are measured using the time-averaged relative error $\|E\|_{\text{abs}}$ in (3.21) for $E = H, M, B, Q$.

In Figure 3.18, the singular values exhibit slow decay both for the state variables and the nonlinear terms. According to the relative cumulative energy criteria (3.3), we have selected $r = 5$ POD modes and $p = 35$ DEIM modes.
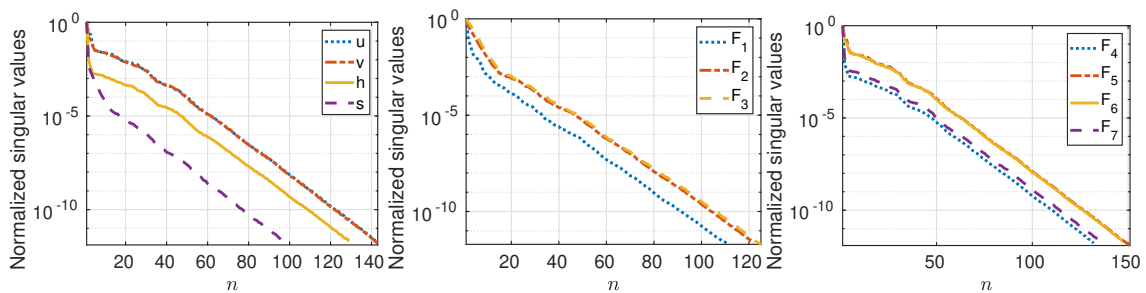

Figure 3.18: Normalized singular values: (left) state variables, (middle and right) nonlinear terms.

Figures 3.19-3.20 indicates that the potential vorticity q and the buoyancy s dynamics are captured well by the ROMs at the final time. Figure 3.21 shows the relative errors in the mass $|M^k - M^0|/M^0$, the Hamiltonian (energy) $|H^k - H^0|/H^0$, the total
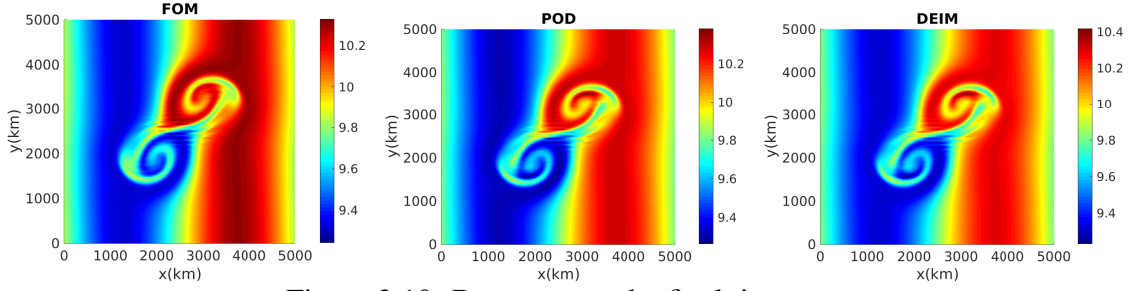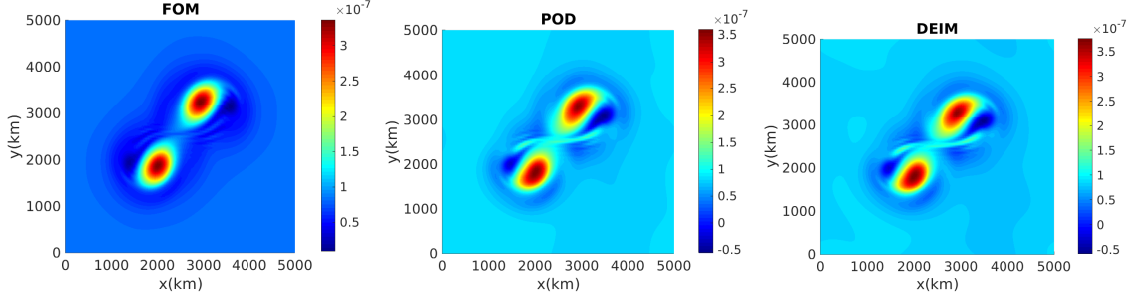
Figure 3.19: Bouyancy at the final time.

Figure 3.20: Potential vorticity at the final time.

potential vorticity $|Q^k - Q^0|/Q^0$, and the buoyancy $|B^k - B^0|/B^0$ versus the time steps, where the total potential vorticity remains constant with the error up to machine precision. Mass, buoyancy, and the energy errors of the ROMs show bounded oscillations over time, i.e., they are conserved almost at the same level of accuracy. All the preserved quantities are well approximated by the POD and POD-DEIM ROMs and do not show any drift over time, which means that the reduced-order solutions are robust in long term computations.

In Table 3.7, we have demonstrated the time-averaged relative L_2 errors of the FOM and ROMs, which shows similar level of accuracy for the POD-DEIM and POD. Moreover, Table 3.8 shows that the preserved quantities are accurately captured by the ROMs. The POD errors are slightly smaller than the POD-DEIM errors in both tables. Nevertheless, the POD-DEIM is much faster than the POD as shown in Table 3.9.

Table 3.7: Time-averaged relative L_2 errors of the state variables.

	$\|\boldsymbol{h} - \widehat{\boldsymbol{h}}\|_{rel}$	$\|\boldsymbol{u} - \widehat{\boldsymbol{u}}\|_{rel}$	$\|\boldsymbol{v} - \widehat{\boldsymbol{v}}\|_{rel}$	$\|\boldsymbol{s} - \widehat{\boldsymbol{s}}\|_{rel}$
POD	8.622e-03	1.502e-01	2.185e-01	7.899e-04
POD-DEIM	1.014e-02	1.737e-01	2.400e-01	7.943e-04

In Table 3.9, we show the computational efficiency ROMs in terms of the wall clock time. The offline computations consist of the time required for the construction of the
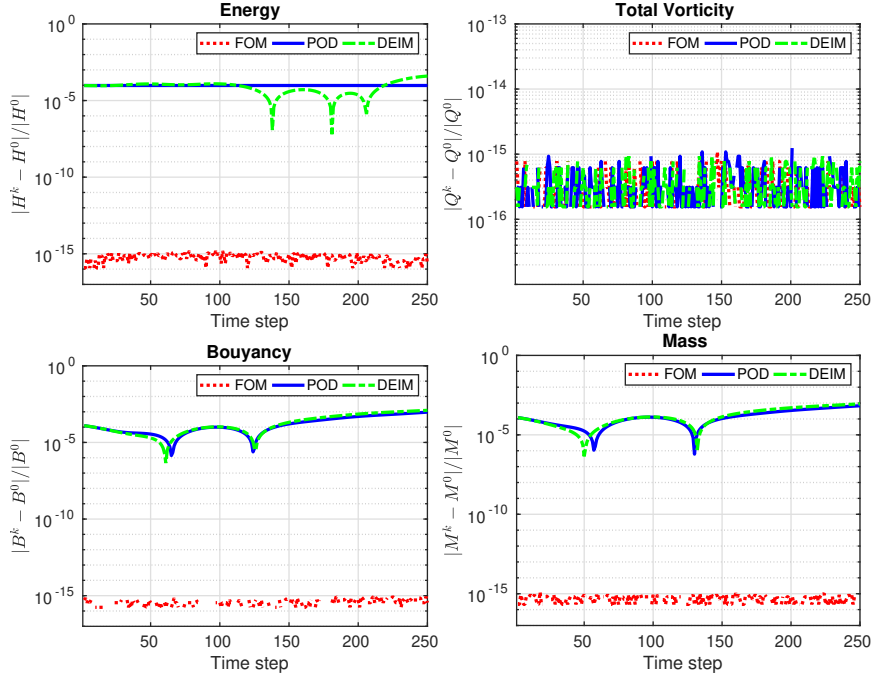
65

Figure 3.21: Relative errors in the conserved quantities.

Table 3.8: Mean relative errors of the conserved quantities by FOM and ROMs.

	$\|H\|_{\text{abs}}$	$\|Q\|_{\text{abs}}$	$\|M\|_{\text{abs}}$	$\|B\|_{\text{abs}}$
FOM	4.768e-15	4.053e-15	2.233e-15	3.267e-16
POD	9.549e-05	3.041e-16	1.834e-04	2.412e-04
POD-DEIM	9.589e-05	3.447e-15	2.440e-04	3.237e-04

Table 3.9: Wall-clock time (in seconds) and speed-up factors.

		Wall-clock time	**Speed-up factors**
FOM		841.2	
POD	offline computation	1.7	
	online computation	31.5	26.7
POD-DEIM	offline computation (POD+DEIM)	10.5	
	online computation	5.8	146.0

precomputed matrices and the basis computation by SVD. On the other hand, the on-line computation includes of the time required for the solution of the reduced models and projections to obtain reduced approximations. Note that the wall clock time of the offline computation for POD-DEIM consists of computation time for DEIM and POD basis as well as the computation time of the reduced tensor calculations. The speed-up factors in Table 3.9 are calculated as the ratio of the time required for FOM solutions over the time required for the online computation of the ROM, which shows that the ROM with DEIM increases the computational efficiency.

CHAPTER 4

DATA DRIVEN NON-INTRUSIVE REDUCED-ORDER MODELLING

Most of the MOR techniques are intrusive, which means they need to access the semi-discretized full order operators so that the reduced-order operators can be obtained by projecting the full order operators onto a low-dimensional subspace using the projection matrices. Intrusive methods sometimes possess less attractive properties in the application, such as they require access to full order operators. When the dynamical system is complex, obtaining an explicit discretized FOM would be difficult. Details of the discretization, solver, and governing equations are generally unavailable when using proprietary software to simulate FOM, and consequently, traditional projection-based intrusive MOR methods are not applicable. The non-intrusive or data-driven reduced models are radically different from the intrusive ROMs. In contrast to the intrusive ROMs, the non-intrusive model reduction techniques aim to learn reduced models from snapshots, i.e., either numerical approximations or measurements of the states and the outputs of the dynamical systems. In this way, the scope of model reduction is extended on settings where the operators of the high-dimensional systems are unavailable. There are several software packages to simulate the SWEs for a given parameter set and an initial condition [48, 38]. In this Chapter, we will investigate applications of the non-intrusive data-driven MOR techniques to SWE, where ROMs are constructed without accessing the full order operators.

Learning dynamical systems from data has become a very active topic, and there is a huge literature on it. Nevertheless, we review the most related ones with reduced systems. Among them, machine learning plays a vital role in analyzing the under-

lying process of the dynamics from the data. A popular approach to constructing a model from only the simulation data is neural networks. By neural networks, it is possible to approximate a large class of functions [66]. Neural networks construct a model by input-output data. They do not need any information about the physical process or discretized operators. A computational bottleneck of the neural networks is the amount of data to construct accurate models, which is large for high-fidelity problems [116]. Moreover, some ideas from compressive sensing have been used to learn the full order operators from a large library of candidate functions [101]. Nevertheless, the method's success massively depends on the compiled library, and generally, one needs to execute computations in the FOM, thus making the method quite challenging in large-scale frames. Dynamic mode decomposition (DMD) learns reduced models for nonlinear dynamical systems, by fitting linear operators to state trajectories with respect to the L_2 norm [100, 107]. Methods based on the Koopman operator have been developed to extend DMD to nonlinear dynamical systems [124]. Reduced models are also constructed by exploiting sparsity in the high-dimensional systems [20, 77, 105]. In the data-fitting surrogate-modeling approaches, regression is used to learn maps from parameters to coefficients of approximations of high dimensional solutions in low-dimensional subspaces. Recently, data-fit non-intrusive regression models with artificial neural networks (ANNs) have been developed for time-dependent dynamical systems [64, 121].

Recently, the operator inference (OpInf) method to assemble ROMs has obtained considerable attention. The OpInf method uses the information of nonlinear terms at the PDE level. In the OpInf methods, the operators representing the ROM are obtained by an optimization problem without requiring the discretized full order operators of the PDEs. The method firstly studied in [90] for polynomial nonlinearities. Later, the method stretched to a class of nonlinear terms, which can be written as quadratic-bilinear (QB) or polynomial nonlinearities by defining new variables in [94, 95]. Lately, in [14] authors generalized the method to nonlinear terms, where the structure of the nonlinear terms is conserved. Recently the DMD is extended to quadratic bilinear systems [52]. The projected trajectories of the OpInf [90] correspond to non-Markovian dynamics in the low-dimensional subspaces even though the high-dimensional trajectories and the corresponding high-dimensional systems

68

are Markovian, as known from, e.g., the Mori–Zwanzig formalism [32, 49]. Dynamical systems are Markovian if future states depend only on the current state and not on previous ones, whereas the non-Markovian systems can be thought of as having a memory so that future states depend on the current and previous states. To recover the Markovian dynamics of the reduced system, in [89] a data sampling scheme has devised. The time stepping iterates between the high-dimensional system and projecting onto the low-dimensional subspaces. The generated trajectories correspond to the low-dimensional Markovian dynamics. It was shown in [89] that under certain conditions, applying operator inference to re-projected trajectories for finite amounts of data and for a large class of systems with polynomial nonlinear terms gives the same operators that are obtained with traditional model reduction methods. Recently, probabilistic a posteriori error estimators are derived for the OpInf with re-projection for linear parabolic PDEs [118]. The non-Markovian OpInf with partial information is investigated [119]. Also, a deep learning version of the OpInf is introduced in [54].

In this Chapter, we apply the OpInf method [90] to the parameterized SWEs. In Section 4.1, the general framework of the OpInf is described. In Section 4.2, we investigate an application of the OpInf method to NTSWE in parametric form. The OpInf is studied in [90] for parametric cases, where the ROMs are built via interpolation for each training parameter. In this study, we present an OpInf framework for the parametric ROMs, assuming that parametric dependency at the PDE level is known. The optimization problem of the OpInf method generally yields a discrete ill-posed least-squares problem. To overcome this bottleneck, we consider the regularized least-squares problem proposed in [90]. The data matrices in the least-squares problem of the OpInf have large condition numbers, leading to ill-conditioned inverse problem. To deal with this, the truncated SVD, Tikhanov regularization, and truncated QR are used. Because decay of the singular values of the data matrices does not provide any information about the choice of the tolerance for the regularization parameter, we solve the least-squares problem in the minimum norm, where the tolerances are determined with the L-curve. In Section 4.3, we lastly investigate the application of the OpInf to RTSWE in parametric form.

Moreover, the intrusive POD Galerkin projection is compared with OpInf using re-projection. Numerical results in Section 4.3.3 show that both the OpInf and the POD

methods are able to predict the parametrized ROMs of RTSWE accurately. Speed-up factors of order two over the FOM have been obtained for both the intrusive and non-intrusive ROMs, whereas the non-intrusive OpInf is more expensive than the POD due to the re-projection. We illustrate the efficiency of the intrusive and non-intrusive ROMs of RTSWE and compare them in terms of capturing the future dynamics of RTSWE outside the range of the training data. Numerical results show that the physical quantities of the RTSWE are conserved for both intrusive and non-intrusive ROMs over time. Moreover, in both of the ROMs, the long-term stability of the reduced trajectories is obtained. The conservation of system physics of FOM, such as the physical quantities by both ROMs, allows that the ROM fit better to data, and stable solutions are obtained in the long-term predictions which are robust to parameter changes.

4.1 Operator Inference for Linear-Quadratic PDEs

In this section, we briefly introduce the operator inference method and discuss the least-squares methods for inferring reduced-order operators.

We consider the following linear-quadratic system of ODEs of the form

$$\dot{\mathbf{w}}(t) = \mathbf{A}\mathbf{w}(t) + \mathbf{H}(\mathbf{w}(t) \otimes \mathbf{w}(t)), \tag{4.1}$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$ is a linear operator, $\mathbf{H} \in \mathbb{R}^{N \times N^2}$ is the matrix which corresponds for quadratic terms, and $\mathbf{w}(t) \in \mathbb{R}^{N \times 1}$ is spatially discretized state vector at time t. The (4.1) arises from spatial discretization of the PDEs with the linear-quadratic nonlinearities such as RSWE and RTSWE.

The snapshot matrix is given as

$$\mathbf{S} = [\mathbf{w}(t_1), \mathbf{w}(t_2), \ldots, \mathbf{w}(t_K)] \in \mathbb{R}^{N \times K}, \tag{4.2}$$

where $\mathbf{w}(t_j)$ is obtained from solving (4.1) at time t_j and K denotes the number of snapshots.

Here we aim to learn the reduced-order approximation of (4.1), so we start by constructing the ROM of (4.1). The first thing in constructing the ROM is to determine the projection matrix $\mathbf{V}_r$, which can be obtained by the SVD of the snapshot matrix (4.2) as follows

$$\mathbf{S} = \mathbf{V}\Sigma\mathbf{U}^T, \tag{4.3}$$

where $\mathbf{V} \in \mathbb{R}^{N \times K}, \Sigma \in \mathbb{R}^{K \times K}, \mathbf{U} \in \mathbb{R}^{K \times K}$, and the projection matrix $\mathbf{V}_r$ is given by the first r columns of $\mathbf{V}$.

Next, we can construct the ROM of (4.1) by using Galerkin projection as follows

$$\dot{\widehat{\mathbf{w}}}(t) = \widehat{\mathbf{A}}\widehat{\mathbf{w}}(t) + \widehat{\mathbf{H}}(\widehat{\mathbf{w}}(t) \otimes \widehat{\mathbf{w}}(t)), \tag{4.4}$$

where $\mathbf{w}(t) \approx \mathbf{V}_r\widehat{\mathbf{w}}(t)$, $\widehat{\mathbf{A}} = \mathbf{V}_r^T\mathbf{A}\mathbf{V}_r \in \mathbb{R}^{r \times r}$, and $\widehat{\mathbf{H}} = \mathbf{V}_r^T\mathbf{H}(\mathbf{V}_r \otimes \mathbf{V}_r) \in \mathbb{R}^{r \times r^2}$ with $r \ll N$.

Our primary goal is to obtain reduced-order operators $\widehat{\mathbf{A}}, \widehat{\mathbf{H}}$ without necessarily accessing the full-order operators $\mathbf{A}, \mathbf{H}$. Since, we already have the trajectories of states $\mathbf{w}(t_j)$ for $j = 1, \ldots, K$ from the solution of the FOM (4.1) or some experiments, using (4.4) the reduced-order operators will satisfy the following least-squares problem:

$$\min_{\widehat{\mathbf{A}} \in \mathbb{R}^{r \times r}, \widehat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}} \sum_{k=1}^{K} \left\| \widehat{\mathbf{A}}\widehat{\mathbf{w}}(t_k) + \widehat{\mathbf{H}}(\widehat{\mathbf{w}}(t_k) \otimes \widehat{\mathbf{w}}(t_k)) - \dot{\widehat{\mathbf{w}}}(t_k) \right\|_2^2, \tag{4.5}$$

where reduced states can be obtained via projection $\widehat{\mathbf{w}}(t_k) = \mathbf{V}_r^T\mathbf{w}(t_k)$. The reduced time derivative of the trajectories $\dot{\widehat{\mathbf{w}}}(t_k)$ can be constructed either using the right-hand side of FOM (4.1) as $\dot{\widehat{\mathbf{w}}}(t_k) = \mathbf{V}_r^T\dot{\mathbf{w}}(t_k)$ or by approximating the time derivatives using some finite difference approximation, e.g., the 5-point approximation

$$\dot{\widehat{\mathbf{w}}}(t_k) = \frac{-\widehat{\mathbf{w}}(t_{k+2}) + 8\widehat{\mathbf{w}}(t_{k+1}) - 8\widehat{\mathbf{w}}(t_{k-1}) + \widehat{\mathbf{w}}(t_{k-2})}{12\Delta t}.$$

To write the least-squares problem (4.5) in a matrix form, let us define the reduced-snapshot matrix $\widehat{\mathbf{S}}$ and reduced-time derivative matrix $\dot{\widehat{\mathbf{S}}}$ as follows

$$\widehat{\mathbf{S}} = [\widehat{\mathbf{w}}(t_1), \widehat{\mathbf{w}}(t_2), \ldots, \widehat{\mathbf{w}}(t_K)] \in \mathbb{R}^{r \times K}, \dot{\widehat{\mathbf{S}}} = \left[\dot{\widehat{\mathbf{w}}}(t_1), \dot{\widehat{\mathbf{w}}}(t_2), \ldots, \dot{\widehat{\mathbf{w}}}(t_K)\right] \in \mathbb{R}^{r \times K}.$$

Using the reduced-snapshot matrix $\widehat{\mathbf{S}}$ and time derivative matrix $\dot{\widehat{\mathbf{S}}}$, (4.5) can be written in the matrix form as

$$\min_{\widehat{\mathbf{A}} \in \mathbb{R}^{r \times r}, \widehat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}} \left\| \widehat{\mathbf{A}}\widehat{\mathbf{S}} + \widehat{\mathbf{H}}(\widehat{\mathbf{S}}\widehat{\otimes}\widehat{\mathbf{S}}) - \dot{\widehat{\mathbf{S}}} \right\|_F^2, \tag{4.6}$$

where $\| \cdot \|_F$ denotes the Frobenius norm $\widehat{\otimes}$ denotes column-wise Kronecker product (Khatri–Rao product).

Transposing (4.6), we obtain

$$\min_{\widehat{\mathbf{A}} \in \mathbb{R}^{r \times r}, \widehat{\mathbf{H}} \in \mathbb{R}^{r \times r^2}} \left\| \widehat{\mathbf{S}}^T \widehat{\mathbf{A}}^T + (\widehat{\mathbf{S}}\widehat{\otimes}\widehat{\mathbf{S}})^T \widehat{\mathbf{H}}^T - \dot{\widehat{\mathbf{S}}}^T \right\|_F^2, \tag{4.7}$$

which can also be written as

$$\min_{\mathbf{O} \in \mathbb{R}^{r \times (r + r^2)}} \left\| \mathbf{D}\mathbf{O}^T - \dot{\widehat{\mathbf{S}}}^T \right\|_F^2, \tag{4.8}$$

where $\mathbf{O} = \begin{bmatrix} \widehat{\mathbf{A}} \ \widehat{\mathbf{H}} \end{bmatrix} \in \mathbb{R}^{r \times (r + r^2)}$ and $\mathbf{D} = \begin{bmatrix} \widehat{\mathbf{S}}^T \ (\widehat{\mathbf{S}}\widehat{\otimes}\widehat{\mathbf{S}})^T \end{bmatrix} \in \mathbb{R}^{K \times (r + r^2)}$.

4.2 OpInf for NTSWE

In this subsection, we consider the OpInf method to learn ROMs for NTSWE from data, e.g., obtained from real-world data or proprietary software. We start the discussion with the linear-quadratic form of the NTSWE in the parametric settings. Next, we construct of the ROM of NTSWE via an intrusive POD method. Then, we introduce the OpInf method to learn the reduced-order operators of NTSWE by simulation data, where we use the information of the parametric dependency at the PDE level. Furthermore, we address computational details for constructing the ROM by the OpInf method in Subsection 4.2.1.

Let us consider the state vector $\mathbf{w} : [0, T] \times \mathcal{D} \to \mathbb{R}^N$ with N degrees of freedom, a parameter vector $\mu \in \mathcal{D} \subset \mathbb{R}^d$, and the time $t \in [0, T] \subset \mathbb{R}$. The NTSWE model (2.10) contains linear and quadratic terms that can be exploited to construct quadratic ROMs. Consequently, let us consider the following linear-quadratic system ODEs:

$$\dot{\mathbf{w}}(t; \mu) = \mathbf{A}(\mu)\mathbf{w}(t; \mu) + \mathbf{H}(\mu)(\mathbf{w}(t; \mu) \otimes \mathbf{w}(t; \mu)), \tag{4.9}$$

where $\mathbf{H}(\mu) \in \mathbb{R}^{N \times N^2}$ to the quadratic terms and $\mathbf{A}(\mu) \in \mathbb{R}^{N \times N}$ corresponds to the linear terms. Let the initial conditions also depend on the parameter μ, i.e., $\mathbf{w}(0, \mu) = \mathbf{w}_0(\mu)$.

Our aim is to construct a parametrized ROM that captures the dynamics of the FOM (4.9) for a given parameter range as follows:

$$\dot{\widehat{\mathbf{w}}}(t; \mu) = \widehat{\mathbf{A}}(\mu)\widehat{\mathbf{w}}(t; \mu) + \widehat{\mathbf{H}}(\mu)(\widehat{\mathbf{w}}(t; \mu) \otimes \widehat{\mathbf{w}}(t; \mu)), \tag{4.10}$$

where $\widehat{\mathbf{A}}(\mu) \in \mathbb{R}^{r \times r}$ and $\widehat{\mathbf{H}}(\mu) \in \mathbb{R}^{r \times r^2}$ with $r \ll N$. When the FOM are available in the matrix-vector form, then intrusive MOR methods can be applied, such as POD [10, 12] and interpolation-based methods [11]. If the full-order operators is available, the reduced-order operators can be constructed with the projection matrix $\mathbf{V} \in \mathbb{R}^{N \times r}$ so that $\mathbf{w}(t, \mu) \approx \mathbf{V}\widehat{\mathbf{w}}(t, \mu)$, for all $t \geq 0$ and $\mu \in \mathcal{D}$ obtained by POD. The reduced-order matrices of (4.10) can be evaluated as follows:

$$\widehat{\mathbf{A}}(\mu) = \mathbf{V}^T \mathbf{A}(\mu)\mathbf{V} \in \mathbb{R}^{r \times r}, \quad \widehat{\mathbf{H}}(\mu) = \mathbf{V}^T \mathbf{H}(\mu)(\mathbf{V} \otimes \mathbf{V}). \tag{4.11}$$

We assume that the full-order operators in (4.9) have a affine dependence on the parameter μ:

$$\mathbf{A}(\mu) = \alpha_1(\mu)\mathbf{A}_1 + \cdots + \alpha_{\mathbf{n_a}}(\mu)\mathbf{A}_{\mathbf{n_a}}, \tag{4.12a}$$

$$\mathbf{H}(\mu) = \eta_1(\mu)\mathbf{H}_1 + \cdots + \eta_{\mathbf{n_h}}(\mu)\mathbf{H}_{\mathbf{n_h}}, \tag{4.12b}$$

where $\alpha_i(\mu), \eta_j(\mu) : \mathbb{R}^d \to \mathbb{R}$ are the smooth functions of the parameter μ and $\mathbf{A}_i \in \mathbb{R}^{N \times N}$, $\mathbf{H}_j \in \mathbb{R}^{N \times N^2}$ are constant matrices. Then, the reduced-order operators in (4.11) can be computed in the offline stage, e.g., $\widehat{\mathbf{A}}(\mu) = \alpha_1(\mu)\widehat{\mathbf{A}}_1 + \cdots + \alpha_{\mathbf{n_a}}(\mu)\widehat{\mathbf{A}}_{\mathbf{n_a}}$, where $\widehat{\mathbf{A}}_i = \mathbf{V}^T \mathbf{A}_i \mathbf{V}$, $i \in \{1, \ldots, \mathbf{n_a}\}$.

However, when the FOM is not available in matrix form constructing reduced-order operators becomes an issue. We assume that data is obtained from proprietary software. We collect simulation data for a training parameter set, $\mu_i \in \mathcal{D}$ for $i = 1, \ldots, M$. Thus, we construct the global snapshot matrix:

$$\mathbf{S}_\mu = [\mathbf{S}(\mu_1), \ldots, \mathbf{S}(\mu_M)], \quad \mathbf{S}(\mu_i) = [\mathbf{w}(t_1; \mu_i), \mathbf{w}(t_2; \mu_i), \ldots, \mathbf{w}(t_K; \mu_i)] \in \mathbb{R}^{N \times K},$$
$$\tag{4.13}$$

where $\mathbf{w}(t_j, \mu_i)$ denotes the state solution at time t_j for the parameter μ_i. The projection matrix $\mathbf{V}$ can be obtained by the SVD of the snapshot matrix

$$\mathbf{S}_\mu = \mathbf{V}_\mu \Sigma_\mu \mathbf{U}_\mu^T, \tag{4.14}$$

73

where $\mathbf{V}_\mu \in \mathbb{R}^{N \times M \cdot K}, \Sigma_\mu \in \mathbb{R}^{M \cdot K \times M \cdot K}, \mathbf{U}_\mu \in \mathbb{R}^{M \cdot K \times M \cdot K}$, and $\mathbf{V}$ is given then by the first r columns of $\mathbf{V}_\mu$.

To learn the reduced-order operators by using the OpInf method, firstly, we project the snapshot matrix $\mathbf{S}_\mu$ onto the dominant subspace spanned by $\mathbf{V}$ as follows:

$$\widehat{\mathbf{S}}_\mu := \mathbf{V}^T \mathbf{S}_\mu = \left[\widehat{\mathbf{S}}(\mu_1), \dots, \widehat{\mathbf{S}}(\mu_M) \right], \tag{4.15}$$

where

$$\widehat{\mathbf{S}}(\mu_i) = [\widehat{\mathbf{w}}(t_1; \mu_i), \widehat{\mathbf{w}}(t_2; \mu_i), \dots, \widehat{\mathbf{w}}(t_K; \mu_i)] \in \mathbb{R}^{r \times K}$$

in which $\widehat{\mathbf{w}}(t_j, \mu_i) := \mathbf{V}^T \mathbf{w}(t_j, \mu_i)$. Moreover, we define

$$\dot{\widehat{\mathbf{S}}}_\mu = \left[\dot{\widehat{\mathbf{S}}}(\mu_1), \dots, \dot{\widehat{\mathbf{S}}}(\mu_M) \right], \tag{4.16}$$

where $\dot{\widehat{\mathbf{S}}}(\mu_i)$ can either be approximated using $\widehat{\mathbf{S}}(\mu_i)$ by employing a finite difference approximation or obtained using the right-hand side of (4.9), see, e.g., [90]. Consequently, the reduced-order operators of the parametric ROM (4.10) are determined by solving the following least-squares problem:

$$\min_{\widehat{\mathbf{A}}_i \in \mathbb{R}^{r \times r}, \widehat{\mathbf{H}}_j \in \mathbb{R}^{r \times r^2}} \sum_{k=1}^{M} \left\| -\dot{\widehat{\mathbf{S}}}(\mu_k)^T + \sum_{i=1}^{\mathbf{n_a}} \left(\alpha_i(\mu_k) \widehat{\mathbf{S}}(\mu_k)^T \widehat{\mathbf{A}}_i^T \right) + \sum_{i=1}^{\mathbf{n_h}} \left(\eta_i(\mu_k) \left(\widehat{\mathbf{S}}(\mu_k) \hat{\otimes} \widehat{\mathbf{S}}(\mu_k) \right)^T \widehat{\mathbf{H}}_i^T \right) \right\|_F^2 \tag{4.17}$$

where $\hat{\otimes}$ denotes the column-wise Kronecker product. Note that the optimization problem (4.17) does not include any explicit information of the FOM; it involves only the projected data. We can rewrite the least-squares problem (4.17) in the standard form as follows:

$$\min_{\mathcal{X} \in \mathbb{R}^{r \times \mathbf{n_a} r + \mathbf{n_h} r^2}} \sum_{k=1}^{M} \left\| \mathcal{A}(\mu_k) \mathcal{X}^T - \dot{\widehat{\mathbf{S}}}(\mu_k)^T \right\|_F^2, \tag{4.18}$$

where

$$\mathcal{X} = \left[\widehat{\mathbf{A}}_1, \dots, \widehat{\mathbf{A}}_{\mathbf{n_a}}, \widehat{\mathbf{H}}_1, \dots, \widehat{\mathbf{H}}_{\mathbf{n_h}} \right],$$
$$\mathcal{A}(\mu_k) = \left[[\alpha_1(\mu_k), \dots, \alpha_{\mathbf{n_a}}(\mu_k)] \otimes \widehat{\mathbf{S}}(\mu_k)^T, [\eta_1(\mu_k), \dots, \eta_{\mathbf{n_h}}] \otimes \left(\widehat{\mathbf{S}}(\mu_k) \hat{\otimes} \widehat{\mathbf{S}}(\mu_k) \right)^T \right].$$

We summarize overall procedure in the Algorithm (2).

Algorithm 2 Operator inference model for parametric NTSWE.

1: **procedure** OPINF

2:     **Input:** States $\mathbf{u}(t_j; \mu_i), \mathbf{v}(t_j; \mu_i), \mathbf{h}(t_j; \mu_i)$ and time derivatives $\dot{\mathbf{u}}(t_j; \mu_i), \dot{\mathbf{v}}(t_j; \mu_i), \dot{\mathbf{h}}(t_j; \mu_i)$

3:     Stack the states into $\mathbf{w}(t_j; \mu_i) = [\mathbf{u}(t_j; \mu_i), \mathbf{v}(t_j; \mu_i), \mathbf{h}(t_j; \mu_i)]^T$. for each $\mu_i \in \mathcal{D}, \ i = 1, \ldots, M$ and $j = 1, \ldots, K$.

4:     Stack the time derivatives into $\dot{\mathbf{w}}(t_j; \mu_i) = [\dot{\mathbf{u}}(t_j; \mu_i), \dot{\mathbf{v}}(t_j; \mu_i), \dot{\mathbf{h}}(t_j; \mu_i)]^T$ for each $\mu_i \in \mathcal{D}, \ i = 1, \ldots, M$ and $j = 1, \ldots, K$.

5:     Construct $\mathbf{S}(\mu_i) = [\mathbf{w}(t_1; \mu_i), \ldots, \mathbf{w}(t_K; \mu_i)]$.

6:     Construct the global snapshot matrix $\mathbf{S}_\mu = [\mathbf{S}(\mu_1), \ldots, \mathbf{S}(\mu_M)]$.

7:     Compute the dominant global POD basis $\mathbf{V}$ of $\mathbf{S}_\mu$ using (4.14).

8:     Construct $\dot{\mathbf{S}}(\mu_i) = [\dot{\mathbf{w}}(t_1; \mu_i), \ldots, \dot{\mathbf{w}}(t_K; \mu_i)]$.

9:     Compute $\widehat{\mathbf{S}}(\mu_i) = \mathbf{V}^T \mathbf{S}(\mu_i)$ and $\dot{\widehat{\mathbf{S}}}(\mu_i) = \mathbf{V}^T \dot{\mathbf{S}}(\mu_i)$.

10:     Solve the least-squares problem (4.18) to obtain operators of the reduced-order system, having the form. (4.10)

11: **end procedure**

4.2.1 Computational Aspects

In this subsection, we discuss about the computational details of the OpInf method (4.18). Computationally, solving the optimization problem (4.18) can be a difficult task because the problem can be highly ill-conditioned and the cost of the solution $\mathcal{O}\left((rK(\mathbf{n_a}r + \mathbf{n_h}r^2)^3 \right)$ grows quickly with the order r of the ROM, and linearly with the number of snapshots [90]. The computational cost of the least-squares problem (4.18) can be decreased by decoupling of the problem.

In the OpInf method, learnt reduced-order operators are solutions of the ill-posed least-squares problem (4.18), where ill-conditioning may occur due to nearly linearly dependence in the columns of the snapshot matrix. If the snapshots at successive time steps are almost linearly dependent due to small step size, the condition number of the snapshot matrix in the least-squares problem increases. Therefore, the least-squares problem arising from the OpInf method needs a proper regularization method. There exist several methods to deal with this problem. A proper and extensively used candidate for this task is the Tikhonov regularization [117]. Small singular values of the

data matrix amplify the noise on the solution of the least-squares problem, which can be filtered by applying Tikhonov regularization. The quality of the solution obtained by the Tikhonov regularization depends on the regularisation parameter selection, which is typically selected by the L-curve [60]. The Tikhonov regularization parameter creates a trade-off between the residual norm and the solution norm. If the parameter is large, the solution becomes overly smooth that may fail to catch some essential details on the desired solution, while a small regularization parameter may produce a solution that is severely affected by the propagated error. Typically, the logarithmic plot of the solution vs the residual norm exhibits an L shaped curve. The corner of the L-curve represents the optimal regularization parameter, which gives a compromise between the minimization solution norm and residual norm [60]. The stability of the OpInf method can be improved by L-curve [115]. Nevertheless, the computation efficient Tikhonov parameter is costly for large-scale problems [21]. Although it can be argued that the least-squares problem (4.18) is in a low dimension, it still can be large when the number of training parameters or/and the number of snapshots is large. In a compact form the Tikhonov regularization applied to (4.18) can be written as follows:

$$\min_{x_i \in \mathbb{R}^{\mathbf{n_a}r + \mathbf{n_h}r^2}} \|\mathcal{A}_\mu x_i - s_i\|_2^2 + \lambda \|x_i\|_2^2, \quad i = 1, \ldots, r. \tag{4.19}$$

Here x_i are the columns of $\mathcal{X}^T$, and s_i are the columns of $\hat{\dot{\mathbf{S}}}_\mu$, and

$$\mathcal{A}_\mu = \left[\mathcal{A}^T(\mu_1), \ldots, \mathcal{A}^T(\mu_M) \right]^T$$

.

In [90], a heuristic method is proposed to deal with the conditioning of the data matrix in which a subset of the snapshots is considered by taking the snapshots in a regular interval e.g., every 5th time-step. In [90], it is also shown that this method can mitigate the ill-conditioning problem to some extent in some instances. Nevertheless, the interval selection should be made so that the essential trajectories are not missed; therefore, the determination of the interval plays an essential role. This problem is referred to as a heuristic column subset selection problem (CSSP) in the literature. The CSSP determines a subset of the most linearly independent columns of a matrix that gives the best knowledge in the matrix.

The QR decomposition of the snapshot matrix with column pivoting (QR-CP) can be used to find a proper subset for the CSSP. Here, we use the tQR method [27] which is favourable for rank deficient problems with the minimum norm solution. One alternative to this approach is the truncated SVD (tSVD) algorithm. The tSVD is also a widely used method due to its best rank-k approximation. Typically, the tSVD method and tQR method produce close solutions. However, the computation of the tSVD is more costly than the tQR algorithm.

Suppose $\mathcal{A}_\mu \in \mathbb{R}^{m \times n}$ is a rank deficient matrix with rank$(\mathcal{A}_\mu) = p$. There always exists a QR-CP factorization of $\mathcal{A}_\mu$ of the form

$$\mathcal{A}_\mu \Pi = QR, \tag{4.20}$$

where $Q \in \mathbb{R}^{m \times m}$ is an orthogonal matrix, $\Pi \in \mathbb{R}^{n \times n}$ is a permutation matrix, and R is an upper triangular matrix of the form

$$R = \begin{pmatrix} R_1 & R_2 \\ 0 & R_3 \end{pmatrix} \approx \begin{pmatrix} R_1 & R_2 \\ 0 & 0 \end{pmatrix}. \tag{4.21}$$

$R_1 \in \mathbb{R}^{p \times p}$ is upper triangular with rank$(R_1) = p$ and $\|R_3\|_2$ is small. The diagonal entries of R in (4.20) satisfy $|R_{ii}| \geq |R_{jj}|$ for $j > i$ so that the effective rank of $\mathcal{A}$ can be obtained by taking the smallest integer p as the rank such that

$$|R_{p+1,p+1}| < \texttt{tol} \cdot |R_{11}|,$$

where $\texttt{tol}$ can be considered as the tolerance for the linear dependency of the columns of data matrix $\mathcal{A}_\mu$. The regularization tolerance of tQR $\texttt{tol}$ can also be obtained by the L-curve [60]. The minimum norm solution by tQR regularization can be determined as done in [27].

Lastly, in this subsection, we shortly discuss the computational details of Algorithm (4). Since obtaining the POD basis is similar for both non-intrusive and intrusive models, we exclude it from the discussion. Furthermore, we assume that time derivatives (4.16) are given. Computation of the projected trajectories at line (9) cost in $\mathcal{O}(rNK)$. Solving the least-squares problem at line (10) with tQR method as r independent problem cost in $\mathcal{O}\left(\left(2(\mathbf{n_a}r + \mathbf{n_h}r^2 - p) + 1 \right) \left(\mathbf{n_a}r + \mathbf{n_h}r^2 \right)^2 \right)$, where p is the effective rank [27].

4.2.2 Numerical Results

In this subsection, we compare the performance of the OpInf method with the intrusive POD method for two numerical test problems. We also show the prediction capabilities of the intrusive POD method and the OpInf method for the parametric and non-parametric NTSWE. Moreover, we compare the non-intrusive techniques for the tQR method with the tolerance 10^{-6} and for the Tikhonov regularization (4.19) with the penalty parameter $\lambda = 0.01$. Determination of the suitable regularization parameters for both Tikhonov and tQR methods have done by the L-curve criteria. The first test problem shows the propagation of the inertia-gravity waves by Coriolis force, known as geostrophic adjustment [113]. The second example demonstrates a shear instability in the form of a roll-up of an unstable shear layer, known as barotropic instability [113].

The spatial derivatives in NTSWE (2.10) is discretized in space by using central finite-differences approximation. The semi-discrete system includes linear and quadratic terms that depends on the parameter $\mu = \theta$:

$$\dot{\mathbf{w}}(t; \mu) = \mathbf{A}(\mu)\mathbf{w}(t; \mu) + \mathbf{H}(\mu)(\mathbf{w}(t; \mu) \otimes \mathbf{w}(t; \mu)), \qquad (4.22)$$

where $\mathbf{H}(\mu) \in \mathbb{R}^{N \times N^2}$ corresponds to the quadratic term, $\mathbf{A}(\mu) \in \mathbb{R}^{N \times N}$ corresponds to the linear terms, $\mu \in \mathcal{D} \subset \mathbb{R}^d$, the state vector $\mathbf{w} : [0, T] \times \mathcal{D} \to \mathbb{R}^N$ with N degrees of freedom, and the time $t \in [0, T] \subset \mathbb{R}$. The full-order operators $\mathbf{A}(\mu)$ and $\mathbf{H}(\mu)$ in (4.12) are dependent to the parameter μ affinely as follows:

$$\alpha_1(\mu) = 1, \quad \alpha_2(\mu) = \Omega^z = \sin(\mu), \quad \alpha_3(\mu) = \Omega^z \Omega^y = \sin(\mu)\cos(\mu), \quad (4.23a)$$

$$\eta_1(\mu) = 1, \quad \eta_2(\mu) = \Omega^y = \cos(\mu), \quad \eta_3(\mu) = (\Omega^y)^2 = (\cos(\mu))^2. \qquad (4.23b)$$

We study the NTSWE with periodic boundary conditions and assume no input or forcing terms in (4.22).

For simulations, we considered the function **ode15s** in MATLAB® with both the relative and absolute error tolerances set to 10^{-8}. In all numerical examples of NTSWE, we have used the 101×101 equidistant grid points to discretize the spatial domain. We also sampled the snapshots at equidistant time instances using the time step $\Delta t = 0.1$. To measure the accuracy of the ROMs, let us define following relative error in the

78

Frobenius norm

$$\mathcal{E} = \frac{||\mathbf{S}_{\text{FOM}} - \mathbf{V}\mathbf{S}_{\text{ROM}}||_F}{||\mathbf{S}_{\text{FOM}}||_F}, \tag{4.24}$$

where $\mathbf{S}_{\text{FOM}} \in \mathbb{R}^{N \times K}$ is the snapshot matrix of the FOM and $\mathbf{S}_{\text{ROM}} \in \mathbb{R}^{r \times K}$ is the snapshot matrix of either the non-intrusive or the intrusive ROMs. The relative error for the parametric case is computed with snapshot matrices by concatenating the trajectories for parameter samples. Typically, the reduced dimension r is determined through the projection error as follows:

$$\mathcal{E}_{\text{proj}} = \frac{||\mathbf{S}_{\text{FOM}} - \mathbf{V}\mathbf{V}^T\mathbf{S}_{\text{FOM}}||_F}{||\mathbf{S}_{\text{FOM}}||_F}. \tag{4.25}$$

Single-layer geostrophic adjustment

The initial conditions are designated in the form of a motionless layer with an upward protrusion of the height field in a periodic domain $[-5, 5] \times [-5, 5]$:

$$h(x, y, 0) = 1 + \frac{1}{2}\exp\left[-\left(\frac{4x}{5}\right)^2 - \left(\frac{4y}{5}\right)^2\right], \quad u(x, y, 0) = 0, \quad v(x, y, 0) = 0.$$

After the collapse of the initial symmetric peak, the inertia-gravity waves propagate with respect to the axes. Nonlinear interactions produce smaller waves, propagating throughout the domain, and the interactions create more complicated patterns [113].

Non-parametric case

Here, we examine the NTSWE for a fixed parameter $\mu = \frac{\pi}{4}$. The snapshots of the FOM are obtained by solving (2.10) in the time domain $[0, T]$ with $T = 60$. The FOM trajectories are concatenated into $\mathbf{w} = [\mathbf{u}, \mathbf{v}, \mathbf{h}]^T \in \mathbb{R}^{30000}$, resulting to a training data of the size $\mathbf{S}_{\text{FOM}} \in \mathbb{R}^{30000 \times 601}$. Then, the non-intrusive OpInf and the intrusive POD are used to predict the height field outside of the training interval at time $T = 80$.

In Figure 4.1, the decay of the first 300 normalized singular values of the snapshot matrix $\mathbf{S}_{\text{FOM}}$ is displayed . The slow decay in the normalized singular values shows the challenge of constructing accurate ROMs for a few POD modes, which is a common problem for hyperbolic PDEs like the NTSWE.

In Figure 4.2, we show the condition number of the data matrix $\mathcal{A}_\mu$, which is severely ill-conditioned. For up to the reduced dimension $r = 24$ the least-squares problem
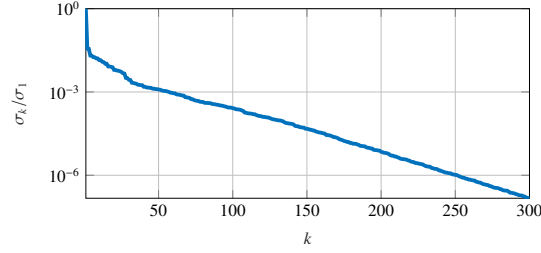
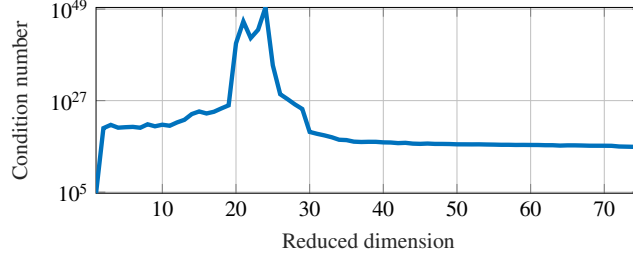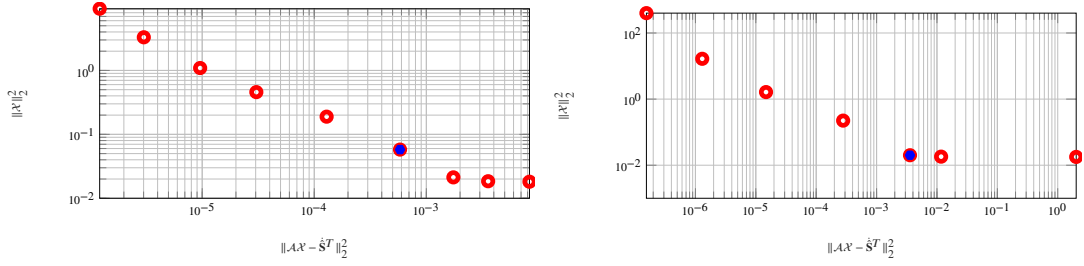Figure 4.1: Single-layer geostrophic adjustment: Normalized singular values.

Figure 4.2: Condition number of the data matrix.

(4.30) becomes overdetermined, and the condition number increasing. On the other hand, when the reduced dimensions greater than $r = 24$, the least-squares problem (4.30) becomes underdetermined, and the condition number is decreasing. When the least-squares problem becomes underdetermined, the uniqueness of the system is lost. Here, the uniqueness of the system is retained due to the minimum-norm solution. Furthermore, the OpInf method without regularization does not give a meaningful result in this problem due to an ill-conditioned data matrix. Truncating the matrix R in (4.21), the condition number is adjusted to approximately $1/\texttt{tol} \approx 10^6$.

In Figure 4.3 , we show the L-curves of tQR and Tikhonov method for the reduce dimension $r = 20$, Tikhonov parameters $\lambda = [10^1, 10^0, \dots, 10^{-7}]$, and tQR tolerances $\texttt{tol} = [10^{-4}, 10^{-5}, \dots, 10^{-10}]$. The vertical axis of the plot is the squared norm of the learned operators, and the horizontal axis is the norm of the residual. If the L-curves are computed for each reduced dimension, the cost remarkably will increase; therefore, the Tikhonov parameter λ and the tQR tolerance $\texttt{tol}$ are chosen close to the corners at the L-curves for $r = 20$ which yields stable solutions from reduce dimension $r = 20$ up to reduce dimension $r = 75$. The taken tQR tolerance and Tikhonov parameter λ are demonstrated as blue dot in Figure 4.3.

We construct a POD-projection matrix $\mathbf{V} \in \mathbb{R}^{30000 \times r}$ with reduced dimension $r = 75$, which results to a projection error (4.25) $\mathcal{E}_{\text{proj}} = 2.07 \cdot 10^{-3}$. Next, we construct ROMs

(a) Tikhonov   (b) tQR

Figure 4.3: L-curves: Tikhonov (left) , tQR (right).

Table 4.1: Single-layer geostrophic adjustment: Comparison of ROMs obtained by POD and OpInf method.

Method	POD	OpInf (Tikhonov regularizer)	OpInf (tQR)
$\mathcal{E}$	$3.27 \cdot 10^{-3}$	$2.07 \cdot 10^{-3}$	$2.07 \cdot 10^{-3}$

using the non-intrusive OpInf and intrusive POD methods. To compare the quality of the non-intrusive ROMs, we have regularized the least-squares problem arising from the OpInf method with both Tikhonov and tQR-based regularizers. In Table 4.1, we demonstrate the accuracy of the non-intrusive and intrusive ROMs using the FOM-ROM error (4.24), where the non-intrusive method gives a better ROM as compared to the intrusive POD method, and both regularizations yield a similar result.

Furthermore, we examine the quality of the ROMs as the reduced-order dimension is increasing. The quality of the ROMs is examined by the relative error $\mathcal{E}$ (4.24) for ROMs obtained through the intrusive and non-intrusive methods in Figure 4.4. The relative error (4.24) arising from the intrusive ROM does not decrease as flatly as in the non-intrusive case. Both regularizers yield equally accurate solutions for all the reduced dimensions of the ROMs. Nevertheless, the penalty parameter of the Tikhonov regularization typically is obtained by the L-curve, which needs the SVD computation of the data matrix [60]. Therefore, the Tikhonov regularization with L-curve is more costly than the tQR.

In the remainder of this subsection for time-domain simulations and prediction, we present the results for the OpInf method with tQR since both the tQR-based and Tikhonov methods results in comparable solutions with respect to the projection error (4.25). In Figure 4.5, the height field obtained from the FOM and ROMs of dimension $r = 75$ at time $T = 60$ are plotted. The figures indicate that both ROMs are very close
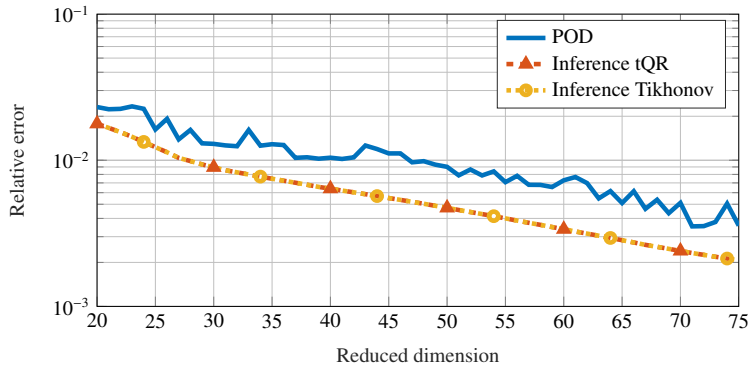
81

Figure 4.4: Single-layer geostrophic adjustment: Relative ROMs errors.

to the FOM solutions, where the non-intrusive ROM is slightly more accurate.

Next, we examine the prediction capabilities of both intrusive and non-intrusive ROMs of dimension $r = 75$. We have trained the ROMs using the snapshots up to time $T = 60$. Then, the height field outside of the training interval at time $T = 80$ is examined in Figure 4.6. Figure 4.6 shows that the height field can be predicted with better accuracy by using the non-intrusive ROM compared to the intrusive ROM.

(a) FOM

(b) Intrusive POD

(c) FOM - POD error

(d) Inference

(e) FOM - inference error
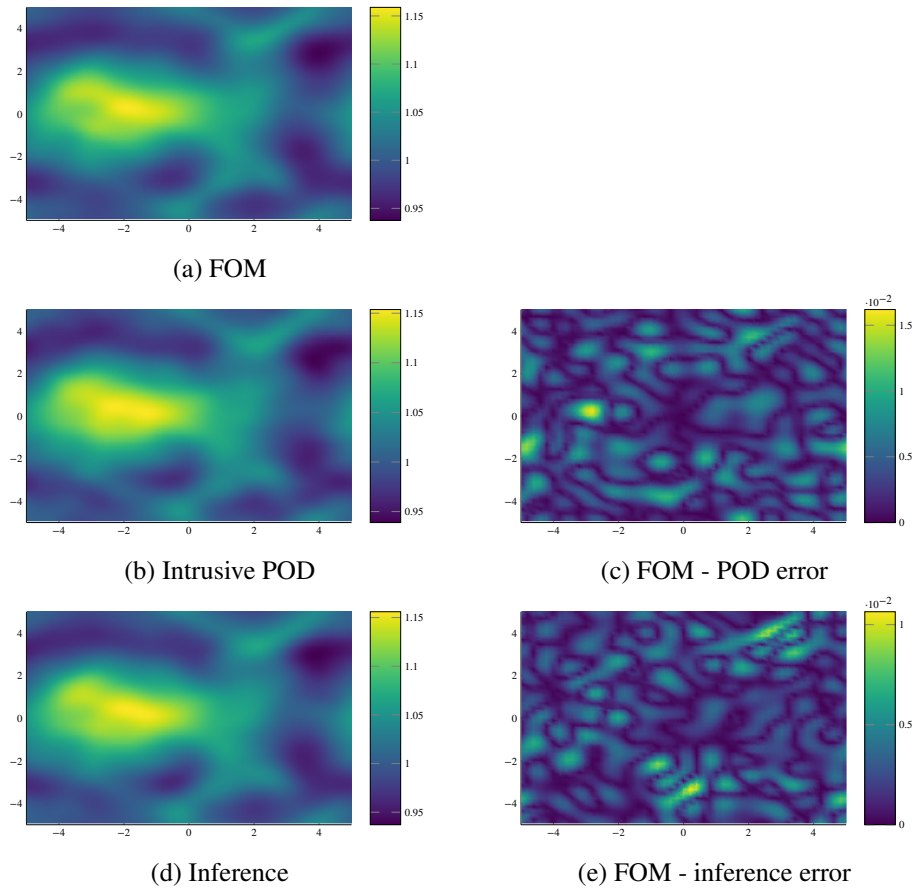
Figure 4.5: Single-layer geostrophic adjustment: Comparison of the height field obtained using the FOM and ROMs of order $r = 75$ at time $T = 60$.
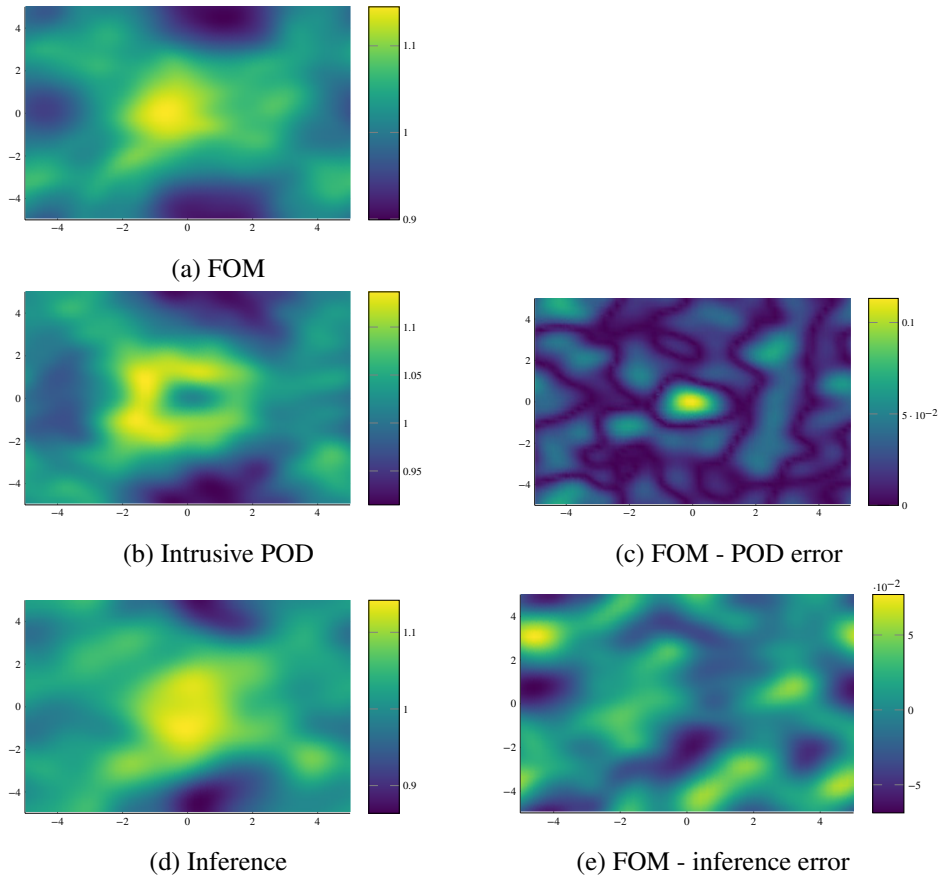
(a) FOM

(b) Intrusive POD

(c) FOM - POD error

(d) Inference

(e) FOM - inference error

Figure 4.6: Single-layer geostrophic adjustment: Prediction of the height field obtained using the FOM and ROMs of order $r = 75$ at time $T = 80$.

Parametric Case

In this subsection, we examine the non-intrusive ROM in the parameter domain of NTSWE as $\mathcal{D} = \left[\frac{\pi}{6}, \frac{\pi}{3}\right] \subset \mathbb{R}$. The time domain is set to $[0, T] \subset R$ for final time $T = 10$. We have generated the trajectories for $M = 5$ equidistantly parameter values $\mu_1, \mu_2, \ldots, \mu_5 \in \mathcal{D}$. The snapshot matrix is constructed as in (4.13), which yields $\mathbf{S}_{\mathrm{FOM}} \in \mathbb{R}^{30000 \times 505}$.

Similar singular values in Figure 4.7 decay is observed to the non-parametric case in Figure 4.1. The accuracy of the non-intrusive ROMs examined for $r = 75$, where we compute the projection error (4.25) with the training data, which is $\mathcal{E}_{\mathrm{proj}} = 2.10 \cdot 10^{-4}$. Table 4.2 shows that the relative errors (4.24) are close to the projection error (4.25) for the both ROMs, which shows that non-intrusive ROMs of both non-intrusive methods have the equivalent level of accuracy. In the parametric setting, we only consider the non-intrusive approach; yet, we observe a similar performance for the parametric setting as in the non-parametric case. We additionally show the relative errors for the non-intrusive ROMs of dimension $r = 25$ to $r = 75$ over the training interval in Figure 4.8. Repeatedly, we observe a similar behaviour as for the non-intrusive ROM; the relative errors in the training interval decrease when the dimension of the ROM increases. Additionally, for dimension $r = 75$, the relative error of the tQR regularizer performs slightly less correct behaviour than the Tikhonov regularizer because of the fixed tolerance of the tQR method.

Next, we investigate the performance of the non-intrusive parametric models on the test parameter values. The test parameter values are considered as the midpoint of two successive training parameter values. In Figure 4.9, we show the relative errors (4.24) of the non-intrusive ROM of dimensions $r = 25$ and $r = 75$ for both test and training parameter values. Figure 4.9 indicates that the accuracy of the non-intrusive
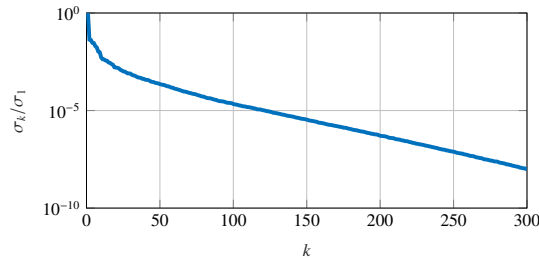

Figure 4.7: Single-layer geostrophic adjustment: Normalized singular values.

Table 4.2: Single-layer geostrophic adjustment: Comparison of ROMs obtained by OpInf method.

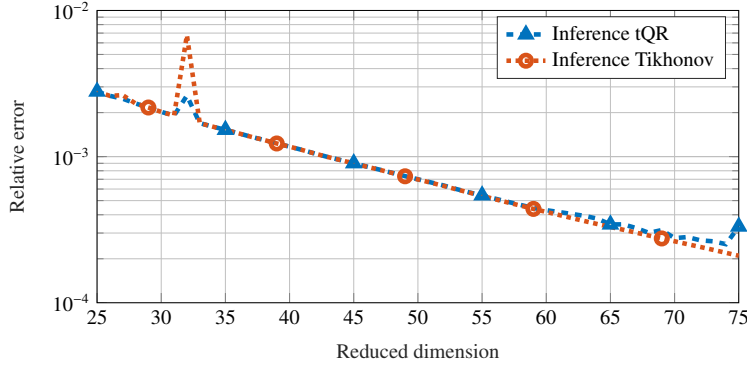| Method	OpInf (Tikhonov regularizer)	OpInf (tQR)
$\mathcal{E}$	$2.10 \cdot 10^{-4}$	$3.32 \cdot 10^{-4}$

Figure 4.8: Single-layer geostrophic adjustment: Relative ROMs errors.

model increases when the dimension r increases for training and testing parameter values.

Finally, we have plotted the height field and the corresponding absolute errors at time $T = 10$ for the parameter $\mu = \frac{5\pi}{24}$ in Figure 4.10 for the non-intrusive model of dimension $r = 75$, which indicates that the non-intrusive model captures the dynamics of the FOM accurately.
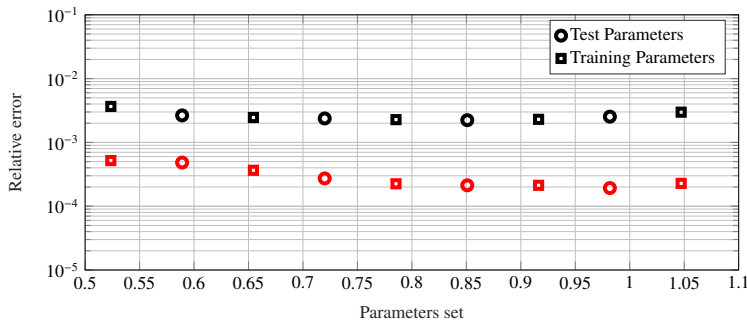

Figure 4.9: Single-layer geostrophic adjustment: Relative error for testing and training parameters; (square): training set, (circle): testing set. (black): reduced dimension $r = 25$, (red): reduced dimension $r = 75$.

(a) FOM

(b) Inference
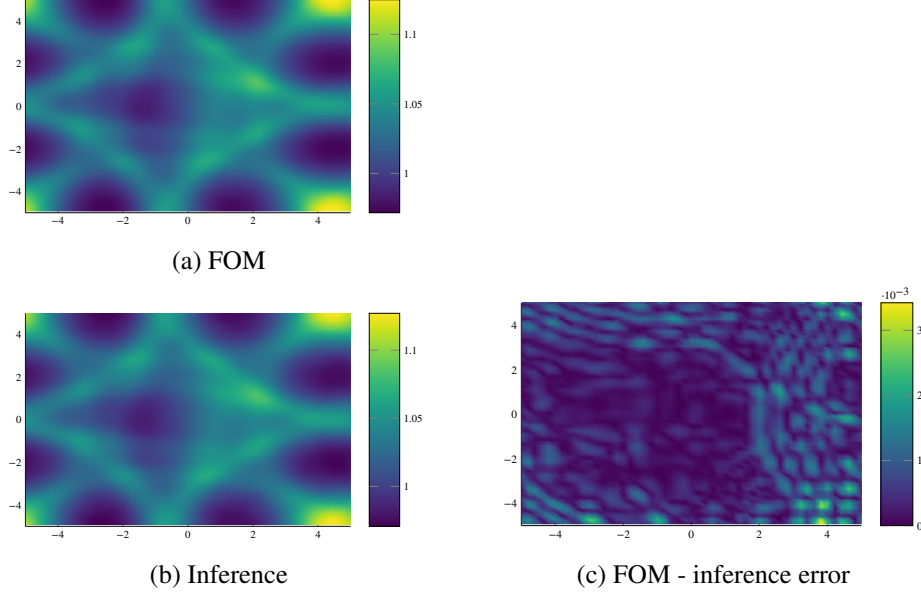

(c) FOM - inference error

Figure 4.10: Single-layer geostrophic adjustment: Comparison of the height field for the parameter $\mu = \frac{5\pi}{24}$ obtained using the FOM and OpInf of order $r = 75$ at time $T = 10$.

Single-layer shear instability

The initial conditions for the second test example on the periodic domain $[0, 10] \times [0, 10]$ are given as:

$$h(x, y, 0) = 1 + \Delta h \sin\left\{\frac{2\pi}{L}\left[y - \Delta y \sin\left(\frac{2\pi x}{L}\right)\right]\right\},$$

$$u(x, y, 0) = -\frac{2\pi \Delta h}{\Omega^z L} \cos\left\{\frac{2\pi}{L}\left[y - \Delta y \sin\left(\frac{2\pi x}{L}\right)\right]\right\},$$

$$v(x, y, 0) = -\frac{4\pi^2 \Delta h \Delta y}{\Omega^z L^2} \cos\left\{\frac{2\pi}{L}\left[y - \Delta y \sin\left(\frac{2\pi x}{L}\right)\right]\right\} \cos\left(\frac{2\pi x}{L}\right),$$

where $\Delta h = 0.2$, $\Delta y = 0.5$ and the dimensionless spatial domain length $L = 10$. This test example illustrates the roll-up of an unstable shear layer [113].

Non-parametric case

The quality of non-intrusive OpInf method is examined in terms of capturing the vorticity dynamics for the parameter value $\mu = \frac{\pi}{4}$. We sample the trajectories as the state vectors concatenated into $\mathbf{w} = [\mathbf{u}, \mathbf{v}, \mathbf{h}]^T \in \mathbb{R}^{30000}$. The FOM (4.22) is solved in the time domain $[0, 60]$, which results the training data set of the size $\mathbf{S}_{\text{FOM}} \in \mathbb{R}^{30000 \times 601}$. The performance of the ROMs in terms of relative errors (4.35) is demonstrated in Figure 4.11, which indicates that the relative errors of the non-intrusive models de-
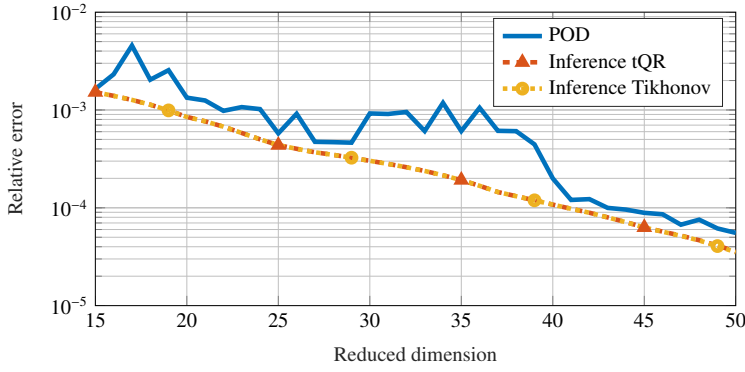
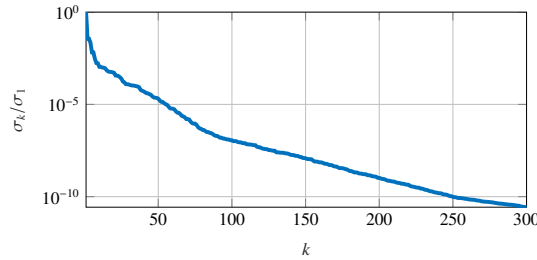Figure 4.11: Single-layer shear instability: Relative ROMs errors.

Figure 4.12: Single-layer shear instability: Normalized singular values.

crease flatly with decreasing singular values in Figure 4.12, whereas the intrusive ROM errors decrease non-smoothly.

The projection error (4.25) for $r = 50$ results, $\mathcal{E}_{\text{proj}} = 3.52 \cdot 10^{-5}$. We also compare the relative errors (4.24) of ROMs of dimension $r = 50$ with the projection error in Table 4.3, which again shows that the non-intrusive models are more accurate.

The potential vorticities obtained from the ROM of dimension $r = 50$ and FOM as well as corresponding absolute error are demonstrated in Figure 4.13, where both the ROM and FOM shows similar roll-up behavior of the vorticity. In Figures 4.13c, 4.13e, the accuracy of non-intrusive and intrusive ROMs of dimension $r = 50$ shows similiar accuracy for the vorticity dynamics.

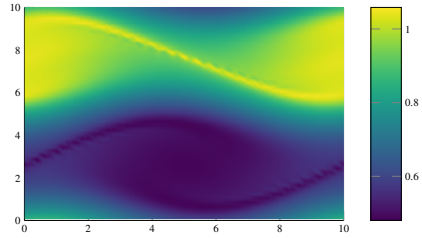We show the prediction capability of the ROMs of dimension $r = 50$ via non-intrusive OpInf and intruive POD methods by training them in the time interval $[0, 60]$, in

Table 4.3: Single-layer shear instability: Comparison of ROMs obtained by POD and OpInf method.
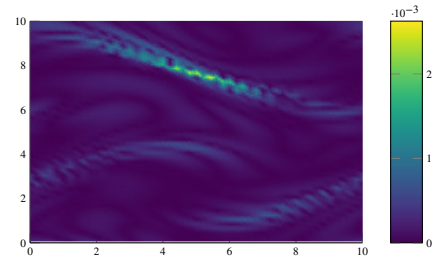
Method	POD	OpInf (Tikhonov regularizer)	OpInf (tQR)
$\mathcal{E}$	$5.54 \cdot 10^{-5}$	$3.52 \cdot 10^{-5}$	$3.58 \cdot 10^{-5}$
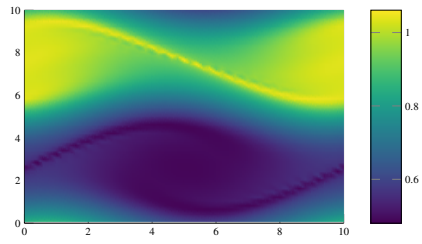
(a) FOM

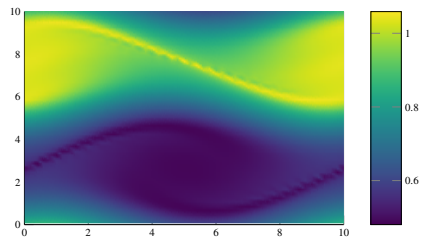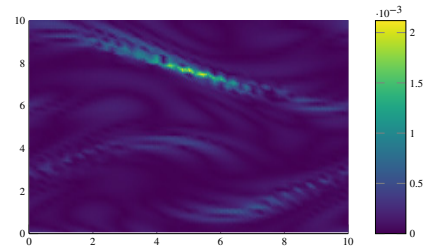(b) Intrusive POD

(c) FOM - POD error

(d) Inference

(e) FOM - inference error

Figure 4.13: Single-layer shear instability: Comparison of the potential vorticity field obtained using the FOM and ROMs of order $r = 50$ at time $T = 60$.

Figure 4.14. The final time for the prediction is set to $T = 80$. Figure 4.14 indicates that the non-intrusive model are more accurate than the intrusive model.

(a) FOM

(b) POD

(c) FOM - POD error

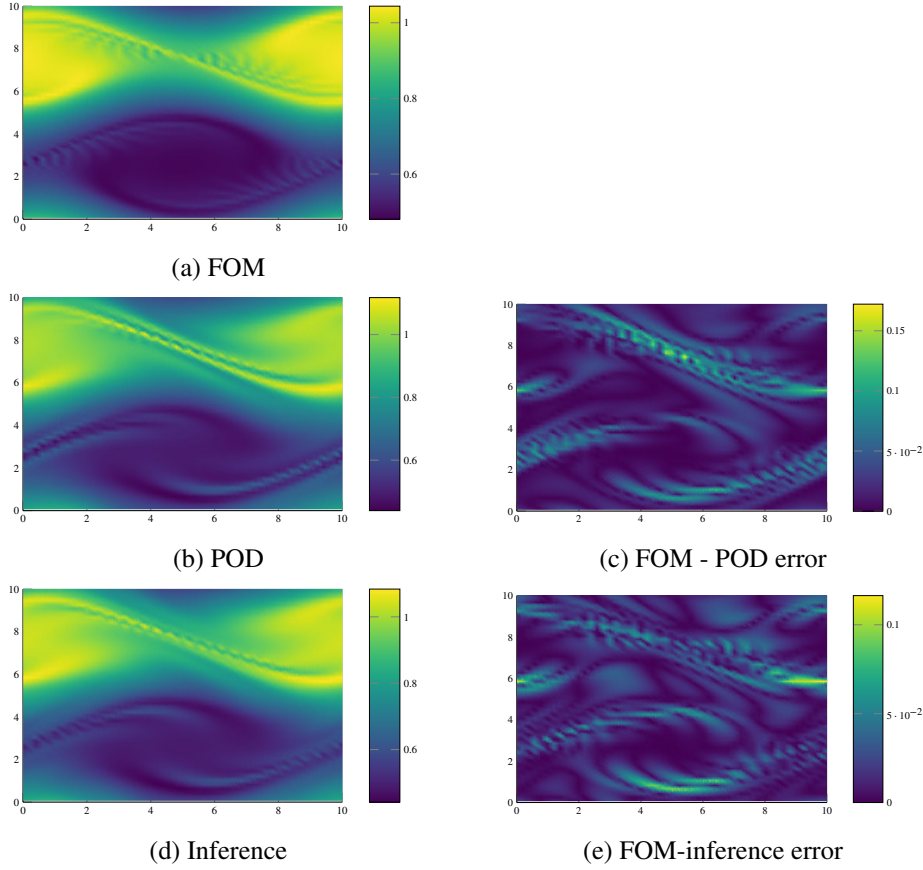(d) Inference

(e) FOM-inference error

Figure 4.14: Single-layer shear instability: Prediction of the potential vorticity field obtained using the FOM and ROMs of order $r = 50$ at time $T = 80$.

Parametric Case

In the last example of NTSWE example, we study the performance of both intrusive and non-intrusive models in terms of the vorticity dynamics by fixing the parameter domain of NTSWE as $\mathcal{D} = \left[\frac{\pi}{6}, \frac{\pi}{3}\right] \subset \mathbb{R}$.

In this example, the initial condition is parameter dependent as well, i.e., dependent on the angular velocity vector Ω^z. We have obtained training trajectories by simulating NTSWE (4.22) on the time domain $[0, T]$ and $M = 5$ equidistantly distributed parameter values $\mu_1, \mu_2, \ldots, \mu_5 \in \mathcal{D}$. The final time of the FOM simulation is set to $T = 30$. The snapshot matrix $\mathbf{S}_{\text{FOM}} \in \mathbb{R}^{30000 \times 1505}$ is again constructed from the concatenated states.

The relative errors (4.24) of the non-intrusive models of dimensions 20 to 65 over the training data set are demonstrated in Figure 4.15, which is smoothly decreasing with the normalized singular values decrease as shown in Figure 4.16. The projection error
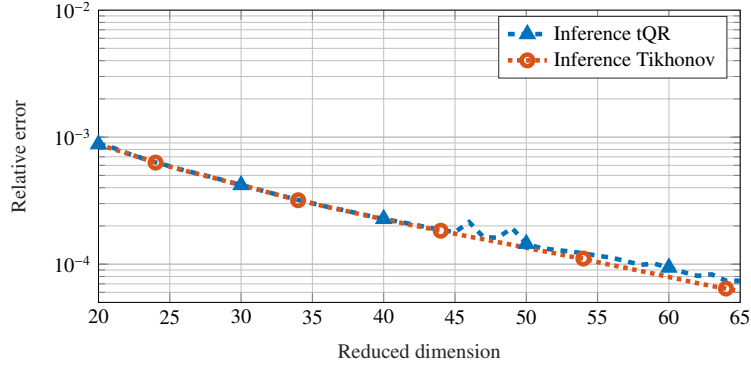
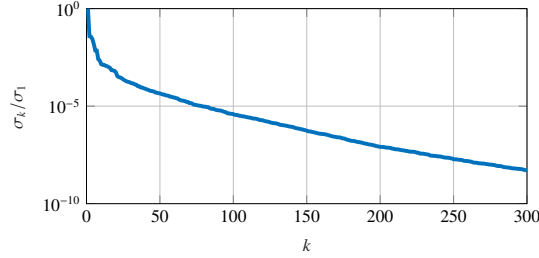Figure 4.15: Single-layer shear instability: Relative ROMs errors.

Figure 4.16: Single-layer shear instability: Normalized singular values.

(4.25) for dimension $r = 65$ is $\mathcal{E} = 6.12 \cdot 10^{-5}$. We have compared the projection error with the relative error (4.24) of the non-intrusive model of dimension $r = 65$ in Table 4.4 , which indicates that both regularizers provide similarly accurate performance.

We again consider the parameter values at the middle points of two successive training parameters for the testing values. Figure 4.17 demonstrates the relative errors for parameter values in the test and training sets for both non-intrusive ROMs of dimension 20 and 65. In Figure 4.17, the regularized solution for the test value $\mu = \frac{9\pi}{48}$ is less accurate comparing to remaining parameter values in the test set for dimension 65. This implies that the truncation tolerance for the QR-CP method can degenerate the accuracy of the ROMs.

Finally, we demonstrate the potential vorticity of the FOM and the non-intrusive model of dimension $r = 65$ as well as the corresponding absolute error at time $T = 30$ for the parameter $\mu = \frac{5\pi}{24}$ in Figure 4.18, which indicates that the non-intrusive model

Table 4.4: Single-layer shear instability (parametric case): Comparison of ROMs obtained by OpInf method.

Method	OpInf (Tikhonov regularizer)	OpInf (tQR)
$\mathcal{E}$	$6.12 \cdot 10^{-5}$	$7.41 \cdot 10^{-5}$

91
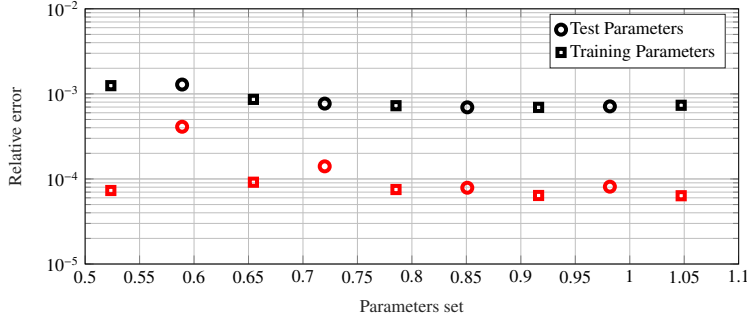
Figure 4.17: Single-layer shear instability: Relative error for testing and training parameters; (square): training set, (circle): testing set. (black): reduced dimension 20, (red): reduced dimension 65.

(a) FOM

(b) Inference

(c) FOM - inference error
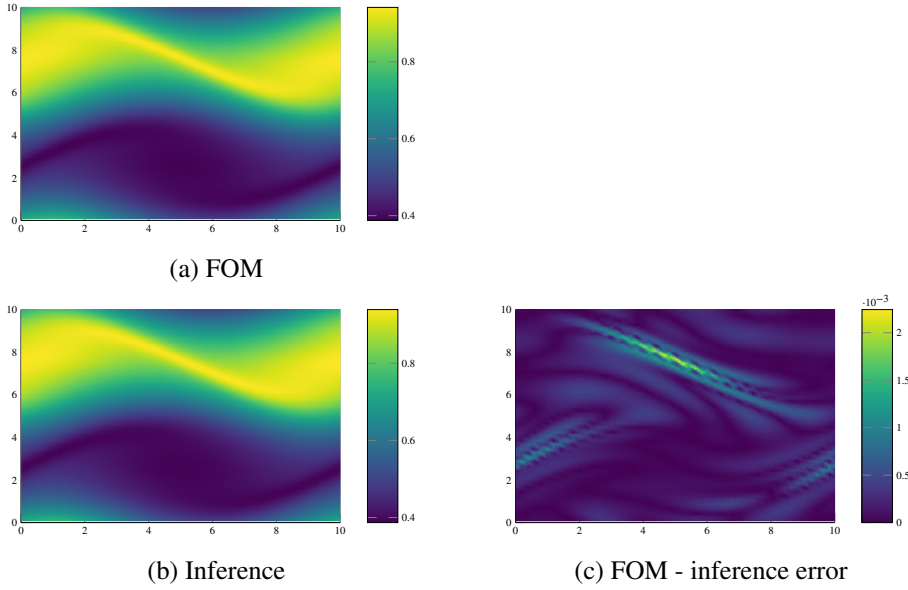
Figure 4.18: Single-layer shear instability: Comparison of the potential vorticity field for the parameter $\mu = \frac{5\pi}{24}$ obtained using the FOM and OpInf of order $r = 65$ at time $T = 30$.

captures the vorticity dynamics of the NTSWE accurately.

4.3 OpInf for RTSWE

In this subsection, we construct an intrusive and non-intrusive ROMs with the OpInf for the RTSWE (2.15) in the parametric form

$$\dot{\mathbf{w}}(\mu) = \mathbf{A}(\mu)\mathbf{w}(\mu) + \mathbf{H}(\mathbf{w} \otimes \mathbf{w}). \qquad (4.26)$$

Note that only the linear terms of the RTSWE depends on the parameter $\mu \in \mathcal{D} \subset \mathbb{R}^d$, whereas the quadratic terms are independent of the parameters.

92

To preserve the coupling structure in both intrusive and non-intrusive ROMs, we have computed the POD basis vectors separately for each the state vector $\mathbf{h}, \mathbf{u}, \mathbf{v}, \mathbf{s}$ as discussed in Section 3.4.

Let us define following trajectories

$$W_h(\mu) = \left[\mathbf{h}^1(\mu), \ldots, \mathbf{h}^K(\mu)\right], W_u(\mu) = \left[\mathbf{u}^1(\mu), \ldots, \mathbf{u}^K(\mu)\right],$$

$$W_v(\mu) = \left[\mathbf{v}^1(\mu), \ldots, \mathbf{v}^K(\mu)\right], W_s(\mu) = \left[\mathbf{s}^1(\mu), \ldots, \mathbf{s}^K(\mu)\right],$$

and the corresponding global snapshot matrices of the concatenated trajectories

$$W_{h\mu} = [W_h(\mu_1), \ldots, W_h(\mu_M)], W_{u\mu} = [W_u(\mu_1), \ldots, W_u(\mu_M)],$$

$$W_{v\mu} = [W_v(\mu_1), \ldots, W_v(\mu_M)], W_{s\mu} = [W_s(\mu_1), \ldots, W_s(\mu_M)].$$

for each state vector and parameter values $\mu_1, \ldots, \mu_M \in \mathcal{D}$.

The POD basis matrices V_h, V_u, V_v, V_s are obtained from the global snapshot matrices $W_{h\mu}, W_{u\mu}, W_{v\mu}, W_{s\mu} \in \mathbb{R}^{N \times MK}$ as in Section 3.4, respectively.

The POD coefficients $\tilde{\mathbf{h}}, \tilde{\mathbf{u}}, \tilde{\mathbf{v}}, \tilde{\mathbf{s}}$ are obtained by the POD matrices as

$$\begin{pmatrix} \mathbf{h} \\ \mathbf{u} \\ \mathbf{v} \\ \mathbf{s} \end{pmatrix} \approx \begin{pmatrix} V_h & & & \\ & V_u & & \\ & & V_v & \\ & & & V_s \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{h}} \\ \tilde{\mathbf{u}} \\ \tilde{\mathbf{v}} \\ \tilde{\mathbf{s}} \end{pmatrix}.$$

Although different number of POD modes can be used for each state, in this study we use equal number of POD modes, i.ei., $r = r_u = r_v = r_h = r_s$. We remark that the POD basis are independent of the parameter $\mu \in \mathcal{D}$.

4.3.1 Proper Orthogonal Decomposition with Galerkin Projection

The reduced-order operators are the determined by the employing Galerkin projection of the FOM (2.32) onto the subspace spanned by POD basis vectors $\Phi_r = diag(V_h, V_u, V_v, V_s)$.

$$\widetilde{\mathbf{A}}(\mu) = \Phi_r^T \mathbf{A}(\mu) \Phi_r \in \mathbb{R}^{r \times r}, \quad \widetilde{\mathbf{H}} = \Phi_r^T \mathbf{H}(\Phi_r \otimes \Phi_r) \in \mathbb{R}^{r \times r^2}. \tag{4.27}$$

The linear and quadratic operators can be constructed as follows

$$
\widetilde{\mathbf{A}}(\mu) = \Phi_r^T \mathbf{A}(\mu) \Phi_r = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & f(\mu) V_u^T V_v & -V_u^T (D_x \mathbf{b})^d V_s \\ 0 & -f(\mu) V_v^T V_u & 0 & -V_v^T (D_y \mathbf{b})^d V_s \\ 0 & 0 & 0 & 0 \end{pmatrix} \in \mathbb{R}^{4r \times 4r},
$$

$$
\widetilde{\mathbf{H}}_1 = \begin{pmatrix} 0 & & & \\ & \widetilde{\mathbf{H}}_1^1 & & \\ & & \widetilde{\mathbf{H}}_1^2 & \\ & & & 0 \end{pmatrix}, \quad \widetilde{\mathbf{H}}_2 = \begin{pmatrix} \widetilde{\mathbf{H}}_2^1 & & & \\ & \widetilde{\mathbf{H}}_2^2 & & \\ & & \widetilde{\mathbf{H}}_2^3 & \\ & & & \widetilde{\mathbf{H}}_2^4 \end{pmatrix},
$$

$$
\widetilde{\mathbf{H}}_4 = \begin{pmatrix} 0 & & & \\ & \widetilde{\mathbf{H}}_4^1 & & \\ & & \widetilde{\mathbf{H}}_4^2 & \\ & & & 0 \end{pmatrix}, \quad \widetilde{\mathbf{H}}_3 = \begin{pmatrix} \widetilde{\mathbf{H}}_3^1 & & & \\ & \widetilde{\mathbf{H}}_3^2 & & \\ & & \widetilde{\mathbf{H}}_3^3 & \\ & & & \widetilde{\mathbf{H}}_3^4 \end{pmatrix},
$$

with the $r \times r^2$ matrices

$$
\begin{aligned}
\widetilde{\mathbf{H}}_1^1 &= -V_u^\top Q(D_x V_u \otimes V_u), & \widetilde{\mathbf{H}}_1^2 &= -V_v^\top Q(D_x V_v \otimes V_u), \\
\widetilde{\mathbf{H}}_2^1 &= -V_h^\top D_x Q(V_h \otimes V_u), & \widetilde{\mathbf{H}}_2^2 &= -V_u^\top Q(V_v \otimes D_y V_u), \\
\widetilde{\mathbf{H}}_2^3 &= -V_v^\top Q(V_v \otimes D_y V_v), & \widetilde{\mathbf{H}}_2^4 &= -V_s^\top Q(D_x V_s \otimes V_u), \\
\widetilde{\mathbf{H}}_3^1 &= -V_h^\top D_y Q(V_h \otimes V_v), & \widetilde{\mathbf{H}}_3^2 &= -\frac{1}{2} V_u^\top Q(V_h \otimes D_x V_s), \\
\widetilde{\mathbf{H}}_3^3 &= -\frac{1}{2} V_v^\top Q(V_h \otimes D_y V_s), & \widetilde{\mathbf{H}}_3^4 &= -V_s^\top Q(D_y V_s \otimes V_v), \\
\widetilde{\mathbf{H}}_4^1 &= -V_u^\top Q(V_s \otimes D_x V_h), & \widetilde{\mathbf{H}}_4^2 &= -V_v^\top Q(V_s \otimes D_y V_h),
\end{aligned} \tag{4.28}
$$

where $Q \in \mathbb{R}^{N \times N^2}$ is the matricized tensor such that $Q(\mathbf{a} \otimes \mathbf{b}) = \mathbf{a} \circ \mathbf{b}$ is satisfied for any vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^N$. Computation of the reduced matricized tensors $\widetilde{\mathbf{H}}_i^j$ is done similar as done in the Section 3.4.

Using the reduced-order operators, resulting ROM can be written as

$$
\dot{\mathbf{w}}(t;\mu) = \widetilde{\mathbf{A}}(\mu)
\begin{bmatrix} 0 \\ \tilde{\mathbf{v}}(t;\mu) \\ \tilde{\mathbf{u}}(t;\mu) \\ 0 \end{bmatrix}
+ \widetilde{\mathbf{H}}_1
\begin{bmatrix} 0 \\ (\tilde{\mathbf{u}}(t;\mu) \otimes \tilde{\mathbf{v}}(t;\mu)) \\ (\tilde{\mathbf{v}}(t;\mu) \otimes \tilde{\mathbf{v}}(t;\mu)) \\ 0 \end{bmatrix}
+ \widetilde{\mathbf{H}}_2
\begin{bmatrix} (\tilde{\mathbf{h}}(t;\mu) \otimes \tilde{\mathbf{u}}(t;\mu)) \\ (\tilde{\mathbf{v}}(t;\mu) \otimes \tilde{\mathbf{u}}(t;\mu)) \\ (\tilde{\mathbf{v}}(t;\mu) \otimes \tilde{\mathbf{v}}(t;\mu)) \\ (\tilde{\mathbf{s}}(t;\mu) \otimes \tilde{\mathbf{u}}(t;\mu)) \end{bmatrix}
$$

$$
\text{(4.29)}
$$

$$
+ \widetilde{\mathbf{H}}_3
\begin{bmatrix} (\tilde{\mathbf{h}}(t;\mu) \otimes \tilde{\mathbf{v}}(t;\mu)) \\ (\tilde{\mathbf{h}}(t;\mu) \otimes \tilde{\mathbf{s}}(t;\mu)) \\ (\tilde{\mathbf{h}}(t;\mu) \otimes \tilde{\mathbf{s}}(t;\mu)) \\ (\tilde{\mathbf{s}}(t;\mu) \otimes \tilde{\mathbf{v}}(t;\mu)) \end{bmatrix}
+ \widetilde{\mathbf{H}}_4
\begin{bmatrix} 0 \\ (\tilde{\mathbf{s}}(t;\mu) \otimes \tilde{\mathbf{h}}(t;\mu)) \\ (\tilde{\mathbf{s}}(t;\mu) \otimes \tilde{\mathbf{h}}(t;\mu)) \\ 0 \end{bmatrix}
$$

which is solved in time by Kahan's method. Note that (4.29) preserves the linear-quadratic structure of the FOM (2.32).

4.3.2 Operator Inference (OpInf) with Re-Projection

In this subsection, we use the OpInf method [90, 89] to learn the reduced-order operators of the RTSWE (2.15) without accessing the full-order operators but only to trajectories and some information about FOM at the PDE level. In the OpInf framework, the main idea is to fit the reduced-order operators by the projected trajectories via least-squares regression. The projected trajectories are obtained by projecting full-order trajectories onto the low-dimensional subspaces spanned by POD basis. Nevertheless, by fitting reduced-order operators via projected trajectories, a closure error is introduced into the learnt operators, i.e., the operators can fail to capture the FOM dynamics in contrast to the intrusive operators. Although the high-dimensional trajectories are Markovian [32, 49], in the low-dimensional subspace, the reduced-order trajectories involve non-Markovian dynamics. The dynamical systems called Markovian when the future states depend only on the current state. On the other hand, non-Markovian systems can be considered as having a memory so that future states depend on the current and previous states. In [89] a data sampling method called re-projection is used to cancel the non-Markovian dynamics so that the trajectories in reduced space correspond to Markovian dynamics. The OpInf with re-projection exactly recovers the intrusive reduced-order operators. The re-projection scheme in [89]

iterates between the time stepping the FOM and projecting onto the low-dimensional subspaces to generate trajectories that correspond to low-dimensional Markovian dynamics. Under certain conditions, applying operator inference to re-projected trajectories gives the same operators that are constructed with the intrusive model reduction in the limit of $r \to N$.

Let us define the projected trajectories for each state as follows

$$\widehat{W}_h(\mu) = V_h^\top W_h(\mu) \in \mathbb{R}^{r \times K}, \quad \widehat{W}_u(\mu) = V_u^\top W_u(\mu) \in \mathbb{R}^{r \times K},$$
$$\widehat{W}_v(\mu) = V_v^\top W_v(\mu) \in \mathbb{R}^{r \times K}, \quad \widehat{W}_s(\mu) = V_s^\top W_s(\mu) \in \mathbb{R}^{r \times K}.$$

In addition, we define the time derivatives and their projections as

$$\dot{W}_h(\mu) = \left[\dot{\mathbf{h}}^1(\mu), \dots, \dot{\mathbf{h}}^K(\mu) \right], \quad \dot{W}_u(\mu) = \left[\dot{\mathbf{u}}^1(\mu), \dots, \dot{\mathbf{u}}^K(\mu) \right],$$
$$\dot{W}_v(\mu) = \left[\dot{\mathbf{v}}^1(\mu), \dots, \dot{\mathbf{v}}^K(\mu) \right], \quad \dot{W}_s(\mu) = \left[\dot{\mathbf{s}}^1(\mu), \dots, \dot{\mathbf{s}}^K(\mu) \right],$$
$$\dot{\widehat{W}}_h(\mu) = V_h^\top \dot{W}_u(\mu), \quad \dot{\widehat{W}}_u(\mu) = V_u^\top \dot{W}_u(\mu),$$
$$\dot{\widehat{W}}_v(\mu) = V_v^\top \dot{W}_v(\mu), \quad \dot{\widehat{W}}_s(\mu) = V_s^\top \dot{W}_s(\mu).$$

The time derivatives can be computed by using finite differences approximation [82] or by evaluation of a right-hand side of the RTSWE (2.32). We have used right-hand side of the RTSWE (2.32) to compute the time derivatives.

Finally, the reduced operators can be learned by solving following least-squares problems for each state separately as follows

$$\min_{\mathcal{X}_j} \sum_{k=1}^M \left\| \mathcal{A}_j(\mu_k)\mathcal{X}_j^\top - \dot{\widehat{W}}_j(\mu_k)^\top \right\|_F^2, \quad j = h, u, v, s, \quad (4.30)$$

with the data matrices

$$\mathcal{A}_h(\mu_k) = [\mathcal{A}_{11}, \mathcal{A}_{12}] \in \mathbb{R}^{K \times 2r^2},$$
$$\mathcal{A}_u(\mu_k) = [\mathcal{A}_{21}, \mathcal{A}_{22}, \mathcal{A}_{23}, \mathcal{A}_{24}] \in \mathbb{R}^{K \times 3r^2 + r},$$
$$\mathcal{A}_s(\mu_k) = [\mathcal{A}_{31}, \mathcal{A}_{32}] \in \mathbb{R}^{K \times 2r^2},$$
$$\mathcal{A}_v(\mu_k) = [\mathcal{A}_{41}, \mathcal{A}_{42}, \mathcal{A}_{43}, \mathcal{A}_{44}] \in \mathbb{R}^{K \times 3r^2 + r},$$

consist of the matrices

$$\mathcal{A}_{11} = (\widehat{W}_h(\mu_k)\hat{\otimes}\widehat{W}_u(\mu_k))^\top, \quad \mathcal{A}_{12} = (\widehat{W}_h(\mu_k)\hat{\otimes}\widehat{W}_v(\mu_k))^\top,$$

$$\mathcal{A}_{21} = (\widehat{W}_u(\mu_k)\hat{\otimes}\widehat{W}_u(\mu_k))^\top, \quad \mathcal{A}_{22} = (\widehat{W}_v(\mu_k)\hat{\otimes}\widehat{W}_u(\mu_k))^\top,$$

$$\mathcal{A}_{23} = (\widehat{W}_h(\mu_k)\hat{\otimes}\widehat{W}_s(\mu_k))^\top, \quad \mathcal{A}_{24} = f(\mu_k)\widehat{W}_v(\mu_k)^\top,$$

$$\mathcal{A}_{31} = (\widehat{W}_u(\mu_k)\hat{\otimes}\widehat{W}_s(\mu_k))^\top, \quad \mathcal{A}_{32} = (\widehat{W}_v(\mu_k)\hat{\otimes}\widehat{W}_s(\mu_k))^\top,$$

$$\mathcal{A}_{41} = (\widehat{W}_u(\mu_k)\hat{\otimes}\widehat{W}_v(\mu_k))^\top, \quad \mathcal{A}_{42} = (\widehat{W}_v(\mu_k)\hat{\otimes}\widehat{W}_v(\mu_k))^\top,$$

$$\mathcal{A}_{43} = (\widehat{W}_h(\mu_k)\hat{\otimes}\widehat{W}_s(\mu_k))^\top, \quad \mathcal{A}_{44} = f(\mu_k)\widehat{W}_u(\mu_k)^\top,$$

and with the learned operators

$$\mathcal{X}_h = \left[\widehat{\mathcal{H}}_{1,1}, \widehat{\mathcal{H}}_{1,2}\right] \in \mathbb{R}^{2r^2 \times r}, \quad \mathcal{X}_u = \left[\widehat{\mathcal{H}}_{2,1}, \widehat{\mathcal{H}}_{2,2}, \widehat{\mathcal{H}}_{2,3}, \widehat{\mathcal{L}}_1(\mu)\right] \in \mathbb{R}^{3r^2+r \times r} \quad (4.31)$$

$$\mathcal{X}_s = \left[\widehat{\mathcal{H}}_{4,1}, \widehat{\mathcal{H}}_{4,2}\right] \in \mathbb{R}^{2r^2 \times r}, \quad \mathcal{X}_v = \left[\widehat{\mathcal{H}}_{3,1}, \widehat{\mathcal{H}}_{3,2}, \widehat{\mathcal{H}}_{3,3}, \widehat{\mathcal{L}}_2(\mu)\right] \in \mathbb{R}^{3r^2+r \times r} \quad (4.32)$$

. The least-squares problem (4.30) can be written in standard form as

$$\min_{\mathcal{X}_j} \left\|\mathcal{A}_{\mu j}\mathcal{X}_j^\top - \dot{\widehat{W}}_{\mu j}\right\|_F^2, \quad j = h, u, v, s \quad (4.33)$$

where $\dot{\widehat{W}}_{\mu j} = \left[\dot{\widehat{W}}_j^T(\mu_1), \dots, \dot{\widehat{W}}_j^T(\mu_M)\right]^T$ and $\mathcal{A}_{\mu j} = \left[\mathcal{A}_j^T(\mu_1), \dots, \mathcal{A}_j^T(\mu_M)\right]^T$ for $j = h, u, v, s$.

Note that each of the least-squares problems (4.30) are independent and are solved separately to improve the computational efficiency [90].

Substituting the recovered operators, the non-intrusive ROM becomes

$$\dot{\widehat{\mathbf{h}}}(t;\mu) = \widehat{\mathcal{H}}_{1,1}(\widehat{\mathbf{h}}(t;\mu) \otimes \widehat{\mathbf{u}}(t;\mu)) + \widehat{\mathcal{H}}_{1,2}(\widehat{\mathbf{h}}(t;\mu) \otimes \widehat{\mathbf{v}}(t;\mu)), \quad (4.34a)$$

$$\dot{\widehat{\mathbf{u}}}(t;\mu) = \widehat{\mathcal{H}}_{2,1}(\widehat{\mathbf{u}}(t;\mu) \otimes \widehat{\mathbf{u}}(t;\mu)) + \widehat{\mathcal{H}}_{2,2}(\widehat{\mathbf{v}}(t;\mu) \otimes \widehat{\mathbf{u}}(t;\mu)) \quad (4.34b)$$

$$+ \widehat{\mathcal{H}}_{2,3}(\widehat{\mathbf{h}}(t;\mu) \otimes \widehat{\mathbf{s}}(t;\mu)) + f(\mu)\widehat{\mathcal{L}}_1\widehat{\mathbf{v}}(t;\mu), \quad (4.34c)$$

$$\dot{\widehat{\mathbf{v}}}(t;\mu) = \widehat{\mathcal{H}}_{3,1}(\widehat{\mathbf{u}}(t;\mu) \otimes \widehat{\mathbf{v}}(t;\mu)) + \widehat{\mathcal{H}}_{3,2}(\widehat{\mathbf{v}}(t;\mu) \otimes \widehat{\mathbf{v}}(t;\mu)) \quad (4.34d)$$

$$+ \widehat{\mathcal{H}}_{3,3}(\widehat{\mathbf{h}}(t;\mu) \otimes \widehat{\mathbf{s}}(t;\mu)) + f(\mu)\widehat{\mathcal{L}}_2\widehat{\mathbf{u}}(t;\mu), \quad (4.34e)$$

$$\dot{\widehat{\mathbf{s}}}(t;\mu) = \widehat{\mathcal{H}}_{4,1}(\widehat{\mathbf{u}}(t;\mu) \otimes \widehat{\mathbf{s}}(t;\mu)) + \widehat{\mathcal{H}}_{4,2}(\widehat{\mathbf{v}}(t;\mu) \otimes \widehat{\mathbf{s}}(t;\mu)). \quad (4.34f)$$

We have summarized data sampling by re-projection [89] in Algorithm (3) by preserving the coupling structure of the ROM as done in [95]

97

Algorithm 3 Data sampling via re-projection

1: **procedure** RE-PROJECTION
2:     **Input:** States $\mathbf{w}^j$ $j = 1, \ldots, K$ and POD basis matrix Φ_r.
3:     **for** $j = 1, \ldots, K$ **do**
4:         Set $\mathbf{w}^j_{\text{proj}} = (\mathbf{h}^j_{\text{proj}}, \mathbf{u}^j_{\text{proj}}, \mathbf{v}^j_{\text{proj}}, \mathbf{s}^j_{\text{proj}})^T = \Phi_r \Phi_r^T \mathbf{w}^j$.
5:         Re-project the time derivatives $\dot{\overline{\mathbf{w}}}^j = \Phi_r^T \mathbf{F}(\mathbf{w}^j_{\text{proj}})$ in (2.33).
6:         Re-project the states $\overline{\mathbf{w}}^j = \Phi_r^T \mathbf{w}^j_{\text{proj}}$.
7:     **end for**
8: **return** $\dot{\overline{W}} = \left[\dot{\overline{\mathbf{w}}}^1, \ldots, \dot{\overline{\mathbf{w}}}^K\right]$ and $\overline{W} = \left[\overline{\mathbf{w}}^1, \ldots, \overline{\mathbf{w}}^K\right]$.
9: **end procedure**

Algorithm 4 Operator inference with re-projection for parametric RTSW.

1: **procedure** OPINF
2:     **Input:** States $\mathbf{w}_j(\mu_i)$ and time derivatives $\dot{\mathbf{w}}_j(\mu_i)$ for $i = 1, \ldots, M$ and $j = 1, \ldots, K$.
3:     Construct the trajectories for each state

$$W_h(\mu) = [\mathbf{h}_1(\mu), \ldots, \mathbf{h}_K(\mu)], \quad W_u(\mu) = [\mathbf{u}_1(\mu), \ldots, \mathbf{u}_K(\mu)],$$
$$W_v(\mu) = [\mathbf{v}_1(\mu), \ldots, \mathbf{v}_K(\mu)], \quad W_s(\mu) = [\mathbf{s}_1(\mu), \ldots, \mathbf{s}_K(\mu)].$$

4:     Construct the global snapshot matrices $W_{j\mu} = [W_j(\mu_1), \ldots, W_j(\mu_M)]$ for $j = h, u, v, s$.
5:     Compute the global POD basis Φ_{r_j} of $W_{j\mu}$ for $j = h, u, v, s$.
6:     Sample the data via re-projection and set $\widehat{W}_j = \overline{W}_j$ and $\dot{\widehat{W}}_j = \dot{\overline{W}}_j$ for $j = h, u, v, s$.
7:     Determine the tolerance of **lsqminnorm** tol, using L-curve formula.
8:     Solve the least-squares problem (4.30) to obtain operators of the reduced-order system using $\mathcal{X}_j^T = $**lsqminnorm**.$(\mathcal{A}_{\mu j}, \dot{\widehat{W}}_{\mu j}, tol)$.
9: **end procedure**

The uniqueness of the solution of the least-squares problem (4.33) plays a crucial role in the OpInf method, which is obtained by enforcing the system to be overdetermined, i.e., the number of the time steps should satisfy $K \geq r + r^2$ for linear-quadratic PDEs like the RTSWE (see Corollary 3.2 in [89]) and the data matrices $\mathcal{A}_j, \; j = h, u, v, s$

must have full column rank. If this condition is not satisfied, the least-squares problem (4.33) becomes underdetermined, and the uniqueness of the solution is lost. This issue can be overcome by picking the minimum norm of the solution. The minimum-norm solution enforces uniqueness of the solution by picking smallest $\|\mathcal{X}_j\|_F$ which minimizing the $\|\mathcal{A}_{\mu j}\mathcal{X}_j^\top - \widehat{\dot{W}}_{\mu j}\|_F$. To obtain the minimum-norm solution, one can use the Moore-Penrose pseudoinverse or complete orthogonal decomposition (COD) formulas. We use here MATLAB's routine lsqminnorm as regularizer of the least-squares problem (4.30) to achieve the uniqueness. An ill-conditioned matrix as either a matrix with a well-determined numerical rank or an ill-determined numerical rank, depending on the behavior of the singular value spectrum [59]. Usually, the L-curve criterion [60] is used to determine the tolerance at which the singular values are truncated. Determining regularization tolerance by the L-curve, a good compromise between matching the accurate solution and making the problem well-conditioned is obtained.

Once the reduce operators are recovered from the OpInf method, any time stepping scheme can be used to solve the ROM with the recovered operators. Here, we use Kahan's method as the time integrator for (4.34), because it is cheap, i.e., linearly implicit and preserves the conserved quantities in contrast to the implicit or explicit Euler methods.

The costs of Algorithm (4) are bounded linearly, in the FOM dimension N and the number of time steps K. All the lines in Algorithm (4) have a cost that is independent of the FOM dimension N. Further cost occurs for obtaining the re-projected time derivatives and trajectories. Overall, the computational cost of the Algorithm (4) with re-projection is twice that of OpInf without re-projection.

There is a trade-off between the ROM error and the condition number of the data matrices in increasing reduced dimension r, which decrease the ROM error yet increases the condition number of the data matrix. By sampling in a subset of trajectories, the condition number of the data matrix can be decreased, and the computational efficiency can be increased, i.e. taking every kth snapshot in the data matrix. Another strategy to deal with the condition number is to solve the FOM with different initial

values [90].

4.3.3 Numerical Results

In this subsection, we compare the computational efficiency, accuracy, and the prediction skills of the POD and OpInf methods for the double vortex RTSWE [43] in the parametric and in the non-parametric formin a doubly periodic domain $\Omega = [0, L]^2$ with no bottom topography $(b = 0)$. The initial conditions are given in Section (3.5.4).

Non-parametric case

In the non-parametric case, we consider the Coriolis parameter as $f = 0.00006147\text{s}^{-1}$, which corresponds at the latitude 7, close to equator. The number of time steps fixed to $K = 250$ with the time step-size $\Delta t = 486\text{s}$, that results to the final time $T = 33\text{h}$ 45min [43]. The snapshot matrices are of size 14440×250.

The accuracy of both the intrusive and non-intrusive models are measured using the relative error of the four states

$$\frac{||\Phi_r Z - W||_F}{||W||_F}, \tag{4.35}$$

where $W \in \mathbb{R}^{4N \times K}$ is the snapshot matrix of the FOM and $Z \in \mathbb{R}^{4r \times K}$ is the snapshot matrix of either the intrusive or the non-intrusive model. The average relative errors between ROMs and FOM solutions are determined for each state variable $\boldsymbol{w} = \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{h}, \boldsymbol{s}$ in the time-averaged $L^2(\Omega)$-norms (3.22).

The preservation of discrete quantities (2.27): the energy, buoyancy, mass, and total vorticity obtained from the FOM and ROM solutions are examined using the time-averaged relative error (3.21) for $E \equiv H, M, B, Q$.

First, we show how the regularization tolerance tol in Algorithm 4 is obtained for the regularized least-squares problem (4.33). In Figure 4.19, the singular values of the data matrices decay without showing a gap. Consequently, the L-curve does not give any useful information to determine the regularization tolerances.

The regularization tolerances for the routine **lsqminnorm** are obtained by the L-
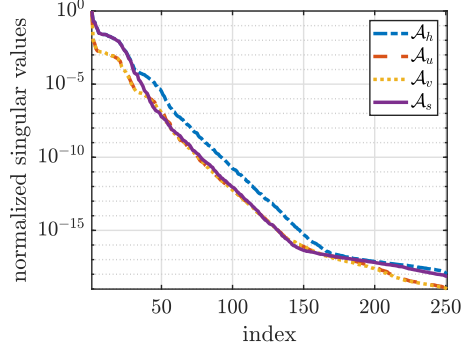
Figure 4.19: Normalized singular values of the data matrices for the reduced dimension $r = 20$.

curves of the data matrices in Figure 4.20 as $1e - 6$ for $j = h, v$ and $1e - 7$ for $j = u, s$.
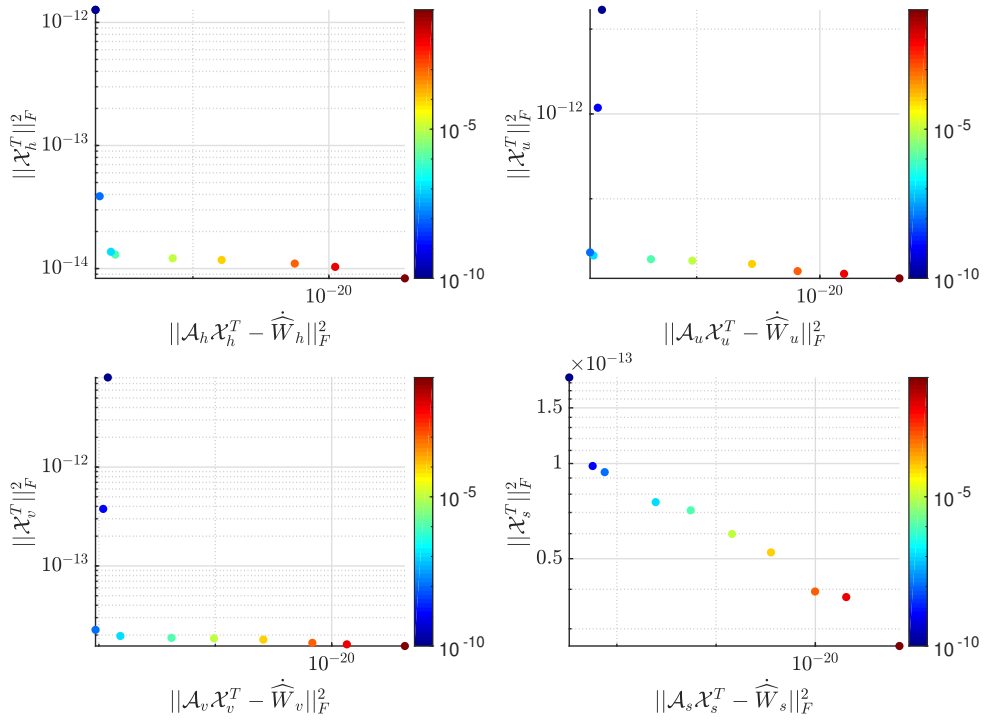

Figure 4.20: L-curves for reduced dimension $r = 20$.

In Figure 4.21, the singular values decay comparatively slow, which is a common characteristic property for the complex fluid dynamic problems with transport and phenomena like the SWEs [88].
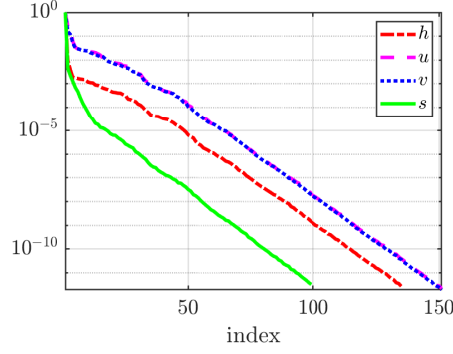
Figure 4.21: Normalized singular values of the snapshot matrices.

In Figure 4.22, the relative errors linked to both the intrusive and non-intrusive (4.35) of the four trajectories are plotted, which shows the errors obtained from both intrusive and non-intrusive models showed similar performance, i.e. trajectories are very close. By increasing dimension r, the POD and OpInf errors decrease in the same manner. For reduced dimension $r = 20$, the average relative errors (3.22) regarding the intrusive POD and non-intrusive OpInf methods are $1.499e - 03$ and $1.485e - 03$, respectively, which indicates that both intrusive and non-intrusive model captures the FOM dynamics with the same level of accuracy.
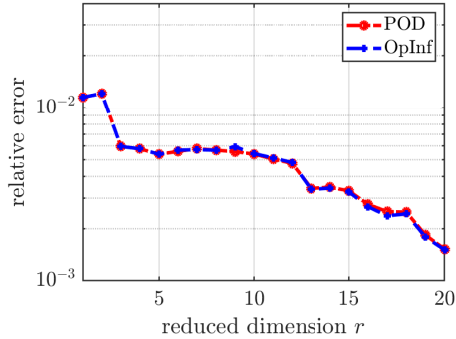
Figure 4.22: Relative errors (4.35) of the POD and OpInf.

We compare the efficiency of intrusive and non-intrusive ROMs by the total computational times in Figure 4.23, where we don't include the computation time for determining the POD basis because it is the same for the POD and OpInf methods. The computational time for the POD method consists of the online computation time and the elapsed time for determining the reduced tensors (4.28), whereas in the OpInf method, it consists of the online computation, elapsed time of determining the re-projected time derivatives and states, and solving the least-squares problem (4.30).

102

Due to the re-projection, the computational cost of the OpInf is larger than the POD. The total computational times for the POD and OpInf method are 1.044s, 2.045s, respectively. In Figure 4.23, we show the elapsed computation for ROMs, which display a speed-up of order 10^2 over the FOM with 585.129s computing time .
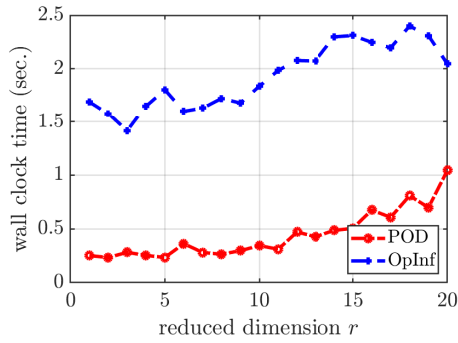
Figure 4.23: Elapsed total computational time of the ROMs.

Both intrusive and non-intrusive models of dimension $r = 20$ regenerate the FOM trajectories accurately at the final time in Figure 4.24. In Figure 4.25, the relative errors in the mass $|M^k - M^0|/|M^0|$, Hamiltonian (energy) $|H^k - H^0|/|H^0|$, total potential vorticity $|Q^k - Q^0|/|Q^0|$, and the buoyancy $|B^k - B^0|/|B^0|$ are shown. The total potential vorticity and total mass are preserved up to machine precision. Both ROMs' energy and buoyancy errors exhibit bounded oscillations over time without any drift, i.e., they are preserved well in the long term. Conservation of the energy and Casimir's by the ROMs is demonstrated in Table 4.5 in terms of the relative errors (3.21).
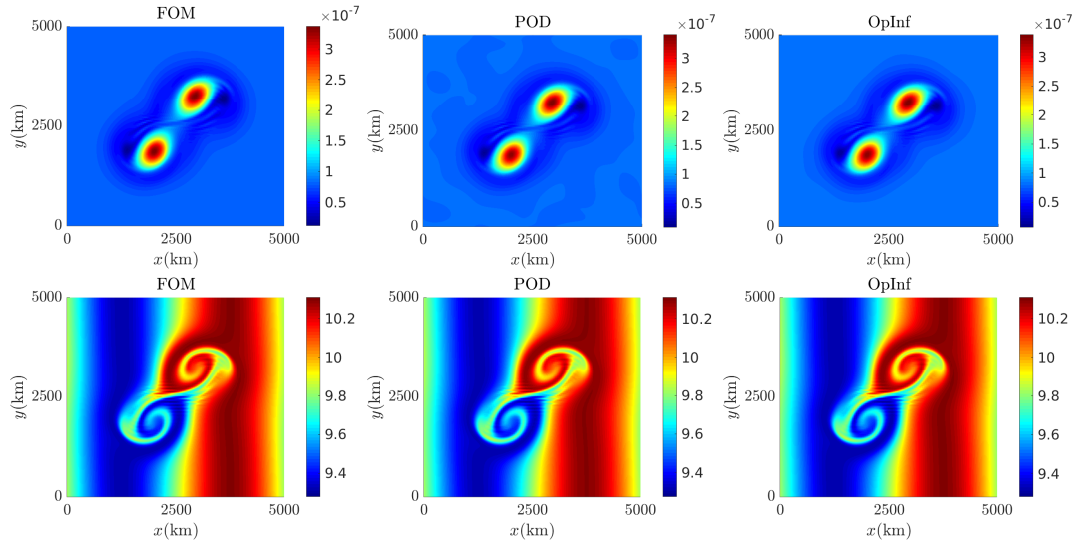
Figure 4.24: (Top) Vorticity and (bottom) bouyancy of the FOM and ROMs at the final time.
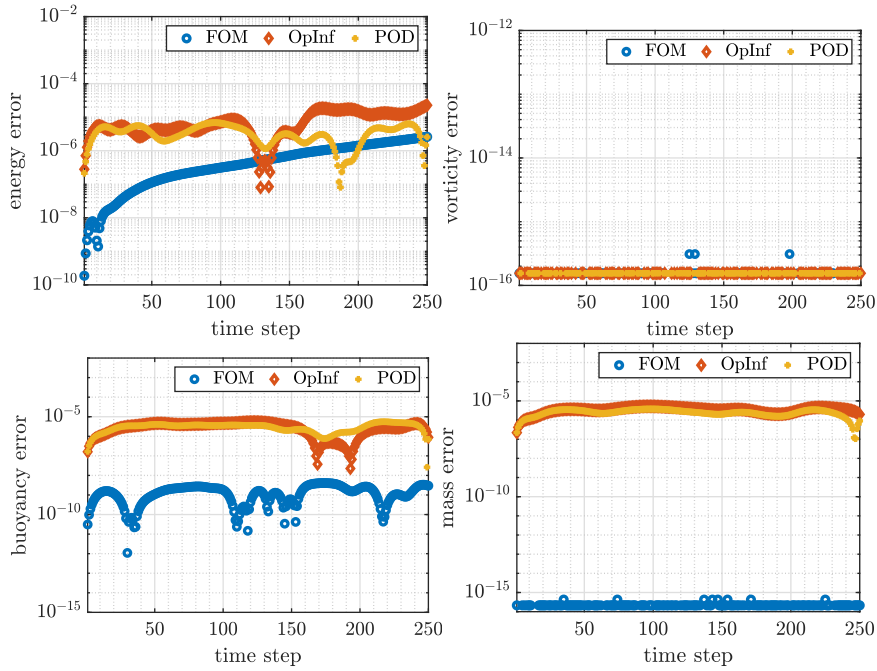
Figure 4.25: Relative errors in the conserved quantities.

Table 4.5: Time-averaged relative errors (3.21) of the conserved quantities.

	Energy	Total Vorticity	Mass	Buoyancy
FOM	7.484e-07	6.392e-17	1.545e-16	1.567e-09
POD	3.489e-06	1.024e-16	2.489e-06	3.053e-06
OpInf	8.114e-06	1.018e-16	3.440e-06	3.050e-06

Forecasting the future dynamics of complex systems is studied with the non-intrusive

104

and intrusive models for the single-injector combustor [83], the quasi-geostrophic equations [86], and the shallow water equations [3]. Here, we investigate the predictive performances of both intrusive and non-intrusive ROM approaches for the RTSWE. Let us define the relative FOM-ROM error at the time step k as

$$\frac{\|\boldsymbol{w}^k - \boldsymbol{w}_r^k\|_{L^2}}{\|\boldsymbol{w}^k\|_{L^2}}, \tag{4.36}$$

where the state vector defined as $\boldsymbol{w} = (\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{h}, \boldsymbol{s})^T$.

In Figures 4.26-4.27, the relative errors (4.36) obtained from the intrusive and non-intrusive models are plotted, where the vertical blue lines indicate the separation of the training and the prediction intervals. We investigate the prediction performance in two cases; when training periods longer than the prediction periods and vice versa. These cases are demonstrated in Figures 4.26 and 4.27. Figure 4.27 shows that when the number of the snapshot in the training data becomes larger than the prediction regime, both ROM predictions improve. On the other hand, increasing ROMs' dimension does not significantly affect the ROM predictions in Figures 4.26-4.27. In both intrusive and non-intrusive models, the ROMs can accurately regenerate the training data and capture much of the overall system behaviour in the prediction period. A summary of the review of ROMs in terms of the averaged relative errors (3.22) is given in Table 4.6. Figures 4.26-4.27, and Table 4.6 show that the extrapolatory predictive performance of both ROMS is not improved with increasing number of reduced dimensions.
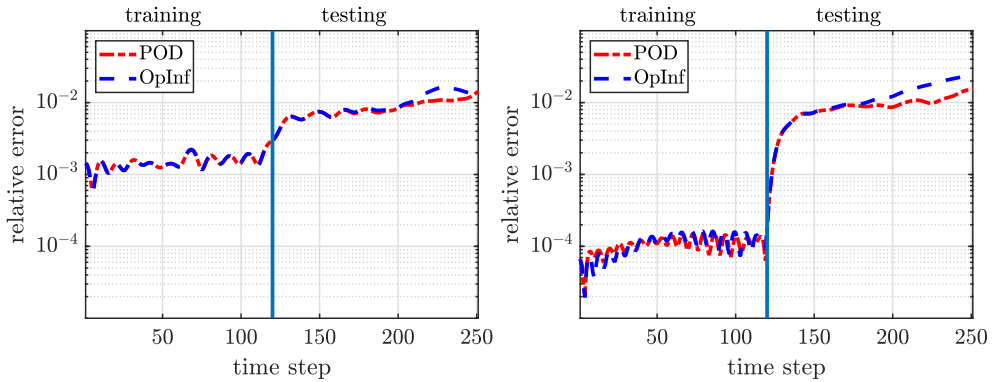

Figure 4.26: Prediction performance of the ROMs trained up to $K = 120$ with the reduced dimension (left) $r = 10$ and (right) $r = 20$.
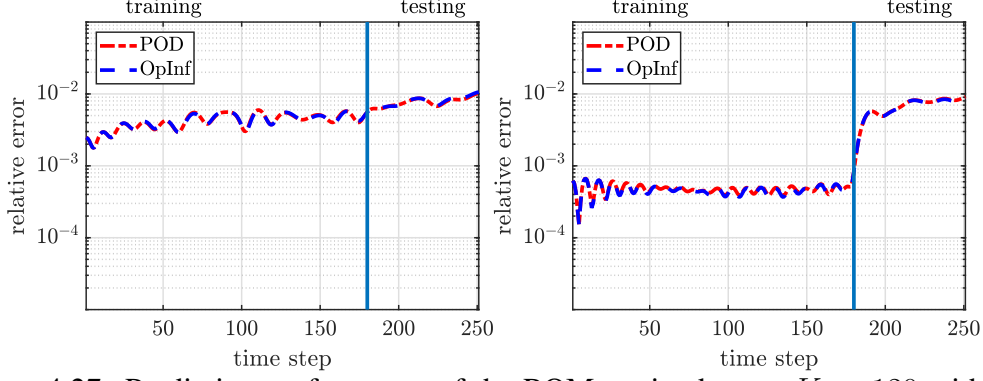
Figure 4.27: Prediction performance of the ROMs trained up to $K = 120$ with the reduced dimension (left) $r = 10$ and (right) $r = 20$.

Table 4.6: Average relative errors (3.22) for training and prediction sets.

		$r = 10$		$r = 20$	
		POD	OpInf	POD	OpInf
$K = 120$	training	1.529e-03	1.523e-03	1.060e-04	1.106e-04
	prediction	8.299e-03	9.487e-03	8.769e-03	1.193e-02
$K = 180$	training	4.250e-03	4.233e-03	4.737e-04	4.637e-04
	prediction	7.691e-03	7.817e-03	6.695e-03	6.587e-03

Parametric case

In the last example, we assume that the Coriolis parameter $f(\mu) = 2\Omega \sin(\mu)$ varying with the latitude μ in the parameter domain $\mathcal{D} := 4\pi/18 \leq \mu \leq 8\pi/18$, i.e., between the 40^0th to 80^0th latitude. We consider a larger time interval, with the number of time steps $K = 300$. The average relative errors in the training phase are computed as

$$\frac{1}{M_{train}} \sum_{i=1}^{M_{train}} \frac{\|\mathbf{\Phi}_r Z(\mu_i^{train}) - W(\mu_i^{train})\|_F}{\|W(\mu_i^{train})\|_F}, \tag{4.37}$$

where $W(\mu_i^{train})$ is the FOM trajectory and $\mathbf{Z}(\mu_i^{train})$ is the trajectory of either obtained by the intrusive reduced model (POD) or recovered model (OpInf) from the re-projected trajectories. Similarly, the accuracy of the parametric model for the test case is measured via the average relative error defined by

$$\frac{1}{M^{test}} \sum_{i=1}^{M_{test}} \frac{\|\mathbf{\Phi}_r Z(\mu_i^{test}) - W(\mu_i^{test})\|_F}{\|W(\mu_i^{test})\|_F}. \tag{4.38}$$

To construct the intrusive and non-intrusive ROMs, we considered $M_{train} = 6$ equidistant parameters $\mu_1^{train}, \ldots, \mu_{M_{train}}^{train} \in \mathcal{D}$ in the parameter domain $\mathcal{D}$. Furthermore, to

decrease the condition number of data matrices, we make the initial conditions randomly perturbed around the centre of the vortices as $oy = 0.1 + \gamma$, where γ is a uniformly distributed random value. Considering the random initial conditions does not necessarily have a physical meaning. However, they show the FOM behaviour over different trajectories to provide complete information about the FOM and decrease the condition number of the data matrices in the OpInf. The least-squares problem (4.30) is again solved with **lsqminnorm** algorithm with the tolerance $tol = 1e - 10$, and by determining subset of the snapshots, i.e., every 2nd snapshot.

In the testing period, we consider $M_{\text{test}} = 7$ randomly distributed parameter values $\mu_1^{test} \ldots, \mu_{M_{test}}^{test} \in \mathcal{D}$. In Figure 4.28, we show the relative errors (4.37) and (4.38) for the training and testing regimes as the dimension r is increasing, which shows that both intrusive and non-intrusive models act likewise, yielding more accurate solutions with the increasing reduced dimension. Figure 4.29 demonstrates that the potential vorticity and buoyancy at the final time for dimension $r = 10$ and the Coriolis parameter μ_1^{test}, which exhibits accurate behavior for the intrusive and non-intrusive ROMs.
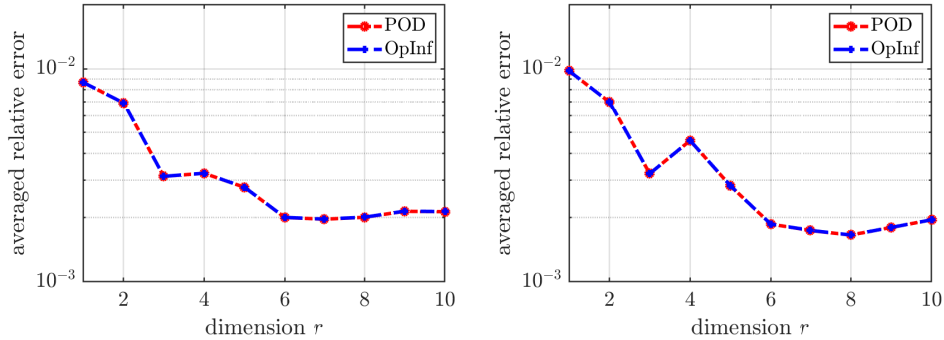

Figure 4.28: (Left) The relative error (4.37) in the training period, (right) the relative error (4.38) in the testing period.
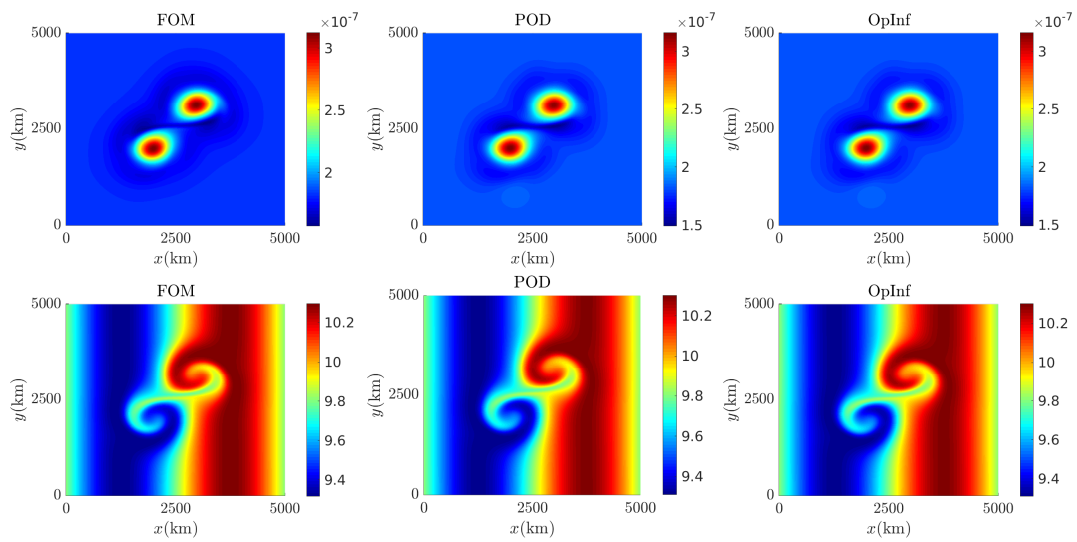
Figure 4.29: (Top) Vorticity and (bottom) bouyancy of the FOM and ROMs at the final time.

CHAPTER 5

SUMMARY & CONCLUSIONS

In this thesis, intrusive and data-driven non-intrusive MOR techniques are studies and developed for the rotating SWEs. The reduced system inherits the structure of the FOM, such as the skew-symmetry and conservation of the Hamiltonian and conserved quantities. Conserving the structure of the SWEs not only results in a physically meaningful reduced model but also provides robust long-time behavior and a stable ROM. The numerical experiments in this thesis illustrate that the proposed methods consistently result in a robust reduced system. Even when highly accurate FOMs are chosen, traditional model reduction techniques may yield an unstable or poorly performed reduced system. Numerical experiments confirm that the conservation of the Hamiltonian structure of the rotating SWEs can significantly enhance the overall dynamics of the reduced system. Furthermore, data-driven ROMs can predict the system's behavior beyond the time horizon and for different parameter values in the test period for the parameterized rotating SWEs.

In Chapter 2, two different kinds of FOMs are considered by preserving the Hamiltonian structure: linear-quadratic systems of ODEs and a skew-gradient system. There are geometric integrators for specific non-canonical Hamiltonian systems, such as the rotating SWEs, unlike the symplectic integrators for the canonical Hamiltonian systems. It is also not possible to preserve the Poisson or symplectic structure and the energy of the dynamical systems at the same time. Therefore, both types of FOMs are integrated with the energy (Hamiltonian) preserving integrators; the linearly implicit Kahan's method, and the fully implicit AVF method. Both integrators preserve the physical quantities of the SWEs well, like the energy, enstrophy, mass, buoyancy, and

vorticity in long-time integration.

In Chapter 3, traditional intrusive ROM techniques, POD, and DEIM with Galerkin projection are applied to the rotating SWEs. Nonlinear terms present a difficulty for ROM from an efficiency standpoint. If they are naively implemented, then the reduced model will not be independent of the FOM's dimension. For polynomial nonlinearities, tensor methods can be used, where nonlinear terms are pre-assembled as tensors. The degree of the polynomial nonlinearity determines the order of the tensor. Thus the complexity of the ROM grows with the degree of the polynomial nonlinearity. In Section 3.4 by utilizing the linear-quadratic structure of the RSWE and tensor techniques, the offline and online phase is separated, which leads to an efficient online computation time of ROMs. For more general nonlinearities, hyper-reduction techniques are required to avoid the growth in complexity from nonlinear terms. For a skew-gradient system with a constant Poisson structure like the KdV equation, the skew-symmetric structure is preserved using the ROMs. Preservation of the state-dependent Poisson structure is more challenging. Applying DEIM to the skew-symmetric Poisson matrix and the gradient of the Hamiltonian, ROMs are constructed for the RTSWE. The offline cost further decreased using tensor techniques.

In Chapter 4, data-driven projection-based ROMs of the NTSWE and RTSWE are constructed in the Markovian and non-Markovian OpInf framework. Since the least-squares problem of the OpInf method may suffer from ill-conditioning, the solutions are regularized using the QR decomposition and with the minimum norm solution as an alternative to Tikhonov regularization. The performance of the inferred models is examined in terms of prediction capabilities. Numerical results in the parametric and non-parametric settings show that both intrusive and data-driven ROMs behave similarly and yield similar reduced solutions. This validates the convergence of the learned operators to the intrusively obtained reduced operators under certain conditions.

In summary, we have shown with a relatively small number of reduced modes, stable, accurate, and fast reduced solutions are obtained; energy and other conserved quantities are well preserved over long-time integration. The reduced system can be identified by reduced energy and reduced conserved quantities that mimic high-fidelity

systems. This results in an overall, correct evolution of the solutions that ensure the robustness of the reduced system. Both reduced models can accurately re-predict the training data and capture much of the overall system behavior in the prediction period. Due to re-projection, the OpInf is more costly than the POD. Nevertheless, speed-up factors of order two are obtained by both ROMs. Moreover, preservation of the conserved quantities by both ROMs indicates the stability of the reduced solutions in long-time integration, which is important for Hamiltonian PDEs like the RTSWE. It also shows the importance of respecting the physics of complex problems in ROM application.

5.1   Future Research Perspectives

Reduced-order modeling is a very active research topic. Below are some research directions that are listed related to the SWEs.

- Using the lifting technique, the potential vorticity term can be treated as a constraint, and an auxiliary equation is added to the system. This particular treatment allows that the nonlinear terms in the skew-symmetric Poisson matrix will be linear, and the SWEs in Hamiltonian form contains only polynomial nonlinearities.

- The methodology in this thesis can be extended to the shallow water magnetohydrodynamic equation, which has a similar Hamiltonian structure as the RTSWE.

- Both intrusive and non-intrusive techniques can be compared with machine learning techniques such as symplectic and Hamiltonian neural networks.

- SWEs can also be presented as hyperbolic systems with conservation laws. ROM techniques can be developed to preserve energy and entropy.

REFERENCES

[1] B. M. Afkham and J. S. Hesthaven. Structure preserving model reduction of parametric Hamiltonian systems. *SIAM Journal on Scientific Computing*, 39(6):A2616–A2644, 2017.

[2] B. M. Afkham, N. Ripamonti, Q. Wang, and J. S. Hesthaven. Conservative model order reduction for fluid flow. In D'Elia M., Gunzburger M., and Rozza G., editors, *Quantification of Uncertainty: Improving Efficiency and Technology*, volume 137 of *Lecture Notes in Computational Science and Engineering*, pages 67–99. Springer, Cham, 2020.

[3] S. E. Ahmed, O. San, D. A. Bistrian, and I. M. Navon. Sampling and resolution characteristics in reduced order models of shallow water equations: Intrusive vs nonintrusive. *International Journal for Numerical Methods in Fluids*, 92(8):992–1036, 2020.

[4] A. Arakawa and V. R. Lamb. A potential enstrophy and energy conserving scheme for the shallow water equations. *Monthly Weather Review*, 109(1):18–36, 1981.

[5] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251, 2008.

[6] C. Bach, D. Ceglia, L. Song, and F. Duddeck. Randomized low-rank approximation methods for projection-based model order reduction of large nonlinear dynamical problems. *International Journal for Numerical Methods in Engineering*, 118(4):209–241, 2019.

[7] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[8] W. Bauer and C. J. Cotter. Energy-enstrophy conserving compatible finite element schemes for the rotating shallow water equations with slip boundary conditions. *Journal of Computational Physics*, 373:171 – 187, 2018.

[9] E. Belanger and A. Vincent. Data assimilation (4d-var) to forecast flood in shallow-waters with sediment erosion. *Journal of Hydrology*, 300(1-4):114–125, 2005.

[10] P. Benner and T. Breiten. Two-sided projection methods for nonlinear model order reduction. *SIAM Journal on Scientific Computing*, 37(2):B239–B260, 2015.

[11] P. Benner and P. Goyal. Interpolation-based model order reduction for polynomial systems. *SIAM Journal on Scientific Computing*, 43(1):A84–A108, 2021.

[12] P. Benner, P. Goyal, and S. Gugercin. $\mathcal{H}_2$-quasi-optimal model order reduction for quadratic-bilinear control systems. *SIAM Journal on Matrix Analysis and Applications*, 39(2):983–1032, 2018.

[13] P. Benner, P. Goyal, J. Heiland, and I. P. Duff. Operator inference and physics-informed learning of low-dimensional models for incompressible flows. e-prints 2010.06701, arXiv, 2020. math.NA.

[14] P. Benner, P. Goyal, B. Kramer, P. Peherstorfer, and K. Willcox. Operator inference for non-intrusive model reduction of systems with non-polynomial nonlinear terms. *Computer Methods in Applied Mechanics and Engineering*, 372:113433, 2020.

[15] P. Benner, M. Ohlberger, A. Cohen, and K. Willcox, editors. *Model reduction and approximation*, volume 15 of *Computational Science & Engineering*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2017. Theory and algorithms.

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

[17] D. A. Bistrian and I. M. Navon. 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, 2015.

[18] D. A. Bistrian and I. M. Navon. The method of dynamic mode decomposition in shallow water and a swirling flow problem. *International Journal for Numerical Methods in Fluids*, 83(1):73–89, 2017.

[19] E. Boss, N. Paldor, and L. Thompson. Stability of a potential vorticity front: from quasi-geostrophy to shallow water. *Journal of Fluid Mechanics*, 315:65–84, 1996.

[20] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.

[21] D. Calvetti, P. C. Hansen, and L. Reichel. L-curve curvature bounds via lanczos bidiagonalization. *Electron. Trans. Numer. Anal*, 14:20–35, 2002.

[22] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623 – 647, 2013.

[23] K. Carlberg, R. Tuminaro, and P. Boggs. Preserving Lagrangian structure in nonlinear model reduction with application to structural dynamics. *SIAM J. Sci. Comput.*, 37(2):B153–B184, 2015.

[24] E. Celledoni, V. Grimm, R. I. McLachlan, D. I. McLaren, D. O'Neale, B. Owren, and G. R. W. Quispel. Preserving energy resp. dissipation in numerical PDEs using the "Average Vector Field" method. *J. Comput. Physics*, 231:6770–6789, 2012.

[25] E. Celledoni, R. I. McLachlan, D. I. McLaren, B. Owren, and G. R. W. Quispel. Discretization of polynomial vector fields by polarization. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 471(2184), 2015.

[26] E. Celledoni, R. I. McLachlan, B. Owren, and G. R. W. Quispel. Geometric properties of Kahan's method. *Journal of Physics A: Mathematical and Theoretical*, 46(2):025201, 2013.

[27] T. F. Chan and P. C. Hansen. Some applications of the rank revealing qr factorization. *SIAM Journal on Scientific and Statistical Computing*, 13(3):727–741, 1992.

[28] S. Chaturantabut, C. Beattie, and S. Gugercin. Structure-preserving model reduction for nonlinear port-Hamiltonian systems. *SIAM Journal on Scientific Computing*, 38(5):B837–B865, 2016.

[29] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[30] S. Chaturantabut and D. C. Sorensen. A state space error estimate for POD-DEIM nonlinear model reduction. *SIAM Journal on Numerical Analysis*, 50(1):46–63, 2012.

[31] A. Chertock, M. Dudzinski, A. Kurganov, and M. Lukáčová-Medvid'ová. Well-balanced schemes for the shallow water equations with Coriolis forces. *Numer. Math.*, 138(4):939–973, 2018.

[32] A. Chorin and P. Stinis. Problem reduction, renormalization, and memory. *Commun. Appl. Math. Comput. Sci.*, 1(1):1–27, 2006.

[33] D. Cohen and E. Hairer. Linear energy-preserving integrators for Poisson systems. *BIT Numerical Mathematics*, 51(1):91–101, 2011.

[34] C.J. Cotter and J. Shipton. Mixed finite elements for numerical weather prediction. *Journal of Computational Physics*, 231(21):7076–7091, 2012.

[35] R. Ştefănescu and I. M. Navon. POD/DEIM nonlinear model order reduction of an ADI implicit shallow water equations model. *Journal of Computational Physics*, 237:95 – 114, 2013.

[36] R. Ştefănescu, A. Sandu, and I. M. Navon. Comparison of pod reduced order strategies for the nonlinear 2D shallow water equations. *International Journal for Numerical Methods in Fluids*, 76(8):497–521, 2014.

[37] M. Dahlby, B. Owren, and T. Yaguchi. Preserving multiple first integrals by discrete gradients. *Journal of Physics A: Mathematical and Theoretical*, 44(30):305205, 2011.

[38] O. Delestre, F. Darboux, F. James, L. Lucas, C. Laguerre, and S. Cordier. FullSWOF: Full shallow-water equations for overland flow. *Journal of Open Source Software*, 2(20):448, 2017.

[39] P. J. Dellar. Hamiltonian and symmetric hyperbolic structures of shallow water magnetohydrodynamics. *Physics of Plasmas*, 9(4):1130–1136, 2002.

[40] P. J. Dellar. Common Hamiltonian structure of the shallow water equations with horizontal temperature gradients and magnetic fields. *Physics of Fluids*, 15(2):292–297, 2003.

[41] P. J. Dellar and R. Salmon. Shallow water equations with a complete coriolis force and topography. *Physics of Fluids*, 17(10):106601, 2005.

[42] D. P. Dempsey and R. Rotunno. Topographic Generation of Mesoscale Vortices in Mixed-Layer Models. *Journal of the Atmospheric Sciences*, 45(20):2961–2978, 1988.

[43] C. Eldred, T. Dubos, and E. Kritsikis. A quasi-Hamiltonian discretization of the thermal shallow water equations. *Journal of Computational Physics*, 379:1 – 31, 2019.

[44] V. Esfahanian and K. Ashrafi. Equation-free/Galerkin-free reduced-order modeling of the shallow water equations based on Proper Orthogonal Decomposition. *Journal of Fluids Engineering*, 131(7):071401–071401–13, 2009.

[45] U. S. Fjordholm, S. Mishra, and E. Tadmor. Well-balanced and energy stable schemes for the shallow water equations with discontinuous topography. *Journal of Computational Physics*, 230(14):5587 – 5609, 2011.

[46] G. J. Gassner, A. R. Winters, and D. A. Kopriva. A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations. *Applied Mathematics and Computation*, 272:291 – 308, 2016.

[47] T. Gerkema and V. I. Shrira. Near-inertial waves in the ocean: beyond the 'traditional approximation'. *Journal of Fluid Mechanics*, 529:195–219, 2005.

[48] T. H. Gibson, A. T. T. McRae, C. J. Cotter, L. Mitchell, and D. A. Ham. *Compatible finite element methods for geophysical flows*. SpringerBriefs in Mathematics of Planet Earth. Springer, Cham, 2019. Automation and implementation using Firedrake.

[49] D. Givon, R. Kupferman, and A. Stuart. Extracting macroscopic dynamics: model problems and algorithms. *Nonlinearity*, 17(6):R55–R127, 2004.

[50] Y. Gong, J. Cai, and Y. Wang. Some new structure-preserving algorithms for general multi-symplectic formulations of Hamiltonian {PDEs}. *Journal of Computational Physics*, 279:80 – 102, 2014.

[51] Y. Gong, Q. Wang, and Z. Wang. Structure-preserving Galerkin POD reduced-order modeling of Hamiltonian systems. *Computer Methods in Applied Mechanics and Engineering*, 315:780 – 798, 2017.

[52] I. V. Gosea and I. P. Duff. Toward fitting structured nonlinear systems by means of dynamic mode decomposition, 2020.

[53] E. Gouzien, N. Lahaye, V. Zeitlin, and T. Dubos. Thermal instability in rotating shallow water with horizontal temperature/density gradients. *Physics of Fluids*, 29(10):101702, 2017.

[54] P. Goyal and P. Benner. Lqresnet: A deep neural network architecture for learning dynamic processes. *arXiv preprint arXiv:2103.02249*, 2021.

[55] W. R. Graham, J. Peraire, and K. Y. Tang. Optimal control of vortex shedding using low-order models. Part I –open-loop model development. *International Journal for Numerical Methods in Engineering*, 44(7):945–972, 1999.

[56] C. Gu. QLMOR: a projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 30(9):1307–1320, 2011.

[57] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration: Structure-preserving algorithms for ordinary differential equations*. Springer Series in Computational Mathematics. Springer, Heidelberg, 2016.

[58] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[59] A. C. Hansen and J. Strain. On the order of deferred correction. *Applied Numerical Mathematics*, 61(8):961 – 973, 2011.

[60] P. C. Hansen. The l-curve and its use in the numerical treatment of inverse problems. computational inverse problems in electrocardiology. *Advances in Computational Bioengineering*, 5:119, 2001.

[61] E. Herrera and S. Morett. On the direction of Coriolis force and the angular momentum conservation. *Revista Brasileira de Ensino de Física*, 38(3), 2016.

[62] J. Hesthaven and C. Pagliantini. Structure-preserving reduced basis methods for Poisson system. *Mathematics of Computation*, 2021.

[63] J. S. Hesthaven and C. Pagliantini. Structure-preserving reduced basis methods for Hamiltonian systems with a state-dependent Poisson structure. *Mathematics of Computation*, 2020.

[64] J. S. Hesthaven and S. Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018.

[65] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge, second edition, 2012.

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

[67] W. Kahan. Unconventional numerical methods for trajectory calculations. Technical report, Computer Science Division and Department of Mathematics, University of California, Berkeley, 1993. Unpublished lecture notes.

[68] W. Kahan and R. C. Li. Unconventional schemes for a class of ordinary differential equatins-with applications to the Korteweg-de Vries equation. *Journal of Computational Physics*, 134(2):316 – 331, 1997.

[69] B. Karasözen, S. Yıldız, and M. Uzunca. Structure preserving model order reduction of shallow water equations. *Mathematical Methods in the Applied Sciences*, 44(1):476–492, 2021.

[70] B. Karasözen and M. Uzunca. Energy preserving model order reduction of the nonlinear schrödinger equation. *Advances in Computational Mathematics*, 44(6):1769–1796, 2018.

[71] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, 2009.

[72] B. Kramer and K. E. Willcox. Nonlinear model order reduction via lifting transformations and proper orthogonal decomposition. *AIAA Journal*, 57(6):2297–2307, 2019.

[73] A. Kurganov, Y. Liu, and V. Zeitlin. Thermal versus isothermal rotating shallow water equations: comparison of dynamical processes by simulations with a novel well-balanced central-upwind scheme. *Geophysical & Astrophysical Fluid Dynamics*, 0(0):1–30, 2020.

[74] A. Kurganov, Y. Liu, and V. Zeitlin. A well-balanced central-upwind scheme for the thermal rotating shallow water equations. *J. Comput. Phys.*, 411:109414, 24, 2020.

[75] S. Leibovich and S.K. Lele. The influence of the horizontal component of earth's angular velocity on the instability of the ekman layer. *Journal of Fluid Mechanics*, 150:41–87, 1985.

[76] P. d. Leva. MULTIPROD TOOLBOX, multiple matrix multiplications, with array expansion enabled. Technical report, University of Rome Foro Italico, Rome, 2008.

[77] J. C. Loiseau and S. L. Brunton. Constrained sparse Galerkin regression. *J. Fluid Mech.*, 838:42–67, 2018.

[78] A. Lozovskiy, M. Farthing, and C. Kees. Evaluation of Galerkin and Petrov-Galerkin model reduction for finite element approximations of the shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 318:537 – 571, 2017.

[79] A. Lozovskiy, M. Farthing, C. Kees, and E. Gildin. POD-based model reduction for stabilized finite element approximations of shallow water flows. *Journal of Computational and Applied Mathematics*, 302:50 – 70, 2016.

[80] P. Lynch. Hamiltonian methods for geophysical fluid dynamics: An introduction, 2002.

[81] J. Marshall and F. Schott. Open-ocean convection: Observations, theory, and models. *Reviews of Geophysics*, 37(1):1–64, 1999.

[82] J. R. Martins and J. T. Hwang. Review and unification of methods for computing derivatives of multidisciplinary computational models. *AIAA Journal*, 51(11):2582–2599, 2013.

[83] S. A. McQuarrie, C. Huang, and K. Willcox. Data-driven reduced-order models via regularised Operator Inference for a single-injector combustion process. *Journal of the Royal Society of New Zealand*, 0(0):1–18, 2021.

[84] Y. Miyatake. Structure-preserving model reduction for dynamical systems with a first integral. *Japan Journal of Industrial and Applied Mathematics*, 36(3):1021–1037, 2019.

[85] P. J. Morrison. Hamiltonian description of the ideal fluid. *Rev. Mod. Phys.*, 70:467–521, 1998.

[86] C. Mou, H. Liu, D. R. Wells, and T. Iliescu. Data-driven correction reduced order models for the quasi-geostrophic equations: a numerical investigation. *International Journal of Computational Fluid Dynamics*, 34(2):147–159, 2020.

[87] N. C. Nguyen, A. T. Patera, and J. Peraire. A "best points" interpolation method for efficient approximation of parametrized functions. *International Journal for Numerical Methods in Engineering*, 73(4):521–543, 2008.

[88] M. Ohlberger and S. Rave. Reduced basis methods: Success, limitations and future challenges. *Proceedings of the Conference Algoritmy*, pages 1–12, 2016.

[89] B. Peherstorfer. Sampling low-dimensional Markovian dynamics for pre-asymptotically recovering reduced models from data with operator inference. *SIAM Journal on Scientific Computing*, 42(5):A3489–A3515, 2020.

[90] B. Peherstorfer and K. Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196 – 215, 2016.

[91] L. Peng and K. Mohseni. Symplectic model reduction of Hamiltonian systems. *SIAM Journal on Scientific Computing*, 38(1):A1–A27, 2016.

[92] D.I. Pontes, P. Goyal, and P. Benner. Balanced truncation for a special class of bilinear descriptor systems. *IEEE Control Syst. Lett.*, 3(3):535–540, 2019.

[93] E. Qian, I. G. Farcas, and K. Willcox. Reduced operator inference for nonlinear partial differential equations, 2021.

[94] E. Qian, B. Kramer, A. N. Marques, and K. E. Willcox. Transform & learn: A data-driven approach to nonlinear model reduction. In *2019 AIAA Aviation and Aeronautics Forum and Exposition, June 17-21, Dallas, TX*, 2019.

[95] E. Qian, B. Kramer, B. Peherstorfer, and K. Willcox. Lift & learn: Physics-informed machine learning for large-scale nonlinear dynamical systems. 2019. Oden Institute Report 19-18.

[96] A. Quarteroni and G Rozza. *Reduced Order Methods for Modeling and Computational Reduction*, volume 9 of *MS&A*. Springer, Milano, 1 edition, 2014.

[97] H. Ranocha. Shallow water equations: split-form, entropy stable, well-balanced, and positivity preserving numerical methods. *GEM Int. J. Geomath.*, 8(1):85–133, 2017.

[98] T. Reis and T. Stykel. Stability analysis and model order reduction of coupled systems. *Math. Comput. Model. Dyn. Syst.*, 13(5):413–436, 2007.

[99] P. Ripa. On improving a one-layer ocean model with thermodynamics. *Journal of Fluid Mechanics*, 303:169–201, 1995.

[100] C. W. Rowley, I. Mezic, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral analysis of nonlinear flows. *Journal of fluid mechanics*, 641(1):115–127, 2009.

[101] S. Rudy, A. Alla, S. L. Brunton, and J. N. Kutz. Data-driven identification of parametric partial differential equations. *SIAM Journal on Applied Dynamical Systems*, 18(2):643–660, 2019.

[102] R. Salmon. Hamiltonian fluid mechanics. *Annual Review of Fluid Mechanics*, 20(1):225–256, 1988.

[103] R. Salmon. Poisson-bracket approach to the construction of energy- and potential-enstrophy-conserving algorithms for the shallow-water equations. *Journal of the Atmospheric Sciences*, 61(16):2016–2036, 2004.

[104] R. Salmon. A general method for conserving energy and potential enstrophy in shallow-water models. *Journal of the Atmospheric Sciences*, 64(2):515–531, 2007.

[105] H. Schaeffer, G. Tran, and R. Ward. Extracting sparse high-dimensional dynamics from limited data. *SIAM Journal on Applied Mathematics*, 78(6):3279–3295, 2018.

[106] W. Schilders, S. Grivet-Talocia, P. Benner, A. Quarteroni, G. Rozza, and L. Silveira. *Snapshot-Based Methods and Algorithms*. De Gruyter, 2021.

[107] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.

[108] L. Sirovich. Turbulence and the dynamics of coherent structures. III. Dynamics and scaling. *Quart. Appl. Math.*, 45(3):583–590, 1987.

[109] R. Ştefănescu, A. Sandu, and I. M. Navon. Comparison of pod reduced order strategies for the nonlinear 2d shallow water equations. *International Journal for Numerical Methods in Fluids*, 76(8):497–521, 2014.

[110] A. L. Stewart and P. J. Dellar. Multilayer shallow water equations with complete Coriolis force. part 1. derivation on a non-traditional beta-plane. *Journal of Fluid Mechanics*, 651:387–413, 2010.

[111] A. L. Stewart and P. J. Dellar. Multilayer shallow water equations with complete coriolis force. part 2. linear plane waves. *Journal of Fluid Mechanics*, 690:16–50, 2012.

[112] A. L. Stewart and P. J. Dellar. Multilayer shallow water equations with complete coriolis force. part 3. hyperbolicity and stability under shear. *Journal of Fluid Mechanics*, 723:289–317, 2013.

[113] A. L. Stewart and P. J. Dellar. An energy and potential enstrophy conserving numerical scheme for the multi-layer shallow water equations with complete Coriolis force. *Journal of Computational Physics*, 313:99 – 120, 2016.

[114] Y. Sugibuchi, T. Matsuo, and S. Sato. Constructing invariant-preserving numerical schemes based on Poisson and Nambu brackets. *JSIAM Letters*, 10:53–56, 2018.

[115] R. Swischuk, B. Kramer, C. Huang, and K. Willcox. Learning physics-based reduced-order models for a single-injector combustion process. *AIAA Journal*, 58(6):2658–2672, 2020.

[116] R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox. Projection-based model reduction: Formulations for physics-based machine learning. *Computers & Fluids*, 179:704–717, 2019.

[117] A. N. Tikhonov and V. Y. Arsenin. *Solutions of ill-posed problems*, volume 14. Winston, Washington, DC, 1977.

[118] W. I. T. Uy and B. Peherstorfer. Probabilistic error estimation for non-intrusive reduced models learned from data of systems governed by linear parabolic partial differential equations. *arXiv e-prints*, page arXiv:2005.05890, May 2020.

[119] W. I. T. Uy and B. Peherstorfer. Operator inference of non-markovian terms for learning reduced models from partially observed state trajectories, 2021.

[120] G. K. Vallis. *Atmospheric and oceanic fluid dynamics*. Cambridge University Press, 2017.

[121] Q. Wang, J. S. Hesthaven, and D. Ray. 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, 2019.

[122] E. S. Warneford and P. J. Dellar. The quasi-geostrophic theory of the thermal shallow water equations. *Journal of Fluid Mechanics*, 723:374–403, 2013.

[123] E. S. Warneford and P. J. Dellar. Thermal shallow water models of geostrophic turbulence in Jovian atmospheres. *Physics of Fluids*, 26(1):016603, 2014.

[124] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley. A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. *J. Nonlinear Sci.*, 25(6):1307–1346, 2015.

[125] G. A. Wimmer, C. J. Cotter, and W. Bauer. Energy conserving upwinded compatible finite element schemes for the rotating shallow water equations. *arXiv e-prints*, 2019.

[126] Y. Xu, J. J. Van Der Vegt, and O. Bokhove. Discontinuous Hamiltonian finite element method for linear hyperbolic systems. *J. Sci. Comput.*, 35(2-3):241–265, 2008.

[127] W. R. Young and L. Chen. Baroclinic Instability and Thermohaline Gradient Alignment in the Mixed Layer. *Journal of Physical Oceanography*, 25(12):3172–3185, 12 1995.

[128] S. Yıldız, P. Goyal, P. Benner, and B. Karasözen. Learning reduced-order dynamics for parametrized shallow water equations from data. *International Journal for Numerical Methods in Fluids*, 93(8):2803–2821, 2021.

[129] V. Zeitlin. *Geophysical fluid dynamics: understanding (almost) everything with rotating shallow water models*. Oxford University Press, Oxford, 2018.

[130] M. Zerroukat and T. Allen. A moist Boussinesq shallow water equations set for testing atmospheric models. *Journal of Computational Physics*, 290:55 – 72, 2015.

[131] R. Zimmermann and K. Willcox. An accelerated greedy missing point estimation procedure. *SIAM Journal on Scientific Computing*, 38(5):A2827–A285, 2016.

[132] D. Zlatko and S. Gugercin. A new selection operator for the discrete empirical interpolation method-improved a priori error bound and extensions. *SIAM Journal on Scientific Computing*, 38(2):A631–A648, 2016.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Yıldız, Süleyman
Nationality: Turkish (TC)
Date and Place of Birth: 01.02.1986, Antakya
Marital Status: Married

EDUCATION

Degree	Institution	Year of Graduation
Ph.D.	Scientific Computing, METU	2021
M.S.	Scientific Computing, METU	2016
B.S.	Mathematics Department, GOP	2012

AWARDS

- 2018 - 2021, 100/2000 Ph.D. Scholarship Program of Turkish Higher Education Council (YOK)

- (METU) Middle East Technical University, "Course Performance Award", 2018

- KYK (Credit and Dormitories Institution)

  2016- 2020, Ph.D. Scholarship

  2015- 2016, Master Scholarship

COMPUTER SKILLS

Proficient in Matlab, Python, Latex, Linux/UNIX

Working Knowledge of Fortran, HTML, JAVA

TEACHING EXPERIENCE

Middle East Technical University, Ankara, TURKEY, 2019

- Lecturer

  – Introduction to MATLAB Programming

PUBLICATIONS

Journal Articles Published/Accepted:

- B. Karasözen, S. Yıldız, M. Uzunca, *Structure Preserving Model Order Reduction of Shallow Water Equations* , Mathematical Methods in the Applied Sciences, 2021, 44, 476-492, https://doi.org/10.1002/mma.6751

- B. Karasözen, G. Mülayim, M. Uzunca, S. Yıldız, *Reduced order modelling of nonlinear cross-diffusion systems*, Applied Mathematics and Computation, 2021, 401: 126058, https://doi.org/10.1016/j.amc.2021.126058

- M. Uzunca, S. Yıldız, B. Karasözen, *Structure-preserving reduced-order modelling of Korteweg de Vries equation*, Mathematics and Computers in Simulation, 2021, Volume 188, Pages 193-211, https://doi.org/10.1016/j.matcom.2021.03.042

- S. Yıldız, P. Goyal, P. Benner, & B. Karasozen, *Learning Reduced-order Dynamics for Parametrized Shallow Water Equations from Data* , International Journal for Numerical Methods in Fluids, 2021, 93(8):2803–2821, https://doi.org/10.1002/fld.4998

Conference Proceedings (Refereed) Accepted:

- S. Yıldız, M. Uzunca, & B. Karasözen, *Structure-preserving Reduced Order Modeling of non-traditional Shallow Water Equation*, to be appear in "Model Reduction of Complex Dynamical Systems", "International Series of Numerical Mathematics", Springer

Journal Articles in Review:

- S. Yıldız, M. Uzunca, & B. Karasözen. *Energy preserving reduced-order modelling of thermal shallow water equation* . Submitted 2021.

- S. Yıldız, M. Uzunca, & B. Karasözen. *Intrusive and non-intrusive reduced order modelling of the rotating thermal shallow water equation.* Submitted 2021.

Conference Poster (Refereed) Published:

- S. Yıldız, P. Goyal, P. Benner, & B. Karasozen. *Data-Driven Learning of Reduced-Order Dynamics for a Parametrized Shallow Water Equation.* 2021 PAMM, 20(S1), e202000360.

SCIENTIFIC MEETINGS

Presentation in Research Talks:

- *Structure Preserving Model Order Reduction for Thermal Shallow Water Equation* SIAM Conference on Computational Science and Engineering (CSE21), March 1-5, 2021.

- *Data-Driven Learning of Reduced-Order Dynamics for a Parametrized Shallow Water Equation,* Learning Models from Data: Model Reduction, System Identification and Machine Learning, GAMM Juniors' Summer School on Applied Mathematics and Mechanics July 27 - 31, 2020 (Poster presentation).

- *Structure-preserving Reduced Order Modeling of non-traditional Shallow Water Equation,* 4th Workshop on Model Reduction of Complex Dynamical Systems - MODRED 2019 - August 28th to 30th, 2019, Graz.

Research Visits:

- 19 - 30 July 2021,

  2021 Gene Golub SIAM Summer School, Muizenberg, South Africa

  – got accepted, planning to attend

  – supported by Society for Industrial and Applied Mathematics (SIAM)

- Febuary 2020-May 2020,

  Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg, GERMANY

  – supervised by Prof. Dr. Peter Benner

  – supported by Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg

- August 2019-November 2019,

  Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg, GERMANY

  – supervised by Prof. Dr. Peter Benner

  – supported by Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg