Tesis Doctoral

Doctorado en Oceanografía y Cambio Global

Advances in the Modelling of Spatio-Temporal Variability from Large to Mesoscale Oceanographic Features Off Northwest Africa: A Finite Element Method Approach with HDG

Inés Hernández García

Las Palmas de Gran Canaria
Junio 2025

ULPGC · UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

ULPGC
Universidad de
Las Palmas de
Gran Canaria

Universidad de Las Palmas de Gran Canaria

Escuela de Doctorado

Doctorado en Oceanografía y Cambio Global

Título de la tesis: Advances in the Modelling of Spatio-Temporal Variability from Large to Mesoscale Oceanographic Features Off Northwest Africa: A Finite Element Method Approach with HDG

Tesis Doctoral presentada por: Inés Hernández García

Dirigida por:

Dr. D. Albert Oliver Serra

Dr. D. Francisco José Machín Jiménez

Las Palmas de Gran Canaria

Junio de 2025

Acknowledgements

I would like to thank my tutor, Ángel Rodríguez, and my directors, Francis Machín, with whom it has always been a pleasure to work, and Albert Oliver, who was a great discovery during this thesis, for their constant support and guidance throughout the entire process. This work would not have been possible without them.

I am also grateful to my colleagues from OFYGA, the Physics Department, and SIANI for their collaboration and support. A very special thanks goes to Laia, Laura, and Inés, who contributed to this thesis through their final degree projects and were wonderful companions throughout this journey. I also wish to thank Eirik and everyone at NMBU who welcomed me during my international stay. Their hospitality and support made the experience both enriching and unforgettable. Finally, I would like to thank Younghun Kang and Charles Troupin, who served as external reviewers, for their comments and suggestions, which greatly helped refine the structure and improve the details of this thesis. Also, this work was possible thanks to the financial support from the Canarian Agency for Research, Innovation and Information Society of the Regional Ministry of Economy, Knowledge and Employment (ACIISI).

I would like to acknowledge all my teachers, final degree projects advisors, classmates, and friends from the undergraduate and master's programs in Ciencias del Mar, who have inspired me and motivated my path toward a PhD. I am especially thankful to the close friends who supported me right after the lockdown in 2020, and made the university feel less empty.

To my friends, thank you for always being there and supporting me through this journey. Finally, I would like to thank my family for their unconditional love and encouragement every step of the way.

Funding sources

Work co-financed by the Canarian Agency for Research, Innovation and Information Society of the Regional Ministry of Economy, Knowledge and Employment and by the European Social Fund (ESF) Canary Islands Integrated Operational Program 2014-2020, Axis 3 Priority Theme 74 (85%).

Canarias
avanza
con Europa

Gobierno de Canarias
Consejería de Universidades,
Ciencia e Innovación y Cultura

Agencia Canaria de Investigación,
Innovación y Sociedad
de la Información

UNIÓN EUROPEA
Fondo Social Europeo
El FSE invierte en tu futuro

Abstract

The ocean exhibits a wide range of physical processes across different spatial and temporal scales. In the North Atlantic Ocean, these processes range from large-scale currents associated with the global thermohaline circulation, of approximately 10 000 km, to microscale turbulence at the cm level. The temporal scales extend from hourly fluctuations to decadal variations. Due to this complexity, observational data often lack the resolution and synopticity needed to fully capture these dynamics, making numerical modelling an essential tool for complementing observations and improving our understanding of oceanic physical structures.

A useful approach to analyze ocean dynamics is through the Shallow Water Equations (SWE), an approximation of the Navier–Stokes equations that assumes a shallow fluid layer in which the horizontal scale greatly exceeds the vertical. These equations cannot be solved analytically and require the use of numerical methods.

The Finite Element Method (FEM) is a well-suited tool to create powerful models. FEM is based on the variational formulation of the system of Partial Differential Equations, and offers flexibility in the mesh design, permitting the use of unstructured meshes, and computational efficiency. The Hybridizable Discontinuous Galerkin (HDG) method belongs to the FEM family and provides all its advantages while also ensuring local conservation and efficient solution strategies.

Implicit–explicit (IMEX) time integration schemes allow for larger time-step sizes than fully explicit methods, while being less computationally demanding than fully implicit schemes. Strong Stability Preserving (SSP) schemes further ensure that the non-linear stability properties of the system are maintained, a crucial requirement when dealing with hyperbolic systems such as the SWE.

In this thesis, we analyze the suitability of a numerical model of the Shallow Water Equations based on the HDG spatial discretization and an IMEX

SSP time discretization for oceanographic problems. The model is implemented using the Firedrake finite element library in Python.

Resumen

El océano presenta una amplia gama de procesos físicos a diferentes escalas espaciales y temporales. En el Océano Atlántico Norte, estos procesos van desde las corrientes a gran escala, asociadas a la circulación termohalina mundial, de aproximadamente 10 000 km, hasta la turbulencia a microescala, a nivel de cm. Las escalas temporales abarcan desde fluctuaciones horarias hasta variaciones decenales. Debido a esta complejidad, los datos observacionales carecen a menudo de la resolución y sinopticidad necesarias para captar plenamente esta dinámica, lo que convierte a la modelización numérica en una herramienta esencial para complementar las observaciones y mejorar nuestra comprensión de las estructuras físicas oceánicas.

Un enfoque útil para analizar la dinámica del océano es el de las ecuaciones de aguas someras (SWE, por sus siglas en inglés), una aproximación de las ecuaciones de Navier-Stokes que supone una capa de fluido poco profunda en la que la escala horizontal supera ampliamente a la vertical. Estas ecuaciones no pueden resolverse analíticamente y requieren del uso de métodos numéricos.

El método de los elementos finitos (FEM, por sus siglas en inglés) es una herramienta muy conveniente para crear modelos potentes. El FEM se basa en la formulación variacional de las ecuaciones diferenciales que definen el problema, y ofrece flexibilidad en el diseño de la malla, permitiendo el uso de mallas no estructuradas, y eficiencia computacional. El método Galerkin discontinuo hibridizable (HDG, por sus siglas en inglés) pertenece a la familia del FEM y ofrece todas sus ventajas, al tiempo que garantiza la conservación local y estrategias de solución eficientes.

Los esquemas de discretización temporal implícitos-explícitos (IMEX) permiten tamaños de paso temporal mayores que los métodos totalmente explícitos, mientras que son menos exigentes desde el punto de vista computacional que los esquemas totalmente implícitos. Los esquemas Strong Stability Preserving (SSP) garantizan, además, el mantenimiento de las propiedades de estabilidad no lineal del sistema, un requisito crucial cuando se trabaja con sistemas hiperbólicos como las SWE.

En esta tesis se analiza la idoneidad de un modelo numérico de las Ecuaciones de Aguas Someras basado en la discretización espacial HDG y una discretización temporal IMEX SSP para problemas oceanográficos. El modelo se implementa utilizando la librería de elementos finitos Firedrake en Python.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In oceanography, modelling is a key tool to complement the observational data, in order to understand the complex physical structures of the ocean. The dynamics of the ocean are usually represented by the Shallow Water Equation system. Finite Element methods are a convenient tool to create powerful models. Particularly, Hybridizable Discontinuous Galerkin (HDG) methods offer all the advantadges of Finite Elements while also ensuring local conservation and efficient solution strategies. In this thesis, we aim to implement a HDG SWE model, with the goal of improving our ability to predict, interpret, and manage oceanic phenomena.

1.1 State of the art

In the ocean, the main structure at the global scale is the thermohaline circulation, accompanied by the subtropical gyres, equatorial currents, and intense western boundary currents at each basin. Moving to finer scales, at the mesoscale the main processes are coastal upwelling systems, the filaments associated to them, ocean fronts and mesoscale eddies. Finally, the turbulence is the main process at the submesoscale (Talley et al., 2011).

In the North Atlantic Ocean the main water masses interacting between the surface and 1000 m are the North Atlantic Central Water (NACW) and the South Atlantic Central Water (SACW), and, at intermediate levels, we can find the Antarctic Intermediate Water (AAIW) and the Mediterranean Water (MW) (Tomczak, 1981).

Regarding the physical structures, the North Atlantic Ocean is dominated by the North Atlantic Subtropical Gyre. In the East part, the water flows southward in the Canary Current, and westward in the North Equatorial

Current (Hernández-Guerra et al., 2005; Machín et al., 2006; Pelegrí et al., 2017; Pérez et al., 2001; Vélez-Belchí et al., 2017). The NACW and SACW form a sharp front at the area off the Northwest African coast (Barton, 1987; Burgoa et al., 2021; Tomczak, 1981). Long-lived anti-cyclonic and cyclonic eddies have been found in the path of the Canary Current (Burgoa et al., 2021; Hagen, 1985; Sangrà et al., 2009). Some intrusions of MW moving into the Atlantic Ocean form mediterranean water eddies (meddies) (Zenk et al., 1991). There is also a prominent upwelling system at the African Coast (Mittelstaedt, 1983; Pelegrí et al., 2017).

Observational data fails to capture in detail these complex processes with many different spatial and temporal scales. In order to understand them, modelling is a fundamental tool in oceanography.

The Navier-Stokes (NS) equations (Navier, 1823; Stokes, 1845) are a set of equations that describe the motion of a fluid, with a great variety of structures that can occur simultaneously on different time scales and lengthscales (Kämpf, 2009; Kundu et al., 2024). They consist on the momentum equations, the continuity equation, advection-diffusion equations, and the equation of state of the water (Kämpf, 2009).

The Navier-Stokes equations are capable to model fast and slow waves in fluids, among other processes. Slow waves, with a velocity of around $30\,\mathrm{m\,s^{-1}}$ correspond to the large scale patterns, advection, and the Coriolis force. The fast waves, with the speed of sound in the air, $300\,\mathrm{m\,s^{-1}}$, called acoustic waves, are related to the non-linear dynamics (Betteridge, 2020).

The Shallow Water Equations (SWE) are an approximation of the Navier-Stokes equations for an incompressible shallow layer of water (Kämpf, 2009; Kundu et al., 2024). They describe the evolution of a layer of fluid where the horizontal scale is much larger than the vertical scale, under gravitational and inertial forces (Betteridge et al., 2021; Valseth & Dawson, 2021). The NS and SWE can not be solved analytically and require the use of numerical methods (Betteridge, 2020).

Traditionally, oceanographic models are solved using the Finite Differences (FD) (LeVeque, 2007) and Finite Volume (FV) (LeVeque, 2002) methods. Only some prediction models are based on Finite Element Methods (FEM) (Zienkiewicz et al., 2013), but this method is becoming increasingly more popular. The main limitations of FD methods are their high computational expense, due to the need for strict meshes, and non-optimal scalability for larger models (Ortleb, 2017). FEM is discretized using integrals and offers a flexible mesh formed by non-overlapping elements (Marras et al., 2015).

FEM methods offer high parallel efficiency, and provide flexibility, due to the use of unstructured meshes that adjust to the geometry of the bathymetry and coastline, and the wide range of scales covered by the physical processes (Kubatko et al., 2009; Marras et al., 2015). Besides the spatial discretization, FEM has also been recently used for the time discretization of the model (Valseth & Dawson, 2021). However, FEM methods were initially not widely adopted in computational fluid dynamics (CFD) due the forming of instabilities and the lack of inherent local conservation properties (Pachev, 2024). Traditional Galerkin formulations often led to oscillations in advection-dominated problems. To overcome these limitations, various stabilization techniques were developed and incorporated into the finite element framework. The Streamline Upwind Petrov-Galerkin (SUPG) method (Brooks & Hughes, 1982) introduced artificial diffusion along the streamlines to suppress numerical oscillations and enhance stability in convection-dominated flows. Other approaches, such as the Galerkin/Least-Squares (GLS) method (T. J. R. Hughes et al., 1989), the Space-Time Galerkin/Least-Squares (STGLS) method (Shakib & Hughes, 1991) and the Subgrid Scale (SGS) method (T. J. Hughes, 1995), aimed to improve stability and accuracy by discretizing both space and time using finite elements, and modeling the effect of unresolved scales. The Characteristic Galerkin method (Douglas & Russell, 1982; Löhner et al., 1984) combined the finite element method with the method of characteristics to better capture advective transport. The Taylor-Galerkin (TG) method (Donea, 1984) employed a Taylor series expansion in time to derive a stabilized scheme, introducing numerical diffusion through higher-order time derivatives. A review of al these methods can be found in Codina (1998), Donea and Huerta (2004), and T. J. R. Hughes et al. (2010). Despite their success in improving stability and accuracy, these stabilized FEM methods do not guarantee local conservation of fluxes. More recently, the Discontinuous Petrov-Galerkin (DPG) method (Savant et al., 2019) was proposed, which ensures stability by systematically optimizing the test functions to produce well-posed discrete systems (Pachev, 2024; Valseth & Dawson, 2021).

As an alternative to the stabilization techniques, Discontinuous Galerkin (DG) methods introduced local conservation properties inherent to the method, as well as better handling of discontinuities, which are essential in geophysical flows (Costa-Solé et al., 2019; Kang et al., 2020). However, the increased number of degrees of freedom and the cost of numerical flux evaluations across element interfaces make DG methods expensive for large-scale simulations (Kirby et al., 2012).

DG methods (Cockburn, 2004; Hesthaven & Warburton, 2008) provide local conservation. It was initially developed by Reed and Hill (1973) for the neutron transport equation, and has since been used to solve different Partial Differential Equations (PDE) systems. They are a robust tool in the solution of differential problems (Donea & Huerta, 2004; Marras et al., 2015). DG solves the equation system by solving the elements individually, making the solution discontinuous at their interfaces, and imposing a flux, ensuring conservation among them (Costa-Solé et al., 2019; Costa-Solé, 2020;

Javadzadeh Moghtader, 2016). It provides intrinsic stability at the discontinuities and sharp fronts, due to the numerical fluxes (Costa-Solé, 2020; Javadzadeh Moghtader, 2016). DG contains the characteristics of Finite Volume and Finite Elements methods, and offers advantages such as high accuracy, and inherent global and local conservation along the elements, among others. Its element-wise construction naturally lends itself to parallel computation (Bui-Thanh, 2016; Kubatko et al., 2009; Samii et al., 2019). Like FEM, DG can also be applied to unstructured meshes (Betteridge, 2020; Betteridge et al., 2021; Valseth & Dawson, 2021). However, DG has a higher computational cost than FEM methods, due to the increase in degrees of freedom, particularly in time-dependent systems (Bui-Thanh, 2015; Kang et al., 2020; Kubatko et al., 2009).

Hybridizable Discontinuous Galerkin (HDG) methods emerged later as a natural evolution, retaining the advantageous properties of DG while improving computational efficiency through static condensation (Samii et al., 2019; Valseth & Dawson, 2021). Although HDG belongs to the FEM family, its properties and advantages have led to the widespread use of methods from the finite element family in fluid dynamics computation (Cockburn, 2023). For example, the HDG method was developed for the Navier-Stokes equations by Fernández (2019), Nguyen et al. (2011), Ueckermann and Lermusiaux (2016), and Vo (2017). Cockburn (2023) presents a review of the evolution of HDG methods. The HDG method was introduced recently by Cockburn et al. (2009). HDG increases the number of unknowns, compared to DG, creating new nodes in the skeleton of the mesh (Betteridge et al., 2021). Despite increasing the number of unknowns, HDG achieves improved efficiency through a decoupling strategy, solving the system in the skeleton and then recovering the solution within the elements (Samii et al., 2019). This approach, known as Hybridization or Static Condensation, was initially developed for linear elasticity problems (Fraeijs de Veubeke, 1965) and later applied to Galerkin methods. It reduces the computational cost associated with the DG method while maintaining its properties, finding applications across various fields for PDEs like the Poisson equation, convection-diffusion equation, Stokes equation, Euler and Navier-Stokes equations, Maxwell equation, acoustics and elastodynamics and Helmholtz equation, among others (Bui-Thanh, 2016). HDG is a tool in development that is undergoing continuous enhancements (Betteridge et al., 2021; Cockburn, 2023; Ellmenreich et al., 2025; Felipe et al., 2024; Kang et al., 2020; Kirk et al., 2022; Pachev, 2024).

Regarding the modelization of the SWE, FEM has previously been applied using stabilization techniques (Danilov et al., 2004; Dawson et al., 2006; Hanert et al., 2003; Kolar et al., 1996; Le Roux et al., 1998, 2000; Lee & Froehlich, 1987; Piggott et al., 2007; Walters & Barragy, 1997; Walters, 2005; Walters & Carey, 1983; Williams & Zienkiewicz, 1981). With the development of DG, which has inherent local conservation, DG models became increasingly popular for the Navier–Stokes and SWE equations (Aizinger & Dawson, 2002;

Ern et al., 2007; Giraldo & Warburton, 2007; Giraldo et al., 2002; Kubatko et al., 2009; Pan et al., 2021; Restelli & Giraldo, 2009). DG methods were first applied to the SWE by Schwanenberg and Köngeter (2000) and are still a developing field in oceanographic applications (Dawson et al., 2024; Kang et al., 2020; Kärnä et al., 2018; Ricardo et al., 2024; Wichitrnithed et al., 2024; Wu et al., 2024).

Although the HDG method was introduced in 2009, its application to the SWE system is still very recent. Currently, only a handful of studies have implemented HDG for SWE, and there are some aspects, such as the formulation of boundary conditions, that are not yet unified or well established. The first HDG formulation for linear SWE was proposed by Bui-Thanh (2016), setting the foundations for the imposition of numerical fluxes using Godunov-type solvers. Later, Samii et al. (2019) extended the formulation to the nonlinear SWE, and Kang et al. (2020) introduced an implicit-explicit (IMEX) HDG scheme tailored for hyperbolic systems. Arabshahi (2016) studied the nonlinear SWE with a space-time HDG method. Betteridge et al. (2021) proposed a variant of this scheme. Some efforts have been made to create tools to solve SWE models with the DG and HDG methods, such as Dawson et al. (2024), Gibson et al. (2020), Kärnä et al. (2018), and Pachev (2024). In this thesis, we focus specifically on the application of HDG methods to the SWE.

Time-dependent PDEs, such as the SWE, also require a temporal discretization (Kang et al., 2020). As stated previously, the SWE system can be separated into different components with different wave velocities (Betteridge, 2020). Fast waves, like gravity waves, limit the allowed time-step size in explicit schemes. This leads to a large number of steps, therefore, increasing the computational cost. On the other hand, implicit methods can become expensive because of the resolution of non-linear systems (Bui-Thanh, 2016; Kang et al., 2020).

To solve this issue, we can use Implicit-Explicit (IMEX) methods. IMEX methods were introduced by Crouzeix (1980) for evolving parabolic equations and have been widely used. IMEX methods allow for flexibility in the time-step. Treating the slow terms explicitly and the fast terms implicitly balance the time-step size restriction from the fast waves, and the computational cost (Betteridge et al., 2021). Therefore, the SWE can be solved with an IMEX scheme, with the faster linear terms treated implicitly, and the non-linear terms treated explicitly (Kang et al., 2020).

Hyperbolic equation systems, like the SWE, need a discretization scheme that is robust and ensures stability. Strong Stability Preserving (SSP) time discretizations were developed for hyperbolic time dependent PDEs and ODEs, and are particularly recommended for hyperbolic PDEs with shocks (Gottlieb et al., 2011). They are a strong tool to preserve the stability of the system. Explicit SSP methods were first used by Gottlieb and Shu (1998), Shu

(1988), and Shu and Osher (1988), and were called Total Variation Diminishing (TVD) discretizations (Gottlieb et al., 2011). In Shu (1988) high order multistep SSP and first order SSP Runge Kutta methods are shown. A general SSP theory for multi-stage methods applied to non-linear equations has been developed by Spijker (2007).

1.2 Goals

The overall goal of this work is to build our own HDG FEM model for the SWE that can be used to simulate oceanic structures, taking advantage of the FEM and HDG characteristics. Some specific goals and contributions of this thesis are:

- Identify and assess the current needs in oceanographic modelling.
- Derive the SWE and establish the applicability of the SWE approximations for the physical processes relevant to the area of interest.
- Evaluate the performance of SSP time discretization methods, including both IMEX and non-IMEX, for the HDG SWE model.
- Formulate the weak imposition of boundary conditions in the HDG discretization of the SWE.
- Implement a one layer HDG model of the SWE to simulate coastal processes.
- Analyze the error convergence of the one layer model for benchmark problems to assess the accuracy of the method.
- Extend the one layer model to a multilayer formulation to accurately simulate the complex physical processes occurring in the area of interest.

The development presented in this thesis is not a continuation of previous modeling work within the research group, but a first implementation from scratch of a fully functional HDG model for SWE. This required a deep understanding of the model: deriving the shallow water equations from the constitutive equations of fluids, recognizing their mathematical structure as hyperbolic conservation laws, connecting this structure to the HDG spatial discretization, and mastering the IMEX-SSPRK time discretization. Beyond theory, starting from scratch also meant translating this knowledge into functional code and validating it against analytical and realistic test cases. While more time-consuming, this ground-up approach allowed for a comprehensive and original contribution that lays the groundwork for future applications of HDG in ocean modeling.

1.3 Outline

In this thesis we address the need for efficient models in oceanography and implement a HDG SWE model with SSP temporal discretization. The thesis is organized as follows. In Chapter 2 we analyze different sampling methods used on an oceanographic cruise in the North-East Atlantic Ocean, focusing on sinopticity and the spatial scale of the methods, as a measure of their efficiency capturing the oceanographic structures. The analysis proves the need for a more detailed ocean model. In Chapter 3 we develop the Shallow Water Equations, starting from the conservation equations and considering the Navier-Stokes and Shallow Water approximations. Then, we define a linear and non-linear version of the SWE. Finally, we study its properties as a conservation law. Chapter 4 contains the spatial discretization for the SWE model, using the HDG method. To understand the HDG method, we introduce first the FEM method. Then, it is improved using Discontinuous Galerkin, and, finally, Hybridizable Discontinuous Galerkin. The HDG SWE model is then written in bilinear forms. The numerical fluxes for DG and HDG are calculated using the Godunov method, and choosing the Lax-Friedrichs fluxes. Finally, the boundary conditions are explained. Then, in Chapter 5, the temporal discretization for the model is developed, using Implicit-Explicit (IMEX) methods and Strong Stability Preserving (SSP) schemes. After that, a Static Condensation approach is used for the resolution of the equation system. Chapter 6 shows the results of the model for different numerical examples. The accuracy stability of the model is calculated for three examples with an analytical solution, a translating vortex, linear standing wave, and a linear Kelvin wave. A lake at rest problem is used to make sure that the SWE model is well-balanced. Then, we solve other complex examples that are of significant interest in oceanography. Finally, in Chapter 7, we summarize the work of this thesis, the goals we have achieved and the future work.

Chapter 2

Evaluation of ocean sampling and modelling approaches to capture oceanographic features off Northwest Africa

In this chapter, we analyze the ability of the ocean models and sampling tools to capture the physical processes at various spatial scales. We also describe the physical structures found in the North Atlantic Ocean. The dynamic processes of the ocean have a wide range of spatial scales, from a few centimeters to tens of thousands of kilometers, and temporal scales, from hours to decades. In order to understand the dynamics of the ocean we need a tool capable of capturing all these scales simultaneously while maintaining synopticity.

The content of this chapter has been published during the development of this thesis in Hernández-García et al. (2024).

2.1 High resolution observations in an ocean front system

Traditionally, ocean analysis has relied on datasets obtained from ship-based measurements and moorings. In addition, remote-sensing technologies have gained widespread usage and continue to advance. However, current remote-sensing technologies primarily focus on the ocean's surface. Over the past few decades, subsurface floats such as Argo floats, remotely operated vehicles (ROVs), and autonomous underwater vehicles (AUVs) like gliders have emerged, offering a vast array of applications in oceanography (Bachmayer et al., 2004, 2006; Hernández-García et al., 2018; Ramos et al., 2018; Rudnick et al., 2004).

These state-of-the-art sampling systems now enable access to ocean phenomena across a wide range of scales, from the size of the ocean basin (approximately 10 000 km) to microscales (just a few centimeters). They are highly portable and suitable for sampling intermittent and localized phenomena, such as upwelling events. Notably, gliders offer relative control over horizontal location, allowing scientists to strategically select where to conduct profiles. Furthermore, compared to other AUVs, gliders boast lower energy consumption enabling longer and deeper operations. However, gliders are limited in terms of the sensors they can carry, necessitating the use of sensors that are compact, lightweight, and have low power consumption (Bachmayer et al., 2004; Rudnick et al., 2004).

The Cape Verde Frontal Zone (CVFZ) represents a dynamically complex region situated in the eastern North Atlantic Subtropical Gyre (20°W to 30°W, $\sim$ 20°N). This front is defined by the strong interleaving of two main water masses at surface and central levels in the North Atlantic Ocean - North Atlantic Central Water (NACW) and South Atlantic Central Water (SACW) (Barton, 1987; Burgoa et al., 2021; Pelegrí et al., 2017; Tomczak, 1981).

These water masses are defined by a range of temperature and salinity values, and can be represented with an uniform density. NACW is characterized by an almost linear relationship between temperature and salinity within the range of 11 to 18.65 °C and 35.47 to 36.76, while SACW exhibits this relationship within the range of 9.7 to 15.25 °C and 35.177 to 35.7. SACW also presents a minimum in oxygen levels, dropping below 1.5 $\mathrm{mL\,L^{-1}}$, whereas NACW exhibits low nutrient levels and high oxygen concentration (Hagen, 1985; Tomczak, 1981; Zenk et al., 1991). Below these water masses, Antarctic Intermediate Water (AAIW) becomes apparent at intermediate layers, characterized by a minimum in both salinity and oxygen. Mediterranean Water (MW), found at depths of 800-1200 m and in the form of meddies, stands out for its notably warmer and more saline properties compared to AAIW (Bashmachnikov et al., 2015; Pérez et al., 2001; Zenk et al., 1991).

The Cape Verde Front (CVF) is traditionally defined by the intersection of the 36.0 isohaline with the 150 m depth isobath (Barton, 1987; Zenk et al., 1991). This front persists throughout the year, experiencing high spatial and temporal variability and extending from 20°W to 30°W, at latitudes between 15°N and 22°N, from Cape Blanc to Cape Verde Islands (Burgoa et al., 2021; Mittelstaedt, 1983; Tomczak & Hughes, 1980; Zenk et al., 1991). A compilation of CVF positions from synoptic surveys by Zenk et al. (1991) suggested that spatial variations of the front are at least on the order of 300 km in the region off Cape Blanc.

This highly meandering thermohaline front is marked by the presence of intrusions, filaments from the upwelling system, and eddy activity (Martínez-Marrero et al., 2008). The upwelling between 20°N and 25°N exhibits maximum intensity during spring and autumn, featuring sharp gradients in temperature ($<3\,^{\circ}$C) and salinity (<0.08) at depths down to 600 m. The temperature decrease with depth compensates for the salinity decrease, resulting in both variables effectively counterbalancing their effects on density and maintaining the front's dynamic stability. In this highly variable region, high-definition sampling methods prove crucial to capture all the processes observed during sampling. It is also important to consider the sampling velocity concerning the timescales of oceanic structures. Slower sampling methods might inadvertently present temporal variability as spatial structure, challenging the perception of a vertical section as a precise snapshot of oceanic conditions (P. Hughes & Barton, 1974; Martínez-Marrero et al., 2008; Pérez-Rodríguez et al., 2001; Rudnick et al., 2004).

2.2 Methods

2.2.1 Sampling methods

The FLUXES-II cruise took place in the CVFZ between November 2nd and November 24th, 2017, utilizing the BIO Sarmiento de Gamboa. The primary objective of the FLUXES project was to investigate the hydrographic characteristics of the CVF, using both traditional and innovative instruments. The expedition involved various measurements including CTD stations, SeaSoar observations, glider deployments, ADCP recordings, and turbulence measurements, among others (Burgoa et al., 2021; Campanero et al., 2022). The spatial distribution of the data is shown in Figure 2.1.

Mercator Model

The numerical model outputs were sourced from the Copernicus Marine Environment Monitoring Service dataset GLOBAL_ANALYSIS_FORECAST _PHY _001_024 (European Union-Copernicus Marine Service, 2016). This model, which provides daily outputs, is a numerical forecasting model with temperature, salinity, currents, sea level, mixed layer depth and ice parameters (von Schuckmann et al., 2016). The dataset utilized for this study corresponds to the model outputs covering the same spatial locations and dates as the glider data. It offers a horizontal resolution of $\frac{1}{12}^{\circ}$ (0.083°) and comprises 50 vertical levels spanning from the sea surface to 5500 m depth.

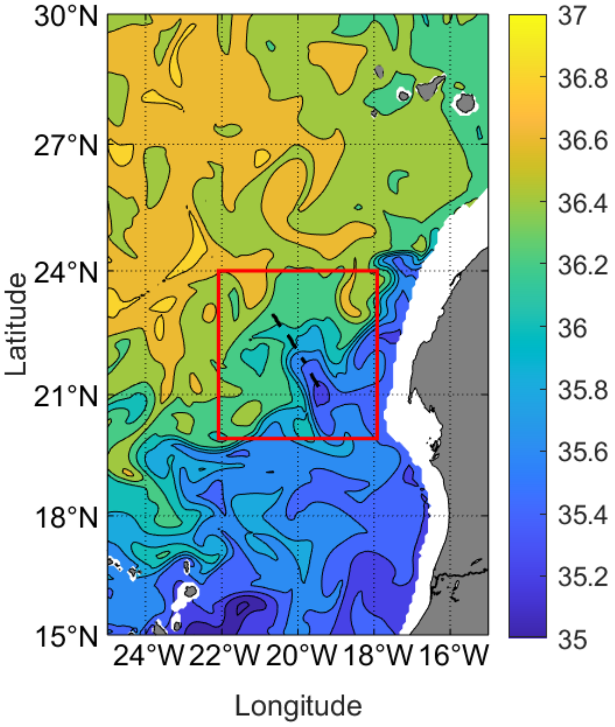


FIGURE 2.1: Map illustrating the study area for this project. Salinity data at 150 m is depicted, emphasizing the isohaline of 36 with the white contour. The black dashed line represents the transect used for this analysis, corresponding to a portion of the FLUXES-II cruise's sampling. The red square highlights the area sampled by the FLUXES-II survey. The map was generated using global Mercator model data of November 13th 2017.

Ship CTD

A total of 48 stations were sampled using the CTD rosette system, with 36 stations being utilized for this study, between November 10th and November 20th. These measurements encompassed depths from the sea surface down to 1500 m. The stations were approximately 9.26 km apart, with a vertical resolution of about 1 m (Burgoa et al., 2021; Campanero et al., 2022).

SeaSoar

The SeaSoar, a vehicle performing undulations on the water column, was deployed in a continuous towed manner behind the research vessel. The transect T5 from the cruise, conducted on November 8th, was used. In this transect the water column was sampled from the surface to approximately 400 m depth, with a horizontal resolution of approximately 2.5 km and a vertical resolution of 0.5 m (Allen et al., 2002; Burgoa et al., 2021; Campanero et al., 2022; Hales & Takahashi, 2002; Pollard, 1986).

Gliders

Two gliders, AUVs, travelling by adjusting their buoyancy to traverse the water column, were deployed between November 4th and November 21st. These gliders sampled from the surface to 1000 m, with a horizontal spatial resolution of approximately 1 km and a vertical resolution of around 1 m. The dataset from the glider Bio-584 was chosen for this study as both gliders provided similar observations (Bachmayer et al., 2006; Burgoa et al., 2021; Campanero et al., 2022).

Sea Level and Surface Velocity from Satellite Data

Sea Level Anomaly (SLA) and the velocity field estimated from satellite observations were obtained from the Copernicus Marine Environment Monitoring Service dataset SEALEVEL_ GLO_PHY_L4_REP_OBSERVATIONS_008 _047 (European Union-Copernicus Marine Service, 2021). The dataset integrates sea level anomaly data from different altimeter missions, achieving a horizontal resolution up to $\frac{1}{4}^{\circ}$.

2.2.2 Data Processing

The variables considered in this study encompass potential temperature, practical salinity, potential density, and oxygen. Practical salinity from the Ship CTD was calibrated by analyzing 51 water samples. Oxygen measurements from the CTD stations were calibrated using in situ samples, ensuring a precision of $\pm 0.53\,\mu\text{mol}\,\text{kg}^{-1}$ (Burgoa et al., 2021). Glider data were processed using the Glider Toolbox from the Balearic Islands Coastal Observing and Forecasting System (SOCIB), incorporating thermal lag correction and quality control (Troupin et al., 2015).

The location of the CVF within the depth range of 100 m to 650 m was calculated using the methodology described by Burgoa et al. (2021), defining the front location based on salinity values representing a 50% contribution of NACW and SACW.

2.2.3 Wavelet analysis

The comparison of sampling methods is conducted based on the scales they can effectively capture, allowing for a focused analysis of various processes. We utilized wavelet analysis to objectively quantify the scales accessed by the different sampling methods. Wavelet analysis is a technique that decomposes a time series into a time-frequency space, aiding in the determination of the primary frequencies of the signal. It provides a clear visual representation of the signals, presenting local values for the amplitude and phase for each harmonic of a dataset. Thus, wavelets are highly valuable for describing non-stationary processes that cannot be adequately detected using classical Fourier Transform techniques. Furthermore, wavelet analysis can be applied to signals of any size, whereas Fourier Transform necessitates a predetermined size (Baliunas et al., 1997; Combes et al., 1990; Lau & Weng, 1995; Meyers et al., 1993; Torrence & Compo, 1998; Venkata-Ramana et al., 2013; Weng & Lau, 1994).

Wavelet analysis was pioneered by Grossmann and Morlet (1984), Morlet (1983), and Morlet et al. (1982). It has found extensive use in climatic studies, climatic time series analysis (Baliunas et al., 1997; Collineau & Brunet, 1993; Gao & Li, 1993; Gollmer et al., 1995; Grinsted et al., 2004; Gu & Philander, 1995; Jevrejeva et al., 2003; Kumar & Foufoula-Georgiou, 1993; Lau & Weng, 1995; Mak, 1995; Moron et al., 1998; Torrence & Compo, 1998; Wang & Wang, 1996; Yiou et al., 2000), various ocean processes (Camayo & Campos, 2006; Farge, 1992; Gamage & Blumen, 1993; Gamage & Hagelberg, 1993; Garel et al., 2016; Liu, 1994; Meyers et al., 1993), medical research, seismic signals, image processing, and more (Torrence & Compo, 1998).

In this study, we utilized the complex continuous Morlet wavelet method. A complex wavelet enables the detection of both amplitude and phase. Continuous wavelets are particularly useful for scale analysis, although they can be redundant on larger scales (Lau & Weng, 1995; Torrence & Compo, 1998). The wavelet analysis was applied to all the different data sources: Mercator model outputs, Ship CTD, SeaSoar, and gliders.

After characterizing the primary features within the domain, we proceeded to analyze the scales captured by various sampling methods. The wavelet analysis was conducted at specific depths within the central water region: 150, 350, and 600 m (note that the SeaSoar did not sample beyond 400 m). The salinity variable was chosen for the wavelet analyses at these depths due to its well-defined signature of the frontal zone. The depth of 150 m holds significance, traditionally defining the CVF by the 36.0 isohaline at this depth (Zenk et al., 1991). The data series were obtained within the same latitudinal range spanning approximately 200 km, with the assumption that the identified features should remain consistent across the CVF, regardless of the observation methodology. The figures are presented in a NW to SE orientation.

The horizontal resolution for each sampling method, calculated as the mean distance between two consecutive profiles, is detailed in Table 2.1. Amplitudes smaller than the Nyquist frequency, which is twice the horizontal spatial resolution for each method, were excluded from the analysis.

TABLE 2.1: Spatial and temporal resolution of the different sampling methods and the numerical model

Sampling method	Horizontal spatial resolution (km)	Vertical range (m)	Sampling period (days)
Mercator model	11.5	1	17
Ship CTD	9.3	1	5
SeaSoar	2.5	0.5	1
Glider	0.9	1	17

2.3 Results

2.3.1 TS Diagram and Water Masses

Figures 2.2A and 2.2B depict the TS diagram obtained along the transect using the glider. The primary water masses identified are NACW and SACW. SACW is slightly less saline and colder than the NACW at equivalent depths. The surface data exhibited the highest variability, attributable to insolation during the summer and autumn, as well as the potential influence of water exported in filaments from the African upwelling system. Some mixed AAIW was observed at intermediate depths (deeper than 800 meters).

The majority of the data points correspond to NACW, as evidenced by the TS frequency diagram in Figure 2.2B. The TS diagrams obtained through the different sampling methods in this study exhibit consistent patterns.

2.3.2 Vertical Sections

Mercator Model

The model outputs were extracted based on the latitude and longitude range provided by the glider, as well as its temporal range, starting from the north on November 4th and ending in the south on November 21st. The model provides data down to 5500 m depth, but for this analysis, we used data down to 1000 m.

The temperature ranged from 5 to 24.8 °C and the salinity from 34.9 to 37.1 (Figures 2.3A and 2.3B). The water column appears stratified based on the temperature. On the other hand, the salinity distribution exhibited more

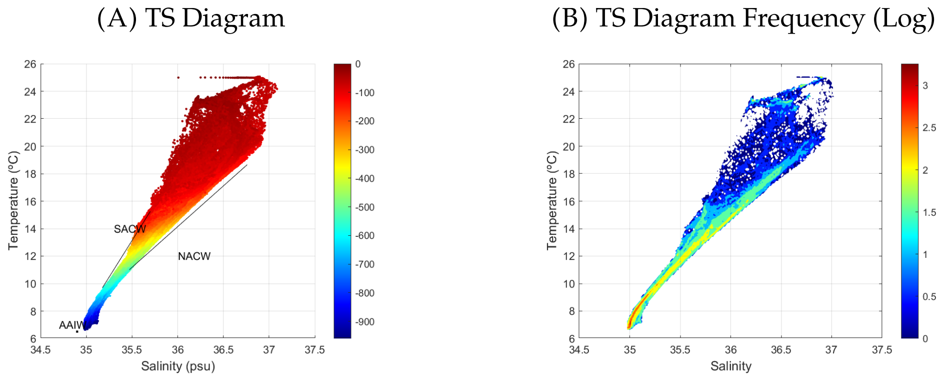


FIGURE 2.2: (A) TS diagram obtained from the glider and (B) frequency distribution of each TS cell for the glider data. Reference lines for SACW and NACW are adopted from (Tomczak, 1981). The AAIW point is sourced from (Pérez et al., 2001). Dots are color-coded based on their depth in meters in (A) and logarithm of the frequency in (B).

variability than the temperature. In the first half of the track, there was more saline water in the surface layer, while the temperature remained relatively constant in this layer throughout the sampling.

Ship CTD

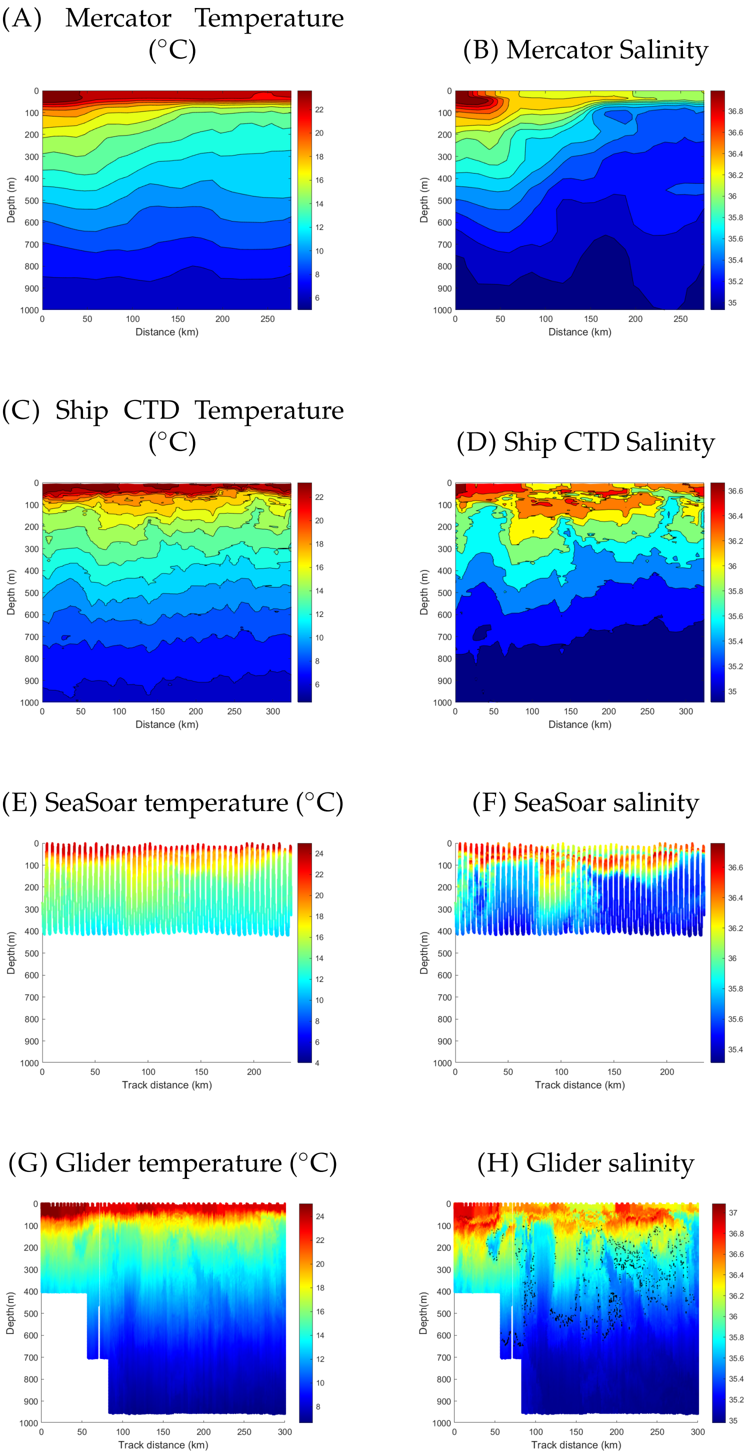
The ship performed 36 stations from 20.5°N to almost 23°N, with a separation of 9.5 km. The observations started on November 11th and ended on November 16th, reaching depths down to 1500 m, although for this analysis we only used data down to 1000 m.

The temperature ranged from 4.5 to 24.5 °C, and the salinity from 34.9 to 36.9 (Figures 2.3C and 2.3D). The temperature and salinity distributions presented intrusions of SACW, which is colder and less salty than NACW, at depths ranging from 100 m to 500 m. These intrusions were found at approximately 50 km and 150-200 km. Notably, the second intrusion appeared to be tilted.

The last 75 km (>200 km), at the lowest latitude, showed more saline water in the surface layer. The temperature in the surface layer was similar along the whole track.

SeaSoar

For this study, we utilized the transect T5 from the SeaSoar grid. It began in the north and proceeded southward, similar to all the other sampling


(A) Mercator Temperature (°C)

(B) Mercator Salinity

(C) Ship CTD Temperature (°C)

(D) Ship CTD Salinity

(E) SeaSoar temperature (°C)

(F) SeaSoar salinity

(G) Glider temperature (°C)

(H) Glider salinity

FIGURE 2.3: Vertical section of (A, D, E, G) potential temperature and (B, D, F, H) salinity obtained by the Mercator model, Ship CTD, SeaSoar and glider, along the track. The orientation is from north (left) to south (right). The black dots indicate the position of the Cape Verde Front between 100 to 650 m according to Burgoa et al. (2021).

methods, from November 8th to November 9th. The SeaSoar sampled down to a depth of 400 m.

The temperature ranged from 10.8 to 23.7 °C and the salinity from 35.3 to 36.7 (Figures 2.3E and 2.3F). In the surface layer, the water was warmer and saltier along the first 75 km than within the rest of the track.

The vertical sections revealed 3 intrusions of SACW, occurring between 25-75 km, 125-175 km, and >200 km. These intrusions began at a depth of 100 m and extended to the SeaSoar's maximum depth.

Glider

The glider initiated its southward track at 23°N and continued until 21°N. It continuously dived from the surface down to a depth of 1000 m between November 4th and November 21st.

The temperature ranged from 6.6 to 25.1 °C and the salinity from 34.9 to 37 (Figures 2.3G and 2.3H). High temperatures were observed within the surface layer (<100 m) along the entire section.

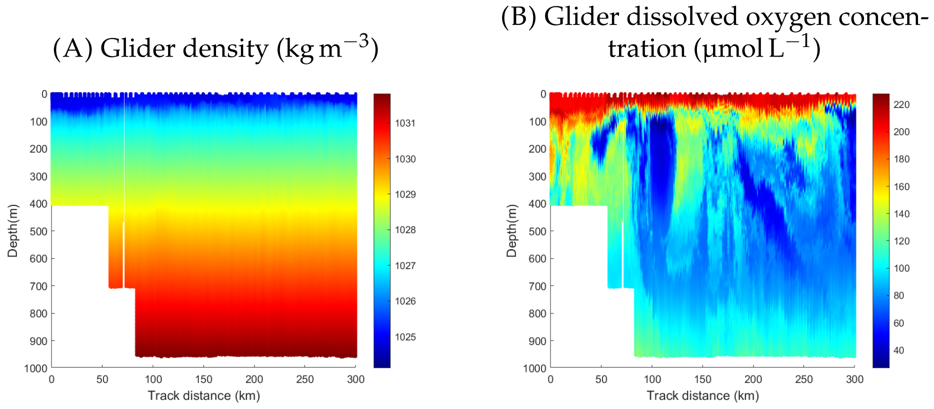
The vertical sections displayed 3 large intrusions of SACW, occurring between 75-125 km, 175-225 km, and >275 km. The second intrusion was noticeably tilted to the north. The intrusions were captured in greater detail than on the CTD and SeaSoar observations and seemed to be located between 100 and 800 m depth.

The location of the Cape Verde Front is indicated following the methodology developed by Burgoa et al. (2021), highlighting the limits between NACW and SACW.

The density distribution exhibited a relatively flat pattern along the entire section, as expected for a density-compensated front (Figure 2.4A). The density ranged from 1024.1 kg m^{-3} to 1031.8 kg m^{-3}.

The oxygen distribution (Figure 2.4B) displayed the same intrusions as the salinity and temperature distributions. The SACW intrusions were marked by a lower value of oxygen concentration. In the surface layer, the oxygen concentration remained high throughout the entire track. The oxygen concentration ranged from 26.8 to 227.8 µmol L^{-1}.

Notably, the model dataset vertical sections lacked clear features in contrast to the in situ sampling data, which exhibited similar oceanographic structures. Specifically, the Ship CTD results appeared coarsest in resolution,


(A) Glider density (kg m^{-3})

(B) Glider dissolved oxygen concentration (µmol L^{-1})

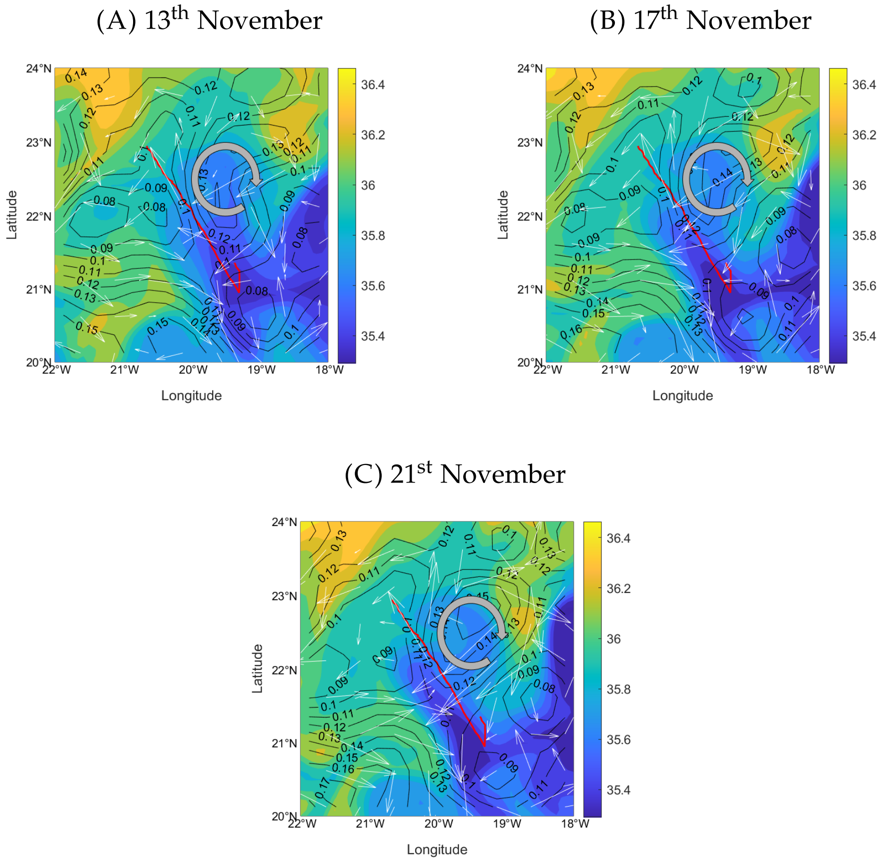
FIGURE 2.4: Vertical section of (A) density and (B) dissolved oxygen concentration obtained by the glider along its track. The orientation is from north (left) to south (right).

while the glider data offered the most detailed insights. A consistent finding across these methods was the stronger variability in the surface layer, attributed to atmospheric interaction and upwelling filaments.

Tilted intrusion

Additionally, a significant feature observed was the lateral intrusion of SACW into NACW. This intriguing tilted intrusion manifests at greater depths compared to the vertical intrusions, pointing to its possible origin as a vertical intrusion subsequently tilted by external forces. In Figure 2.5, an amalgamation of data from various datasets, including the Mercator model, SLA, and surface velocity from satellite altimetry on three distinct dates, provides a glimpse into the temporal evolution at the CVF, precisely where the tilted intrusion was identified. A dynamic signature emerges, unveiling the presence of an anticyclonic eddy previously characterized by Navarro et al. (2018), evident in both SLA and resulting surface velocities. This anticyclonic eddy could induce a baroclinic behavior, resulting in varying velocities near the surface versus higher depths, generating an along-depth variable drag. This, in turn, could distort a vertical front, ultimately giving rise to a tilted intrusion.

This feature of the tilted intrusion of SACW into NACW and its formation mechanism might have been overlooked relying solely on data from Ship CTD or SeaSoar. The interpretation of this phenomenon as a coherent oceanographic structure necessitates high-rate spatial sampling. Previous studies on the CVF have reported analogous structures. For instance, Barton (1987) identified an anticyclonic eddy with a diameter of approximately 20 km in the CVF. Besides, they also noted meandering patterns of around 200 km at the front, although a direct comparison to our results is limited by


(A) 13th November

(B) 17th November

(C) 21st November

FIGURE 2.5: Temporal evolution of the eddy found during the sampling, at the days (A) November 13th, (B) November 17th and (C) November 21st. The salinity data (colored map) are taken from the Mercator model, at 222.5 m depth. The SLA data (m), represented by the black contours, and the velocity field (m s^{-1}), represented by the white arrows, are taken from satellite data. The big grey curved arrow indicates the location of the eddy. The red line indicates the path of the glider.

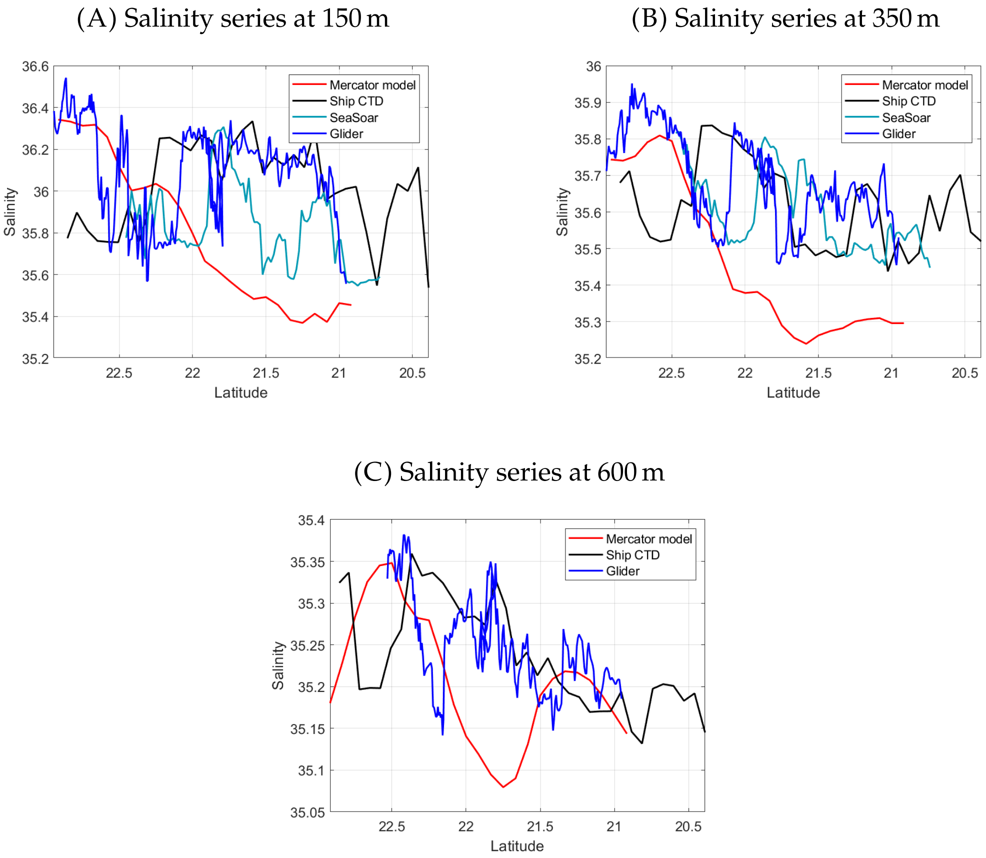
the length of our sampling track. Similarly, Koshlyakov and Grachev (1973) found an anticyclonic eddy at 16°N, 33°W with a diameter ranging from 90-200 km.

2.3.3 Wavelet analysis of in-situ signals

At 150 m, salinity fluctuated between 35.4 and 36.6 with a slightly decreasing trend (Figure 2.6A). At the 350 m sampling salinity ranges from 35.2 to 36, and a clearer decreasing trend than in 150 m was found (Figure 2.6B). At 600 m the salinity varies between 35 and 35.4, narrowing the range with depth, and a consistent decreasing trend was observed across all sampling methods (Figure 2.6C).

The SeaSoar and the glider show the higher-frequency variability in all the sampled depths, while the model shows the smoother salinity series.

It is important to note that the position of the front varied among the different sampling methods due to the highly variable dynamics of the front and the fact that the system was sampled at different moments.


(A) Salinity series at 150 m

(B) Salinity series at 350 m

(C) Salinity series at 600 m

FIGURE 2.6: Salinity series from the Mercator model, Ship CTD, SeaSoar and glider at (A) 150 m, (B) 350 m and (C) 600 m, starting at the NW (left) and finishing at the SE (right).

Analysis performed at 150 m

The wavelet analysis revealed that the numerical model captured scales within a relatively narrow band ranging from 23 to 64 km (Figure 2.7A, Table 2.2). However, these scales were not constant along the entire track, and no significant scales are identified.

The Ship CTD captured scales within a wider range of 18.6 to 128 km (Figure 2.7B, Table 2.2). The range was slightly narrower at the beginning of the signal, broadening towards the end of the path covered, capturing smaller scales. A significant scale was identified at around 32 km towards the end of the track.

The SeaSoar captured scales within the range of 5 to 64 km (Figure 2.7C, Table 2.2). However, scales smaller than 16 km appeared intermittently along the track. The scale of 32-64 km was significant at the middle of the track.

The glider sampling captured scales in the widest range, spanning from 1.8 to 128 km (Figure 2.7D, Table 2.2). Scales smaller than 8 km were not constant during the track. Significant scales around 2-4 km and at 32 km were identified at some points in the first third of the track. Additionally, a significant scale of 64 km was obtained at the beginning and middle of the track.

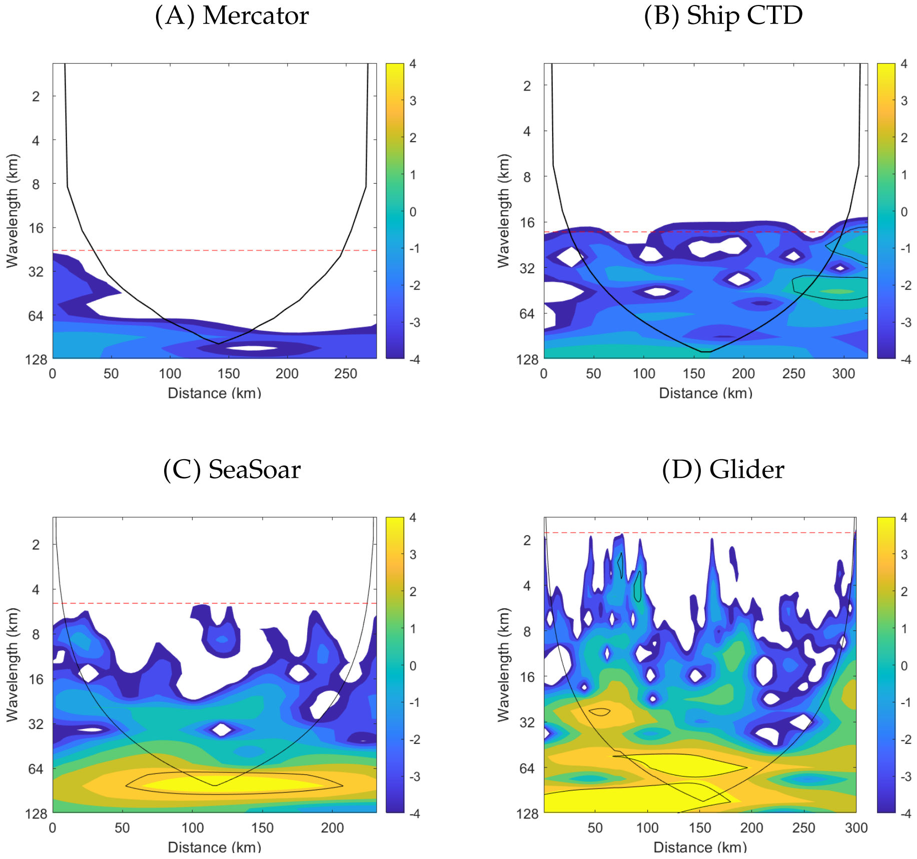


FIGURE 2.7: Wavelet analysis from the model (A), Ship CTD (B), SeaSoar (C) and glider (D) salinity at 150 m. The wavelet results on the area under the black line may be affected by edge effects and are not considered as valid. The red dotted line indicates the minimum accepted wavelength, 2*horizontal resolution (the horizontal Nyquist frequency). The black contour indicates the significant results. The Y and Z axis are logarithmic. The Z values represent the base 2 logarithm of the wavelet power spectrum. The signals start at the NW (left) and finish at the SE (right).

Analysis performed at 350 m

The Mercator model captured scales ranging from 23 and 64 km at the NW end, and around 64 km at the rest of the path (Figure 2.8A, Table 2.2), yet no significant scales were identified. The smaller scales were absent in the second half of the track.

The Ship CTD displayed scales between 18.6 and 128 km (Figure 2.8B, Table 2.2). The range slightly broadened towards the end of the track for smaller scales. However, scales smaller than 64 km were not consistently present.

The SeaSoar captured scales between 5 and 64 km (Figure 2.8C, Table 2.2). The range remained constant throughout the signal, but scales smaller than 32 km appeared intermittently. The analysis did not reveal any significant scales.

The glider exhibited the widest range of scales, spanning from 1.8 and 128 km (Figure 2.8D, Table 2.2). Towards the final segment of the transect (>200 km), the range narrowed down to 4-128 km. A significant scale is evident in the 96-128 km range for the second third of the track.

Analysis performed at 600 m

The results from the analysis at 600 m indicate that the Mercator model's sampling captured scales between 48 and 64 km, without yielding any significant scales (Figure 2.9A, Table 2.2).

The Ship CTD captured scales ranging from 18.6 to 128 km (Figure 2.9B, Table 2.2). However, the scales smaller than 32 km were not consistently present throughout the signal, and the scales greater than 64 km disappeared towards the end of the track. No significant scales were obtained.

The glider exhibited the widest band of scales, spanning between 1.8 and 128 km (Figure 2.9C, Table 2.2). The scales smaller than 16 km appeared intermittently. However, no significant scales were captured.

In summary, at all depths, the glider captured the smallest scales and recovered the widest range, while the model and the Ship CTD captured the narrowest range. The glider, followed by both the SeaSoar and Ship CTD, exhibited the highest number of significant scales and the highest spatial variability. No significant scales were captured at 600 m. At all the depths, the wavelet results of the model fell significantly below the spatial Nyquist frequency, distinguishing it from the other sampling methods. At 600 m the glider and Ship CTD exhibited a similar scale range to the other depths; however, the range of scales captured by the Mercator model was narrower.

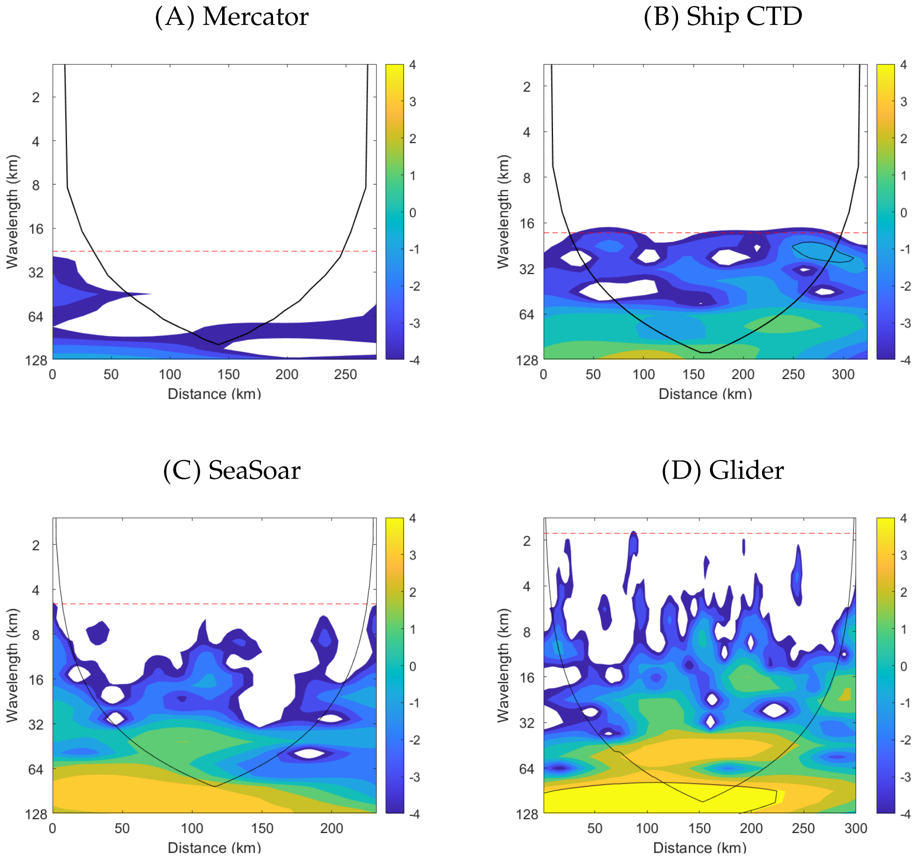


FIGURE 2.8: Wavelet analysis from the model (A), Ship CTD (B), SeaSoar (C) and glider (D) salinity at 350 m. The wavelet results on the area under the black line may be affected by edge effects and are not considered as valid. The red dotted line indicates the minimum accepted wavelength, 2*horizontal resolution (the horizontal Nyquist frequency). The black contour indicates the significant results. The Y and Z axis are logarithmic. The Z values represent the base 2 logarithm of the wavelet power spectrum. The signals start at the NW (left) and finish at the SE (right).

The SeaSoar and glider data at 150 m exhibit significant signals exceeding approximately 60-80 km in the central domain, likely linked to the meandering of the front. A significant scale was observed in the 16-32 km range at the end of the signal. At both 150 m and 350 m, significant signals of approximately 20-25 km were obtained for the Ship CTD in the vicinity of the upwelling, likely associated with lateral intrusions across the front.

In terms of sampling speed, the SeaSoar emerged as the fastest method, with a 14 h sampling, providing the closer approximation to a synoptic view. Conversely, the glider was the slowest in sampling, with a 17 days duration. The Ship CTD achieved the greatest sampling depth among all methods, while the SeaSoar maintained a shallower sampling depth. The glider stood

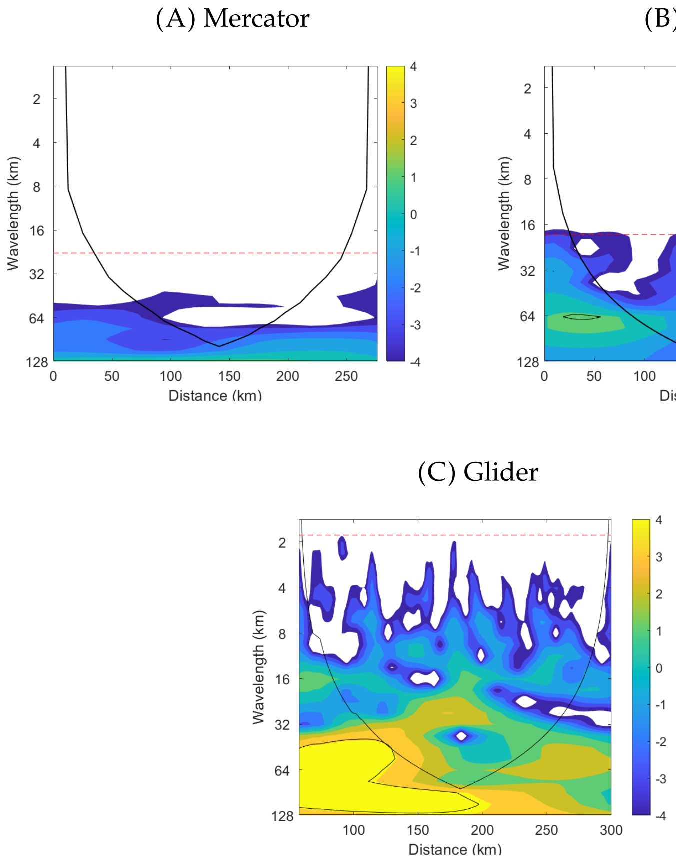


FIGURE 2.9: Wavelet analysis from the model (A), Ship CTD (B) and glider (C) salinity at 600 m. The wavelet results on the area under the black line may be affected by edge effects and are not considered as valid. The red dotted line indicates the minimum accepted wavelength, 2*horizontal resolution (the horizontal Nyquist frequency). The black contour indicates the significant results. The Y and Z axis are logarithmic. The Z values represent the base 2 logarithm of the wavelet power spectrum. The signals start at the NW (left) and finish at the SE (right).

TABLE 2.2: Range of scales (km) captured by each method at each depth.

Depth	Mercator model	Ship CTD	SeaSoar	Glider
150 m	23 - 64	18.6 - 128	5 - 64	1.8 - 128
350 m	23 - 64	18.6 - 128	5 - 64	1.8 - 128
600 m	48 - 64	18.6 - 128	-	1.8 - 128

out for its independence from the ship, enabling diverse sampling paths if required, offering a more flexible deployment and retrieval approach.

The complexity of the CVF, as evidenced in our results, necessitates fast sampling methods with high vertical and horizontal resolutions to attain an accurate, synoptic and comprehensive portrayal. Sampling independence is another major advantage. These considerations emphasize the importance of strategically selecting and integrating sampling methods to effectively capture the smaller scale dynamics of oceanic fronts like the CVF.

These results highlight the limitations of observational data alone and support the need for a robust modelling tool capable of capturing the wide range of ocean dynamics. By integrating numerical modelling with observations, we can obtain a more comprehensive understanding of the physical processes governing the ocean.

Chapter 3

Shallow Water Equations

In this chapter we derive the Shallow Water Equations (SWE). The SWE constitute an approximation of the equations governing the mechanics of a fluid, commonly used in oceanography. It assumes a shallow layer of water, with homogeneous density and a mean velocity acting on the entire layer. They allow to represent the fluid dynamics while using an equation system easier to resolve than the complete Navier-Stokes (NS) set of equations.

Starting from the conservation equations of a fluid, we impose the Navier-Stokes approximations. Then, we assume a shallow layer of water with the horizontal scale greater than the vertical scale. The variables for the equations are defined. We consider some boundary conditions at the surface and at the bottom, and the forces affecting the fluid's motion. The resulting approximations are consistent with the dominant physical processes of interest, as outlined in Chapters 1 and 2.

Then, the Shallow Water Equations are finally derived. We write the SWE with geopotential variables, and define two versions, linear and non-linear SWE. Additionally, the SWE are transformed to an uncoupled conservation law equation system. We calculate the eigenvalues and eigenvectors, defining the velocities of the waves.

3.1 Constitutive equations

The equations governing fluid mechanics are the conservation of mass, momentum and energy. The constitutive equations depend on the density, velocity, pressure and temperature. Additionally, for the ocean, the conservation of salt, and the Equation of State for Seawater relating salinity, temperature, and pressure, could be considered (Kämpf, 2009; Kundu et al., 2024).

The conservation of energy can be expressed as the conservation of density. Since the Navier-Stokes and Shallow Water equations use an incompressible fluid approximation, the mass and momentum conservation can provide the general description for a Newtonian fluid (Cushman-Roisin, 2011).

The conservation of mass, also known as the continuity equation, is based on the principle that the mass of fluid particles remains constant over time. Similarly, the conservation of momentum is developed from Newton's second law, which connects the dynamics of an object to the external forces applied to it (Kämpf, 2009; Kundu et al., 2024).

Considering a control volume $V(t)$, with a surface $A(t)$, and $\mathbf{n}$ as the outward normal, the governing equations of a fluid are:

$$\begin{cases} \frac{\partial}{\partial t} \int_{V(t)} \rho(\mathbf{x},t)\, dV = 0 \\ \frac{\partial}{\partial t} \int_{V(t)} \rho(\mathbf{x},t)\mathbf{U}(\mathbf{x},t)\, dV = \int_{V(t)} \rho(\mathbf{x},t)\mathbf{F_b}\, dV + \int_{A(t)} \mathbf{F_s}(\mathbf{n},\mathbf{x},t)\, dA \end{cases} \tag{3.1}$$

where ρ ($\mathrm{kg\,m^{-3}}$) is the density of the fluid, that moves at a velocity $\mathbf{U}$ ($\mathrm{m\,s^{-1}}$). $\rho\mathbf{U}$ represents the momentum per unit volume ($\mathrm{kg\,m\,s^{-1}}$). $\mathbf{F_b}$ and $\mathbf{F_s}$ are the body forces and surface forces, respectively.

We will consider gravity as one of the body forces, and separate $\mathbf{F_b}$ as $\mathbf{F_b} = \mathbf{g} + \mathbf{F}$.

Applying Gauss' theorem and the Reynolds transport theorem to equations (3.1), and developing the integral for an arbitrary volume, the equations become (Kundu et al., 2024):

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho\mathbf{U}) = 0 \\ \frac{\partial}{\partial t}(\rho\mathbf{U}) = -\nabla \cdot (\rho\mathbf{U} \otimes \mathbf{U}) + \mathbf{F_s} + \rho\mathbf{g} + \rho\mathbf{F} \end{cases} \tag{3.2}$$

3.2 Navier-Stokes equations

Starting from equation (3.2) and assuming an incompressible and newtonian fluid with constant density, we can obtain the Navier-Stokes equations.

A water mass is considered to not interact with the surrounding fluid. First, we can consider the water in the ocean as an incompressible fluid, i.e., the inflows and outflows must be balanced, imposing density constant in time $\frac{\partial \rho}{\partial t} = 0$.

$$\begin{cases} \nabla \cdot (\rho \mathbf{U}) = 0 \\ \frac{\partial}{\partial t}(\rho \mathbf{U}) = -\nabla \cdot (\rho \mathbf{U} \otimes \mathbf{U}) + \mathbf{F_s} + \rho \mathbf{g} + \rho \mathbf{F} \end{cases}$$

As stated in chapter 2, water masses are defined by a certain temperature, salinity and density. Thus, the density is also considered constant throughout the fluid, and therefore, it is constant in space $\frac{\partial \rho}{\partial (xyz)} = 0$. The equations can be simplified to:

$$\begin{cases} \nabla \cdot \mathbf{U} = 0 \\ \frac{\partial \mathbf{U}}{\partial t} = -\nabla \cdot (\mathbf{U} \otimes \mathbf{U}) + \frac{1}{\rho}\mathbf{F_s} + \mathbf{g} + \mathbf{F} \end{cases} \tag{3.3}$$

The water is a newtonian fluid, so the pressure must be part of the surface forces $\mathbf{F_s}$, as follows:

$$\mathbf{F_s} = \int_{A(t)} \tau \cdot \mathbf{n}\, dA = \int_{A(t)} (-p\mathbf{I} + \boldsymbol{\sigma}) \cdot \mathbf{n}\, dA \tag{3.4}$$

with p as the hydrostatic pressure, $\mathbf{I}$ as the identity matrix, and $\boldsymbol{\sigma}$ the deviatory stress tensor containing the viscosity, deformation and rotation velocities, and the thermodynamics of the fluid.

Applying Gauss's theorem to $\mathbf{F_s}$ (3.4):

$$\mathbf{F_s} = \int_{A(t)} (-p\mathbf{I} + \boldsymbol{\sigma}) \cdot \mathbf{n}\, dA = \nabla \cdot (-p\mathbf{I} + \boldsymbol{\sigma}) = -\nabla p + \nabla \cdot \boldsymbol{\sigma} \tag{3.5}$$

Substituting equation (3.5) into equation (3.3) we arrive to the Navier-Stokes equations:

$$\begin{cases} \nabla \cdot (\mathbf{U}) = 0 \\ \frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U} \otimes \mathbf{U}) = -\frac{1}{\rho}\nabla p + \frac{1}{\rho}\nabla \cdot \boldsymbol{\sigma} + \mathbf{g} + \mathbf{F} \end{cases} \tag{3.6}$$

where the first one is the continuity equations and the second one is the momentum conservation equation.

3.3 Shallow Water Equations

We will consider now a homogeneous fluid with uniform density, with an average depth H and horizontal scale λ, in a shallow layer (Figure 3.1). For the physical processes in the ocean, including those seen in Chapter 2, we can assume that the horizontal scale is much greater than the vertical scale ($H <<< \lambda$).

To derive the SWE, we will apply boundary conditions at the surface and bottom of the layer, define the relevant forces and hydrostatic pressure, and integrate the Navier–Stokes equations (3.6) along the vertical axis (z), leading to a depth-averaged velocity field.

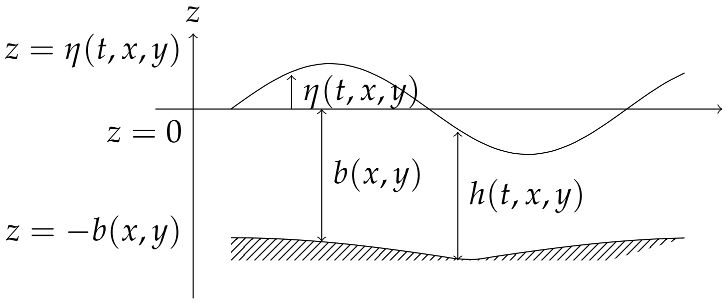


FIGURE 3.1: Shallow water scheme. $\eta(t,x,y)$ is the elevation of the free surface, $b(x,y)$ is the bathymetry, positive downward, and $h(t,x,y)$ is the total depth of the water column, as $h(t,x,y) = b(x,y) + \eta(t,x,y)$.

3.3.1 Boundary conditions

Some boundary conditions at the surface and the bottom of the fluid will be used to obtain the SWE after integrating in z. These are the following:

1. No penetration at the bottom ($z = -b(x,y)$):

   The fluid cannot penetrate the bottom surface. In the case of a water mass within our area of interest, the interaction between water masses, and the water mass and the bottom, can be neglected. When simulating the water masses in the area off Northwest Africa, we can impose no penetration at the bottom and also no penetration between water masses.

$$\mathbf{U} \cdot \mathbf{n_B} = 0 \tag{3.7}$$

with $\mathbf{n_B}$ as the normal outward vector from the bottom:

$$\mathbf{n_B} = \frac{1}{\sqrt{\left(\frac{\partial b}{\partial x}\right)^2 + \left(\frac{\partial b}{\partial y}\right)^2}} \begin{pmatrix} -\frac{\partial b}{\partial x} \\ -\frac{\partial b}{\partial y} \\ -1 \end{pmatrix} \tag{3.8}$$

Substituting (3.8) into (3.7) leads to:

$$\left(U\frac{\partial b}{\partial x} + V\frac{\partial b}{\partial y} + W \right)_{z=-b} = 0 \tag{3.9}$$

2. Stress tensor at the bottom ($z = -b(x, y)$):

   The stress tensor at the bottom, containing the bottom friction, is defined:

$$\begin{cases} \sigma_{bx} = \sigma_{xx}\frac{\partial b}{\partial x} + \sigma_{xy}\frac{\partial b}{\partial y} + \sigma_{xz} \\ \sigma_{by} = \sigma_{xy}\frac{\partial b}{\partial x} + \sigma_{yy}\frac{\partial b}{\partial y} + \sigma_{yz} \end{cases} \tag{3.10}$$

3. Pressure at the surface ($z = \eta(t, x, y)$):

   The pressure at the surface corresponds to the atmospheric pressure, and is considered spatially uniform across the ocean surface:

$$P_{z=\eta} = P_{\text{atm}} \tag{3.11}$$

4. No penetration at the surface ($z = \eta(t, x, y)$):

   There is no outward flow at the fluid surface. In the case of a water mass within our area of interest, the interaction between the water masses and the surface, and between water masses can be neglected as well.

$$\mathbf{U} \cdot \mathbf{n}_\eta = \frac{\partial \eta}{\partial t} \tag{3.12}$$

   with $\mathbf{n}_\eta$ as the normal, outward vector of the surface of the fluid:

$$\mathbf{n}_\eta = \frac{1}{\sqrt{\left(\frac{\partial \eta}{\partial x}\right)^2 + \left(\frac{\partial \eta}{\partial y}\right)^2}} \begin{pmatrix} -\frac{\partial \eta}{\partial x} \\ -\frac{\partial \eta}{\partial y} \\ 1 \end{pmatrix} \tag{3.13}$$

   Substituting (3.13) into (3.12) leads to:

$$\left(\frac{\partial \eta}{\partial t} + U\frac{\partial \eta}{\partial x} + V\frac{\partial \eta}{\partial y} - W \right)_{z=\eta} = 0 \tag{3.14}$$

5. Stress tensor at the surface ($z = \eta(t, x, y)$):

   The stress tensor for the surface friction, for external forcings like, for example, wind stress, is:

$$\begin{cases} \sigma_{sx} = -\sigma_{xx}\frac{\partial \eta}{\partial x} - \sigma_{xy}\frac{\partial \eta}{\partial y} + \sigma_{xz} \\ \sigma_{sy} = -\sigma_{xy}\frac{\partial \eta}{\partial x} - \sigma_{yy}\frac{\partial \eta}{\partial y} + \sigma_{yz} \end{cases} \tag{3.15}$$

3.3.2 Forces into consideration

We must define the forces **F** from equation (3.6). The forces we will develop for the SWE are the Coriolis force, for the body forces, and the stress tensor σ, for the surface forces.

Coriolis force

We need to consider the Earth's non-inertial frame and its rotation, which are relevant for the physics at oceanic time and length scales, by applying the Coriolis acceleration. The Coriolis acceleration, corresponding to the centrifugal force from the Earth's rotation is (Kämpf, 2009; Kundu et al., 2024):

$$\frac{\partial U}{\partial t} = 2\Omega V, \qquad \frac{\partial V}{\partial t} = -2\Omega U$$

In the Earth's rotation, the magnitude of the Coriolis force is dependent on latitude, with f, the Coriolis parameter, defined as $f = 2\Omega \sin\theta$, where θ is the latitude, and Ω the angular velocity of the Earth. For spatial scales of 100 km or less, the curvature of the Earth can be ignored and the value of f can be considered constant. This is called the f-plane approximation.

It deflects the fluid's path to the right on the Northern Hemisphere, and to the left on the Southern Hemisphere, and it becomes 0 at the equator. The Coriolis force can be written as follows:

$$\mathbf{F} = -f \times \mathbf{U}^{\perp} = \begin{bmatrix} f \cdot V \\ -f \cdot U \end{bmatrix}$$

Stress tensor

σ is the deviatoric stress tensor, and depends on the velocity gradient tensor $\frac{\partial u_i}{\partial x_j}$. The term $\frac{\partial u_i}{\partial x_j}$ can be decomposed as the symmetric (**S**) and antisymmetric (**R**) parts:

$$\frac{\partial u_i}{\partial x_j} = \frac{1}{2}S_{ij} + \frac{1}{2}R_{ij}$$

with:

$$S_{ij} = \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \qquad R_{ij} = \frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}$$

By definition, stress only develops on fluid elements that change shape, so, only the symmetric part of the tensor should be considered in the fluid constitutive equation. The antisymmetric part corresponds to the pure rotation of the fluid elements.

Finally, we will define the tensor σ, for an incompressible and newtonian fluid, with μ as the viscosity coefficient, as (Kundu et al., 2024) :

$$\sigma = \mu \left(\frac{\partial U_j}{\partial x_i} + \frac{\partial U_i}{\partial x_j} \right)$$

This expression, expanded, is:

$$\sigma = \mu \begin{pmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} & 2\frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial w}{\partial x} + \frac{\partial u}{\partial z} & \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} & 2\frac{\partial w}{\partial z} \end{pmatrix}$$

3.3.3 Hydrostatic pressure

We need to apply the hydrostatic pressure to the SWE. For a fluid at rest, the downward gravity force and the upward pressure derivative force are balanced. This is called the hydrostatic balance. Generally, processes with the horizontal scale considerably greater than the vertical scale can be considered hydrostatic (Kämpf, 2009).

We will consider a layer of fluid with depth H, with an horizontal bathymetry, setting $z = 0$ at the surface, and η as the displacement of the surface (see Figure 3.1).

We will apply a scale analysis to the z-momentum equation (3.6):

$$\frac{\partial W}{\partial t} = -\left(\frac{\partial(UW)}{\partial x} + \frac{\partial(VW)}{\partial y} + \frac{\partial W^2}{\partial z}\right) - \frac{1}{\rho}\frac{\partial p}{\partial z} + \frac{1}{\rho}\nabla \cdot \sigma - g + F_z \tag{3.16}$$

We consider the typical values for the ocean: $\widetilde{U} = 10^{-1}$ as the horizontal velocity scale, $\widetilde{L} = 10^6$ the horizontal scale, and $\widetilde{H} = 10^3$ the vertical scale. Applying them to the continuity equation, we arrive to $\widetilde{W} = 10^{-4}$ for the vertical velocity scale.

Applying these values to equation (3.16), the terms of the temporal and spatial derivative of W, body forces F_z, and viscosity σ are in the orders between 10^{-15} and 10^{-6} and can be discarded, leaving us with only the pressure derivative and gravity terms, obtaining the equation for the hydrostatic balance:

$$\frac{1}{\rho}\frac{\partial P}{\partial z} = -g \tag{3.17}$$

Then, integrating (3.17) between the free surface, η, and the depth, z, and considering uniform density:

$$\frac{1}{\rho}\int_z^\eta \frac{\partial P}{\partial z}\,dz = \int_z^\eta -g\,dz$$

Solving the integrals we obtain the relation:

$$P(\eta) - P(z) = -\rho g(\eta - z) \tag{3.18}$$

With this relation, the pressure at any point in the water column depends on the height of the water column above that point. We impose the pressure at the surface as the atmospheric pressure (equation (3.11)) into equation (3.18).

$$P(z) = \rho g(\eta - z) + P_{\text{atm}}$$

This results in the horizontal pressure gradients:

$$\frac{\partial p}{\partial x} = \rho g \frac{\partial \eta}{\partial x}; \qquad \frac{\partial p}{\partial y} = \rho g \frac{\partial \eta}{\partial y} \tag{3.19}$$

3.3.4 Derivation of the SWE

Once we have obtained the forces and boundary conditions, we need to apply them to obtain the SWE. The Navier-Stokes equations (3.6) are integrated in the vertical z axis.

Continuity equation

First, we take the continuity equation (3.6) and integrate it vertically through the water column:

$$\nabla \cdot (\mathbf{U}) = 0$$

We must define the depth averaged velocities:

$$\overline{U}(x,y,t) = \frac{1}{h}\int_{-b}^{\eta} U\,dz, \qquad \overline{V}(x,y,t) = \frac{1}{h}\int_{-b}^{\eta} V\,dz \tag{3.20}$$

Integrating the continuity equation gives:

$$\int_{-b}^{\eta} \nabla \cdot (\mathbf{U})\,dz = 0$$

$$\int_{-b}^{\eta} \left(\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} + \frac{\partial W}{\partial z} \right) dz = 0$$

The integral can be separated:

$$\int_{-b}^{\eta} \frac{\partial U}{\partial x}\,dz + \int_{-b}^{\eta} \frac{\partial V}{\partial y}\,dz + \int_{-b}^{\eta} \frac{\partial W}{\partial z}\,dz = 0$$

The third integral has a direct solution. For the integrals for U and V we can apply the Leibniz rule, since η and d depend on (x,y).

$$\frac{\partial}{\partial x}\int_{-b}^{\eta} U\,dz - \left(U_{z=\eta}\frac{\partial \eta}{\partial x} + U_{z=-b}\frac{\partial b}{\partial x}\right) + \frac{\partial}{\partial y}\int_{-b}^{\eta} V\,dz - \left(V_{z=\eta}\frac{\partial \eta}{\partial y} + V_{z=-b}\frac{\partial b}{\partial y}\right) + \left(W_{z=\eta} - W_{z=-b}\right) = 0$$

Now, applying the boundary conditions (3.14) and (3.9), and considering the depth averaged velocities (3.20), we get to the continuity equation for the SWE:

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(h\overline{U}) + \frac{\partial}{\partial y}(h\overline{V}) = 0 \tag{3.21}$$

Momentum conservation equation

We will integrate the left and right sides of the equation independently.

First, the left side of the x-momentum conservation equation, from equation (3.6) is:

$$\frac{\partial U}{\partial t} + \frac{\partial U^2}{\partial x} + \frac{\partial UV}{\partial y} + \frac{\partial UW}{\partial z}$$

Integrating from the bottom to the surface:

$$\int_{-b}^{\eta}\left(\frac{\partial U}{\partial t} + \frac{\partial U^2}{\partial x} + \frac{\partial UV}{\partial y} + \frac{\partial UW}{\partial z}\right) dz$$

Separating the terms of the integral:

$$\int_{-b}^{\eta}\frac{\partial U}{\partial t}\,dz + \int_{-b}^{\eta}\frac{\partial U^2}{\partial x}\,dz + \int_{-b}^{\eta}\frac{\partial UV}{\partial y}\,dz + \int_{-b}^{\eta}\frac{\partial UW}{\partial z}\,dz$$

The last term has a direct integral. The Leibniz rule is applied to the other terms:

$$\frac{\partial}{\partial t}\int_{-b}^{\eta} U\,dz - \left(U_{z=\eta}\frac{\partial \eta}{\partial t} + U_{z=-b}\frac{\partial b}{\partial t}\right) + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^2\,dz - \left(U^2_{z=\eta}\frac{\partial \eta}{\partial x} + U^2_{z=-b}\frac{\partial b}{\partial x}\right) + \frac{\partial}{\partial y}\int_{-b}^{\eta} UV\,dz - \left(UV_{z=\eta}\frac{\partial \eta}{\partial y} + UV_{z=-b}\frac{\partial b}{\partial y}\right) + UW_{z=\eta} - UW_{z=-b}$$

Applying the boundary conditions (3.14) and (3.9):

$$\frac{\partial}{\partial t}\int_{-b}^{\eta} U\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^2\,dz + \frac{\partial}{\partial y}\int_{-b}^{\eta} UV\,dz$$

The velocities U and V can be separated into the average and the deviation.

$$U(x,y,z,t) = \overline{U}(x,y,t) + U^*(x,y,z,t)$$

Applying the distributive property the integrals can be expanded:

$$\begin{aligned}\frac{\partial}{\partial t}\int_{-b}^{\eta} U\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} \overline{U}\cdot\overline{U}\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} \overline{U}\cdot U^*\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot\overline{U}\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot U^*\,dz \\ + \frac{\partial}{\partial x}\int_{-b}^{\eta} \overline{U}\cdot\overline{V}\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} \overline{U}\cdot V^*\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot\overline{V}\,dz + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot V^*\,dz\end{aligned}$$

Since the depth-averaged velocity $\overline{U}$ does not depend on z it can be taken out of the integrals. Applying its definition (3.20):

$$\begin{aligned}\frac{\partial}{\partial t}(h\overline{U}) + \frac{\partial}{\partial x}(h\overline{U}^2) + \frac{\partial}{\partial x}\left(\overline{U}\cdot\int_{-b}^{\eta} U^*\,dz\right) + \frac{\partial}{\partial x}\left(\overline{U}\cdot\int_{-b}^{\eta} U^*\,dz\right) + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot U^*\,dz \\ + \frac{\partial}{\partial y}(h\overline{UV}) + \frac{\partial}{\partial x}\left(\overline{U}\cdot\int_{-b}^{\eta} V^*\,dz\right) + \frac{\partial}{\partial x}\left(\overline{V}\cdot\int_{-b}^{\eta} U^*\,dz\right) + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot V^*\,dz\end{aligned}$$

The integral of the deviation of the velocity U^* and U^* must be zero.

$$\frac{\partial}{\partial t}(h\overline{U}) + \frac{\partial}{\partial x}(h\overline{U}^2) + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot U^*\,dz + \frac{\partial}{\partial y}(h\overline{UV}) + \frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot V^*\,dz$$

The terms $\frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot U^*\,dz$ and $\frac{\partial}{\partial x}\int_{-b}^{\eta} U^*\cdot V^*\,dz$ represent the vertical shear and the turbulence effects, and can be neglected, as their size is much smaller than the other terms. This approximation is suitable for the representation of mesoscale and large scale structures observed in Chapter 2.

$$\frac{\partial}{\partial t}(h\overline{U}) + \frac{\partial}{\partial x}(h\overline{U}^2) + \frac{\partial}{\partial y}(h\overline{UV}) \tag{3.22}$$

Second, the right side of the x-momentum conservation equation, from equation (3.6) is:

$$fV - \frac{1}{\rho}\frac{\partial P}{\partial x} + \frac{1}{\rho}\left(\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z}\right)$$

Integrating it:

$$\int_{-d}^{\eta}\left(fV - \frac{1}{\rho}\left(\frac{\partial P}{\partial x} + \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + \frac{\partial \sigma_{xz}}{\partial z}\right)\right) dz$$

Separating the terms:

$$\int_{-d}^{\eta} fV\, dz + \int_{-b}^{\eta} -\frac{1}{\rho}\frac{\partial P}{\partial x}\, dz + \int_{-b}^{\eta} \frac{1}{\rho}\frac{\partial \sigma_{xx}}{\partial x}\, dz + \int_{-b}^{\eta} \frac{1}{\rho}\frac{\partial \sigma_{xy}}{\partial y}\, dz + \int_{-b}^{\eta} \frac{1}{\rho}\frac{\partial \sigma_{xz}}{\partial z}\, dz$$

The density of a water mass is assumed to be constant within it. Considering f and ρ constant with depth, they can be taken outside the integrals:

$$f\int_{-b}^{\eta} V\, dz + \frac{1}{\rho}\left(-\int_{-b}^{\eta}\frac{\partial P}{\partial x}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xx}}{\partial x}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xy}}{\partial y}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xz}}{\partial z}\, dz\right)$$

Applying the hydrostatic pressure (3.19) to the pressure horizontal gradient:

$$f\int_{-b}^{\eta} V\, dz + \frac{1}{\rho}\left(-\rho g\int_{-b}^{\eta}\frac{\partial \eta}{\partial x}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xx}}{\partial x}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xy}}{\partial y}\, dz + \int_{-b}^{\eta}\frac{\partial \sigma_{xz}}{\partial z}\, dz\right)$$

Solving directly the second and the last integral and applying the Leibniz theorem to the terms with σ:

$$\begin{aligned} &f\int_{-b}^{\eta} V\, dz - gh\frac{\partial \eta}{\partial x} + \frac{1}{\rho}\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xx}\, dz - \frac{1}{\rho}\left(\sigma_{xx|z=\eta}\frac{\partial \eta}{\partial x} + \sigma_{xx|z=-b}\frac{\partial b}{\partial x}\right) \\ &+\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{xy}\, dz - \frac{1}{\rho}\left(\sigma_{xy|z=\eta}\frac{\partial \eta}{\partial y} + \sigma_{xy|z=-b}\frac{\partial b}{\partial y}\right) + \frac{1}{\rho}\sigma_{xz|z=\eta} + \frac{1}{\rho}\sigma_{xz|z=-b} \end{aligned}$$

With the boundary conditions (3.15) and (3.10), and substituting the depth-averaged velocities (3.20):

$$fh\overline{V} - gh\frac{\partial \eta}{\partial x} + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xx}\,dz - \sigma_{bx} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{xy}\,dz + \sigma_{sx}\right) \tag{3.23}$$

The second term, considering that $\eta = h - b$, can be transformed applying the chain rule:

$$-gh\frac{\partial \eta}{\partial x} = gh\frac{\partial b}{\partial x} - \frac{1}{2}g\frac{\partial h^2}{\partial x}$$

Then, equation (3.23) is:

$$fh\overline{V} + gh\frac{\partial b}{\partial x} - \frac{1}{2}g\frac{\partial h^2}{\partial x} + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xx}\,dz - \sigma_{bx} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{xy}\,dz + \sigma_{sx}\right) \tag{3.24}$$

Joining both sides of the equation, (3.22) and (3.24), we arrive to the x-momentum equation:

$$\begin{aligned}\frac{\partial}{\partial t}(h\overline{U}) + \frac{\partial}{\partial x}(h\overline{U}^2 + \frac{1}{2}gh^2) + \frac{\partial}{\partial y}(h\overline{UV}) &= fh\overline{V} + gh\frac{\partial b}{\partial x}\\ &+\frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xx}\,dz - \sigma_{bx} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{xy}\,dz + \sigma_{sx}\right)\end{aligned} \tag{3.25}$$

The y-momentum equation is integrated similarly, obtaining:

$$\begin{aligned}\frac{\partial}{\partial t}(h\overline{V}) + \frac{\partial}{\partial x}(h\overline{UV}) + \frac{\partial}{\partial y}(h\overline{V}^2 + \frac{1}{2}gh^2) &= -fh\overline{U} + gh\frac{\partial b}{\partial y}\\ &+\frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xy}\,dz - \sigma_{by} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{yy}\,dz + \sigma_{sy}\right)\end{aligned} \tag{3.26}$$

Shallow Water Equations

Joining the modified equations (3.21), (3.25) and (3.26), we arrive to the Shallow Water Equations. The depth-averaged velocities $\overline{U}$ and $\overline{V}$ will be rewritten as U and V for clarity.

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}(hU^2 + \frac{1}{2}gh^2) + \frac{\partial}{\partial y}(hUV) \\ \qquad = fhV + gh\frac{\partial b}{\partial x} + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xx}\,dz - \sigma_{bx} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{xy}\,dz + \sigma_{sx}\right) \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}(hUV) + \frac{\partial}{\partial y}(hV^2 + \frac{1}{2}gh^2) \\ \qquad = -fhU + gh\frac{\partial b}{\partial y} + \frac{1}{\rho}\left(\frac{\partial}{\partial x}\int_{-b}^{\eta}\sigma_{xy}\,dz - \sigma_{by} + \frac{\partial}{\partial y}\int_{-b}^{\eta}\sigma_{yy}\,dz + \sigma_{sy}\right) \end{cases} \tag{3.27}$$

We will now slightly modify the form of the equations to make them more suitable for the model. First, we will start on equations (3.27), but leaving only the bathymetry gradient term on the right hand side for now:

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}(hU^2 + \frac{1}{2}gh^2) + \frac{\partial}{\partial y}(hUV) = gh\frac{\partial b}{\partial x} \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}(hUV) + \frac{\partial}{\partial y}(hV^2 + \frac{1}{2}gh^2) = gh\frac{\partial b}{\partial y} \end{cases}$$

We will substitute the velocity terms with momentum:

$$U = \frac{hU}{h}, \qquad V = \frac{hV}{h}$$

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}\left(\frac{(hU)^2}{h} + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y}\left(\frac{(hU)(hV)}{h}\right) = gh\frac{\partial b}{\partial x} \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}\left(\frac{(hU)(hV)}{h}\right) + \frac{\partial}{\partial y}\left(\frac{(hV)^2}{h} + \frac{1}{2}gh^2\right) = gh\frac{\partial b}{\partial y} \end{cases}$$

Next, the absolute water depth h is replaced by the sum of the sea-level elevation η and the bathymetry b when h is not multiplying (U, V).

$$h = (b + \eta)$$

$$\begin{cases} \frac{\partial(b+\eta)}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}\left(\frac{(hU)^2}{(b+\eta)} + \frac{1}{2}g(b+\eta)^2\right) + \frac{\partial}{\partial y}\left(\frac{(hU)(hV)}{(b+\eta)}\right) = g(b+\eta)\frac{\partial b}{\partial x} \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}\left(\frac{(hU)(hV)}{(b+\eta)}\right) + \frac{\partial}{\partial y}\left(\frac{(hV)^2}{(b+\eta)} + \frac{1}{2}g(b+\eta)^2\right) = g(b+\eta)\frac{\partial b}{\partial y} \end{cases}$$

Expanding the squares and reorganizing the terms, we arrive to:

$$\begin{cases} \frac{\partial(b+\eta)}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}\left(\frac{(hU)^2}{(b+\eta)} + g\eta b + \frac{1}{2}g\eta^2\right) + \frac{\partial}{\partial y}\left(\frac{(hU)(hV)}{(b+\eta)}\right) = g\eta\frac{\partial b}{\partial x} \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}\left(\frac{(hU)(hV)}{(b+\eta)}\right) + \frac{\partial}{\partial y}\left(\frac{(hV)^2}{(b+\eta)} + g\eta b + \frac{1}{2}g\eta^2\right) = g\eta\frac{\partial b}{\partial y} \end{cases}$$

The bathymetry b is considered constant in time, and can be removed from the time derivative on the left side:

$$\begin{cases} \frac{\partial \eta}{\partial t} + \frac{\partial}{\partial x}(hU) + \frac{\partial}{\partial y}(hV) = 0 \\ \frac{\partial}{\partial t}(hU) + \frac{\partial}{\partial x}\left(\frac{(hU)^2}{(b+\eta)} + g\eta b + \frac{1}{2}g\eta^2\right) + \frac{\partial}{\partial y}\left(\frac{(hU)(hV)}{(b+\eta)}\right) = g\eta\frac{\partial b}{\partial x} \\ \frac{\partial}{\partial t}(hV) + \frac{\partial}{\partial x}\left(\frac{(hU)(hV)}{(b+\eta)}\right) + \frac{\partial}{\partial y}\left(\frac{(hV)^2}{(b+\eta)} + g\eta b + \frac{1}{2}g\eta^2\right) = g\eta\frac{\partial b}{\partial y} \end{cases}$$

Finally, we will work with geopotential variables, multiplying our η, b, hu, and hv by gravity (g).

$$\phi = g\eta, \quad \phi_B = gb, \quad u = ghU = (\phi + \phi_B)U, \quad v = ghV = (\phi + \phi_B)V$$

$$\begin{cases} \frac{\partial}{\partial t}\left(\frac{\phi}{g}\right) + \frac{\partial}{\partial x}\left(\frac{u}{g}\right) + \frac{\partial}{\partial y}\left(\frac{v}{g}\right) = 0 \\ \frac{\partial}{\partial t}\left(\frac{u}{g}\right) + \frac{\partial}{\partial x}\left(\frac{\frac{u^2}{g^2}}{\left(\frac{\phi+\phi_B}{g}\right)} + g\frac{\phi}{g}\frac{\phi_B}{g} + \frac{1}{2}g\left(\frac{\phi}{g}\right)^2\right) + \frac{\partial}{\partial y}\left(\frac{\left(\frac{u}{g}\right)\left(\frac{v}{g}\right)}{\left(\frac{\phi+\phi_B}{g}\right)}\right) = g\frac{\phi}{g}\frac{1}{g}\frac{\partial \phi_B}{\partial x} \\ \frac{\partial}{\partial t}\left(\frac{v}{g}\right) + \frac{\partial}{\partial x}\left(\frac{\left(\frac{u}{g}\right)\left(\frac{v}{g}\right)}{\left(\frac{\phi+\phi_B}{g}\right)}\right) + \frac{\partial}{\partial y}\left(\frac{\frac{v^2}{g^2}}{\left(\frac{\phi+\phi_B}{g}\right)} + g\frac{\phi}{g}\frac{\phi_B}{g} + \frac{1}{2}g\left(\frac{\phi}{g}\right)^2\right) = g\frac{\phi}{g}\frac{1}{g}\frac{\partial \phi_B}{\partial y} \end{cases}$$

The common term $\frac{1}{g}$ can be taken out of all of the terms, leaving us with the final equation, without the forcings:

$$\begin{cases} \frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x}(u) + \frac{\partial}{\partial y}(v) = 0 \\ \frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{u^2}{\phi+\phi_B} + \phi_B\phi + \frac{1}{2}\phi^2\right) + \frac{\partial}{\partial y}\left(\frac{uv}{\phi+\phi_B}\right) = \phi\frac{\partial \phi_B}{\partial x} \\ \frac{\partial v}{\partial t} + \frac{\partial}{\partial x}\left(\frac{uv}{\phi+\phi_B}\right) + \frac{\partial}{\partial y}\left(\frac{v^2}{\phi+\phi_B} + \phi_B\phi + \frac{1}{2}\phi^2\right) = \phi\frac{\partial \phi_B}{\partial y} \end{cases}$$

Recovering the Coriolis force, the final form of the SWE we are using in the model is:

$$\begin{cases} \frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{u} = 0 \\ \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \left(\frac{\mathbf{u} \otimes \mathbf{u}}{\phi + \phi_B} + \left(\phi_B \phi + \frac{1}{2}\phi^2 \right) \mathbf{I} \right) = \phi \nabla \phi_B - f \mathbf{u}^T \end{cases}$$

The equations can be written in compact form as:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{q}) = \mathbf{s}(\mathbf{q}) \tag{3.28}$$

$$\mathbf{q} = \begin{bmatrix} \phi \\ u \\ v \end{bmatrix}, \qquad \mathbf{s}(\mathbf{q}) = \begin{bmatrix} 0 \\ \phi \frac{\partial \phi_B}{\partial x} + f v \\ \phi \frac{\partial \phi_B}{\partial x} - f u \end{bmatrix}$$

$$\mathbf{F}(\mathbf{q}) = \begin{bmatrix} u & v \\ \frac{u^2}{\phi + \phi_B} + \phi_B \phi + \frac{1}{2}\phi^2 & \frac{uv}{\phi + \phi_B} \\ \frac{uv}{\phi + \phi_B} & \frac{V^2}{\phi + \phi_B} + \phi_B \phi + \frac{1}{2}\phi^2 \end{bmatrix}$$

This form of the PDE is known as the conservative form, and corresponds to a conservation law. In section 3.4, we will explore conservation laws in greater detail and discuss its mathematical properties.

3.3.5 Linearized SWE

In order to linearize the SWE, the non-linear terms of the velocity are removed. This assumption is valid when the processes are represented only by gravity waves. The tensor σ is also considered non-linear. The SWE (3.28) can be linearized to:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}_{\mathrm{L}}(\mathbf{q}) = \mathbf{s}(\mathbf{q}) \tag{3.29}$$

$$\mathbf{F}_{\mathrm{L}}(\mathbf{q}) = \begin{bmatrix} u & v \\ \phi_B \phi & 0 \\ 0 & \phi_B \phi \end{bmatrix}$$

As we have determined in Chapters 2 and 3, the Navier-Stokes and shallow water hypotheses can accurately represent the area of interest in this thesis. If we were to represent the microscale, where the turbulence is a key process, we can not neglect those terms. In order to simulate a body of water with high stratification, like the central waters with a thermocline, the temperature, salinity and density are not considered constant in depth, and we need to use a multilayer model, dividing the water mass into smaller layers. For the multilayer SWE model the pressure for the intermediate and bottom layers at their surface is not the atmospheric pressure, but depends on the superior layer. Finally, the linear approximation is not suitable for fully capturing the complex processes occurring in the area of interest, however, it can still be applied to simpler simulations.

3.4 SWE as a conservation law

In this section, we will show that a system of conservation laws, like the SWE, can be rewritten as an uncoupled system of advection equations. We will also establish the connection between the characteristic speeds of these advection equations and the eigenvalues of the Jacobian of the Flux vector of the conservation laws. For simplicity, we use a one dimensional version of the SWE.

SWE, in the compact form (3.28), is a conservation law.

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{q}) = 0 \tag{3.30}$$

To transform the system into an uncoupled system, first we can apply the chain rule to the spatial derivative:

$$\frac{\partial \mathbf{F}(\mathbf{q})}{\partial x} = \frac{\partial \mathbf{F}(\mathbf{q})}{\partial \mathbf{q}} \cdot \frac{\partial \mathbf{q}}{\partial x}$$

Then, defining $\mathbf{A}$ as the Jacobian matrix of $\mathbf{F}$:

$$\mathbf{A} = \frac{\partial}{\partial \mathbf{q}}(\mathbf{F}, \mathbf{n}) \tag{3.31}$$

The conservation law (3.30) can be written as (Bui-Thanh, 2015):

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{q}}{\partial x} = 0 \tag{3.32}$$

To decouple the system, we need to diagonalize the matrix $\mathbf{A}$, calculating the eigenvalues (λ) and eigenvectors ($\mathbf{p}$), which satisfy $|\mathbf{A} - \lambda\mathbf{I}| = 0$ and $\mathbf{A}\mathbf{p} = \lambda\mathbf{p}$:

$$\mathbf{A} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1} \tag{3.33}$$

with $\mathbf{D}$ as the diagonal matrix of eigenvalues and $\mathbf{P}$ the matrix of eigenvectors.

Now, we can define a new unknown $\mathbf{w}$ for the uncoupled system:

$$\mathbf{w} = \mathbf{P}^{-1}\mathbf{q} \tag{3.34}$$

If we derive (3.34) spatially we obtain:

$$\frac{\partial \mathbf{w}}{\partial x} = \mathbf{P}^{-1}\frac{\partial \mathbf{q}}{\partial x} \tag{3.35}$$

Substituting (3.35) into the conservation law equation (3.32), it becomes:

$$\mathbf{P}\frac{\partial \mathbf{w}}{\partial t} + \mathbf{A}\mathbf{P}\frac{\partial \mathbf{w}}{\partial x} = 0$$

Then, if the terms are premultiplied by $\mathbf{P}^{-1}$ we obtain:

$$\mathbf{P}^{-1}\mathbf{P}\frac{\partial \mathbf{w}}{\partial t} + \mathbf{P}^{-1}\mathbf{A}\mathbf{P}\frac{\partial \mathbf{w}}{\partial x} = 0$$

Now, applying the relation (3.33) and considering that $\mathbf{P}^{-1} \cdot \mathbf{P} = \mathbf{I}$, we arrive to the system:

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{D}\frac{\partial \mathbf{w}}{\partial x} = 0$$

Since $\mathbf{D}$ is the diagonal matrix of eigenvalues, this system can be separated as:

$$\begin{cases} \frac{\partial w_1}{\partial t} + \lambda_1 \frac{\partial w_1}{\partial x} = 0 \\ \dots \\ \frac{\partial w_n}{\partial t} + \lambda_n \frac{\partial w_n}{\partial x} = 0 \end{cases}$$

This is a decoupled system, formed by transport equations. The eigenvalues (λ_i) of the flux Jacobian matrix correspond to the characteristic wave speeds of the system, and thus determine the direction and magnitude of wave propagation. A system is considered hyperbolic when all its eigenvalues are real.

3.4.1 Calculation of eigenvalues for the SWE

As a hyperbolic partial differential equations system, the SWE admits wave-like solutions that propagate at finite speeds. To find these speeds, we need to calculate the eigenvalues of the matrix $\mathbf{A}$ for the SWE's linear and non-linear versions. To derive the Jacobian matrix $\mathbf{A}$, we will express it as $\mathbf{A} = \mathbf{A_1} \cdot n_x + \mathbf{A_2} \cdot n_y$, where:

$$\mathbf{A_1} = \frac{\partial}{\partial \mathbf{q}}(\mathbf{F_x}), \qquad \mathbf{A_2} = \frac{\partial}{\partial \mathbf{q}}(\mathbf{F_y})$$

Linear SWE

For our linear SWE (3.29), $\mathbf{A_1}$ and $\mathbf{A_2}$ are:

$$\mathbf{A_1} = \begin{bmatrix} 0 & 1 & 0 \\ \phi_B & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad \mathbf{A_2} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ \phi_B & 0 & 0 \end{bmatrix}$$

The eigenvalues will be, from lowest to highest:

$$\begin{cases} \lambda_1 = -\sqrt{\phi_B}\sqrt{n_x^2 + n_y^2} = -\sqrt{\phi_B} \\ \lambda_2 = 0 \\ \lambda_3 = \sqrt{\phi_B}\sqrt{n_x^2 + n_y^2} = \sqrt{\phi_B} \end{cases}$$

with the maximum eigenvalue $\lambda_{\max} = \sqrt{\phi_B}$.

Non-linear SWE

For the non-linear SWE (3.28), $\mathbf{A_1}$ and $\mathbf{A_2}$ are:

$$\mathbf{A_1} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{u^2}{(\phi+\phi_B)^2} + \phi_B + \phi & \frac{2u}{(\phi+\phi_B)} & 0 \\ -\frac{uv}{(\phi+\phi_B)^2} & \frac{v}{(\phi+\phi_B)} & \frac{u}{(\phi+\phi_B)} \end{bmatrix}, \quad \mathbf{A_2} = \begin{bmatrix} 0 & 0 & 1 \\ -\frac{uv}{(\phi+\phi_B)^2} & \frac{v}{(\phi+\phi_B)} & \frac{u}{(\phi+\phi_B)} \\ -\frac{v^2}{(\phi+\phi_B)^2} + \phi_B + \phi & 0 & \frac{2v}{(\phi+\phi_B)} \end{bmatrix}$$

The eigenvalues are:

$$\begin{cases} \lambda_1 = |\mathbf{U} \cdot \mathbf{n}| - \sqrt{(\phi + \phi_B)} \\ \lambda_2 = |\mathbf{U} \cdot \mathbf{n}| \\ \lambda_3 = |\mathbf{U} \cdot \mathbf{n}| + \sqrt{(\phi + \phi_B)} \end{cases}$$

and the maximum eigenvalue is $\lambda = |\mathbf{U} \cdot \mathbf{n}| + \sqrt{(\phi + \phi_B)}$. Remember that $\mathbf{U}$ corresponds to the primitive velocity ($\mathbf{u} = gh\mathbf{U}$).

The analysis of the eigenvalues of both the linear and non-linear versions of the SWE confirms the hyperbolic nature of the system, since all eigenvalues are real.

Chapter 4

Hybridizable Discontinuous Galerkin method

The Hybridizable Discontinuous Galerkin (HDG) is a method from the family of Finite Elements (FE) methods, used to solve PDEs. FEM methods provide advantages, such as permitting flexibility in the mesh. Discontinuous Galerkin (DG) methods make the domain discontinuous between the elements, ensuring both global and local conservation. Then, HDG methods reduce the computational cost.

In this chapter, we obtain the weak form of the equations, define the discretized domain, and the bases for the function spaces. Then, the DG and HDG methods are applied to the Shallow Water Equations (SWE).

DG and HDG require the use of a numerical flux between elements. A numerical flux is calculated using the Godunov method, by solving the Riemann problem, and then the parameters are chosen using the Lax-Friedrichs scheme. The boundary conditions for the model are described. Finally, the equations are written in bilinear forms, and the splitting of the velocities is developed.

4.1 Finite Elements

Finite Element Methods (FEM) are a numerical tool used for solving partial differential equations (PDEs) by discretizing the domain into elements through an unstructured mesh. It provides a continuous solution function over the entire domain. The flexibility in the mesh makes FEM particularly suitable for simulations requiring adaptability to complex geometries (Sayas, 2015).

The FEM method is based on the weak form of the equation, and the Galerkin method for discretization, specifically using Lagrange bases.

4.1.1 Weak form

In FEM, the equations are typically reformulated into their weak form, also known as the variational formulation, relaxing the derivative requirements.

This reformulation is essential for several reasons. First, the weak form allows the solution to belong to a broader class of functions that may not possess classical derivatives, which is particularly important when dealing with sharp gradients or discontinuities, as often found in geophysical flows. By integrating the equations against test functions and applying integration by parts, the order of the derivatives on the solution is reduced, enabling the use of simpler basis functions such as piecewise polynomials. Additionally, the weak form naturally incorporates boundary conditions into the formulation and provides a framework that ensures local and global conservation properties. It also lays the foundation for applying the Galerkin method, which is at the core of the finite element approach.

Now, we will change the SWE from the strong form of a conservation equation, to the weak form. To do so, we will start from the SWE in the conservation law form (3.28).

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{q}) = \mathbf{s}(\mathbf{q}), \qquad \mathbf{q} = [\phi, u, v]$$

We apply the inner product, with a test function v, in both sides of the equation. This equation is integrated along the domain Ω.

$$\int_\Omega \frac{\partial \mathbf{q}}{\partial t} \cdot \mathbf{v}\, dx + \int_\Omega \nabla \cdot \mathbf{F}(\mathbf{q}) \cdot \mathbf{v}\, dx = \int_\Omega \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx \qquad \forall \mathbf{v} \in V$$

$$\mathbf{v} = \left[\psi, \omega_1, \omega_2\right]$$

with the test functions defined in an infinite space $V = \{\mathbf{v} \in L^2\}$. The equation must be true for any test function v.

The integral of the divergence $\nabla \cdot \mathbf{F}(\mathbf{q})$ is expanded by applying Gauss's theorem, obtaining the weak form of the equation:

$$\int_\Omega \frac{\partial \mathbf{q}}{\partial t} \cdot \mathbf{v}\, dx - \int_\Omega \mathbf{F}(\mathbf{q}) \cdot \nabla \cdot \mathbf{v}\, dx + \int_{\partial\Omega} (\mathbf{F}(\mathbf{q}) \cdot \mathbf{n}) \cdot \mathbf{v}\, dS = \int_\Omega \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx \qquad \forall \mathbf{v} \in V \tag{4.1}$$

Finally, to solve the PDE problem a boundary condition must be specified. A boundary of the type $\mathbf{q}(x)|_{\partial\Omega} = C$ is a Dirichlet boundary and can be added to the formulation explicitly, imposing directly the boundary values in those nodes. A boundary defined $\mathbf{q}'(x)|_{\partial\Omega} = C$ is a Neumann boundary and must be added to the formulation implicitly, throughout the integral forms of the weak form (Brenner & Scott, 2008).

4.1.2 Galerkin Method

The weak form must be discretized into a finite subspace of V in order to solve the system. We define a finite subspace V_h with $\mathbf{v_h}$ as the basis functions, such as:

$$\mathbf{v_h} \in V_h$$

In the Galerkin method, the continuous solution $\mathbf{q}$ is also defined in the space V_h. This minimizes the approximation error generated with the discretization and can be expressed as (Brenner & Scott, 2008):

$$\mathbf{q_h} = \sum_{i=1}^{n} \mathbf{q}_i \cdot \mathbf{v_i}$$

Now, the unknown we are solving for is not the function $\mathbf{q}$, but the coefficients $\mathbf{q}_i$. Then, equation (4.1) becomes:

$$\begin{aligned} \int_\Omega \frac{\partial \mathbf{q_h}}{\partial t} \cdot \mathbf{v_h}\, dx - \int_\Omega \mathbf{F}(\mathbf{q_h}) \cdot \nabla \cdot \mathbf{v_h}\, dx + \int_{\partial\Omega} (\mathbf{F}(\mathbf{q_h}) \cdot n) \cdot \mathbf{v_h}\, dS \\ = \int_\Omega \mathbf{s}(\mathbf{q_h}) \cdot \mathbf{v_h}\, dx \qquad \forall \mathbf{q_h}, \mathbf{v_h} \in V_h \end{aligned} \tag{4.2}$$

4.1.3 Continuous Galerkin

The original FEM method, also known as Continuous Galerkin (CG) method, defines the finite space V_h, discretizing the domain Ω using a mesh, located in $\mathbb{R}^d$, with d as the dimension. The boundaries of the domain are $\partial\Omega$. A CG mesh is shown in Figure 4.1.

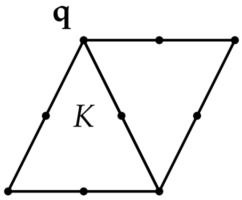


FIGURE 4.1: Structure of a CG mesh. The points represent the nodes of the mesh, where the unknowns $\mathbf{q}$ are imposed. The elements K are situated in the interior of the triangles.

The domain is discretized into a mesh τ, formed by non-overlapping polyhedral elements K with element faces ∂K. The discretized domain Ω_h, an approximation of the original domain Ω, is the sum of all the elements:

$$\Omega_h = \bigcup_{e=1}^{N_{el}} K_e$$

The skeleton of the mesh ε is the union of the edges of the elements. It includes the interfaces between elements ε^{int} and the boundary faces $\varepsilon^{\text{bound}}$, shown in Figure 4.2, and can be defined as:

$$\varepsilon^{\text{int}} = \bigcup_{K\in\tau} (\partial K \backslash \partial\Omega), \qquad \varepsilon^{\text{bound}} = \bigcup_{K\in\tau} (\partial K \cap \partial\Omega), \qquad \varepsilon = \varepsilon^{\text{int}} \cup \varepsilon^{\text{bound}} \quad (4.3)$$

with $\cup$ as the union (all elements from both sets), $\backslash$ as the difference (all elements in one set that are not in the other), and $\cap$ as the intersection (the elements common to both sets).

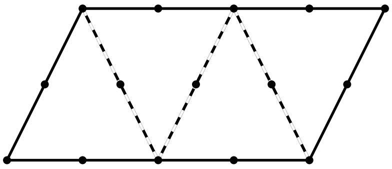


FIGURE 4.2: Boundaries of a CG mesh. The continuous line represents the boundary faces, $\varepsilon^{\text{bound}}$, and the dashed line corresponds to the interior boundaries between the elements, ε^{int}.

Once the mesh is defined, the bases of the space V_h, contained in the mesh, must be defined. In CG the bases ψ must be polynomial, continuous in the domain, piecewise functions. They must be linearly independent (Brenner & Scott, 2008). The functions ψ_i are associated to nodes x_i, and must satisfy:

$$\psi_i(x_j) = \delta_{ij}$$

with δ_{ij} as the Kronecker delta:

$$\delta_{ij} = \begin{cases} 1 & \text{if} \quad i = j \\ 0 & \text{if} \quad i \neq j \end{cases}$$

so the value of the function ψ_i is 1 on the associated node x_i, and 0 in the rest of the nodes (Figure 4.3).

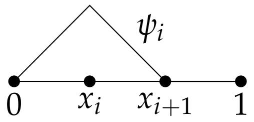


FIGURE 4.3: Scheme of the bases for FEM. The black dots represent the nodes of the mesh. The line represents the function of the base ψ_i, corresponding to node x_i, with the value of 1 in that node and 0 in the rest.

The CG method uses Lagrangian bases. Lagrangian bases satisfy these requirements and are defined as (Brenner & Scott, 2008):

$$\Psi_i(x) = \prod_{j=1, j\neq i}^{n} \frac{x - x_j}{x_i - x_j} = \frac{x - x_1}{x_i - x_1} \cdots \frac{x - x_{i-1}}{x_i - x_{i-1}} \cdot \frac{x - x_{i+1}}{x_i - x_{i+1}} \cdots \frac{x - x_n}{x_i - x_n}$$

The CG method for the SWE is:

$$\int_\Omega \frac{\partial \mathbf{q_h}}{\partial t} \cdot \mathbf{v_h}\, dx - \int_\Omega \mathbf{F}(\mathbf{q_h}) \cdot \nabla \cdot \mathbf{v_h}\, dx + \int_{\partial\Omega} (\mathbf{F}(\mathbf{q_h}) \cdot n) \cdot \mathbf{v_h}\, dS = \int_\Omega \mathbf{s}(\mathbf{q_h}) \cdot \mathbf{v_h}\, dx \qquad \forall \mathbf{q_h}, \mathbf{v_h} \in V_{h,\Psi} \tag{4.4}$$

where the space $V_{h,\Psi}$ is formed by the basis Ψ_i.

4.2 Discontinuous Galerkin

While FEM ensures global conservation between the domain and its boundaries, it does not guarantee local conservation within individual elements

(Sayas, 2015). In DG the elements K of the domain are solved individually, allowing for discontinuities between them, ensuring both global and local conservation, which is a significant advantage over FEM. The solution is approximated by element-by-element polynomial functions, without global continuity requirement, making the solution discontinuous at element interfaces (Javadzadeh Moghtader, 2016).

DG combines advantages of classical Finite Volume (FV) and FEM methods (Bui-Thanh, 2015). DG methods offer several advantages, such as high accuracy, suitability for arbitrary meshes, global mass conservation, conditionally stable finite element discretizations, and local h and p-adaptivity. The element-by-element discretization makes the method compact, making it suitable for parallelization (Kubatko et al., 2009; Valseth & Dawson, 2021). For sufficiently smooth solutions, higher-order DG methods are highly efficient, exhibiting exponential convergence rates as the polynomial degree of the approximation increases (Betteridge et al., 2021).

In the DG discretization a numerical flux $\mathbf{F}^*(\mathbf{q})$ must be introduced linking the elements at the element boundaries ∂K. It will be calculated in section 4.4. The numerical flux might add artificial diffusion, which is not straightforward to remove (Betteridge et al., 2021). Therefore, it is important to choose an appropriate flux.

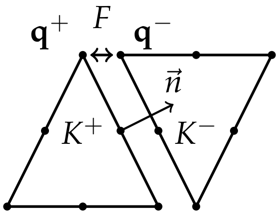


FIGURE 4.4: Structure of a DG mesh. The points represent the nodes of the mesh, where the unknowns $\mathbf{q}$ are imposed. The elements K^+ and K^- are situated in the interior of the triangles. The arrow represents the normal outward vector of element K^+. The arrow with two points corresponds to the flux F linking both elements.

For the DG spatial discretization we will assign K^+ and K^- to both elements sharing an edge ε, with $\mathbf{q}^+$ and $\mathbf{q}^-$ as the solution in their degrees of freedom, and $\mathbf{F}^*(\mathbf{q})$ as the flux linking the elements (as indicated in Figure 4.4).

We need to define some operators between a cell K^+ and its neighbour K^-, with $\mathbf{n}$ the unit outward normal vector of each cell, such that $\mathbf{n}^+ = -\mathbf{n}^-$. For a scalar s and a vector $\mathbf{a}$ we will define the average and jump, respectively:

$$\begin{cases} \{\{s\}\} = \frac{1}{2}(s^+ + s^-); \quad \{\{\mathbf{a}\}\} = \frac{1}{2}(\mathbf{a}^+ + \mathbf{a}^-) \\ [\![s]\!] = s^+\mathbf{n}^+ + s^-\mathbf{n}^- = s^+\mathbf{n}^+ - s^-\mathbf{n}^+; \quad [\![\mathbf{a}]\!] = \mathbf{a}^+\mathbf{n}^+ + \mathbf{a}^-\mathbf{n}^- = \mathbf{a}^+\mathbf{n}^+ - \mathbf{a}^-\mathbf{n}^+ \end{cases}$$

The SWE (4.2) with the DG method becomes:

$$\int_K \frac{\partial \mathbf{q}}{\partial t} \cdot \mathbf{v}\, dx - \int_K \mathbf{F}(\mathbf{q}) \cdot \nabla \cdot \mathbf{v}\, dx + \int_{\partial K} (\mathbf{F}^*(\mathbf{q}) \cdot \mathbf{n}) \cdot \mathbf{v}\, dS$$
$$= \int_K \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx \qquad \forall \mathbf{v} \in V_h, \quad \forall K \in \tau \tag{4.5}$$

The flux $\mathbf{F}^*(\mathbf{q})$ will be calculated in section 4.4.

4.3 Hybridizable Discontinuous Galerkin

In DG methods, the increase in degrees of freedom, compared to classic FEM, makes them computationally expensive (Kubatko et al., 2009). The introduction of the numerical flux at element boundaries leads to off-diagonal matrix entries, complicating the use of semi-implicit time-stepping schemes (Betteridge et al., 2021).

HDG introduces a new set of unknowns in order to reduce the cost of the method. It reduces the number of coupled degrees of freedom substantially and provides optimal convergence for convection-diffusion problems. The new unknowns are introduced on the mesh skeleton through a new variable, $\hat{\mathbf{q}}$ defined on the cell boundaries (Betteridge et al., 2021; Bui-Thanh, 2015, 2016).

This approach allows for a two-step solution process known as static condensation. First, the system is solved on the skeleton. Once the skeleton unknowns are solved, the DG unknowns can be recovered in an element-by-element fashion, completely independent of each other. This decoupling method is further developed in section 5.5. The decoupling leads to significant computational savings, particularly for large-scale simulations (Betteridge et al., 2021; Bui-Thanh, 2015, 2016).

The new set of unknowns $\hat{\mathbf{q}}$ does not belong to the discretized volume space V_h. We need to define a trace space W_h, with the associated test functions $\hat{\mathbf{v}}$.

$$\hat{\mathbf{v}} = [\hat{\psi}, \hat{\omega}_1, \hat{\omega}_2]$$

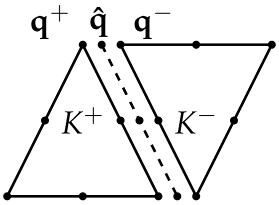


FIGURE 4.5: Structure of an HDG mesh. The points represent the nodes of the mesh, where the unknowns $\mathbf{q}$ are imposed. The dotted line is the skeleton of the mesh, associated to the unknowns $\hat{\mathbf{q}}$. The elements K^+ and K^- are situated in the interior of the triangles.

$$\hat{\mathbf{q}}, \hat{\mathbf{v}} \in W_h, \qquad W_h = \{\hat{\mathbf{v}} \in L^2\}$$

Now, for each element, we need to define the numerical flux at the boundary. The numerical flux will be $\hat{\mathbf{F}}(\mathbf{q}, \hat{\mathbf{q}})$ and will depend on the solution $\mathbf{q}$ in the element, and the skeleton solution $\hat{\mathbf{q}}$ (see Figure 4.5). It will be calculated in section 4.4.

Since a new unknown $\hat{\mathbf{q}}$ has been added, a new equation is needed to close the system. The equation balancing the solution $\mathbf{q}$ at the elements and at the skeleton $\hat{\mathbf{q}}$ is added.

The equations for the SWE (4.2) with the HDG method are:

$$\begin{cases} \int_K \frac{\partial \mathbf{q}}{\partial t} \cdot \mathbf{v}\, dx - \int_K \mathbf{F}(\mathbf{q}) \cdot \nabla \cdot \mathbf{v}\, dx + \int_{\partial K} (\hat{\mathbf{F}}(\mathbf{q}) \cdot n) \cdot \mathbf{v}\, dx \\ \qquad\qquad = \int_K \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx \qquad \forall \mathbf{v} \in V_h, \quad \forall K \in \tau \\ \int_\varepsilon \{\{\hat{\mathbf{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}\}\} \cdot \hat{\mathbf{v}}\, dS = 0 \qquad \forall \hat{\mathbf{v}} \in W_h \end{cases} \tag{4.6}$$

The spatial discretization of the Shallow Water Equations using the HDG method yields a system of ordinary differential equations (ODEs) in time, which must then be discretized using a suitable time integration scheme.

4.4 Numerical fluxes

In this section we will obtain the DG and HDG numerical fluxes using the Godunov method, by solving the Riemann problem for a generic conservation law equation. Specifically, we use the Lax-Friedrichs scheme. Then, the numerical fluxes for the SWE are obtained, using the corresponding eigenvalues.

4.4.1 Calculation of numerical fluxes

The numerical fluxes $\mathbf{F}^*$ and $\hat{\mathbf{F}}$ will be calculated using the Godunov method. Using a proper numerical flux ensures a stable and convergent solution for the PDE (Samii et al., 2019).

Remembering that a conservation law can be written as (3.32) :

$$\frac{\partial \mathbf{q}}{\partial t} + \mathbf{A}\frac{\partial \mathbf{q}}{\partial x} = 0$$

The Riemann problem, for a conservation law, using a PDE with an initial condition that creates a discontinuity at $x = 0$, is:

$$\begin{cases} \frac{\partial \mathbf{q}(x,t)}{\partial t} + \mathbf{A}\frac{\partial \mathbf{q}(x,t)}{\partial x} = 0 \\ \mathbf{q}(x,0) = \mathbf{q_0}(x) = \begin{cases} \mathbf{q}^- & \text{if} \quad x < 0 \\ \mathbf{q}^+ & \text{if} \quad x > 0 \end{cases} \end{cases} \tag{4.7}$$

Assigning $\mathbf{q} = \mathbf{q}^-$ for $\mathbf{n} < 0$, and $\mathbf{q} = \mathbf{q}^+$ for $\mathbf{n} > 0$, we obtain a Riemann problem, shown in Figure 4.6. We will solve the Riemann problem to obtain the numerical fluxes.

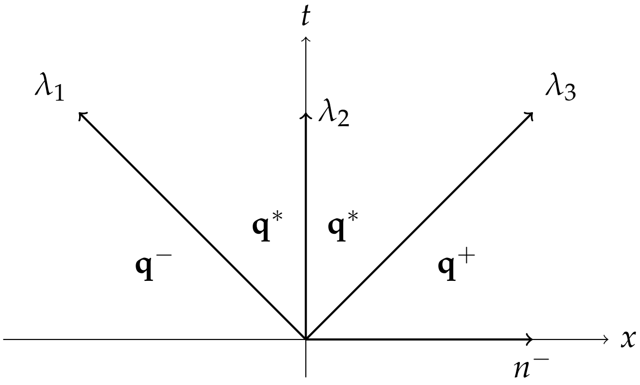


FIGURE 4.6: Scheme of the Riemann problem. λ_1, λ_2 and λ_3 are the eigenvalues, corresponding to the velocities of the equation system. $\mathbf{q}^-$ is the solution imposed for $\mathbf{n} < 0$, and $\mathbf{q}^+$ the solution for $\mathbf{n} > 0$. $\mathbf{q}^*$ is the unknown to be solved.

The eigenvectors P are forming a base for this problem. The initial conditions for the Riemann problem (4.7) can be decomposed using them, with α_i as the coefficients (Bui-Thanh, 2015):

$$\begin{cases} \mathbf{q}^- = \sum_{i=1}^m \alpha_i^- \mathbf{P_i} \\ \mathbf{q}^+ = \sum_{i=1}^m \alpha_i^+ \mathbf{P_i} \end{cases} \tag{4.8}$$

Then, applying (4.8), the solution to the Riemann problem for a small time t is:

$$\mathbf{q}^* = \sum_{i=1}^{I} \alpha_i^- \mathbf{P_i} + \sum_{i=I+1}^{m} \alpha_i^+ \mathbf{P_i} \tag{4.9}$$

with m as the number of elements of $\mathbf{q}$.

The negative eigenvalues must be associated to $\mathbf{q}^-$ and the positive eigenvalues with $\mathbf{q}^+$. I is defined such as λ_I<0 and $\lambda_{I+1} > 0$, for the ordered eigenvalues.

The Riemann solution (4.9) can be expressed in terms of either the left or the right state, as follows:

$$\mathbf{q}^* = \mathbf{q}^- + \sum_{i=1}^{I}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i}; \qquad \sum_{i=1}^{I}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i} = \mathbf{q}^* - \mathbf{q}^- \tag{4.10}$$

$$\mathbf{q}^* = \mathbf{q}^+ + \sum_{i=I+1}^{m}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i}; \qquad \sum_{i=I+1}^{m}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i} = \mathbf{q}^* - \mathbf{q}^+$$

Remembering $\mathbf{Aq} = \mathbf{F}(\mathbf{q})$, the following relation is true:

$$\mathbf{F}^* \cdot \mathbf{n} = \mathbf{Aq}^*$$

Then, the flux on the boundary of an element K^- is:

$$\mathbf{F}^* \cdot \mathbf{n}^- = \mathbf{Aq}^- + \mathbf{A}\sum_{i=1}^{I}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i} \tag{4.11}$$

The positive part of $\mathbf{A}$ has no contribution to the second term. The flux (4.11) can also be expressed as:

$$\mathbf{F}^* \cdot \mathbf{n}^- = \mathbf{Aq}^- - |\mathbf{A}|\sum_{i=1}^{I}(\alpha_i^+ - \alpha_i^-)\mathbf{P_i} \tag{4.12}$$

with $|\mathbf{A}| = \mathbf{P}|\mathbf{D}|\mathbf{P}^{-1}$. Substituting the Riemann solution (4.10) into the sum term of this equation, the flux (4.12) becomes:

$$\mathbf{F}^* \cdot \mathbf{n}^- = \mathbf{A}\mathbf{q}^- - |\mathbf{A}|(\mathbf{q}^* - \mathbf{q}^-)$$

Replacing $\mathbf{A}\mathbf{q} = \mathbf{F}(\mathbf{q})$ and rearranging the signs:

$$\mathbf{F}^* \cdot \mathbf{n}^- = \mathbf{F}(\mathbf{q}^-) \cdot \mathbf{n}^- + |\mathbf{A}|(\mathbf{q}^- - \mathbf{q}^*) \tag{4.13}$$

We have obtained the flux $\mathbf{F}^* \cdot \mathbf{n}$. However, $\mathbf{q}^*$ is still unknown. To eliminate it from the flux, we also develop the flux for element K^+:

$$\mathbf{F}^* \cdot \mathbf{n}^+ = \mathbf{F}(\mathbf{q}^+) \cdot \mathbf{n}^+ + |\mathbf{A}|(\mathbf{q}^+ - \mathbf{q}^*) \tag{4.14}$$

Then, subtracting the fluxes (4.13) and (4.14), and applying the relation $\mathbf{n}^- = -\mathbf{n}^+$, we obtain the symmetric form of the Godunov flux, linking each side of the edge between two elements.

$$\mathbf{F}^*(\mathbf{q}^+, \mathbf{q}^-) \cdot \mathbf{n}^- = \{\{\mathbf{F}(\mathbf{q})\}\} \cdot \mathbf{n}^- + \frac{1}{2}|\mathbf{A}|(\mathbf{q}^- - \mathbf{q}^+) \tag{4.15}$$

In the case of the HDG flux, it links the element and the skeleton of the mesh. Treating $\mathbf{q}^*$ from the Godunov flux as the extra unknown and solving it in the skeleton of the mesh ($\hat{\mathbf{q}}$), it is no longer unknown, so we can apply directly the flux equation (4.13):

$$\hat{\mathbf{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n} = \mathbf{F}(\mathbf{q}) \cdot \mathbf{n} + |\mathbf{A}|(\mathbf{q} - \hat{\mathbf{q}}) \tag{4.16}$$

Applying the Lax-Friedrichs scheme, the fluxes only use the maximum eigenvalue $\lambda_{\max}$. The fluxes (4.15) and (4.16) become:

$$\mathbf{F}^{*\mathrm{LF}}(\mathbf{q}^+, \mathbf{q}^-) \cdot \mathbf{n}^- = \{\{\mathbf{F}(\mathbf{q})\}\} \cdot \mathbf{n}^- + \frac{1}{2}\lambda_{\max}(\mathbf{q}^- - \mathbf{q}^+) \tag{4.17}$$

$$\hat{\mathbf{F}}^{\mathrm{LF}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n} = \mathbf{F}(\mathbf{q}) \cdot \mathbf{n} + \lambda_{\max}(\mathbf{q} - \hat{\mathbf{q}}) \tag{4.18}$$

Finally, remembering that the HDG scheme needs a conservation condition that ensures that the method is locally conservative, it will be imposed using the $\hat{\mathbf{F}}$ flux:

$$\langle [\![\mathbf{F}(\mathbf{q}) \cdot \mathbf{n} + \lambda_{\max}(\mathbf{q} - \hat{\mathbf{q}})]\!], \hat{\mathbf{v}} \rangle_e = 0 \tag{4.19}$$

with $\langle a, b\rangle_e = \int_e abdx$.

4.4.2 Numerical fluxes for the SWE

For the linear SWE, using the eigenvalues obtained in Section 3.4, the Lax-Friedrichs (Rusanov) DG linear flux $\mathbf{F}_{\mathrm{L}}^*$ (4.17) , and HDG linear flux $\hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q})$ (4.18), are:

$$\mathbf{F}_{\mathrm{L}}^*(\mathbf{q}^+, \mathbf{q}^-) \cdot \mathbf{n} = \{\{\mathbf{F}_{\mathrm{L}}(\mathbf{q})\}\} \cdot \mathbf{n} + \frac{1}{2}\sqrt{\phi_B}\langle\langle \mathbf{q}\rangle\rangle$$

$$\hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q}) \cdot \mathbf{n} = \mathbf{F}_{\mathrm{L}}(\mathbf{q}) \cdot \mathbf{n} + \sqrt{\phi_B}(\mathbf{q} - \hat{\mathbf{q}})$$

The Lax-Friedrichs fluxes for the non-linear SWE, (4.17) and (4.18), are:

$$\mathbf{F}^*(\mathbf{q}^+, \mathbf{q}^-) \cdot \mathbf{n} = \{\{\mathbf{F}(\mathbf{q})\}\} \cdot \mathbf{n} + \frac{1}{2}\lambda_{\max}\langle\langle \mathbf{q}\rangle\rangle$$

$$\hat{\mathbf{F}}(\mathbf{q}) \cdot \mathbf{n} = \mathbf{F}(\mathbf{q}) \cdot \mathbf{n} + \lambda_{\max}(\mathbf{q} - \hat{\mathbf{q}})$$

with $\lambda_{\max} = \max\left[|\mathbf{U}^+ \cdot \mathbf{n}| + \sqrt{(\phi^+ + \phi_B)}, |\mathbf{U}^- \cdot \mathbf{n}| + \sqrt{(\phi^- + \phi_B)}\right]$.

4.5 Boundary conditions for HDG

The boundary conditions are applied in the outside faces of the mesh, defined as $\varepsilon^{\text{bound}} = \partial K \cap \partial\Omega$ (equation (4.3)).

In the DG and HDG equations for the SWE, (4.5) and (4.6), the integral at ∂K is defined for all the faces (ε). To impose the boundary conditions, we need to separate it into the element interfaces ε^{int} and the exterior boundary faces $\varepsilon^{\text{bound}}$. For the interfaces between the elements, the conservation conditions remain the same. For the boundary faces we impose the boundary conditions in the matrix $\mathbf{B}$.

$$\int_{\varepsilon}\{\{\hat{\mathbf{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}\}\} \cdot \hat{\mathbf{v}}\, dS = \int_{\varepsilon^{\text{int}}}\{\{\hat{\mathbf{F}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}\}\} \cdot \hat{\mathbf{v}}\, dS + \int_{\varepsilon^{\text{bound}}} \mathbf{B}(\mathbf{q}, \hat{\mathbf{q}}, \mathbf{q}_\infty) \cdot \hat{\mathbf{v}}\, dS$$

The matrix $\mathbf{B}$, containing the boundary values $\mathbf{q}_\infty$, is defined as:

$$\mathbf{B} = \mathbf{A}^+\mathbf{q} - |\mathbf{A}|\hat{\mathbf{q}} - \mathbf{A}^-\mathbf{q}_\infty$$

with

$$\begin{cases} \mathbf{A}^+ = \frac{1}{2}(\mathbf{A} + |\mathbf{A}|) \\ \mathbf{A}^- = \frac{1}{2}(\mathbf{A} - |\mathbf{A}|) \end{cases}$$

$\mathbf{A}$ is the Jacobian of the matrix $\mathbf{F}$, as defined in equation (3.31), in section 3.4. Choosing the Lax-Friedrichs stabilization parameter, we can apply (Samii et al., 2019):

$$|\mathbf{A}| = \lambda_{\max} I = \begin{pmatrix} \lambda_{\max} & 0 & 0 \\ 0 & \lambda_{\max} & 0 \\ 0 & 0 & \lambda_{\max} \end{pmatrix}$$

4.5.1 Periodic boundary conditions

For the periodic boundary conditions all the element faces are treated as interior faces. It is imposed by stating $\mathbf{q}_\infty|_{\Gamma 1} = \mathbf{q}_\infty|_{\Gamma 2}$, where Γ_1 and Γ_2 are the opposite connected edges with the boundary condition. This is applied by stating $\mathbf{B}|_{\Gamma 1} = \mathbf{F}^*|_{\Gamma 1}$ and $\mathbf{B}|_{\Gamma 2} = \mathbf{F}^*|_{\Gamma 2}$. Periodic boundary conditions are already incorporated in the mesh libraries.

4.5.2 Wall boundary conditions

Wall boundary conditions are treated by imposing a reflection of the variables, using an imaginary "ghost" element next to the boundary element (Kang et al., 2020). Considering K^- as the element next to the boundary, and K^+ as the imaginary element, it must be:

$$\begin{cases} \phi^+ = \phi^- \\ u^+ = u^- - 2(\mathbf{u}^- \cdot \mathbf{n})n_1 \\ v^+ = v^- - 2(\mathbf{u}^- \cdot \mathbf{n})n_2 \end{cases} \tag{4.20}$$

Substituting (4.20) into the conservation condition (4.19), in transforms into:

$$\begin{cases} \langle \mathbf{u}\cdot\mathbf{n} + \sqrt{\phi_B}(\phi - \hat{\phi}),\, \hat{\mathbf{v}}\rangle_{\partial K^- \cap \partial\Omega} = 0 \\ \langle \sqrt{\phi_B}(\mathbf{u}^T - \hat{\mathbf{u}}^T), \hat{\mathbf{v}}^T\rangle_{\partial K^- \cap \partial\Omega} = 0 \\ \hat{\mathbf{u}}\cdot\mathbf{n} = 0 \end{cases}$$

with T indicating the tangential part.

4.6 Bilinear forms for the SWE

The equations for the HDG SWE (4.6) can be written using bilinear forms. We will use them in the next chapters, for simplicity.

$$\begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{L}(\mathbf{q},\mathbf{v}) \\ \Xi(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = 0 \end{cases} \tag{4.21}$$

with the terms:

$$\mathbf{M}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{q}\cdot\mathbf{v}\,dx \qquad \text{Mass matrix}$$

$$\mathbf{L}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{F}(\mathbf{q})\cdot\nabla\cdot\mathbf{v}\,dx - \int_{\partial K}(\hat{\mathbf{F}}(\mathbf{q})\cdot\mathbf{n})\cdot\mathbf{v}\,dS + \int_K \mathbf{s}(\mathbf{q})\cdot\mathbf{v}\,dx$$

$$\Xi(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = \int_{\varepsilon^{\text{int}}} \{\{\hat{\mathbf{F}}(\mathbf{q},\hat{\mathbf{q}})\cdot\mathbf{n}\}\}\cdot\hat{\mathbf{v}}\,dS + \int_{\varepsilon^{\text{bound}}} \mathbf{B}(\mathbf{q},\hat{\mathbf{q}},\mathbf{q}_\infty)\cdot\hat{\mathbf{v}}\,dS = 0$$

If we want to use the linear SWE, we replace the matrix $\mathbf{F}$ for $\mathbf{F}_{\mathrm{L}}$.

$$\begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{L}_{\mathrm{L}}(\mathbf{q},\mathbf{v}) \\ \Xi_{\mathrm{L}}(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = 0 \end{cases} \tag{4.22}$$

with

$$\mathbf{L}_{\mathrm{L}}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{F}_{\mathrm{L}}(\mathbf{q})\cdot\nabla\cdot\mathbf{v}\,dx - \int_{\partial K}(\hat{\mathbf{F}_{\mathrm{L}}}(\mathbf{q})\cdot\mathbf{n})\cdot\mathbf{v}\,dS + \int_K \mathbf{s}(\mathbf{q})\cdot\mathbf{v}\,dx$$

$$\Xi_{\mathrm{L}}(\mathbf{q}, \hat{\mathbf{q}}, \hat{\mathbf{v}}) = \int_{\varepsilon^{\mathrm{int}}} \{\{\hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q}, \hat{\mathbf{q}}) \cdot \mathbf{n}\}\} \cdot \hat{\mathbf{v}}\, dS + \int_{\varepsilon^{\mathrm{bound}}} \mathbf{B}(\mathbf{q}, \hat{\mathbf{q}}, \mathbf{q}_{\infty}) \cdot \hat{\mathbf{v}}\, dS = 0$$

4.6.1 Velocity splitting

The non-linear terms and the linear terms have different velocities, as proven with the eigenvalues in the section 3.4, in the previous chapter.

The full matrix **F** can be separated into the linear and non-linear parts. We define a non-linear bilinear form, **N**.

$$\begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{N}(\mathbf{q}, \mathbf{v}) + \mathbf{L}_{\mathrm{L}}(\mathbf{q}, \mathbf{v}) \\ \Xi_{\mathrm{L}}(\mathbf{q}, \hat{\mathbf{q}}, \hat{\mathbf{v}}) = 0 \end{cases} \tag{4.23}$$

$$\mathbf{N}(\mathbf{q}, \mathbf{v}) = \int_{K} (\mathbf{F}(\mathbf{q}) - \mathbf{F}_{\mathrm{L}}(\mathbf{q})) \cdot \nabla \cdot \mathbf{v}\, dx - \int_{\partial K} \big((\hat{\mathbf{F}}(\mathbf{q}) - \hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q})) \cdot \mathbf{n}\big) \cdot \mathbf{v}\, dS$$

In the general DG and HDG discretizations, the non-linear term will be approached with a DG discretization, independently of the linear part.

The Shallow Water Equations are discretized in space using the HDG method, with Lax-Friedrichs numerical fluxes defined accordingly. This completes the spatial discretization, which is ultimately expressed in bilinear form. These systems are expanded in Appendix B, and the temporal discretization is developed in Chapter 5.

Chapter 5

Implicit-explicit Strong Stability Preserving Runge-Kutta schemes

In this chapter, we will develop the temporal discretization for the SWE system. In the previous chapter, we arrived at the Ordinary Differential Equation (ODE) dependent on time (5.1), that can be discretized using numerical methods.

$$\frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{N}(\mathbf{q},\mathbf{v}) + \mathbf{L}_{\mathrm{L}}(\mathbf{q},\mathbf{v}) \tag{5.1}$$

As we have said in the previous chapters, very different time scales, from slow flows to fast waves, coexist in geophysical fluids dynamics (Cushman-Roisin, 2011). The term $\mathbf{N}$ from the SWE has a slower dynamic and is not stiff, while the term $\mathbf{L}_{\mathrm{L}}$ has a faster dynamic and is stiff, containing the gravity waves. The faster term limits the acceptable time-step size and makes the system lose its stability. To avoid this limitation, we apply an implicit-explicit (IMEX) method. With this method, the two terms are treated differently. The slower term is discretized with an explicit scheme and the faster with an implicit scheme, allowing for a larger time-step (Betteridge et al., 2021; Kang et al., 2020).

The equation modelling the physical processes in oceanography can have discontinuities, and lose the stability. Classical IMEX methods require a small time-step to maintain the stability. Strong Stability Preserving (SSP) methods provide stability to PDEs with sharp gradients or discontinuities, enforcing a stability condition to stop the system from growing without a bound.

First, we present general IMEX methods and we exemplify it with a θ method combined with an explicit Euler. Then, Strong Stability Preserving (SSP) methods are shown. Finally, IMEX SSP are applied to the SWE system.

5.1 IMEX Runge-Kutta methods

IMEX method were introduced by Crouzeix (1980) for evolving parabolic equations and have been widely developed and improved by Ascher et al. (1997), Conde et al. (2017), and Higueras et al. (2014), among others.

IMEX methods were developed for ODEs that contain two terms with different dynamics and stiffness. A term is stiff if explicit time methods are numerically unstable, unless the time-step used is extremely small. This happens to the fast term of the ODEs, restricting the allowed time-step size (Kang et al., 2020). For stiff ODEs, treating the fast waves implicitly helps relax the restriction on the time-step size, but involves solving a system of equations, increasing the computational cost (Betteridge et al., 2021; Conde et al., 2017).

IMEX methods treat the fast term implicitly and the slow term explicitly, balancing the time-step size restriction and the computational cost of implicit schemes (Betteridge et al., 2021; Conde et al., 2017).

For the spatially discretized SWE system (5.1) a s-stage IMEX Runge-Kutta (RK) method can be written, as a combination of backward and forward Euler steps (Betteridge et al., 2021; Conde et al., 2017), in the form:

$$\begin{cases} \mathbf{M}(\mathbf{Q^i}) = \mathbf{M}(\mathbf{q^n}) + \Delta t \sum_{j=1}^{i-1} a_{i,j} \mathbf{N}(\mathbf{Q^j}) + \Delta t \sum_{j=1}^{i} \tilde{a}_{i,j} \mathbf{L}_{\mathrm{L}}(\mathbf{Q^j}) & \text{for} \quad i = 1, ..s \\ \mathbf{M}(\mathbf{q^{n+1}}) = \mathbf{M}(\mathbf{q^n}) + \Delta t \sum_{i=1}^{s} b_i \mathbf{N}(\mathbf{Q^i}) + \Delta t \sum_{i=1}^{s} \tilde{b}_i \mathbf{L}_{\mathrm{L}}(\mathbf{Q^i}) \end{cases} \tag{5.2}$$

The unknown $\mathbf{Q^i}$ represent the intermediate stages of the Runge-Kutta method, located between the time-steps n and $n+1$. The goal is to compute the next time step, $\mathbf{q^{n+1}}$, as a weighted average of these intermediate solutions.

The values of $a_{i,j}$, b_i, $\tilde{a}_{i,j}$ and $\tilde{b}_i$ depend on the chosen time discretization scheme. These values are stored in the Butcher tableaux, within the matrices and vectors $\mathbf{a}$, $\tilde{\mathbf{a}}$, $\mathbf{b}$, and $\tilde{\mathbf{b}}$.

TABLE 5.1: Generic 2-stage Butcher Tableau

c_1	a_{11}	a_{12}
c_2	a_{21}	a_{22}
	b_1	b_2

In equation (5.2), the coefficients $a_{i,j}$ correspond to the explicit part of the method. They are defined only for $j < i$, ensuring that only previously computed stages are used. On the other hand, the coefficients $\tilde{a}_{i,j}$ are defined up to $j = i$, corresponding to the implicit part when $j = i$.

There is another way to write this method, known as the Shu-Osher form (Gottlieb et al., 2011; Shu & Osher, 1988). Because the Shu–Osher form of a RK method is not unique, while the Butcher form is (Conde et al., 2017), we will use the Butcher form. However, we will use the Shu-Osher coefficients for the stability preservation analysis, in section 5.2.1. The Shu-Osher formulation is:

$$\begin{cases} \mathbf{M}(\mathbf{Q^i}) = v_1\mathbf{M}(\mathbf{q^n}) + \sum_{j=1}^{s}\left(\alpha_{i,j}\mathbf{Q^j} + \Delta\beta_{i,j}\mathbf{N}(\mathbf{Q^j})\right) + \sum_{j=1}^{s+1}\left(\tilde{\alpha}_{i,j}\mathbf{Q^j} + \Delta\tilde{\beta}_{i,j}\mathbf{L}_{\mathrm{L}}(\mathbf{Q^j})\right) \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for} \quad i = 1,..s \\ \mathbf{M}(\mathbf{q^{n+1}}) = v_{s+1}\mathbf{M}(\mathbf{q^n}) + \sum_{j=1}^{s}\left(\alpha_{s+1,j}\mathbf{Q^j} + \Delta\beta_{s+1,j}\mathbf{N}(\mathbf{Q^j})\right) \\ \qquad\qquad + \sum_{j=1}^{s+1}\left(\tilde{\alpha}_{s+1,j}\mathbf{Q^j} + \Delta\tilde{\beta}_{s+1,j}\mathbf{L}_{\mathrm{L}}(\mathbf{Q^j})\right) \end{cases}$$

In this thesis we have used the IMEX RK method ARS2(2,3,2) (Ascher et al., 1997) of 2^{nd} order, with 2 stages for the implicit scheme, and 3 stages for the explicit scheme; and the IMEX method ARS3(4,4,3) (Ascher et al., 1997), with 3^{rd} order, and 4 stages for both the explicit and implicit scheme. The Butcher coefficients for these methods are written in Appendix A.

However, classical IMEX methods are not robust enough and often require a small time-step in order to remain stable. To address this issue, some more complex schemes with characteristics like stability preserving properties are developed in the next sections.

5.1.1 θ method

We now apply a specific IMEX Euler method to the SWE. The explicit Euler method is used for the explicit term N, and the θ method for the implicit term $\mathbf{L}_{\mathrm{L}}$. Applying this scheme to (5.1) gives:

$$\frac{\mathbf{M}(\mathbf{q^{n+1}},\mathbf{v}) - \mathbf{M}(\mathbf{q^n},\mathbf{v})}{\Delta t} = \mathbf{N}(\mathbf{q^n},\mathbf{v}) + \theta\mathbf{L}_{\mathrm{L}}(\mathbf{q^{n+1}},\mathbf{v}) + (1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{q^n},\mathbf{v}) \tag{5.3}$$

with $\theta \in [0,1]$. Moving the unknown terms to the left-hand side, and the known terms to the right-hand side, equation (5.3) becomes:

$$\mathbf{M}(\mathbf{q^{n+1}},\mathbf{v}) - \Delta t\theta\mathbf{L}_{\mathrm{L}}(\mathbf{q^{n+1}},\mathbf{v}) = \mathbf{M}(\mathbf{q^n},\mathbf{v}) + \Delta t\mathbf{N}(\mathbf{q^n},\mathbf{v}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{q^n},\mathbf{v}) \tag{5.4}$$

Using $\theta = 0$ we could obtain a fully explicit Euler scheme, and if $\theta = 1$ a fully implicit Euler scheme. If $\theta = 0.5$, the scheme is a Crank-Nicholson method for the term $\mathbf{L}_{\mathrm{L}}(\mathbf{q}, \mathbf{v})$.

As we have stated, we can write the IMEX discretizations using Butcher Tableaux. The Butcher coefficients for the θ method are:

$$\mathbf{a} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} 0 & 0 \\ 1-\theta & \theta \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \tilde{\mathbf{b}} = \begin{pmatrix} 1-\theta \\ \theta \end{pmatrix} \tag{5.5}$$

Substituting the coefficients of this method (5.5) into the Butcher form (5.2), we obtain the system:

$$\begin{cases} \mathbf{M}(\mathbf{Q^1}) = \mathbf{M}(\mathbf{q^n}) \\ \mathbf{M}(\mathbf{Q^2}) = \mathbf{M}(\mathbf{q^n}) + \Delta t \mathbf{N}(\mathbf{Q^1}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{Q^1}) + \Delta t\theta \mathbf{L}_{\mathrm{L}}(\mathbf{Q^2}) \\ \mathbf{M}(\mathbf{q^{n+1}}) = \mathbf{M}(\mathbf{q^n}) + \Delta t \mathbf{N}(\mathbf{Q^1}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{Q^1}) + \Delta t\theta \mathbf{L}_{\mathrm{L}}(\mathbf{Q^2}) \end{cases} \tag{5.6}$$

From the first equation in (5.6), we can deduce $\mathbf{Q^1} = \mathbf{q^n}$. Rewriting the second equation, by subtracting the third equation from the second, it becomes:

$$\begin{aligned} \mathbf{M}(\mathbf{Q^2}) - \mathbf{M}(\mathbf{q^{n+1}}) = & \Big(\mathbf{M}(\mathbf{q^n}) + \Delta t \mathbf{N}(\mathbf{Q^1}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{Q^1}) + \Delta t\theta \mathbf{L}_{\mathrm{L}}(\mathbf{Q^2})\Big) \\ & - \Big(\mathbf{M}(\mathbf{q^n}) + \Delta t \mathbf{N}(\mathbf{Q^1}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{Q^1}) + \Delta t\theta \mathbf{L}_{\mathrm{L}}(\mathbf{Q^2})\Big) \\ \mathbf{M}(\mathbf{Q^2}) - \mathbf{M}(\mathbf{q^{n+1}}) = & \ 0 \end{aligned} \tag{5.7}$$

In equation (5.7) we obtain $\mathbf{M}(\mathbf{Q^2}) = \mathbf{M}(\mathbf{q^{n+1}})$, therefore, we can deduce that $\mathbf{Q^2} = \mathbf{q^{n+1}}$. Substituting $\mathbf{Q^1}$ and $\mathbf{Q^2}$ in the third equation of (5.6), we obtain the final equation for $\mathbf{q^{n+1}}$, that corresponds to the equation for the θ method (5.4):

$$\mathbf{M}(\mathbf{q^{n+1}}) = \mathbf{M}(\mathbf{q^n}) + \Delta t \mathbf{N}(\mathbf{Q^n}) + \Delta t(1-\theta)\mathbf{L}_{\mathrm{L}}(\mathbf{Q^n}) + \Delta t\theta \mathbf{L}_{\mathrm{L}}(\mathbf{Q^{n+1}})$$

5.2 SSP methods

The exact solution of hyperbolic conservation laws, such as equation (3.28) can develop sharp gradients or discontinuities, which might cause difficulties in numerical simulations. This makes necessary the use of high order discretizations with stability properties. Stability is needed to stop the system from growing without a bound (LeVeque, 2007).

Explicit Strong Stability Preserving (SSP) methods were first used by Gottlieb and Shu (1998), Shu (1988), and Shu and Osher (1988), and were called Total Variation Diminishing (TVD) discretizations. They were developed for hyperbolic time-dependent PDEs and ODEs, and are particularly recommended for hyperbolic PDEs with shocks (Gottlieb et al., 2011).

SSP methods guarantee the existence of a non-empty region of absolute monotonicity, compared to non-SSP methods, allowing for larger time steps while preserving stability. The region of absolute monotonicity defines the range of time-step sizes that ensure the numerical solution does not introduce artificial growth or oscillations. This property is especially relevant in the context of hyperbolic problems, due to the sharp gradients or discontinuities (Ascher et al., 1997; Conde et al., 2017).

In this section, we focus exclusively on explicit SSP Runge-Kutta (SSP RK) methods. IMEX SSP schemes will be discussed in later sections. To understand the stability preservation, we will analyze the stability properties of explicit SSP methods, with an ODE defined as (5.8).

$$\frac{\partial \mathbf{q}}{\partial t} = \mathbf{G}(\mathbf{q}) \tag{5.8}$$

Some SSP RK methods can be written as a convex combination of forward Euler steps, each of which individually satisfies a given non-linear stability property. The higher order SSP RK schemes inherit these stability properties, under the same time-step restriction, for all the intermediate stages, ensuring that the stability property is maintained throughout the integration process (Conde et al., 2017; Gottlieb et al., 2011).

The stability region of explicit Runge-Kutta schemes can increase by considering methods with three or more stages (Ferracina & Spijker, 2004a, 2004b; Higueras et al., 2014). However, SSP RK schemes are limited to fourth-order accuracy, while implicit schemes can reach up to sixth order, mainly due to constraints imposed by non-linear stability conditions (Conde et al., 2017; Gottlieb et al., 2011).

In this thesis we have applied the SSP RK methods Shu-Osher (Isherwood et al., 2018), a 3^{rd} order explicit scheme, SSPIRK33 (Isherwood et al., 2018; Ketcheson et al., 2009), a 3^{rd} order diagonally implicit scheme with 3 stages, and SSPIRK43 (Isherwood et al., 2018; Ketcheson et al., 2009), a diagonally implicit scheme with 3^{rd} with 4 stages. The Butcher coefficients for the schemes are presented in Appendix A.

5.2.1 Stability preservation

Total Variation Diminishing (TVD) schemes are the origin of the SSP discretization, as used in Shu and Osher (1988). The total variation of a solution is defined as follows,

$$TV(\mathbf{q^n}) = \Sigma_h |\mathbf{q^n_{h+1}} - \mathbf{q^n_h}|$$

where $\mathbf{q^n_{h+1}}$ is the solution for time n in the element $h+1$, and $\mathbf{q^n_{h+1}}$ the solution for time n in the element h.

A scheme is considered TVD if the solution in the next time-step ($\mathbf{q^{n+1}}$) satisfies:

$$TV(\mathbf{q^{n+1}}) \leq TV(\mathbf{q^n})$$

Similarly, for hyperbolic problems, the linear stability requirement usually leads to a ratio between Δt and Δx. (Conde et al., 2017; Gottlieb et al., 2011). The stability condition for equation (5.8) must satisfy:

$$||\mathbf{q^n} + \Delta t \mathbf{G}(\mathbf{q^n})|| \leq ||\mathbf{q^n}||, \qquad 0 \leq \Delta t \leq \Delta t_{\text{FE}} \tag{5.9}$$

with Δt_{FE} as the forward Euler stable time-step, depending on the spatial discretization, and $||\cdot||$ the norm (Conde et al., 2017). This must be true for any Δt smaller than Δt_{FE}. Then, a method is SSP if it satisfies:

$$||\mathbf{q^{n+1}}|| \leq ||\mathbf{q^n}||$$

Accordingly, the time-step of the SSP method must satisfy the following ratio, with the Strong Stability preserving coefficient C (Conde et al., 2017; Gottlieb et al., 2011):

$$\Delta t \leq C \Delta t_{\text{FE}}$$

$$C = \begin{cases} \min_{i,j} \left(\frac{\alpha_{i,j}}{\beta_{i,j}}\right) & \text{if } \alpha_{i,j}, \beta_{i,j} \text{ are non-negative} \\ 0 \quad \text{otherwise} \end{cases}$$

C is calculated with the Shu-Osher coefficients α and β. To consider a method SSP, it must fulfill $C > 0$, and the coefficients α and β must be non-negative. Moreover, the coefficients from the Butcher form must satisfy $\mathbf{a} \geq 0$ and $\mathbf{b} \geq 0$ to have a non-zero SSP coefficient (Gottlieb et al., 2011).

The goal of SSP methods is to maximize the value of C, since it allows for a larger time-step (Conde et al., 2017).

5.3 IMEX SSPRK discretizations

IMEX SSP methods were first used by Pareschi and Russo (2005), as a tool to solve large ODEs originated by the spatial discretization of hyperbolic PDEs (Gottlieb et al., 2011). In addition, Higueras et al. (2014) introduced some SSP IMEX methods with the SSP conditions in both the implicit and explicit terms (Conde et al., 2017).

For equation (5.1), with the two differentiated terms $\mathbf{N}$ and $\mathbf{L}_\text{L}$, SSP RK IMEX methods must satisfy the Euler condition (5.9) independently, for both terms $\mathbf{N}$ and $\mathbf{L}_\text{L}$:

$$\begin{cases} ||\mathbf{q}^\mathbf{n} + \Delta t \mathbf{N}(\mathbf{q}^\mathbf{n})|| \leq ||\mathbf{q}^\mathbf{n}||, & 0 \leq \Delta t \leq \Delta t_\text{FE} \\ ||\mathbf{q}^\mathbf{n} + \Delta t \mathbf{L}_\text{L}(\mathbf{q}^\mathbf{n})|| \leq ||\mathbf{q}^\mathbf{n}||, & 0 \leq \Delta t \leq K \Delta t_\text{FE} \end{cases}$$

For the SSP condition, the value of K must be > 1. A larger value of K implies a more permissive time-step for the implicit term $\mathbf{L}_\text{L}$.

The IMEX SSP methods. can be expressed using equation (5.2) with the corresponding SSP coefficients, as we have seen in this chapter. In this thesis we have used some variations of the IMEX RK SSP method SSP2(3,3,2), a 2^{nd} order scheme with 3 implicit and explicit stages (Betteridge et al., 2021; Higueras et al., 2014). The Butcher coefficients for the schemes are thoroughly presented in Appendix A.

5.4 Temporal discretization for the SWE system

IMEX DG methods have been proven to have more advantages for the SWE than the fully-explicit or fully-implicit DG methods (Kang et al., 2020).

In our SWE system (4.23), the term $\mathbf{L}_\text{L}$ is stiff and associated with fast dynamics, such as gravity waves. It will be treated implicitly to avoid severe time-step restrictions. On the other hand, the non-stiff term **N** represents the slower dynamics and can be discretized explicitly. The IMEX SSP temporal discretization scheme applied to the SWE system results in:

$$\begin{cases} \mathbf{M}(\mathbf{Q^i},\mathbf{v}) = \mathbf{M}(\mathbf{q^n},\mathbf{v}) + \Delta t\sum_{j=1}^{i-1} a_{i,j}\mathbf{N}(\mathbf{Q^j},\mathbf{v}) + \Delta t\sum_{j=1}^{i} \tilde{a}_{i,j}\mathbf{L}_{\mathrm{L}}(\mathbf{Q^j},\mathbf{v}) \quad \text{for} \quad i = 1,..s \\ \mathbf{M}(\mathbf{q^{n+1}},\mathbf{v}) = \mathbf{M}(\mathbf{q^n},\mathbf{v}) + \Delta t\sum_{i=1}^{s} b_i\mathbf{N}(\mathbf{Q^i},\mathbf{v}) + \Delta t\sum_{i=1}^{s} \tilde{b}_i\mathbf{L}_{\mathrm{L}}(\mathbf{Q^i},\mathbf{v}) \end{cases} \tag{5.10}$$

The complete bilinear form of the SWE (4.21) can be written for non-IMEX SSP RK methods, similarly to equation (5.10), as:

$$\begin{cases} \mathbf{M}(\mathbf{Q^i},\mathbf{v}) = \mathbf{M}(\mathbf{q^n},\mathbf{v}) + \Delta t\sum_{j=1}^{i-1} a_{i,j}\mathbf{L}(\mathbf{Q^j},\mathbf{v}) \quad \text{for} \quad i = 1,..s \\ \mathbf{M}(\mathbf{q^{n+1}},\mathbf{v}) = \mathbf{M}(\mathbf{q^n},\mathbf{v}) + \Delta t\sum_{i=1}^{s} b_i\mathbf{L}(\mathbf{Q^i},\mathbf{v}) \end{cases} \tag{5.11}$$

For the HDG discretization, the flux equation does not require a temporal scheme, and is associated to the unknowns of the system in each step of the solver.

5.5 Resolution of the system: static condensation

The stages $(\mathbf{Q^i})$ of equation (5.10) can be resolved, along with the flux equation Ξ, by applying a static condensation decomposition (Cockburn, 2016).

First, multiplying (5.10) by $\mathbf{M^{-1}}$, we obtain the system we will resolve:

$$\begin{gathered} \mathbf{Q}^i - \Delta t\,\tilde{a}_{i,i}\,\mathbf{M^{-1}}\mathbf{L}_{\mathrm{L}\,i} = \mathbf{q}^n + \Delta t\,\mathbf{M^{-1}}\sum_{j=1}^{i-1}\left(a_{ij}\,\mathbf{N}_j + \tilde{a}_{ij}\,\mathbf{L}_{\mathrm{L}\,j}\right) \\ \left\langle [\![\hat{\mathbf{F}}_{\mathbf{L}}(\mathbf{Q}^i,\hat{\mathbf{Q}}^i)\cdot\mathbf{n}]\!],\hat{\mathbf{v}}\right\rangle_{\varepsilon} = 0 \end{gathered} \tag{5.12}$$

The system (5.12) can be written in matrix form as:

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{Q^i} \\ \hat{\mathbf{Q}}^{\mathbf{i}} \end{pmatrix} = \begin{pmatrix} \mathbf{R_1} \\ \mathbf{R_2} \end{pmatrix} \tag{5.13}$$

with the terms:

$$\mathbf{A} = \mathbf{I} - \Delta t \tilde{a}_{i,i} \mathbf{M}^{-1} \frac{\partial \mathbf{L}_{\mathrm{L}}}{\partial \mathbf{Q}}, \quad \mathbf{B} = -\Delta t \tilde{a}_{i,i} \mathbf{M}^{-1} \frac{\partial \mathbf{L}_{\mathrm{L}}}{\partial \hat{\mathbf{Q}}}, \quad \mathbf{C} = \frac{\partial \Xi}{\partial \mathbf{Q}}, \quad \mathbf{D} = \frac{\partial \Xi}{\partial \hat{\mathbf{Q}}}$$

$$\mathbf{R_1} = \mathbf{q^n} + \Delta t \mathbf{M}^{-1} \sum_{j=1}^{i-1} \left(a_{i,j} \mathbf{N^j} + \tilde{a}_{i,j} \mathbf{L}_{\mathrm{L}}^{\mathbf{j}} \right), \quad \mathbf{R_2} = 0$$

Isolating the unknown $\mathbf{Q^i}$ in the left-hand side, on the first line of the system (5.13) we obtain:

$$\mathbf{Q^i} = \mathbf{A}^{-1} \left(\mathbf{R_1} - \mathbf{B} \hat{\mathbf{Q}}^{\mathbf{i}} \right) \tag{5.14}$$

Replacing (5.14) into the second line of (5.13) we obtain an equation to solve for $\hat{\mathbf{Q}}^{\mathbf{i}}$.

$$\left(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B} \right) \cdot \hat{\mathbf{Q}}^{\mathbf{i}} = \mathbf{R_2} - \mathbf{C}\mathbf{A}^{-1}\mathbf{R_1} \tag{5.15}$$

We have arrived, in equation (5.15) to a linear equation system, that can be solved. Once we have obtained the trace unknown $\hat{\mathbf{Q}}^{\mathbf{i}}$, the element unknown $\mathbf{Q^i}$ can be recovered through equation (5.14). The solver methods for equation (5.15) is indicated in Appendix B.

With the spatial and temporal discretization strategies fully developed, and the solution algorithm based on static condensation established, the HDG model is now completed.

Chapter 6

Numerical results

In this chapter we reproduce some oceanographic problems. In order to do so, the model is implemented using the Firedrake library on Python (Ham et al., 2023), designed for solving PDEs through FEM methods. Firedrake allows the use of weak forms in Unified Form Language (UFL) and compiles them into optimized C code for computation. It relies on PETSc as its core tool for linear algebra, offering various solver options, and supports diverse types of unstructured meshes and finite element (FE) spaces (Betteridge, 2020; Betteridge et al., 2021).

The UFL (Alnæs et al., 2014), created as part of the FEniCS project (Logg et al., 2012), is a domain-specific language for expressing weak formulations of PDEs, making it suitable for numerical solutions. The Portable, Extensible Toolkit for Scientific Computation (PETSc) (Balay et al., 2024) provides an extensive collection of linear and non-linear solvers for scientific applications (Betteridge, 2020; Betteridge et al., 2021).

The system is solved by applying a Static Condensation reduction, shown in section 5.5. The system for the hybrid unknowns $\hat{\mathbf{q}}$ is then solved by applying a Generalized Minimal Residual (GMRES) (Saad & Schultz, 1986) method, with a tolerance of $1e^{-10}$ with an algebraic multigrid (AMG) preconditioner. The AMG system, then, is solved using the Chebyshev iterative method (Golub & Varga, 1961) with a block Jabobi preconditioner. The specific solver parameters and algorithms are indicated in Appendix B.

In the following sections, this chapter presents a series of numerical experiments designed to validate and illustrate the performance of the HDG model applied to the Shallow Water Equations. The selected test cases progressively increase in complexity. To validate the model, in the problems that have an analytic solution we have calculated the error convergence using the formulas in Appendix C.

We begin with three problems that have known analytical solutions - a moving vortex, a linear standing wave, and a linear Kelvin wave - in which we calculate the error of the solution, to verify the accuracy and convergence properties of the model. These are followed by classical benchmark problems in hydrodynamics, including the lake-at-rest scenario and its dynamic counterpart, the shallow lake, where we progressively introduce additional physical effects such as bottom topography. We then explore wind forcing through a wind-driven shallow lake experiment. Then, we test the model's response to an initial perturbation in the water surface elevation, with and without the Coriolis force, assessing its capacity to resolve transient free surface dynamics. Finally, we reproduce a more complex problem, representing a coastal inlet.

Together, these experiments provide a comprehensive assessment of the model's robustness, accuracy, and versatility in capturing geophysical flows. While the current model successfully simulates the physical processes observed in the area of interest, a multilayer extension is required to fully represent all the processes involved.

6.1 Problem 1 with analytical solution: Moving vortex

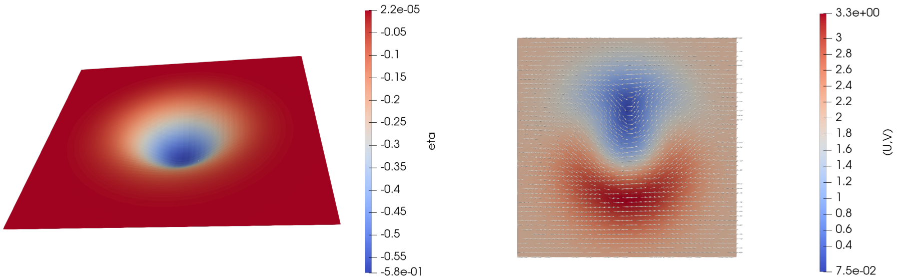
To assess the accuracy and stability of the model, we will reproduce a vortex translation test from Kang et al. (2020) and Ricchiuto and Bollermann (2009). This problem has an exact analytical solution for the non-linear form of the SWE, allowing the error calculation. The problem consists on the simulation of a vortex, translating along the domain at a constant speed in a balanced flow, while maintaining its shape without the presence of external forcings.

6.1.1 Initial conditions

The initial conditions (Figure 6.1) are:

$$\begin{cases} \eta = -\frac{\beta^2}{32\pi^2} \exp^{-2(r^2-1)} \\ (U,V) = (U_\infty, V_\infty) + \frac{\beta}{2\pi} \exp^{-(r^2-1)} \cdot (y_t, x_t) \end{cases}$$

with $\beta = 5$ as the vortex strength, $(u_\infty, v_\infty) = (1,0)$ the reference horizontal velocity, $(x_c, y_c) = (0,0)$ the center of the vortex, $x_t = x - x_c - u_\infty t$ and $y_t = y - y_c - v_\infty t$, and $r^2 = x_t^2 + y_t^2$. Gravity is $g = 2\,\mathrm{m\,s^{-2}}$.



(A) η

(B) $\mathbf{U}$

FIGURE 6.1: Initial conditions of the free surface elevation (η) and the velocity ($\mathbf{U}$) for the moving vortex problem. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

The domain is $\Omega = [-2, 2] \times [-2, 2]$ m (Figure 6.2). The bathymetry is flat, with $b = 1$ m. The exact solution is imposed at the boundary condition.

6.1.2 Results

The mesh for the results (Figure 6.2) is formed by quadrilateral elements of the size $\Delta x = \Delta y = 0.25$ m, with the polynomial order $p = 3$. The time step used for the simulation is $\Delta t = 1 \times 10^{-3}$ s, and the final time is $t = 2$ s.

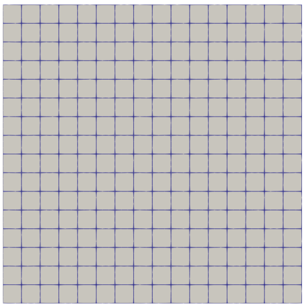
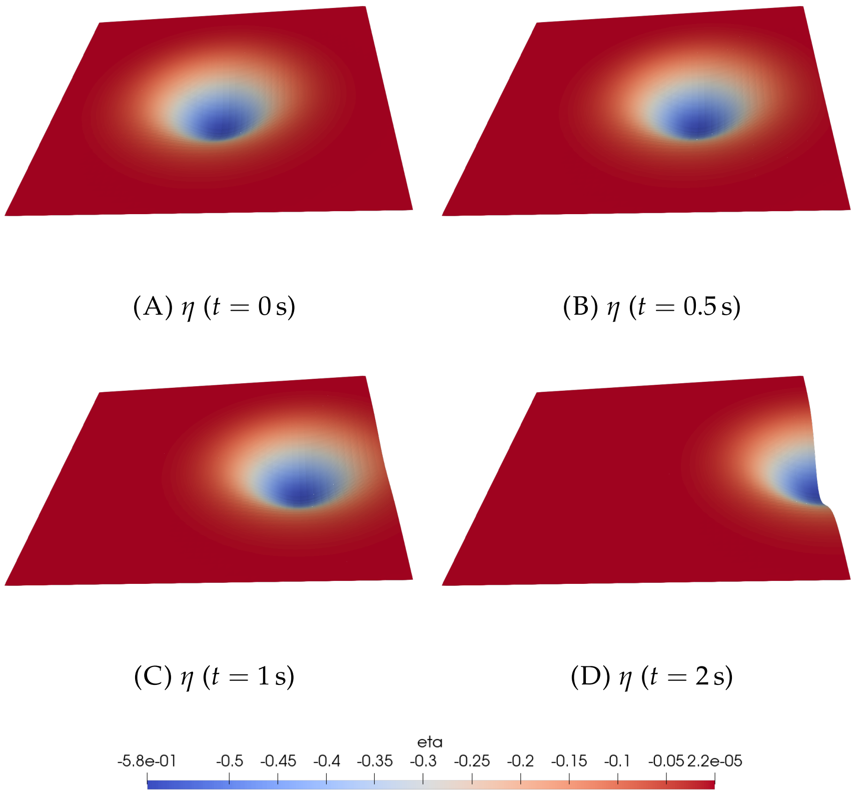


FIGURE 6.2: Mesh for the moving vortex problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [-2, 2] \times [-2, 2]$ m and the element size is $\Delta x = \Delta y = 0.25$ m.

The results for the temporal evolution of the free surface elevation η and velocity are shown in Figures 6.3 and 6.4. The vortex successfully maintains its shape as it propagates to the right, eventually reaching the domain boundary. The velocity field behaves accordingly, preserving the characteristics of the vortex throughout the simulation.



(A) η ($t = 0\,\mathrm{s}$)

(B) η ($t = 0.5\,\mathrm{s}$)

(C) η ($t = 1\,\mathrm{s}$)

(D) η ($t = 2\,\mathrm{s}$)

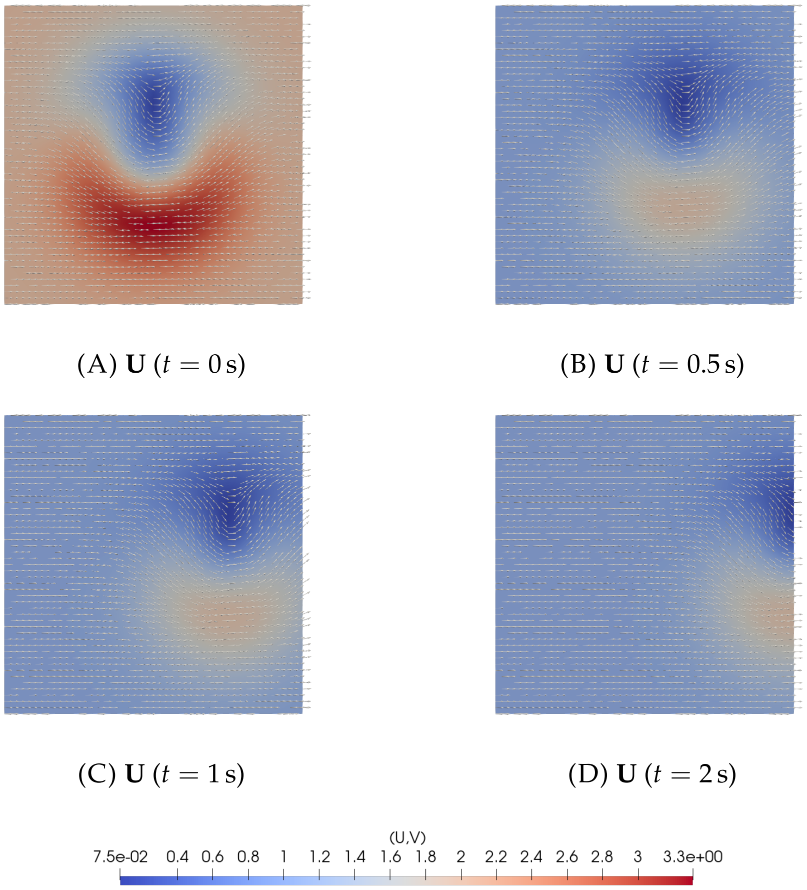
FIGURE 6.3: Results for the evolution of the free surface elevation (η) for the moving vortex problem, at the times $t = 0\,\mathrm{s}$, $t = 0.5\,\mathrm{s}$, $t = 1\,\mathrm{s}$ and $t = 2\,\mathrm{s}$.

6.1.3 Error convergence

For the convergence test, the orders for the polynomial basis functions are $p = 2, 3, 4, 5$, and the number of intervals $N = 8, 12, 16, 20$; with the corresponding time steps $\Delta t = 1 \times 10^{-3}\,\mathrm{s}$ for $p = 2$ and $p = 3$, $\Delta t = 2 \times 10^{-4}\,\mathrm{s}$ for $p = 4$, and $\Delta t = 5 \times 10^{-5}\,\mathrm{s}$ for $p = 5$. The convergence test is run for all the time discretization schemes shown in Appendix A. The error is calculated at $t = 2\,\mathrm{s}$.

The spatial convergence tests (Figures 6.5, 6.6, and 6.7) show that the error decreases as the number of elements increases and the element size is reduced. However, for the Crank–Nicolson scheme, the explicit and implicit Euler methods, and their IMEX counterparts, the convergence of the error deviates from the expected behavior for $p = 3$, $p = 4$, and $p = 5$. Among these, the Crank–Nicolson and IMEX Crank–Nicolson schemes perform better, while the explicit and implicit Euler methods (and their IMEX versions) show the worst convergence.

For the free surface elevation η, the observed convergence rates, corresponding to the slope of the plots, are approximately 2.7, 4.1, 4.7 and 6.4


(A) **U** ($t = 0\,\mathrm{s}$)

(B) **U** ($t = 0.5\,\mathrm{s}$)

(C) **U** ($t = 1\,\mathrm{s}$)

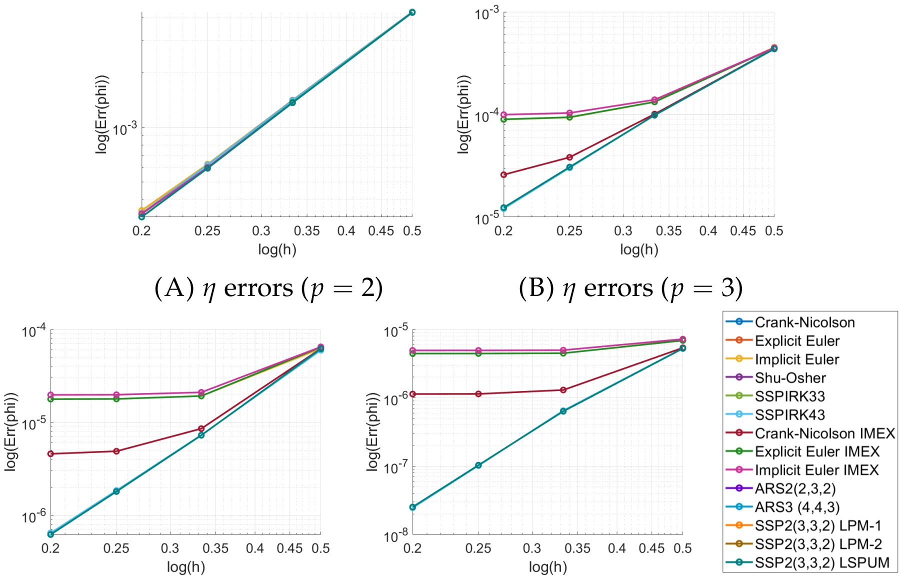
(D) **U** ($t = 2\,\mathrm{s}$)

FIGURE 6.4: Results for the evolution of the velocity (**U**) for the moving vortex problem, at the times $t = 0\,\mathrm{s}$, $t = 0.5\,\mathrm{s}$, $t = 1\,\mathrm{s}$ and $t = 2\,\mathrm{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

for polynomial orders $p = 2, 3, 4, 5$, respectively (Table 6.1). These results are consistent with the expected optimal rate of h^{p+1} for η convergence, as reported in the literature (Bui-Thanh, 2016; Samii et al., 2019). The velocity error convergence rates (Table 6.1) are 2.4, 4.6, 3.9 and 6.9 for the same values of p, which suggests a convergence of approximately $h^{p+0.5}$, a behavior that is consistent with the theoretical results found in Bui-Thanh (2016). Similarly, for the energy error, the convergence rates are 2.5, 4.5, 3.9 and 6.9 (see Table 6.1), also indicating optimal convergence.

TABLE 6.1: Error results for the moving vortex problem using the Shu-Osher method. The errors are calculated for the number of intervals $N = 8, 12, 16, 20$ and the polynomial orders $p = 2, 3, 4, 5$.

p	Nel	Error(η)	order(η)	Error($\mathbf{U}$)	Order($\mathbf{U}$)	Error($\sqrt{E}$)	Order($\sqrt{E}$)
2	8	$4.30{\cdot}10^{-3}$	NaN	$1.17{\cdot}10^{-2}$	NaN	$8.83{\cdot}10^{-3}$	NaN
	12	$1.40{\cdot}10^{-3}$	2.76	$4.34{\cdot}10^{-3}$	2.45	$3.22{\cdot}10^{-3}$	2.48
	16	$6.21{\cdot}10^{-4}$	2.83	$2.16{\cdot}10^{-3}$	2.43	$1.59{\cdot}10^{-3}$	2.47
	20	$3.37{\cdot}10^{-4}$	2.74	$1.23{\cdot}10^{-3}$	2.50	$9.05{\cdot}10^{-4}$	2.52
3	8	$4.42{\cdot}10^{-4}$	NaN	$3.18{\cdot}10^{-3}$	NaN	$2.27{\cdot}10^{-3}$	NaN
	12	$9.84{\cdot}10^{-5}$	3.71	$4.75{\cdot}10^{-4}$	4.69	$3.43{\cdot}10^{-4}$	4.67
	16	$3.02{\cdot}10^{-5}$	4.11	$1.27{\cdot}10^{-4}$	4.59	$9.20{\cdot}10^{-5}$	4.57
	20	$1.21{\cdot}10^{-5}$	4.11	$4.55{\cdot}10^{-5}$	4.58	$3.33{\cdot}10^{-5}$	4.55
4	8	$5.96{\cdot}10^{-5}$	NaN	$1.32{\cdot}10^{-4}$	NaN	$1.03{\cdot}10^{-4}$	NaN
	12	$7.29{\cdot}10^{-6}$	5.19	$3.30{\cdot}10^{-5}$	3.43	$2.39{\cdot}10^{-5}$	3.60
	16	$1.84{\cdot}10^{-6}$	4.78	$1.07{\cdot}10^{-5}$	3.91	$7.70{\cdot}10^{-6}$	3.94
	20	$6.43{\cdot}10^{-7}$	4.72	$4.23{\cdot}10^{-6}$	4.17	$3.03{\cdot}10^{-6}$	4.18
5	8	$5.29{\cdot}10^{-6}$	NaN	$7.78{\cdot}10^{-5}$	NaN	$5.52{\cdot}10^{-5}$	NaN
	12	$6.48{\cdot}10^{-7}$	5.18	$5.51{\cdot}10^{-6}$	6.53	$3.92{\cdot}10^{-6}$	6.52
	16	$1.02{\cdot}10^{-7}$	6.43	$7.47{\cdot}10^{-7}$	6.95	$5.33{\cdot}10^{-7}$	6.94
	20	$2.46{\cdot}10^{-8}$	6.37	$1.58{\cdot}10^{-7}$	6.96	$1.13{\cdot}10^{-7}$	6.95



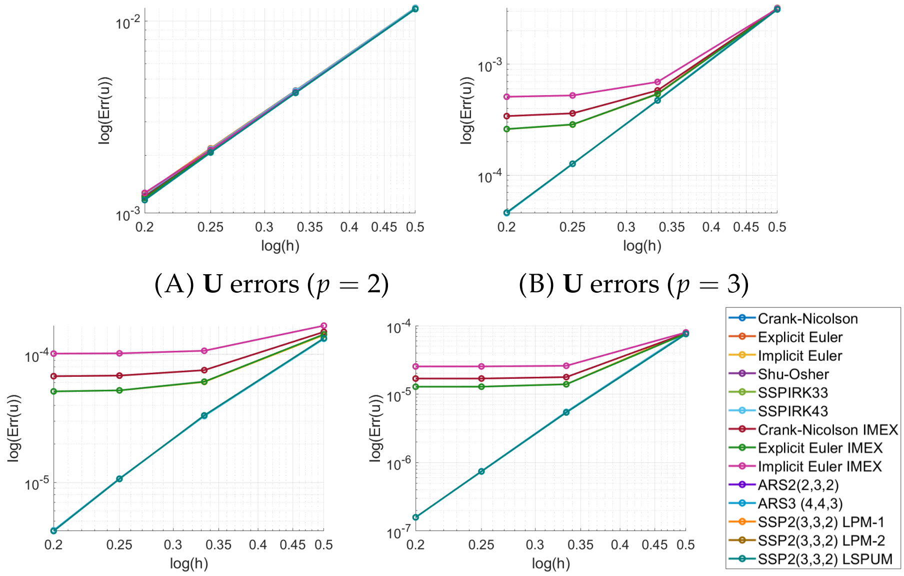
(A) η errors ($p = 2$)

(B) η errors ($p = 3$)

(C) η errors ($p = 4$)

(D) η errors ($p = 5$)

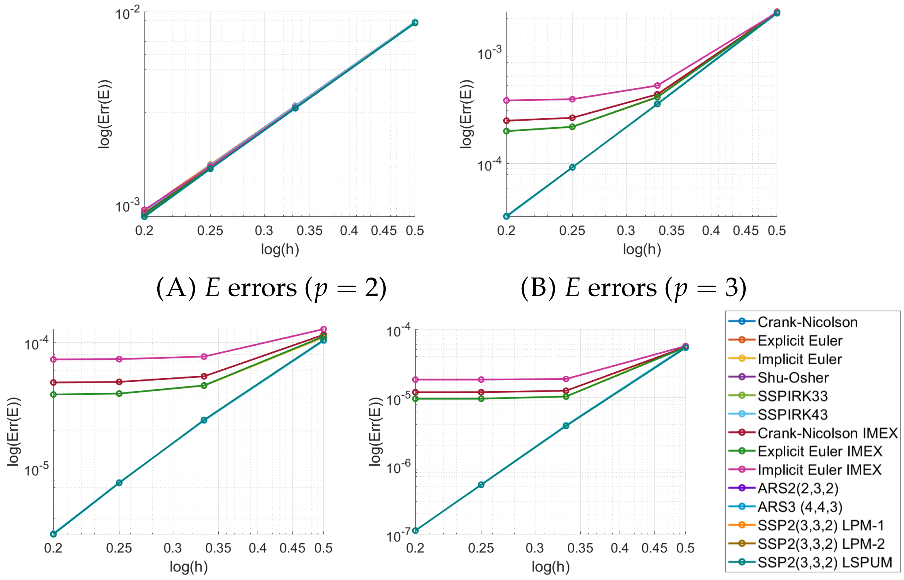
FIGURE 6.5: h-convergence results for the moving vortex problem for the free surface (η) error. Log-log scale plot of the L2 error (Err) and the mesh sizes (h) with the different temporal discretizations, using the polynomial orders $p = 2, 3, 4, 5$.



(A) **U** errors ($p = 2$) (B) **U** errors ($p = 3$)

(C) **U** errors ($p = 4$) (D) **U** errors ($p = 5$)

FIGURE 6.6: *h*-convergence results for the moving vortex problem for the velocity (**U**). Log-log scale plot of the L2 error (*Err*) and the mesh sizes (*h*) with the different temporal discretizations, using the polynomial orders $p = 2, 3, 4, 5$.



(A) E errors ($p = 2$) (B) E errors ($p = 3$)

(C) E errors ($p = 4$) (D) E errors ($p = 5$)

FIGURE 6.7: *h*-convergence results for the moving vortex problem for the energy (E). Log-log scale plot of the L2 error (*Err*) and the mesh sizes (*h*) with the different temporal discretizations, using the polynomial orders $p = 2, 3, 4, 5$.

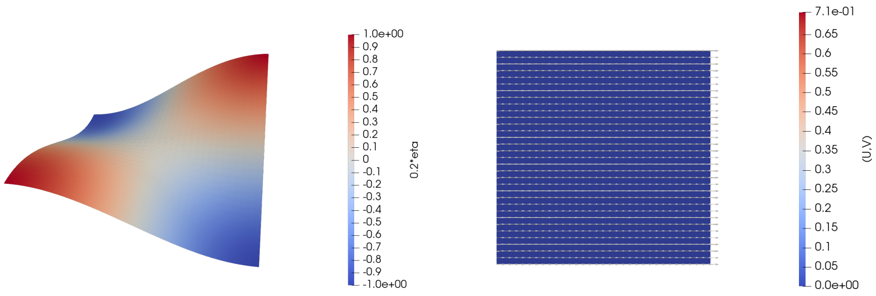
6.2 Problem 2 with analytical solution: Linear standing wave

We also study the convergence in a problem consisting on a linear standing wave, from Bui-Thanh (2016) and Iskandarani et al. (1995). This problem has an exact solution, so the error can be calculated. The interest of this simulation resides in the oscillatory nature of the wave.

6.2.1 Initial conditions

The initial conditions (Figure 6.8) are:

$$\begin{cases} \eta = \frac{\cos(\pi x)\cdot\cos(\pi y)\cdot\cos(\sqrt{2}\pi t)}{g} \\ U = \frac{1}{\sqrt{2}}\sin(\pi x)\cdot\cos(\pi y)\cdot\sin(\sqrt{2}\pi t) \\ V = \frac{1}{\sqrt{2}}\cos(\pi x)\cdot\sin(\pi y)\cdot\sin(\sqrt{2}\pi t) \end{cases}$$



(A) η

(B) $\mathbf{U}$

FIGURE 6.8: Initial conditions of the free surface elevation (η) and the velocity ($\mathbf{U}$) for the linear standing wave problem. The elevation (η) is scaled by a factor of 0.2 to improve visibility. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

The domain is $\Omega = [0,1] \times [0,1]$ m (see Figure 6.9). We choose a flat bathymetry of $b = 1$ m. Wall boundary conditions are applied on all the exterior boundaries. The external forcings, such as Coriolis force, bottom friction, and wind stress are not present.

6.2.2 Results

The mesh is constructed of quadrilateral elements of $\Delta x = \Delta y = 0.125\,\text{m}$ (Figure 6.9), with the polynomial order $p = 4$. The time step is $\Delta t = 1 \times 10^{-6}\,\text{s}$ and the end time is $t = 10\,\text{s}$.

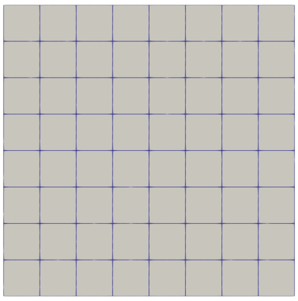
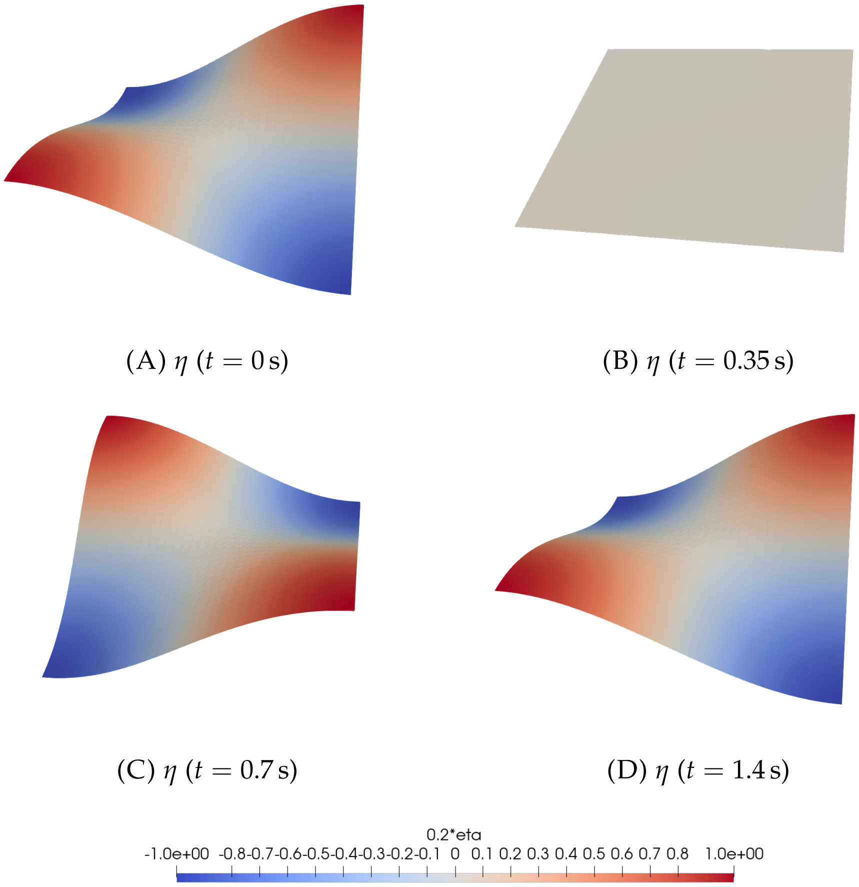


FIGURE 6.9: Mesh for the linear standing wave problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [0, 1] \times [0, 1]$ m and the element size is $\Delta x = \Delta y = 0.125\,\text{m}$.

The results for this problem (Figures 6.10 and 6.11) confirm that the model behaves as expected. The wave reflects symmetrically within the boundaries of the square domain, forming a standing wave pattern that persists over time. The results are shown up to $t = 1.4\,\text{s}$, corresponding to a complete cycle of the wave.


0.2*eta
-1.0e+00 -0.8 -0.7 -0.6 -0.5 -0.4 -0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 1.0e+00

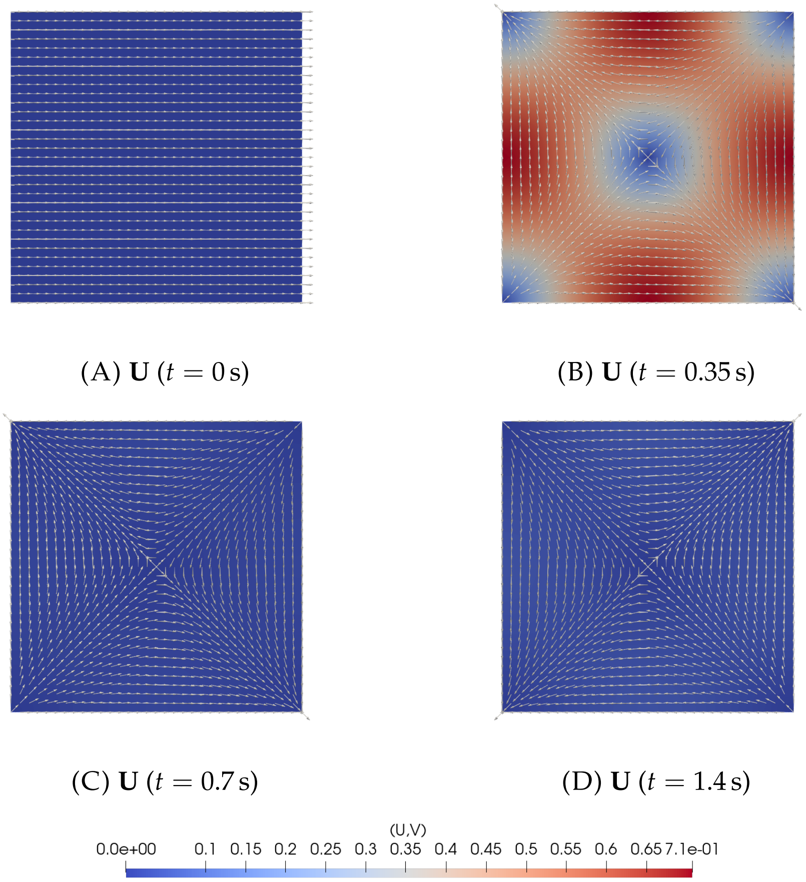
(A) η ($t = 0\,\mathrm{s}$)

(B) η ($t = 0.35\,\mathrm{s}$)

(C) η ($t = 0.7\,\mathrm{s}$)

(D) η ($t = 1.4\,\mathrm{s}$)

FIGURE 6.10: Results for the evolution of the free surface elevation (η) for the linear standing wave problem, at the times $t = 0\,\mathrm{s}$, $t = 0.35\,\mathrm{s}$, $t = 0.7\,\mathrm{s}$ and $t = 1.4\,\mathrm{s}$. The elevation (η) is scaled by a factor of 0.2 to improve visibility.


(A) **U** ($t = 0\,\text{s}$)

(B) **U** ($t = 0.35\,\text{s}$)

(C) **U** ($t = 0.7\,\text{s}$)

(D) **U** ($t = 1.4\,\text{s}$)

FIGURE 6.11: Results for the evolution of the velocity (**U**) for the linear standing wave problem, at the times $t = 0\,\text{s}$, $t = 0.35\,\text{s}$, $t = 0.7\,\text{s}$ and $t = 1.4\,\text{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.2.3 Error convergence

For the convergence test, the chosen polynomial orders for the basis functions are $p = 1, 2, 3, 4$, and the number of intervals are $N = 2, 4, 8$. The time discretization method used is the Shu-Osher (see Appendix A). The time step is $\Delta t = 1 \times 10^{-6}\,\text{s}$, and the error is calculated in the time $t = 10\,\text{s}$.

The spatial convergence tests for this problem (Figure 6.12) also shows that the error decreases as the mesh is refined, indicating optimal convergence behavior for the η, velocity, and energy errors.

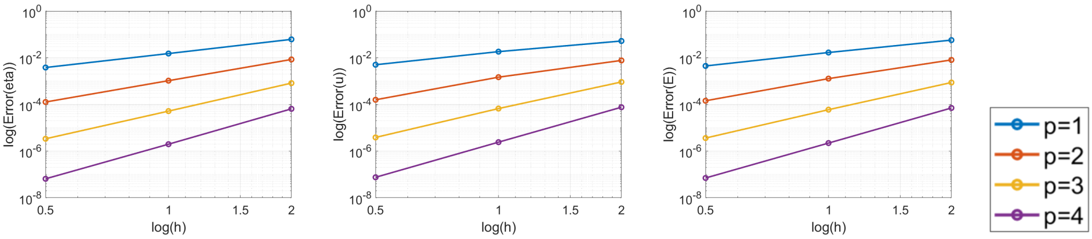
For the free surface elevation, the observed h-convergence rates are approximately 2, 3, 4 and 5 for the polynomial orders $p = 1, 2, 3, 4$, respectively (see Table 6.2). These results are consistent with the expected optimal rate of h^{p+1} reported in the literature. The velocity error convergence rates are 1.6, 3.2, 4.1 and 5 (Table 6.2), suggesting a convergence of approximately $h^{p+0.5}$,

TABLE 6.2: Error results for the linear standing wave problem using the Shu-Osher discretization. The errors are calculated for the number of intervals $N = 2, 4, 8$ and the polynomial orders $p = 1, 2, 3, 4$.

p	Nel	Error(η)	order(η)	Error(**U**)	Order(**U**)	Error($\sqrt{E}$)	Order($\sqrt{E}$)
1	2	$6.14{\cdot}10^{-2}$	NaN	$5.22{\cdot}10^{-2}$	NaN	$5.70{\cdot}10^{-2}$	NaN
	4	$1.51{\cdot}10^{-2}$	2.02	$1.82{\cdot}10^{-2}$	1.52	$1.68{\cdot}10^{-2}$	1.77
	8	$3.81{\cdot}10^{-3}$	1.99	$5.01{\cdot}10^{-3}$	1.86	$4.45{\cdot}10^{-3}$	1.91
2	2	$8.45{\cdot}10^{-3}$	NaN	$7.67{\cdot}10^{-3}$	NaN	$8.07{\cdot}10^{-3}$	NaN
	4	$1.04{\cdot}10^{-3}$	3.02	$1.47{\cdot}10^{-3}$	2.38	$1.27{\cdot}10^{-3}$	2.66
	8	$1.27{\cdot}10^{-4}$	3.04	$1.58{\cdot}10^{-4}$	3.22	$1.43{\cdot}10^{-4}$	3.15
3	2	$8.18{\cdot}10^{-4}$	NaN	$9.15{\cdot}10^{-4}$	NaN	$8.68{\cdot}10^{-4}$	NaN
	4	$5.16{\cdot}10^{-5}$	3.99	$6.64{\cdot}10^{-5}$	3.79	$5.94{\cdot}10^{-5}$	3.87
	8	$3.36{\cdot}10^{-6}$	3.94	$3.84{\cdot}10^{-6}$	4.11	$3.61{\cdot}10^{-6}$	4.04
4	2	$6.45{\cdot}10^{-5}$	NaN	$7.57{\cdot}10^{-5}$	NaN	$7.04{\cdot}10^{-5}$	NaN
	4	$1.97{\cdot}10^{-6}$	5.03	$2.41{\cdot}10^{-6}$	4.97	$2.20{\cdot}10^{-6}$	5.00
	8	$6.48{\cdot}10^{-8}$	4.93	$7.53{\cdot}10^{-8}$	5.00	$7.02{\cdot}10^{-8}$	4.97

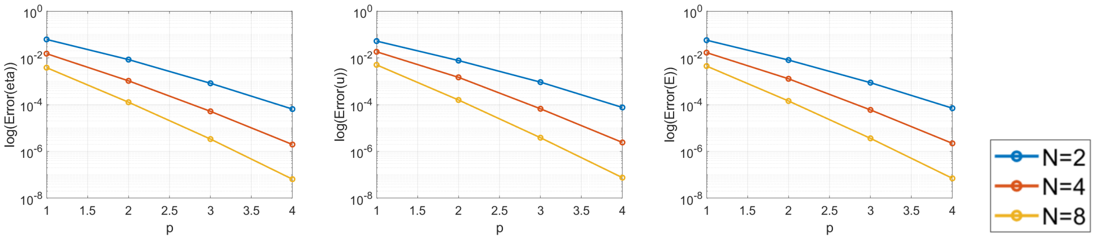
which aligns with the behavior observed in previous studies. Similarly, for the energy error, the convergence rates are 1.8, 3.1, 4 and 5 (Table 6.2), also indicating near-optimal convergence and following the trend observed with the error for the velocity.

The p-convergence curves (Figure 6.13) appear almost linear when plotted on a semi-logarithmic scale, which indicates that the p-convergence has an exponential rate, as seen in Bui-Thanh (2016).



(A) η errors (B) **U** errors (C) E errors

FIGURE 6.12: h-convergence results for the linear standing wave problem using the Shu-Osher discretization. Log-log scale plot of the L2 error (Err) and the mesh sizes (h), for the free surface (η) error (A), the velocity (**U**) error (B), and the energy (E) error (C), using the polynomial orders $p = 1, 2, 3, 4$.



(A) η errors (B) **U** errors (C) E errors

FIGURE 6.13: p-convergence results for the linear standing wave problem using the Shu-Osher discretization. Semi-log scale plot of the L2 error *Err* and the polynomial order p, for the free surface (η) error (A), the velocity (**U**) error (B), and the energy (E) error (C), using the number of intervals $N = 2, 4, 8$.

6.3 Problem 3 with analytical solution: Linear Kelvin wave

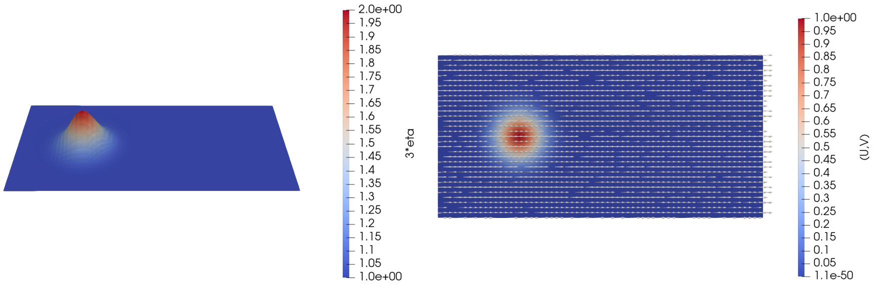
This third problem is also used to study the error convergence of the model. We now simulate a linear equatorial Kelvin wave translating inside the domain, following the conditions described in Bui-Thanh (2016) and Eskilsson and Sherwin (2004). The wave must propagate eastward throughout the domain, while maintaining its shape.

6.3.1 Initial conditions

The initial conditions (Figure 6.14) are:

$$\begin{cases} \eta = \dfrac{1+\exp\left(-\frac{y^2}{2}\right)\exp\left(-\frac{(x+5-t)^2}{2}\right)}{g} \\ U = \exp\left(-\frac{y^2}{2}\right)\exp\left(-\frac{(x+5-t)^2}{2}\right) \\ V = 0 \end{cases}$$

The domain for the problem is $\Omega = [-10, 10] \times [-5, 5]$ m (Figure 6.15). We take a flat bathymetry of $b = 1\,\mathrm{m}$. The boundary conditions on the x-direction are periodic and wall boundary conditions are imposed in the y-direction. The Coriolis force is imposed with a β plane approximation ($f = f_0 + \beta(y - y_m)$), with $f_0 = 0$, $y_m = 0$ and $\beta = 1$.



(A) η

(B) $\mathbf{U}$

FIGURE 6.14: Initial conditions of the free surface elevation (η) and the velocity ($\mathbf{U}$) for the linear Kelvin wave problem. The elevation (η) is scaled by a factor of 3 to improve visibility. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.3.2 Results

The mesh for the problem is formed by quadrilateral elements of $\Delta x = \Delta y = 1.25\,\text{m}$, and polynomial order $p = 4$, with a time step of $\Delta t = 1 \times 10^{-4}\,\text{s}$ (Figure 6.15). The end time is $t = 10\,\text{s}$.

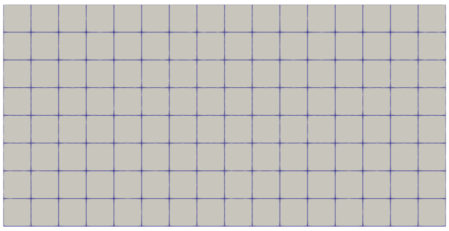
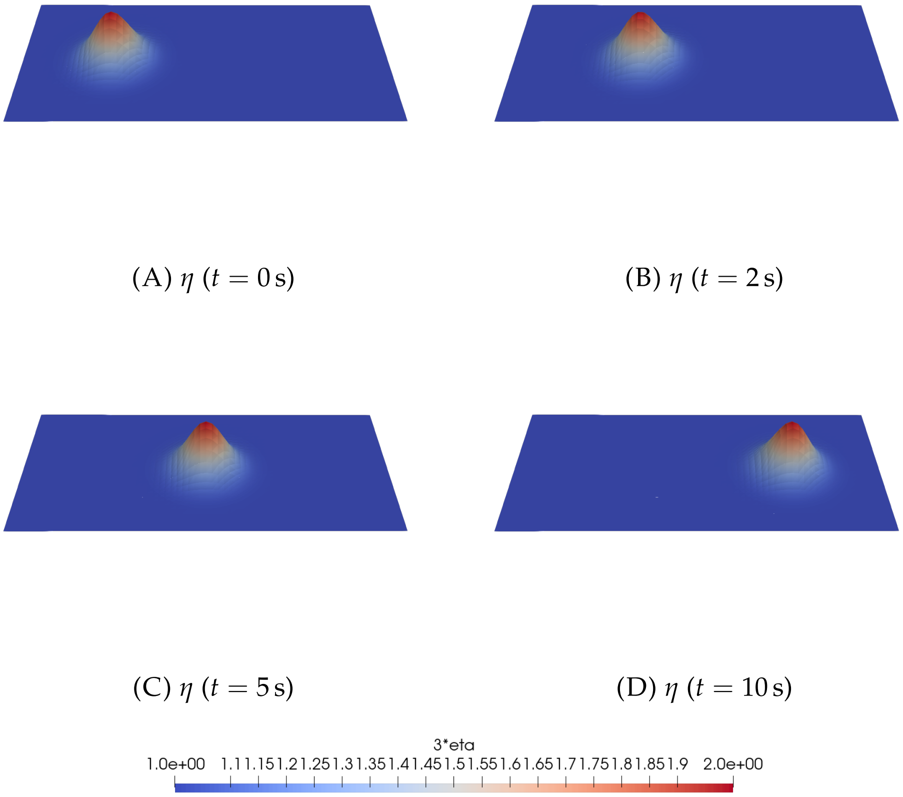


FIGURE 6.15: Mesh for the linear Kelvin wave problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [-10, 10] \times [-5, 5]$ m and the element size is $\Delta x = \Delta y = 1.25\,\text{m}$.

The results for the temporal evolution of the free surface elevation η and velocity are shown in Figures 6.16 and 6.17. The wave moves eastward, from the left of the domain. Similarly, the velocity behaves as expected, as the wave is transported.


3*eta
1.0e+00 1.11.151.21.251.31.351.41.451.51.551.61.651.71.751.81.851.9 2.0e+00

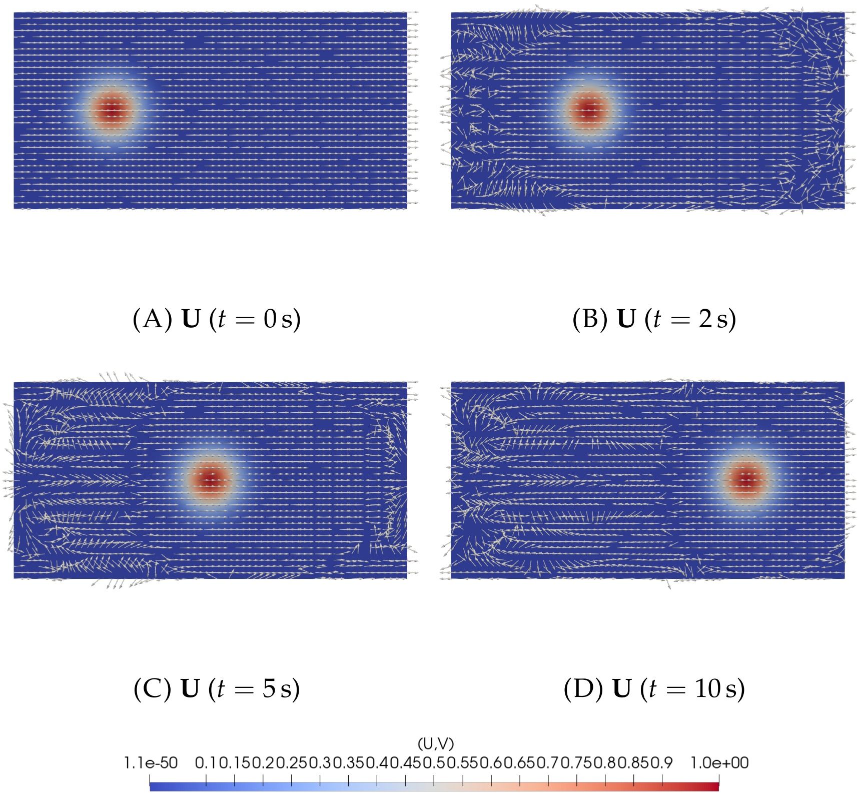
(A) η $(t = 0\,\mathrm{s})$

(B) η $(t = 2\,\mathrm{s})$

(C) η $(t = 5\,\mathrm{s})$

(D) η $(t = 10\,\mathrm{s})$

FIGURE 6.16: Results for the evolution of the free surface elevation (η) for the linear Kelvin wave problem, at the times $t = 0\,\mathrm{s}$, $t = 2\,\mathrm{s}$, $t = 5\,\mathrm{s}$ and $t = 10\,\mathrm{s}$. The elevation (η) is scaled by a factor of 3 to improve visibility.


(A) $\mathbf{U}$ ($t = 0\,\mathrm{s}$)

(B) $\mathbf{U}$ ($t = 2\,\mathrm{s}$)

(C) $\mathbf{U}$ ($t = 5\,\mathrm{s}$)

(D) $\mathbf{U}$ ($t = 10\,\mathrm{s}$)

FIGURE 6.17: Results for the evolution of the velocity ($\mathbf{U}$) for the linear Kelvin wave problem, at the times $t = 0\,\mathrm{s}$, $t = 2\,\mathrm{s}$, $t = 5\,\mathrm{s}$ and $t = 10\,\mathrm{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.3.3 Error convergence

The polynomial orders for the convergence test are $p = 1, 2, 3, 4$ and the number of intervals are $N = 2, 4, 8, 16$. The time discretization method used is the Crank-Nicolson (see Appendix A). The time step is $\Delta t = 1 \times 10^{-4}\,\mathrm{s}$. The error is calculated at $t = 10\,\mathrm{s}$.

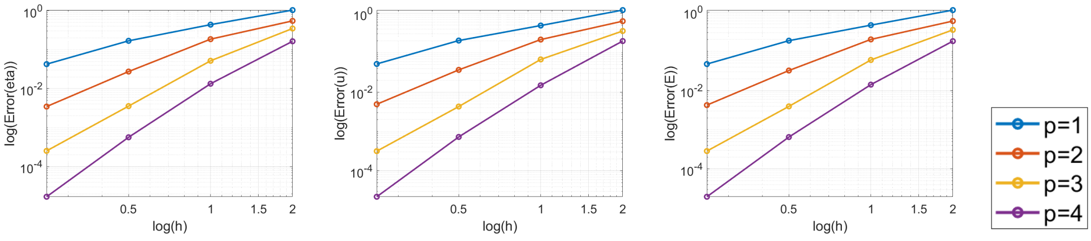
The spatial convergence tests (Figure 6.18) show that the error behaves with optimal convergence. For the free surface elevation, velocity, and energy errors, the observed h-convergence rates are approximately 2, 2.9, 3.8 and 5 for $p = 1$, 2, 3 and 4 (see Table 6.3). The results for the elevation, similarly to the previous problems, are consistent with the expected optimal rate h^{p+1} reported in the literature. In the case of the velocity and the energy, the optimal rate of $h^{p+0.5}$ is also reached.

The p-convergence curves (Figure 6.19) appear almost linear on the semi-logarithmic scale, indicating that the p-convergence has an exponential rate.

TABLE 6.3: Error results for the linear Kelvin wave problem using the Crank-Nicolson time discretization. The errors are calculated for the number of intervals $N = 2, 4, 8, 16$ and the polynomial orders $p = 1, 2, 3, 4$.

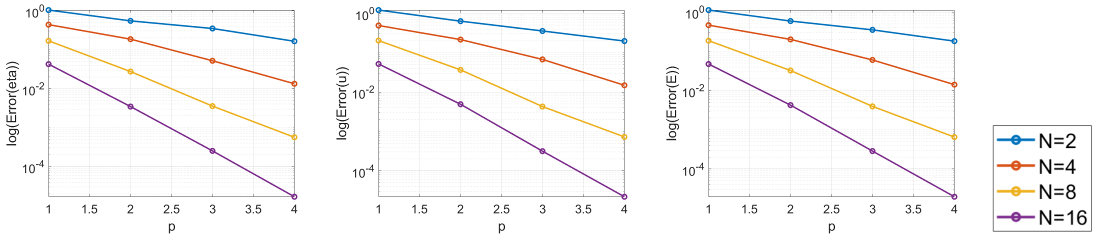
p	Nel	Error(η)	order(η)	Error($\mathbf{U}$)	Order($\mathbf{U}$)	Error($\sqrt{E}$)	Order($\sqrt{E}$)
1	2	1.03	NaN	1.21	NaN	1.13	NaN
	4	$4.37{\cdot}10^{-1}$	1.24	$4.90{\cdot}10^{-1}$	1.30	$4.64{\cdot}10^{-1}$	1.28
	8	$1.68{\cdot}10^{-1}$	1.38	$2.04{\cdot}10^{-1}$	1.27	$1.87{\cdot}10^{-1}$	1.31
	16	$4.24{\cdot}10^{-2}$	1.99	$5.19{\cdot}10^{-2}$	1.97	$4.74{\cdot}10^{-2}$	1.98
2	2	$5.46{\cdot}10^{-1}$	NaN	$6.32{\cdot}10^{-1}$	NaN	$5.91{\cdot}10^{-1}$	NaN
	4	$1.86{\cdot}10^{-1}$	1.56	$2.16{\cdot}10^{-1}$	1.55	$2.01{\cdot}10^{-1}$	1.55
	8	$2.74{\cdot}10^{-2}$	2.76	$3.67{\cdot}10^{-2}$	2.56	$3.24{\cdot}10^{-2}$	2.64
	16	$3.47{\cdot}10^{-3}$	2.98	$4.91{\cdot}10^{-3}$	2.90	$4.25{\cdot}10^{-3}$	2.93
3	2	$3.48{\cdot}10^{-1}$	NaN	$3.56{\cdot}10^{-1}$	NaN	$3.52{\cdot}10^{-1}$	NaN
	4	$5.18{\cdot}10^{-2}$	2.75	$6.76{\cdot}10^{-2}$	2.40	$6.02{\cdot}10^{-2}$	2.55
	8	$3.56{\cdot}10^{-3}$	3.86	$4.30{\cdot}10^{-3}$	3.97	$3.95{\cdot}10^{-3}$	3.93
	16	$2.53{\cdot}10^{-4}$	3.82	$3.16{\cdot}10^{-4}$	3.76	$2.86{\cdot}10^{-4}$	3.78
4	2	$1.64{\cdot}10^{-1}$	NaN	$1.98{\cdot}10^{-1}$	NaN	$1.82{\cdot}10^{-1}$	NaN
	4	$1.33{\cdot}10^{-2}$	3.62	$1.49{\cdot}10^{-2}$	3.73	$1.41{\cdot}10^{-2}$	3.68
	8	$5.66{\cdot}10^{-4}$	4.56	$7.27{\cdot}10^{-4}$	4.36	$6.52{\cdot}10^{-4}$	4.44
	16	$1.69{\cdot}10^{-5}$	5.07	$2.20{\cdot}10^{-5}$	5.05	$1.96{\cdot}10^{-5}$	5.05

In high-order methods like HDG, accuracy can be improved either by refining the mesh (h-refinement) or by increasing the polynomial order (p-refinement). While h-refinement is generally effective for resolving localized features or discontinuities, p-refinement is often more efficient, as it can achieve exponential convergence with fewer degrees of freedom (Bui-Thanh, 2016). This is confirmed by the results shown in Figure 6.13 and 6.19, where the error decreases rapidly as p increases, demonstrating the advantage of using high-order approximations in this problem.



(A) η errors (B) $\mathbf{U}$ errors (C) E errors

FIGURE 6.18: h-convergence results for the linear Kelvin wave problem using the Crank-Nicolson time discretization. Log-log scale plot of the L2 error (Err) and the mesh sizes (h), for the free surface (η) error (A), the velocity ($\mathbf{U}$) error (B), and the energy (E) error (C), using the polynomial orders $p = 1, 2, 3, 4$.



(A) η errors (B) **U** errors (C) E errors

FIGURE 6.19: p-convergence results for the linear Kelvin wave problem using the Crank-Nicolson time discretization. Semi-log scale plot of the L2 error Err and the polynomial order p, for the free surface (η) error (A), the velocity (**U**) error (B), and the energy (E) error (C), using the number of intervals $N = 2, 4, 8, 16$.

6.4 Problem 4: Lake at rest

We now apply a lake at rest test to the model for the non-linear SWE. A numerical scheme is said to be well-balanced if it is capable of preserving the steady state solution, such as the lake-at-rest configuration, throughout the simulation. In this problem, we simulate a variation of the steady lake problem proposed by Valseth and Dawson (2021) with a discontinuous bathymetry. The numerical method must handle the sharp discontinuities in the bathymetry.

6.4.1 Initial conditions

For this problem the initial conditions are:

$$\begin{cases} \eta = 0 \\ (U, V) = (0, 0) \end{cases}$$

The discontinuous bathymetry (Figure 6.20) is defined as:

$$b = \begin{cases} 2 - 1.8, & \text{if } |x - x_c| < 0.025, \ |y - y_c| < 0.025 \\ 2 - 1.4, & \text{if } |x - x_c| < 0.075, \ |y - y_c| < 0.075 \\ 2 - 1.0, & \text{if } |x - x_c| < 0.175, \ |y - y_c| < 0.175 \\ 2 - 0, & \text{otherwise} \end{cases}$$

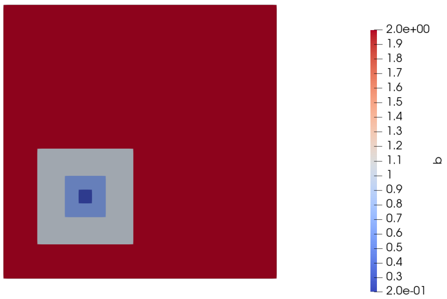


FIGURE 6.20: Discontinuous bathymetry ($b(x, y)$) for the lake at rest problem.

where $(x_c, y_c) = (0.3, 0.3)$ is the mountain center.

The domain is $\Omega = [0, 1] \times [0, 1]$ m (Figure 6.21). We apply wall boundary conditions. We impose bottom friction by setting the damping coefficient $\tau_b = 1s^{-1}$.

6.4.2 Results

We use a mesh with triangular elemnts of the size $\Delta x = \Delta y = 0.025$ m and polynomial order $p = 1$ (Figure 6.21), with the time step $\Delta t = 0.1$ s. The final time is $t = 2$ s.

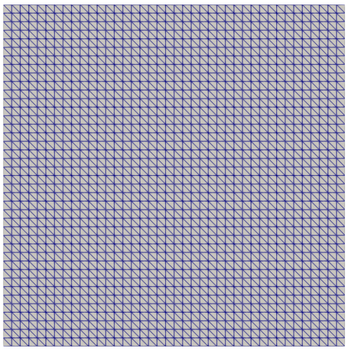
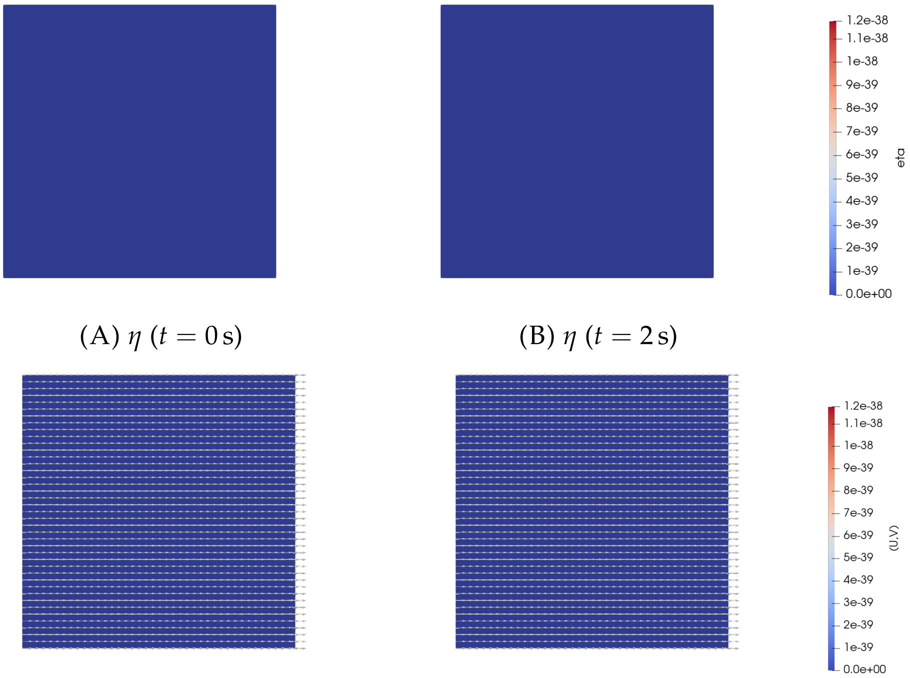


FIGURE 6.21: Mesh for the lake at rest problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [0, 1] \times [0, 1]$ m and the element size is $\Delta x = \Delta y = 0.025$ m.

As we can see in the results (Figure 6.22), the lake at rest state is well preserved by the end of the simulation at $t = 2s$, proving that the method is, indeed, well-balanced. Note that although the velocity arrows appear to point eastward, the actual velocity magnitude is zero. This occurs because the arrows are normalized - i.e., all arrows are drawn with the same length regardless of magnitude. As a result, even when the velocity is zero, default arrows pointing to the right are still displayed, while the magnitude is accurately shown through the color shading.



(A) η ($t = 0\,\mathrm{s}$)

(B) η ($t = 2\,\mathrm{s}$)

(C) $\mathbf{U}$ ($t = 0\,\mathrm{s}$)

(D) $\mathbf{U}$ ($t = 2\,\mathrm{s}$)

FIGURE 6.22: Results for the evolution of the free surface elevation (η) (A,B) and velocity ($\mathbf{U}$) (C,D) for the lake at rest problem, at the times $t = 0\,\mathrm{s}$ and $t = 2\,\mathrm{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.5 Problem 5: Shallow lake

In this section, we simulate the evolution of surface waves in a square shallow lake, initially perturbed at the center by a localized discontinuity. This disturbance generates long circular gravity waves that propagate outward across the domain. The problem setup, taken from Kämpf (2009), is solved using the linear form of the shallow water equations, in order to analyze the fluid behavior in a closed domain. This test allows us to observe the surface waves caused by a perturbation, and to assess the model's ability to capture wave propagation and reflection in idealized geometries.

6.5.1 Initial conditions

The initial conditions (Figure 6.23) are:

$$\begin{cases} \eta = \begin{cases} 1 \text{ if } (x,y) = (250,250) \\ 0 \text{ otherwise} \end{cases} \\ (U,V) = (0,0) \end{cases}$$

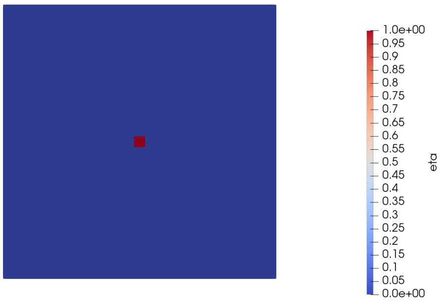


FIGURE 6.23: Initial condition of the free surface elevation (η) for the shallow lake problem.

The domain is $\Omega = [0,500] \times [0,500]$ m (Figure 6.24), with the depth of $\phi_B = 10$ m. The boundary conditions are wall boundaries.

6.5.2 Results

The mesh is formed by quadrilateral elements of polynomial order $p = 1$ and size $\Delta x = \Delta y = 10$ m (Figure 6.24). The time step is $\Delta t = 0.1$ s. The final time is $t = 100$ s.

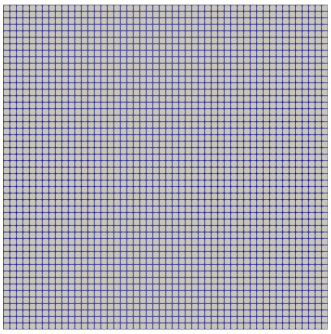
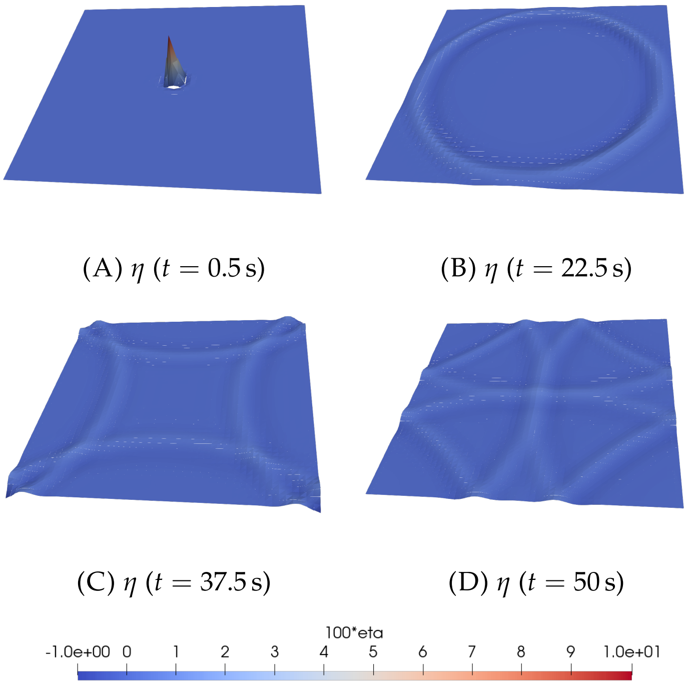


FIGURE 6.24: Mesh for the shallow lake problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [0, 500] \times [0, 500]$ m and the element size is $\Delta x = \Delta y = 10$ m.

The results (Figures 6.25 and 6.26) show how the initial perturbation generates gravity waves that propagate radially, outward, from the center of the lake. As expected in a closed lake, the waves reflect off the boundaries and interact with one another, producing complex interference patterns. The velocity field exhibits the same behavior as the free surface elevation. The simulation preserves this symmetry throughout the domain and over time, highlighting model's ability to accurately represent wave propagation and reflection. This behavior is observed up to $t = 100$ s; however, only the evolution up to $t = 50$ s is shown here to avoid redundancy.



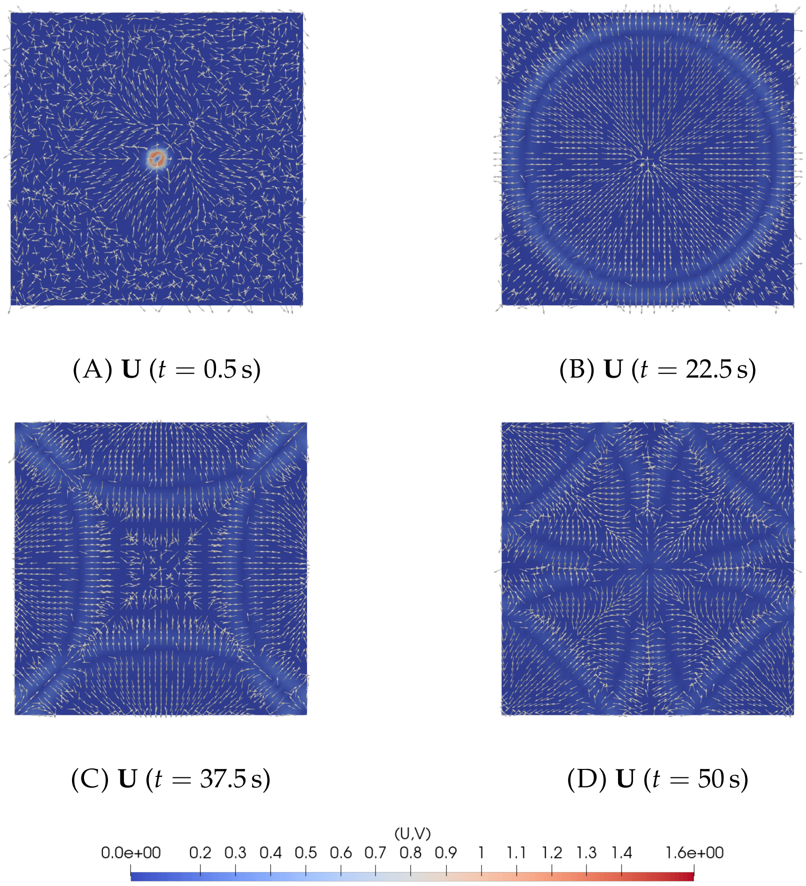
(A) η $(t = 0.5\,\mathrm{s})$

(B) η $(t = 22.5\,\mathrm{s})$

(C) η $(t = 37.5\,\mathrm{s})$

(D) η $(t = 50\,\mathrm{s})$

FIGURE 6.25: Results for the evolution of the free surface elevation (η) for the shallow lake problem, at the times $t = 0.5\,\mathrm{s}$, $t = 22.5\,\mathrm{s}$, $t = 37.5\,\mathrm{s}$ and $t = 50\,\mathrm{s}$. The elevation (η) is scaled by a factor of 100 to improve visibility.


(A) $\mathbf{U}$ ($t = 0.5\,\mathrm{s}$)

(B) $\mathbf{U}$ ($t = 22.5\,\mathrm{s}$)

(C) $\mathbf{U}$ ($t = 37.5\,\mathrm{s}$)

(D) $\mathbf{U}$ ($t = 50\,\mathrm{s}$)

FIGURE 6.26: Results for the evolution of the velocity ($\mathbf{U}$) for the shallow lake problem, at the times $t = 0.5\,\mathrm{s}$, $t = 22.5\,\mathrm{s}$, $t = 37.5\,\mathrm{s}$ and $t = 50\,\mathrm{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.5.3 Adding bathymetry

In this section, a non-flat bathymetry (Figure 6.27) is introduced by adding a submerged mount off-center, at the southwest of the domain. This allows us to study the influence of topography on the wave propagation.

$$\phi_B = \begin{cases} 10 - H \cdot (x - x_{\min}) \cdot (x - x_{\max}) \cdot (y - y_{\min}) \cdot (y - y_{\max}) \\ \qquad \text{if} \quad x_{\min} < x < x_{\max}, y_{\min} < y < y_{\max} \\ 10 \quad \text{otherwise} \end{cases}$$

with $H = 20 \cdot 10^{-6}$ as the mount height, $x_{\min} = 50m$, $x_{\max} = 100m$, $y_{\min} = 50m$, and $y_{\max} = 100m$.

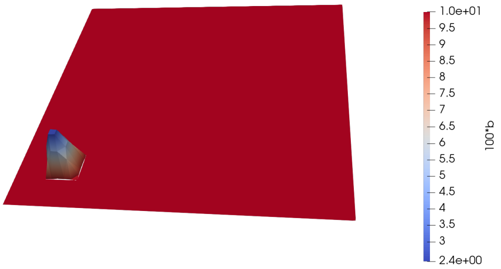
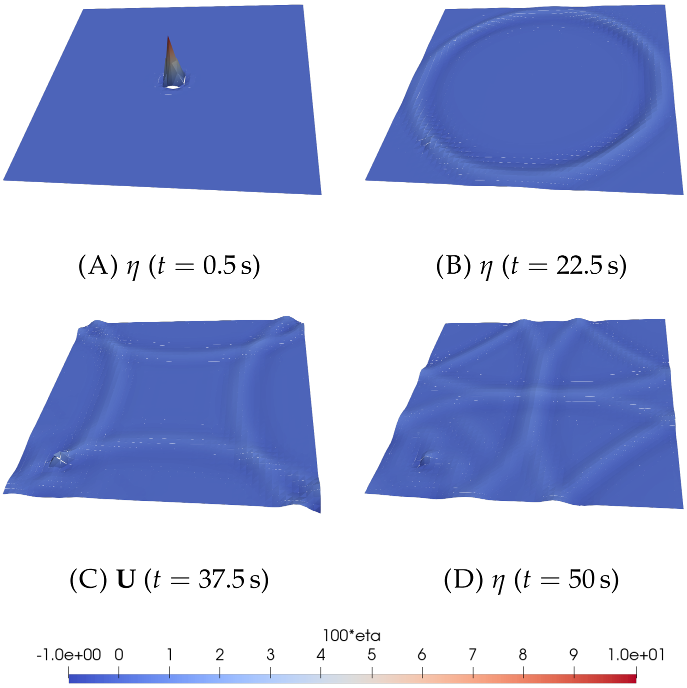


FIGURE 6.27: Bathymetry ($b(x,y)$) for the shallow lake problem. The height is scaled by a factor of 100 to improve visibility.



(A) η ($t = 0.5$ s)

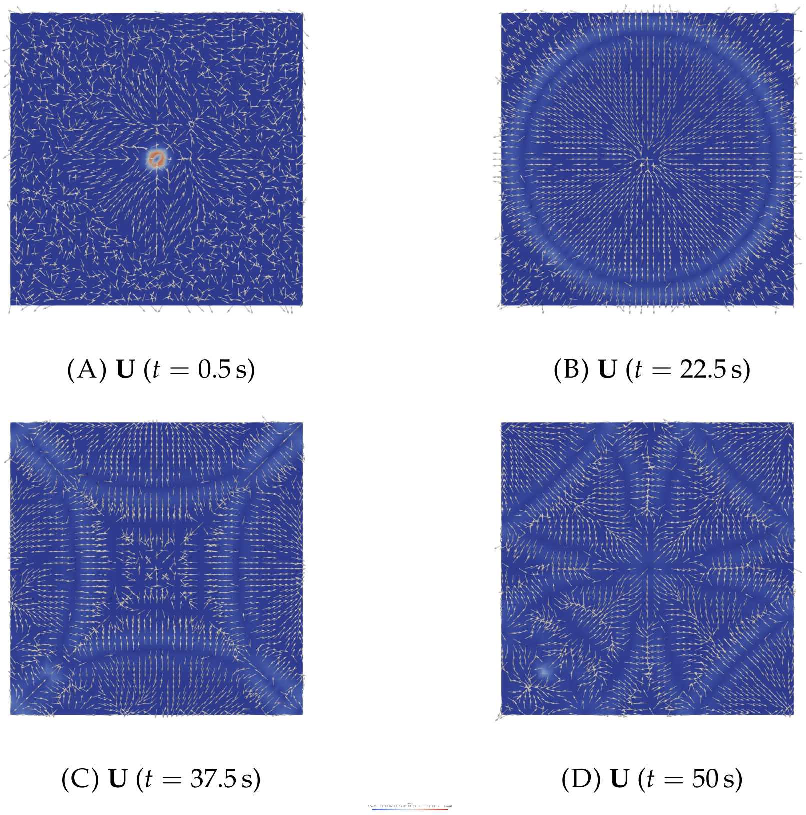
(B) η ($t = 22.5$ s)

(C) $\mathbf{U}$ ($t = 37.5$ s)

(D) η ($t = 50$ s)

FIGURE 6.28: Results for the evolution of the free surface elevation (η) for the shallow lake problem with bathymetry, at the times $t = 0.5$ s, $t = 22.5$ s, $t = 37.5$ s and $t = 50$ s. The elevation (η) is scaled by a factor of 100 to improve visibility.

The results (Figures 6.28 and 6.29) show a perturbation of the wave symmetry caused by the submerged mount. The waves are partially concentrated and refracted around the topographic feature, as observed in the free surface elevation η. The velocity field also shows higher magnitudes in the area of the mount.



(A) $\mathbf{U}$ ($t = 0.5\,\mathrm{s}$)

(B) $\mathbf{U}$ ($t = 22.5\,\mathrm{s}$)

(C) $\mathbf{U}$ ($t = 37.5\,\mathrm{s}$)

(D) $\mathbf{U}$ ($t = 50\,\mathrm{s}$)

FIGURE 6.29: Results for the evolution of the velocity ($\mathbf{U}$) for the shallow lake problem with bathymetry, at the times $t = 0.5\,\mathrm{s}$, $t = 22.5\,\mathrm{s}$, $t = 37.5\,\mathrm{s}$ and $t = 50\,\mathrm{s}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.6 Problem 6: Shallow lake with wind-driven flow

In this section, we simulate a shallow lake with an island located off-center, based on the test case from Kämpf (2009). The system starts at rest, and a constant wind stress is applied at the surface, initiating a wind-driven circulation. This setup allows us to observe the effects of external forcing on the shallow water model. The asymmetric placement of the island leads to the formation of complex flow patterns.

6.6.1 Initial conditions

The initial conditions for this problem are:

$$\begin{cases} \eta = 0 \\ (U, V) = (0, 0) \end{cases}$$

The bathymetry (Figure 6.30B) is defined by the following function, and then smoothed:

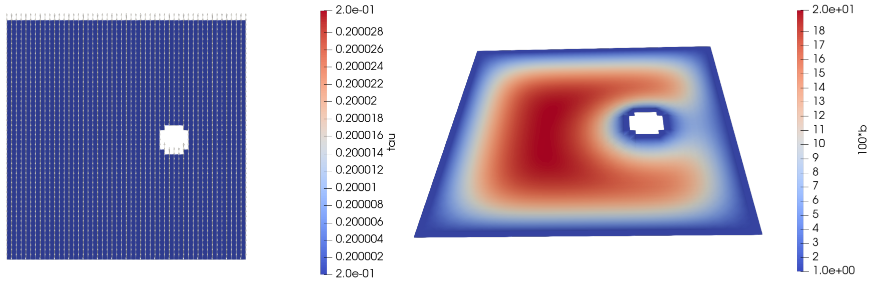
$$\phi_B = \begin{cases} 1, & \text{if } (0 < x < 300), (0 < y < 300), (4900 < x < 5000), (4900 < y < 5000) \\ 20, & \text{if } (300 \leq x \leq 4900), (300 \leq y \leq 4900) \\ 1, & \text{if } \sqrt{(x - x_c)^2 + (y - y_c)^2} < 500 \\ -0.2, & \text{if } \sqrt{(x - x_c)^2 + (y - y_c)^2} < 400 \\ -10 & \text{if } x = 0, x = 5000, y = 0, y = 5000 \end{cases}$$

Where $(x_c, y_c) = (2600, 3600)$ are the coordinates of the center of the island.

The wind stress in the y-direction increases linearly with time (Figure 6.30A), from $\tau_y = 0\,\text{Pa}$ until reaching $\tau_y = 0.2\,\text{Pa}$ in 2 days.

$$\begin{cases} \tau_x = 0 \\ \tau_y(t) = \min\left(\frac{0.2 \cdot t}{(2 \cdot 24 \cdot 3600)}, 0.2\right) \end{cases}$$

The domain is $\Omega = [0, 5000] \times [0, 5000]$ m (Figure 6.31). We impose wall boundary conditions, and the bottom friction is $\boldsymbol{\tau_b} = 1 \times 10^{-3}\,\text{s}^{-1}$.



(A) τ (B) $b(x,y)$

FIGURE 6.30: Maximum wind stress (τ) and bathymetry ($b(x,y)$) for the lake with wind-driven flow problem. The height of the bathymetry is scaled by a factor of 100 to improve visibility.

6.6.2 Results

We use quadrilateral elements of $\Delta x = \Delta y = 100\,\text{m}$ and $p = 1$ (Figure 6.31). The time step is $\Delta t = 3\,\text{s}$ and the final time is $t = 5\,\text{d}$.

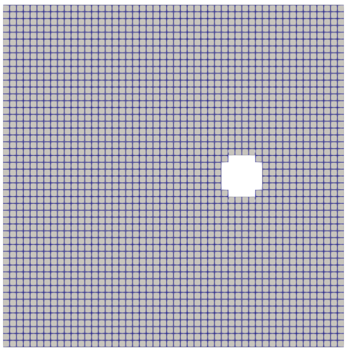
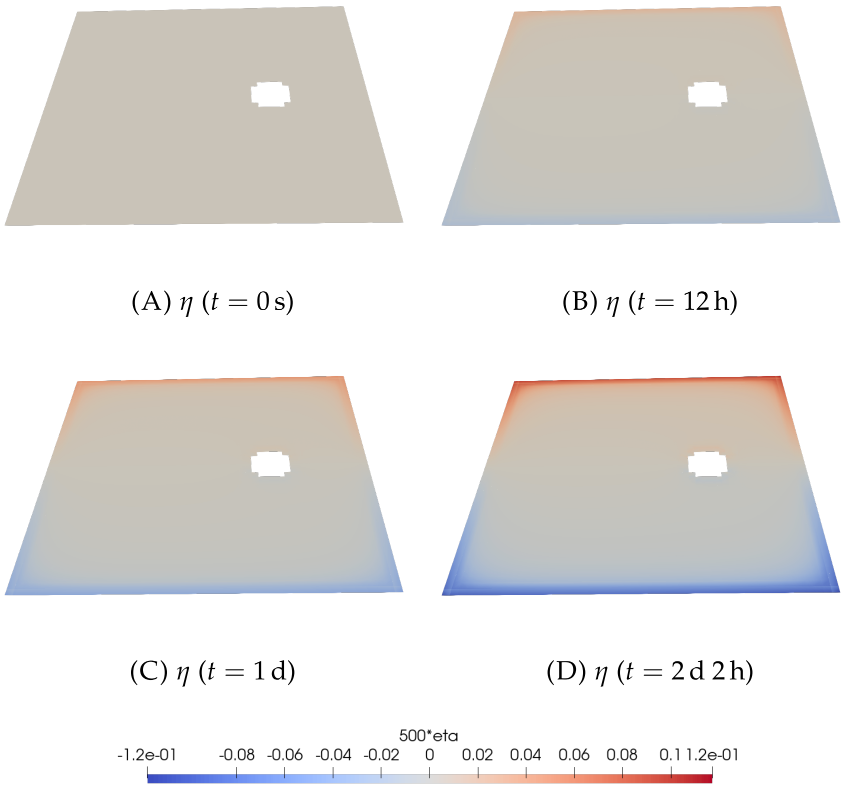


FIGURE 6.31: Mesh for the lake with wind-driven flow problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [0, 5000] \times [0, 5000]$ m and the element size is $\Delta x = \Delta y = 100\,\text{m}$.

The elevation and velocity fields of the wind-driven flow (Figures 6.32 and 6.33) behave as expected. As the wind stress gradually increases, the flow progressively adjusts, generating a cyclonic circulation around the island and an anticyclonic gyre on the western side of the domain. After the wind stress reaches its maximum constant value at $t = 2\,\text{d}$, the system continues to evolve and eventually approaches a steady state. The circulation stabilizes approximately 2 days and 2 hours into the simulation, once the wind forcing has become steady and the internal dynamics have had time to adjust.



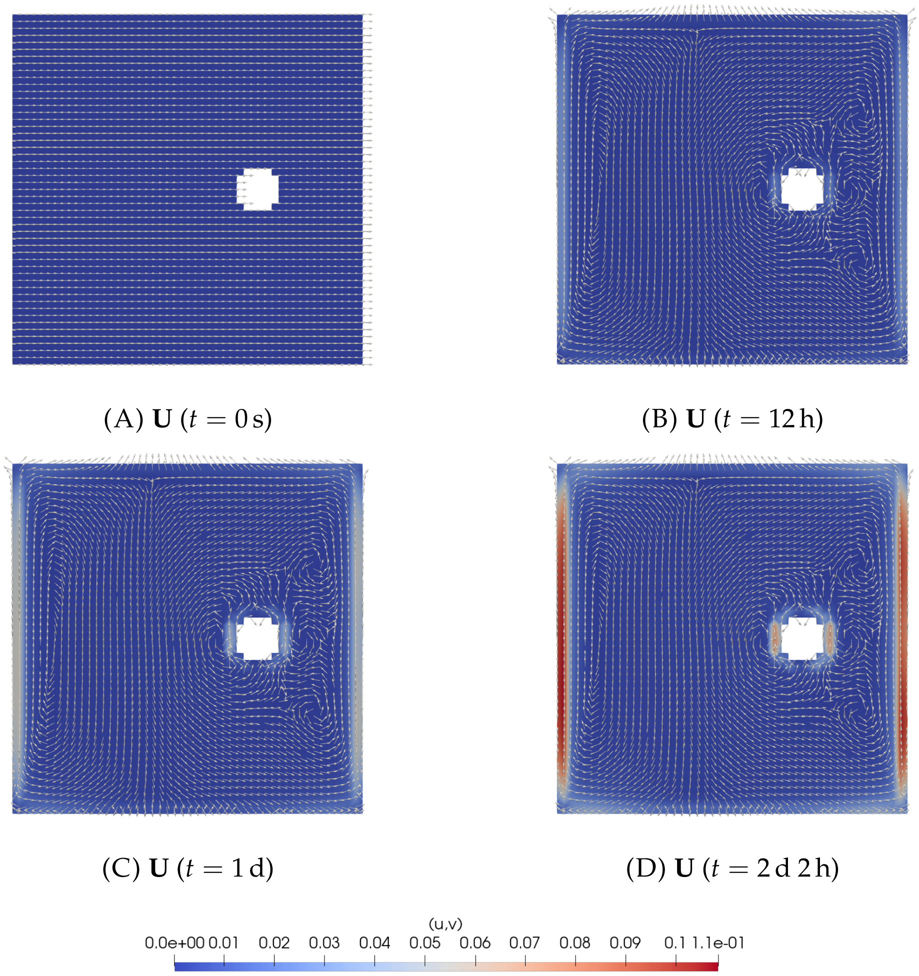
(A) η $(t = 0\,\mathrm{s})$

(B) η $(t = 12\,\mathrm{h})$

(C) η $(t = 1\,\mathrm{d})$

(D) η $(t = 2\,\mathrm{d}\;2\,\mathrm{h})$

FIGURE 6.32: Results for the evolution of the free surface elevation (η) for the lake with wind-driven flow problem, at the times $t = 0\,\mathrm{s}$, $t = 12\,\mathrm{h}$, $t = 1\,\mathrm{d}$ and $t = 2\,\mathrm{d}\;2\,\mathrm{h}$. The elevation ($\eta$) is scaled by a factor of 500 to improve visibility.



(A) $\mathbf{U}$ ($t = 0\,\text{s}$)

(B) $\mathbf{U}$ ($t = 12\,\text{h}$)

(C) $\mathbf{U}$ ($t = 1\,\text{d}$)

(D) $\mathbf{U}$ ($t = 2\,\text{d}\;2\,\text{h}$)

FIGURE 6.33: Results for the evolution of the velocity ($\mathbf{U}$) for the lake with wind-driven flow problem, at the times $t = 0\,\text{s}$, $t = 12\,\text{h}$, $t = 1\,\text{d}$ and $t = 2\,\text{d}\;2\,\text{h}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.7 Problem 7: Water height perturbation

In this problem, we simulate the propagation of smooth gravity waves in a closed square domain, using the non-linear shallow water equations. The problem setup, based on Dumbser and Casulli (2013) and Kang et al. (2020), starts from an initial perturbation in the water height centered in the domain, which triggers the outward propagation of circular gravity waves. Unlike the previous linear case (section 6.5), here we consider the full non-linear dynamics of the system, which allows us to capture additional effects such as non-linear wave interactions and amplitude-dependent wave speeds. We will solve the problem with and without the Coriolis force, to observe its effects.

6.7.1 Initial conditions

The initial conditions are:

$$\begin{cases} \eta = \exp^{-\frac{x^2+y^2}{2\sigma^2}} \\ (U, V) = (0, 0) \end{cases}$$

with $\sigma = 0.1$.

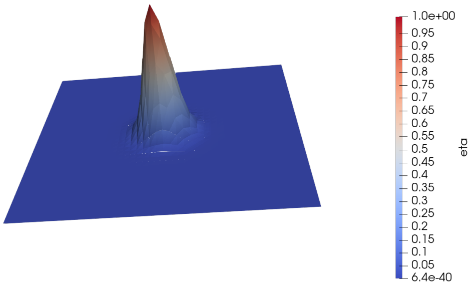


FIGURE 6.34: Initial condition of the free surface elevation (η) for the water height perturbation problem.

The domain is a square $\Omega = [-1, 1] \times [-1, 1]$ m (Figure 6.35), with a flat bathymetry of $b = 100$ m. We apply a wall boundary condition. The Coriolis force is applied imposing $f = 6.6\,\mathrm{s}^{-1}$.

6.7.2 Results

The mesh is formed by quadrilateral elements of the size $\Delta x = \Delta y = 0.1$ m and the polynomial order $p = 2$ (Figure 6.35), and the time step is $\Delta t = 10^{-4} s$. The final time of the simulation is $t = 0.5$.

In the results for this problem (Figures 6.36 and 6.37), we can observe how the initial perturbation, in the center, creates waves propagating outwardly. Then, the waves reflect off the boundaries, and overlap, forming interference patterns. This is observed in the elevation (η) and the velocity. In the results we can observe how the Coriolis force deflects the currents clockwise, rotating the structures formed by the wave patterns.

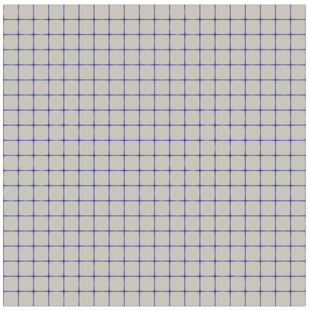
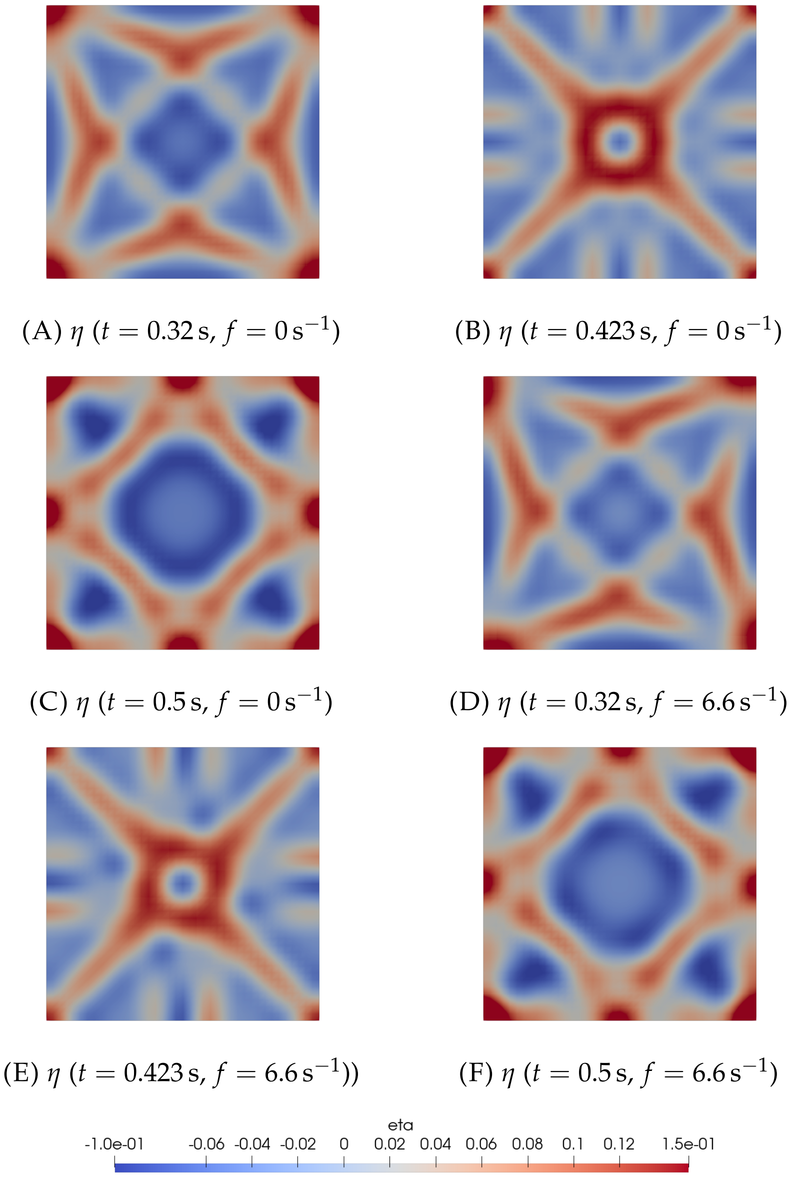


FIGURE 6.35: Mesh for the water height perturbation problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [-1, 1] \times [-1, 1]$ m and the element size is $\Delta x = \Delta y = 0.1\,\text{m}$.



(A) η ($t = 0.32\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

(B) η ($t = 0.423\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

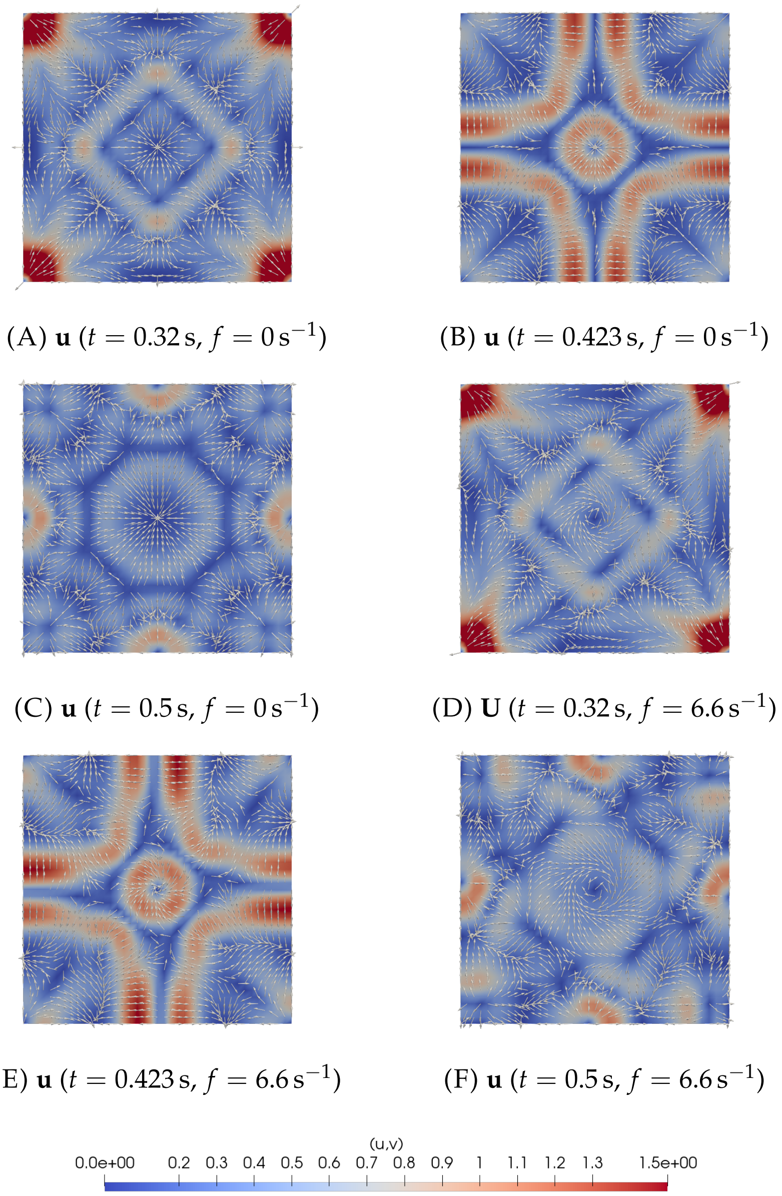
(C) η ($t = 0.5\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

(D) η ($t = 0.32\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$)

(E) η ($t = 0.423\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$))

(F) η ($t = 0.5\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$)

FIGURE 6.36: Results for the evolution of the free surface elevation (η) for the water height perturbation problem, at the times $t = 0.32\,\mathrm{s}$, $t = 0.423\,\mathrm{s}$ and $t = 0.5\,\mathrm{s}$, with $f = 0\,\mathrm{s}^{-1}$ (A, B, C) and $f = 6.6\,\mathrm{s}^{-1}$ (D, E, F).



(A) $\mathbf{u}$ ($t = 0.32\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

(B) $\mathbf{u}$ ($t = 0.423\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

(C) $\mathbf{u}$ ($t = 0.5\,\mathrm{s}$, $f = 0\,\mathrm{s}^{-1}$)

(D) $\mathbf{U}$ ($t = 0.32\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$)

(E) $\mathbf{u}$ ($t = 0.423\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$)

(F) $\mathbf{u}$ ($t = 0.5\,\mathrm{s}$, $f = 6.6\,\mathrm{s}^{-1}$)

FIGURE 6.37: Results for the evolution of the geopotential momentum ($\mathbf{u}$) for the water height perturbation, at the times $t = 0.32\,\mathrm{s}$, $t = 0.423\,\mathrm{s}$ and $t = 0.5\,\mathrm{s}$, with $f = 0\,\mathrm{s}^{-1}$ (A, B, C) and $f = 6.6\,\mathrm{s}^{-1}$ (D, E, F). The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

6.8 Problem 8: Coastal inlet

Lastly, we simulate a more complex and realistic coastal scenario. This problem consists of an inflow of water propagating from the open ocean into an interior bay, connected through an idealized inlet. The bathymetry represents a typical coastal configuration: the ocean region is deeper, while the interior bay is shallower, and an ebb shoal is present near the inlet, representing the sediment deposits common in those areas. When the flow enters the bay through the inlet, the strong shear and topographic interactions form eddies and complex flow structures. This test, based on the setups from Kubatko et al. (2006) and Samii et al. (2019), is particularly useful for observing interactions with coastal topography under tidal forcing.

6.8.1 Initial conditions

The initial conditions for this problem are:

$$\begin{cases} \eta = 0 \\ (U, V) = (0, 0) \end{cases}$$

The bathymetry (Figure 6.38) varies linearly, with $b = 19\,\mathrm{m}$ at the left, open ocean boundary, and $b = 5\,\mathrm{m}$ at the right, the entrance of the inlet. There is also an ebb shoal with a maximum height of 2 m at the center. The bathymetry is defined by:

$$b(x, y) = b_{\text{linear}}(x) + b_{\text{ebb}}(x, y)$$

with:

$$b_{\text{linear}}(x) = \begin{cases} 19, & x \leq 0 \\ 5, & x \geq 2250 \\ 19 - \dfrac{14}{2250}x, & \text{otherwise} \end{cases}$$

$$b_{\text{ebb}}(x, y) = (2 - b_{\text{linear}}(x_c)) \cdot \exp\left(-\frac{(x - x_c)^2 + (y - y_c)^2}{2\left(\frac{r}{2.5}\right)^2} \right)$$

with $(x_c, y_c) = (1875, 1500)$ m as the center of the shoal, and $r = 350$ m as the radius.

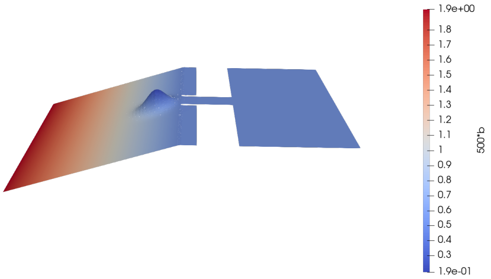


FIGURE 6.38: Bathymetry ($b(x, y)$) for the coastal inlet problem. The height is scaled by a factor of 500 to improve visibility.

The domain of this problem is a shape formed by an outside rectangular region, an interior rectangular region, and a narrow channel connecting them (see Figure 6.39). We apply a wall boundary condition in all the boundaries, except for the west boundary, where the inflow of water following a M2 tidal is imposed.

6.8.2 Results

The mesh is formed by triangular elements of $\Delta x = \Delta y = 50$ m (Figure 6.39). The polynomial order of the elements is $p = 1$. The time step is $\Delta t = 0.1$ s. The final time is $t = 6$ h 30 min.

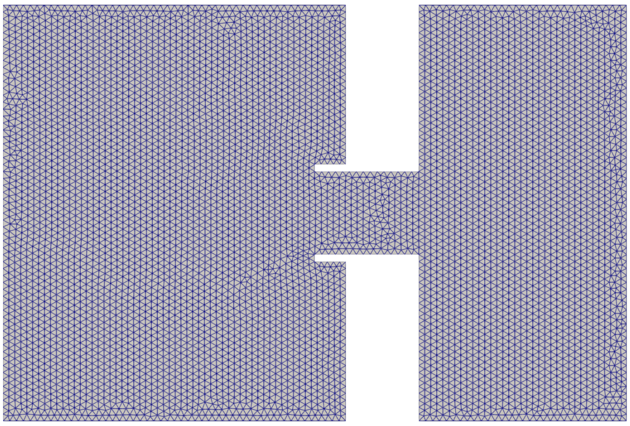
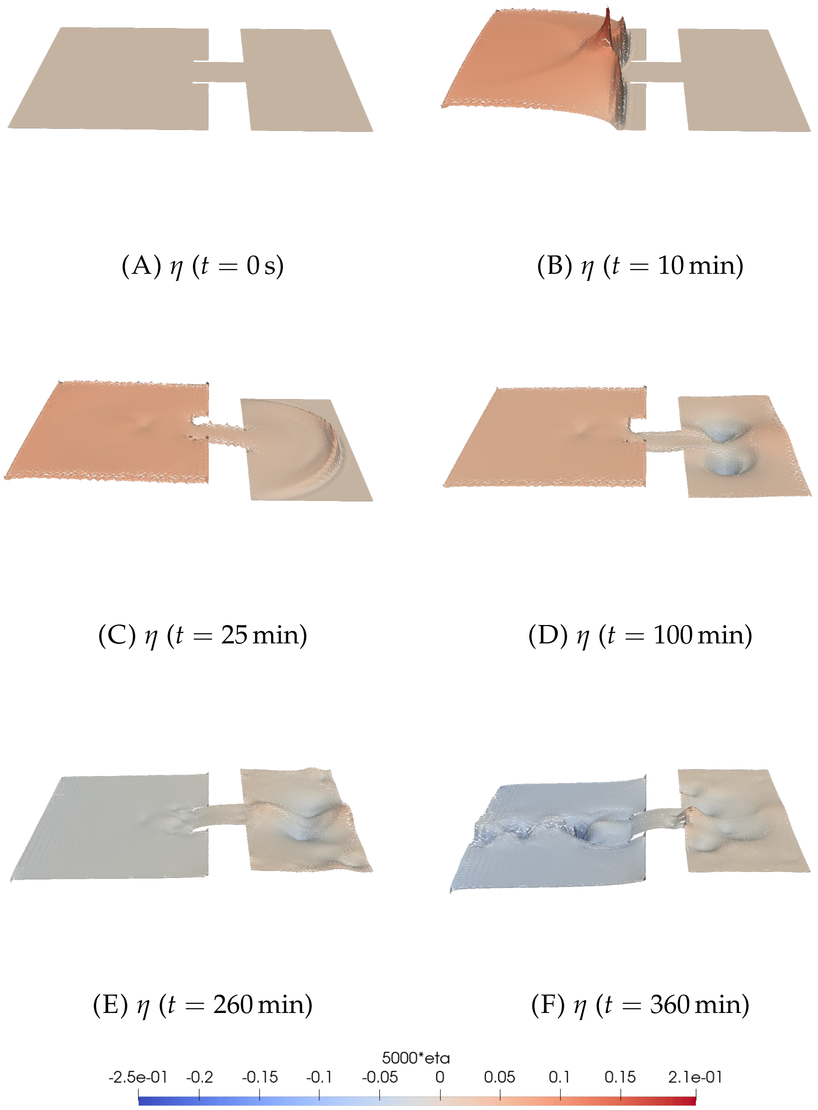


FIGURE 6.39: Mesh for the water height perturbation problem. The blue lines indicate the edges of the elements. The domain is $\Omega = [0, 3000] \times [0, 4500]$ m and the element size is $\Delta x = \Delta y = 50$ m.

The results for the free surface elevation and velocity are shown in Figures 6.40 and 6.41. The simulation shows that the water inflow into the bay generates strong shear layers at the inlet, leading to the formation of eddies due to local instabilities. These instabilities are intensified by the interaction with the ebb shoal, where the depth gradient induces additional vorticity generation. As the flow moves seaward, a von Kármán vortex street develops by $t = 360\,\text{min}$, which is a classical manifestation of flow separation behind an obstacle, occurring when inertial forces dominate over viscous forces (high Reynolds number regime). This behavior is consistent with observations of real coastal inlets, where tidal currents interacting with topography commonly produce eddy shedding and organized vortex patterns. The results suggest that the model is capable of capturing the key physical processes involved in coastal inlet dynamics.



(A) η ($t = 0\,\text{s}$)

(B) η ($t = 10\,\text{min}$)

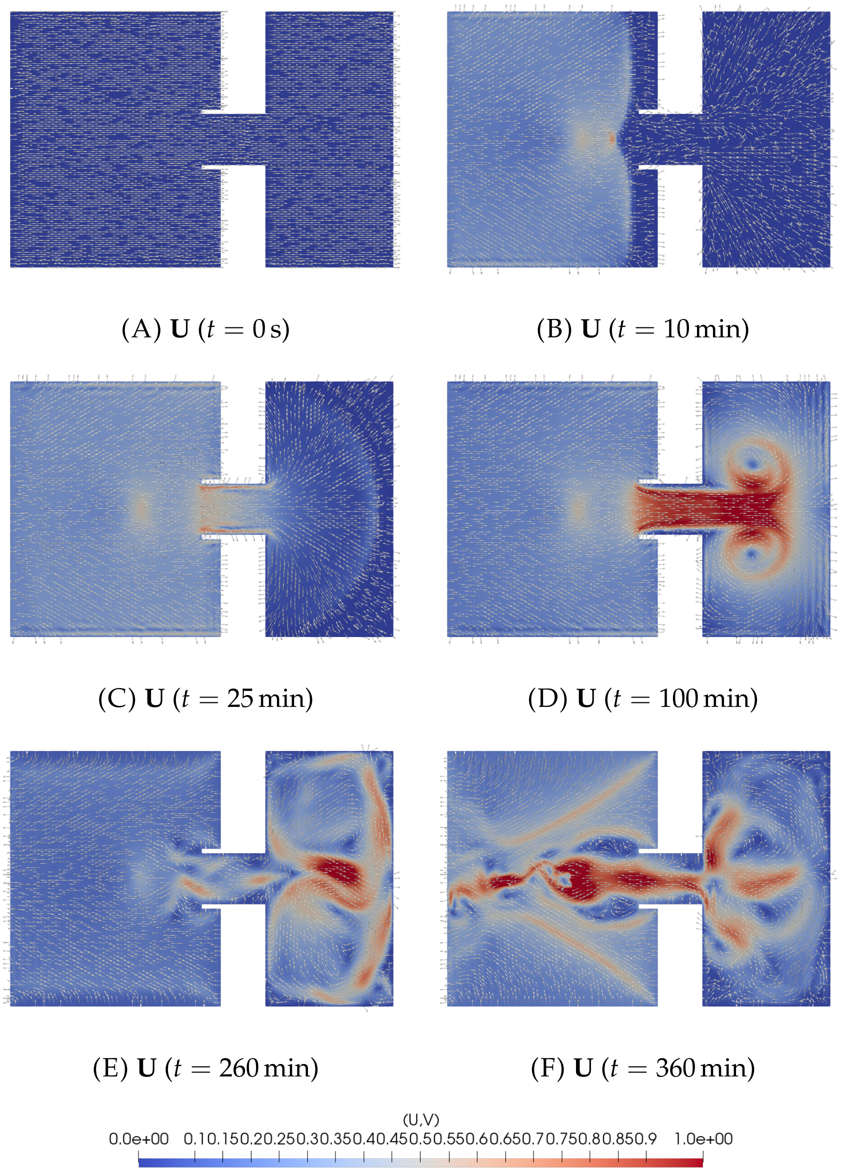
(C) η ($t = 25\,\text{min}$)

(D) η ($t = 100\,\text{min}$)

(E) η ($t = 260\,\text{min}$)

(F) η ($t = 360\,\text{min}$)

FIGURE 6.40: Results for the evolution of the free surface elevation (η) for the coastal inlet problem, at the times $t = 0\,\text{s}$, $t = 10\,\text{min}$, $t = 25\,\text{min}$, $t = 100\,\text{min}$, $t = 260\,\text{min}$ and $t = 360\,\text{min}$. The elevation (η) is scaled by a factor of 5000 to improve visibility.


(A) **U** ($t = 0\,\mathrm{s}$)

(B) **U** ($t = 10\,\mathrm{min}$)

(C) **U** ($t = 25\,\mathrm{min}$)

(D) **U** ($t = 100\,\mathrm{min}$)

(E) **U** ($t = 260\,\mathrm{min}$)

(F) **U** ($t = 360\,\mathrm{min}$)

FIGURE 6.41: Results for the evolution of the velocity (**U**) for the coastal inlet, at the times $t = 0\,\mathrm{s}$, $t = 10\,\mathrm{min}$, $t = 25\,\mathrm{min}$, $t = 100\,\mathrm{min}$, $t = 260\,\mathrm{min}$ and $t = 360\,\mathrm{min}$. The velocity intensity is represented in the background field and the direction of the velocity is represented with the normalized vectors.

Chapter 7

Conclusions

In this thesis we have observed the need for a strong model for physical oceanography. Oceanographic physical structures, represented by the SWE, require a robust method, with a spatial and a temporal discretization that can support the different scales happening simultaneously. We have studied the state of the art of numerical modelling in oceanography and FEM methods.

We have derived the linear and non linear SWE. We have developed and implemented a HDG FEM model from scratch, using Firedrake, to solve the SWE. We have calculated the numerical fluxes with the Lax-Friedrichs scheme, and imposed the boundary conditions weakly. We have assesed various IMEX and SSP temporal discretizations for the model. The model can be further extended and combined with additional features, such as coupling with biogeochemical models or incorporating other developments, to simulate a wider range of oceanic processes.

We have used this model to solve some theoretical numerical examples, from benchmark simple problems to more realistic complex problems, representing the structures found in the ocean, and observed its efficacy. The analysis of the error convergence has shown the effectiveness of the model. As we have shown in this thesis, IMEX SSP methods for the time discretization provide the flexibility in the time step and the effectiveness needed for these models.

Although being a goal of this thesis, we could not accomplish the formulation of the multilayer HDG SWE model due to time constraints, however, this thesis lays the foundation for future additions and improvements to the model.

We are currently working on the publication of a paper about the weak imposition of the boundary conditions, setting a common framework for the formulation, based on the works from Bui-Thanh (2016), Kang et al. (2020), and Samii et al. (2019).

The HDG model, along with the different boundary conditions and IMEX SSP time discretizations, has proved to effectively represent the SWE dynamics for simple numerical examples.

7.1 Main contributions

The main contributions of this thesis are:

1. We have analyzed the data from an oceanographic research cruise and identified the need for numerical methods, due to the limited synopticity and the insufficient spatial and temporal resolution of the sampling techniques, in the area of interest of this thesis.
2. We have presented a coherent theoretical development of HDG methods for the SWE that unifies previous work by Bui-Thanh (2016), Kang et al. (2020), and Samii et al. (2019).
3. We have formulated the weak imposition strategy of the boundary conditions in both IMEX and non-IMEX formulations, and are currently writing a scientific paper on this subject.
4. We have assesed the suitability of various SSP time discretization schemes, for the IMEX and non-IMEX discretizations.
5. We have implemented a one layer HDG model for the SWE, and validated it using standard benchmark problems. This model provides a foundation for further developments in this research area.

7.2 Academic contributions and international experiences

During the development of this thesis we have published a paper, corresponding to the material in Chapter 2 :

- Hernández-García I., Coca J., Ramos A., Rodríguez-Santana A., Machín F. 2024. Assessing hydrological sampling approaches in the Cape Verde frontal zone in November 2017. Sci. Mar. 88(4): e090. https://doi.org/10.3989/scimar.05509.090

We have made the following contributions to international conferences within this thesis:

- Exploring hydrographical structures in the Cape Verde Frontal Zone through in situ data and numerical modelling during Fluxes-II survey (oral communication). Hernández-García I., Machín F. and Coca J. VII International Symposium on Marine Sciences, Barcelona, Spain, 01/03 July 2020.
- Hybridisable Discontinuous Galerkin Skill for Ocean Modelling (oral communication). Hernández-García I., Oliver A. and Machín F. Expanding Ocean Frontiers 2022, Las Palmas de Gran Canaria, Spain, 06/08 July 2022.
- Hybridisable Discontinuous Galerkin applications for oceanographical Shallow Water Equations (oral communication). Hernández-García I., Oliver A. and Machín F. Congress on Numerical Methods in Engineering 2022, Las Palmas de Gran Canaria, Spain, 12/14 September 2022.
- Use of HDG oceanic models to study eddy formation in coastal upwelling areas (oral communication). Hernández-García I., Oliver A. and Machín F. EGU General Assembly 2023, Vienna, Austria, 23/28 April 2023.

As part of my thesis, I undertook an international research stay:

- Department of Data Science, Faculty of Science and Technology, Norwegian University of Life Sciences (NMBU), Ås, Norway. 12 September 2023 to 15 December 2023.

7.3 Future work

The work in this thesis opens several directions for extending this line of work and improving the HDG SWE model in the long term. Some of these future directions are:

1. **Optimizing the implementation and solving methods:** With the HDG SWE model already implemented, we can evaluate different combinations of solver methods for the Static Condensation scheme and analyze their computational efficiency. In particular, we can focus on the parallel implementation, for both the linear and non-linear SWE.

2. **Implementing more boundary conditions:** Starting from the current implementation, we can incorporate additional types of boundary conditions, such as slip wall and absorbing boundary conditions, that can be relevant for simulating various ocean processes.

3. **Extend the one layer model to a multilayer formulation:** A multilayer SWE model enables the simulation of ocean fronts, upwelling systems, complex stratified water columns, and other scenarios involving interactions between different water masses. This is key to simulate the physical processes in the area of interest.

4. **Expanding the model to reproduce more complex structures:** Incorporating additional features into the current HDG SWE model, it becomes possible to simulate more complex oceanic features. For example, implementing a coastal wet–dry algorithm allows for the representation of flooding events. Moreover, coupling the physical model with biological, chemical, or sediment transport models would expand its applications and represent complex physical–biogeochemical processes.

5. **Implementing *h-p* adaptivity techniques:** Implementing adaptive algorithms that combine mesh refinement (*h*-adaptivity) with polynomial order enrichment (*p*-adaptivity) to optimize computational resources. We will use highly refined elements with lower-order polynomials in high-gradient regions (eddies, fronts, and sharp transitions) and larger elements with higher-order polynomials elsewhere. We will identify an appropriate error indicator to detect which regions require refinement and analyze the optimal refinement strategies for each element: refine the mesh, increase polynomial order, or combine both approaches. We will also explore *h*-adaptivity for tracking the wet/dry interface that would improve the shoreline modelling and inundation simulations.

Appendix A

Coefficients of the Butcher Tableaux for the temporal discretizations

This appendix contains the coefficients for the Butcher Tableaux with the values $a_{i,j}$, b_i, $\tilde{a}_{i,j}$ and $\tilde{b}_i$ for the temporal discretizations in Chapter 5. Some IMEX, SSP RK, and IMEX SSP methods are shown. The tables are chosen from the references Betteridge et al. (2021), Higueras et al. (2014), Isherwood et al. (2018), Kang et al. (2020), and Ketcheson et al. (2009).

The IMEX tables use a notation $k(s, \sigma, p)$, in that s: steps of the implicit scheme, σ: steps of the explicit scheme, k: order of the explicit scheme, p: order of the IMEX scheme. Similarly, the explicit tables use the notation σ, k.

The properties of some of the IMEX SSP method are indicated as: L : the implicit method is L-stable, S : the stability region for the explicit part contains an interval on the imaginary axis, P : the amplification factor g for the implicit method is always positive, U : the IMEX RK method features uniform convergence, and M : the IMEX RK method has a nontrivial region of absolute monotonicity (Higueras et al., 2014).

A.1 θ discretization

IMEX method from Betteridge et al. (2021), that corresponds to the θ method.

$$\mathbf{a} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} 0 & 0 \\ 1-\theta & \theta \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \tilde{\mathbf{b}} = \begin{pmatrix} 1-\theta \\ \theta \end{pmatrix}$$

For the SSP RK discretization (5.11), the tables are used as $\mathbf{a} = \tilde{\mathbf{a}}$ and $\mathbf{b} = \tilde{\mathbf{b}}$.

A.2 ARS2(2,3,2) discretization

IMEX method from Betteridge et al. (2021) and Kang et al. (2020).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 \\ \gamma & 0 & 0 \\ \delta & 1-\delta & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 1-\gamma & \gamma \end{pmatrix} \quad \mathbf{b} = \tilde{\mathbf{b}} = \begin{pmatrix} 0 \\ 1-\gamma \\ \gamma \end{pmatrix}$$

with

$$\gamma = 1 - \frac{1}{\sqrt{2}} \qquad \delta = -\frac{2}{3}\sqrt{2}$$

A.3 ARS3(4,4,3) discretization

IMEX method from Betteridge et al. (2021) and Kang et al. (2020).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 & 0 \\ \frac{11}{18} & \frac{1}{18} & 0 & 0 & 0 \\ \frac{5}{6} & -\frac{5}{6} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{7}{4} & \frac{3}{4} & -\frac{7}{4} & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{6} & \frac{1}{2} & 0 & 0 \\ 0 & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{3}{2} & -\frac{3}{2} & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

$$\mathbf{b} = \begin{pmatrix} \frac{1}{4} \\ \frac{7}{4} \\ \frac{3}{4} \\ -\frac{7}{4} \\ 0 \end{pmatrix} \quad \tilde{\mathbf{b}} = \begin{pmatrix} 0 \\ \frac{3}{2} \\ -\frac{3}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}$$

A.4 Shu-Osher discretization

SSP RK method from Isherwood et al. (2018).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0.25 & 0.25 & 0 \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} \frac{1}{6} \\ \frac{1}{6} \\ \frac{2}{3} \end{pmatrix}$$

A.5 SSPIRK33 discretization

SSP RK method from Isherwood et al. (2018) and Ketcheson et al. (2009).

$$\mathbf{a} = \begin{pmatrix} \frac{1}{2} - \frac{\sqrt{2}}{4} & 0 & 0 \\ \frac{\sqrt{2}}{4} & \frac{1}{2} - \frac{\sqrt{2}}{4} & 0 \\ \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{4} & \frac{1}{2} - \frac{\sqrt{2}}{4} \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix}$$

A.6 SSPIRK43 discretization

SSP RK method from Isherwood et al. (2018) and Ketcheson et al. (2009).

$$\mathbf{a} = \begin{pmatrix} \frac{1}{2} - \frac{\sqrt{15}}{10} & 0 & 0 & 0 \\ \frac{\sqrt{15}}{15} & \frac{1}{2} - \frac{\sqrt{15}}{10} & 0 & 0 \\ \frac{\sqrt{15}}{15} & \frac{\sqrt{15}}{15} & \frac{1}{2} - \frac{\sqrt{15}}{10} & 0 \\ \frac{\sqrt{15}}{15} & \frac{\sqrt{15}}{15} & \frac{\sqrt{15}}{15} & \frac{1}{2} - \frac{\sqrt{15}}{10} \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.25 \end{pmatrix}$$

A.7 SSP2(3,3,2)-LSPUM discretization

IMEX SSP method from Higueras et al. (2014).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 \\ \frac{5}{6} & 0 & 0 \\ \frac{11}{24} & \frac{11}{24} & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} \frac{2}{11} & 0 & 0 \\ \frac{205}{462} & \frac{2}{11} & 0 \\ \frac{2033}{4620} & \frac{21}{110} & \frac{2}{11} \end{pmatrix} \quad \mathbf{b} = \tilde{\mathbf{b}} = \begin{pmatrix} \frac{24}{55} \\ \frac{1}{5} \\ \frac{4}{11} \end{pmatrix}$$

A.8 SSP2(3,3,2)-LPM-1 discretization

IMEX SSP method from Higueras et al. (2014).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} \frac{2}{11} & 0 & 0 \\ \frac{2829}{9317} & \frac{2}{11} & 0 \\ \frac{148529}{428582} & \frac{7}{23} & \frac{2}{11} \end{pmatrix} \quad \mathbf{b} = \tilde{\mathbf{b}} = \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix}$$

A.9 SSP2(3,3,2)-LPM-2 discretization

IMEX SSP method from Higueras et al. (2014).

$$\mathbf{a} = \begin{pmatrix} 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix} \quad \tilde{\mathbf{a}} = \begin{pmatrix} \frac{2}{11} & 0 & 0 \\ \frac{2583}{13310} & \frac{2}{11} & 0 \\ \frac{39731}{139755} & \frac{10}{21} & \frac{2}{11} \end{pmatrix} \quad \mathbf{b} = \tilde{\mathbf{b}} = \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix}$$

Appendix B

Implementation

This appendix contains the full expanded equations for the bilinear forms of the linear and non-linear SWE, the implementation of these terms, and the solver parameters used for the model resolution.

B.1 Bilinear forms for the HDG SWE

The equation system for the HDG SWE system, written in the most expanded form, and separated by the bilinear forms (4.21), (4.22) and (4.23), is developed in this section.

$$\begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{L}(\mathbf{q},\mathbf{v}) \\ \Xi(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = 0 \end{cases} \qquad \begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{L}_{\mathrm{L}}(\mathbf{q},\mathbf{v}) \\ \Xi_{\mathrm{L}}(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = 0 \end{cases} \qquad \begin{cases} \frac{\partial \mathbf{M}(\mathbf{q},\mathbf{v})}{\partial t} = \mathbf{N}(\mathbf{q},\mathbf{v}) + \mathbf{L}_{\mathrm{L}}(\mathbf{q},\mathbf{v}) \\ \Xi_{\mathrm{L}}(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = 0 \end{cases}$$

B.1.1 Complete bilinear forms for the non-linear SWE

$$\mathbf{M}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{q}\cdot\mathbf{v}\,dx = \begin{pmatrix} \phi\cdot\psi \\ \mathbf{u}\cdot\boldsymbol{\omega} \end{pmatrix}_K$$

$$
\begin{aligned}
\mathbf{L}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{F}(\mathbf{q}) \cdot \nabla \cdot \mathbf{v}\, dx + \int_K \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx - \int_{\partial K} (\hat{\mathbf{F}}(\mathbf{q}) \cdot \mathbf{n}) \cdot \mathbf{v}\, dS = \\
\begin{pmatrix} \mathbf{u} \cdot \nabla \psi \\ \left(\frac{\mathbf{u} \otimes \mathbf{u}}{\phi + \phi_B} + \left(\phi_B \phi + \frac{1}{2} \phi^2 \right) I \right) \cdot \nabla \cdot \boldsymbol{\omega} \end{pmatrix}_K + \begin{pmatrix} 0 \\ (\phi \nabla \phi_B + f \cdot \mathbf{u}^{\perp}) \cdot \boldsymbol{\omega} \end{pmatrix}_K \\
- \left(\left(\begin{pmatrix} \hat{\mathbf{u}} \\ \frac{\hat{\mathbf{u}} \otimes \hat{\mathbf{u}}}{\hat{\phi} + \phi_B} + \left(\phi_B \hat{\phi} + \frac{1}{2} \hat{\phi}^2 \right) I \end{pmatrix} + \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \begin{pmatrix} (\phi - \hat{\phi})\, \mathbf{n} \\ (\mathbf{u} - \hat{\mathbf{u}})\, \mathbf{n} \end{pmatrix} \right) \cdot \begin{pmatrix} \mathbf{n} \cdot \psi \\ \mathbf{n} \cdot \boldsymbol{\omega} \end{pmatrix} \right)_{dK}
\end{aligned}
$$

$$
\begin{aligned}
\Xi(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = \int_{\varepsilon^{\text{int}}} \{\{ \hat{\mathbf{F}}(\mathbf{q},\hat{\mathbf{q}}) \cdot \mathbf{n} \}\} \cdot \hat{\mathbf{v}}\, dS + \int_{\varepsilon^{\text{bound}}} \mathbf{B}(\mathbf{q},\hat{\mathbf{q}},\mathbf{q}_\infty) \cdot \hat{\mathbf{v}}\, dS = \\
\left(\left(\begin{pmatrix} \hat{\mathbf{u}} \\ \frac{\hat{\mathbf{u}} \otimes \hat{\mathbf{u}}}{\hat{\phi} + \phi_B} + \left(\phi_B \hat{\phi} + \frac{1}{2} \hat{\phi}^2 \right) I \end{pmatrix} + \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \begin{pmatrix} (\phi - \hat{\phi})\, \mathbf{n} \\ (\mathbf{u} - \hat{\mathbf{u}})\, \mathbf{n} \end{pmatrix} \right) \cdot \begin{pmatrix} \mathbf{n} \cdot \hat{\psi} \\ \mathbf{n} \cdot \hat{\boldsymbol{\omega}} \end{pmatrix} \right)_{\varepsilon^{\text{int}}} \\
+ \left(\left(\frac{1}{2} \begin{pmatrix} \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \phi + \mathbf{u} \cdot \mathbf{n} \\ -\phi \frac{\hat{\mathbf{u}} \otimes \hat{\mathbf{u}}}{(\hat{\phi} + \phi_B)^2} \cdot \mathbf{n} + (\hat{\phi} + \phi_B) \phi \mathbf{n} + \frac{\hat{\mathbf{u}} \cdot \mathbf{u}^{\text{flip}}}{\hat{\phi} + \phi_B} \mathbf{n}^{\text{flip}} + \frac{2 \hat{\mathbf{u}} \mathbf{u}}{\hat{\phi} + \phi_B} \mathbf{n} + \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \mathbf{u} \end{pmatrix} \right. \right. \\
- \begin{pmatrix} \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \hat{\phi} \\ \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \hat{\mathbf{u}} \end{pmatrix} \\
\left. - \frac{1}{2} \begin{pmatrix} - \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \phi_\infty + \mathbf{u}_\infty \cdot \mathbf{n} \\ -\phi_\infty \frac{\hat{\mathbf{u}} \otimes \hat{\mathbf{u}}}{(\hat{\phi} + \phi_B)^2} \cdot \mathbf{n} + (\hat{\phi} + \phi_B) \phi_\infty \mathbf{n} + \frac{\hat{\mathbf{u}} \cdot \mathbf{u}_\infty^{\text{flip}}}{\hat{\phi} + \phi_B} \mathbf{n}^{\text{flip}} + \frac{2 \hat{\mathbf{u}} \mathbf{u}_\infty}{\hat{\phi} + \phi_B} \mathbf{n} + \left(|\hat{\mathbf{u}} \cdot \mathbf{n}| + \sqrt{\hat{\phi} + \phi_B} \right) \mathbf{u}_\infty \end{pmatrix} \right) \\
\left. \cdot \begin{pmatrix} \hat{\psi} \\ \hat{\boldsymbol{\omega}} \end{pmatrix} \right)_{\varepsilon^{\text{bound}}}
\end{aligned}
$$

B.1.2 Complete bilinear forms for the linear SWE

$$
\begin{aligned}
\mathbf{L}_{\text{L}}(\mathbf{q},\mathbf{v}) = \int_K \mathbf{F}_{\text{L}}(\mathbf{q}) \cdot \nabla \cdot \mathbf{v}\, dx + \int_K \mathbf{s}(\mathbf{q}) \cdot \mathbf{v}\, dx - \int_{\partial K} (\hat{\mathbf{F}}_{\text{L}}(\mathbf{q}) \cdot \mathbf{n}) \cdot \mathbf{v}\, dS = \\
\begin{pmatrix} \mathbf{u} \cdot \nabla \psi \\ \phi_B \phi \cdot \nabla \cdot \boldsymbol{\omega} \end{pmatrix}_K + \begin{pmatrix} 0 \\ (\phi \nabla \phi_B + f \cdot \mathbf{u}^{\perp}) \cdot \boldsymbol{\omega} \end{pmatrix}_K \\
- \left(\left(\begin{pmatrix} \hat{\mathbf{u}} \\ \phi_B \hat{\phi} \end{pmatrix} + \sqrt{\phi_B} \begin{pmatrix} (\phi - \hat{\phi})\, \mathbf{n} \\ (\mathbf{u} - \hat{\mathbf{u}})\, \mathbf{n} \end{pmatrix} \right) \cdot \begin{pmatrix} \mathbf{n} \cdot \psi \\ \mathbf{n} \cdot \boldsymbol{\omega} \end{pmatrix} \right)_{dK}
\end{aligned}
$$

$$
\begin{aligned}
\Xi_{\mathrm{L}}(\mathbf{q},\hat{\mathbf{q}},\hat{\mathbf{v}}) = \int_{\varepsilon^{\mathrm{int}}} \{\{\hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q},\hat{\mathbf{q}})\cdot\mathbf{n}\}\}\cdot\hat{\mathbf{v}}\,dS + \int_{\varepsilon^{\mathrm{bound}}} \mathbf{B}(\mathbf{q},\hat{\mathbf{q}},\mathbf{q}_\infty)\cdot\hat{\mathbf{v}}\,dS = \\
\left(\left(\begin{pmatrix}\hat{\mathbf{u}}\\ \phi_B\hat{\phi}\end{pmatrix} + \sqrt{\phi_B}\begin{pmatrix}(\phi-\hat{\phi})\,\mathbf{n}\\ (\mathbf{u}-\hat{\mathbf{u}})\,\mathbf{n}\end{pmatrix}\right)\cdot\begin{pmatrix}\mathbf{n}\cdot\hat{\psi}\\ \mathbf{n}\cdot\hat{\omega}\end{pmatrix}\right)_{\varepsilon^{\mathrm{int}}} \\
+\left(\left(\frac{1}{2}\begin{pmatrix}\sqrt{\phi_B}\phi+\mathbf{u}\cdot\mathbf{n}\\ \phi_B\phi\mathbf{n}+\sqrt{\phi_B}\mathbf{u}\end{pmatrix} - \begin{pmatrix}\sqrt{\phi_B}\hat{\phi}\\ \sqrt{\phi_B}\hat{\mathbf{u}}\end{pmatrix} - \frac{1}{2}\begin{pmatrix}-\sqrt{\phi_B}\phi_\infty+\mathbf{u}_\infty\cdot\mathbf{n}\\ \phi_B\phi_\infty\mathbf{n}-\sqrt{\phi_B}\mathbf{u}_\infty\end{pmatrix}\right)\cdot\begin{pmatrix}\hat{\psi}\\ \hat{\omega}\end{pmatrix}\right)_{\varepsilon^{\mathrm{bound}}}
\end{aligned}
$$

B.1.3 Complete bilinear forms for the IMEX SWE

$$
\begin{aligned}
\mathbf{N}(\mathbf{q},\mathbf{v}) = \int_K (\mathbf{F}(\mathbf{q})-\mathbf{F}_{\mathrm{L}}(\mathbf{q}))\cdot\nabla\cdot\mathbf{v}\,dx - \int_{\partial K}((\hat{\mathbf{F}}(\mathbf{q})-\hat{\mathbf{F}}_{\mathrm{L}}(\mathbf{q}))\cdot\mathbf{n})\cdot\mathbf{v}\,dS = \\
\begin{pmatrix}\mathbf{u}\cdot\nabla\psi\\ \left(\frac{\mathbf{u}\otimes\mathbf{u}}{\phi+\phi_B}+\left(\phi_B\phi+\frac{1}{2}\phi^2\right)I\right)\cdot\nabla\cdot\omega\end{pmatrix}_K - \begin{pmatrix}\mathbf{u}\cdot\nabla\psi\\ \phi_B\phi\cdot\nabla\cdot\omega\end{pmatrix}_K \\
-\left(\left(\begin{pmatrix}\hat{\mathbf{u}}\\ \frac{\hat{\mathbf{u}}\otimes\hat{\mathbf{u}}}{\hat{\phi}+\phi_B}+\left(\phi_B\hat{\phi}+\frac{1}{2}\hat{\phi}^2\right)I\end{pmatrix} + \left(|\hat{\mathbf{u}}\cdot\mathbf{n}|+\sqrt{\hat{\phi}+\phi_B}\right)\begin{pmatrix}(\phi-\hat{\phi})\,\mathbf{n}\\ (\mathbf{u}-\hat{\mathbf{u}})\,\mathbf{n}\end{pmatrix}\right)\cdot\begin{pmatrix}\mathbf{n}\cdot\psi\\ \mathbf{n}\cdot\omega\end{pmatrix}\right)_{dK} \\
+\left(\left(\begin{pmatrix}\hat{\mathbf{u}}\\ \phi_B\hat{\phi}\end{pmatrix} + \sqrt{\phi_B}\begin{pmatrix}(\phi-\hat{\phi})\,\mathbf{n}\\ (\mathbf{u}-\hat{\mathbf{u}})\,\mathbf{n}\end{pmatrix}\right)\cdot\begin{pmatrix}\mathbf{n}\cdot\psi\\ \mathbf{n}\cdot\omega\end{pmatrix}\right)_{dK}
\end{aligned}
$$

B.2 Code for the bilinear forms for the full non-linear SWE

The forms from the last section are calculated as shown in the code in this section.

```python
# Bilinear form M
    M = fd.inner(q, v) * fd.dx
```

LISTING B.1: Python script for the bilinear form M

```python
# Source term (Bathymetry and Coriolis)
    bath = phi * fd.grad(phi_b)
    coriolis = f_coriolis * fd.perp(u)
    s = fd.as_vector(
        [
            0,
            bath[0] + coriolis[0],
            bath[1] + coriolis[1],
        ]
    )


```

```
# Nonlinear flux tensor F
  F = fd.as_matrix(
    [
      [u[0], u[1]],
      [
        u[0] * u[0] / (phi + phi_b) + phi * phi / 2 + phi_b * phi,
        u[0] * u[1] / (phi + phi_b),
      ],
      [
        u[0] * u[1] / (phi + phi_b),
        u[1] * u[1] / (phi + phi_b) + phi * phi / 2 + phi_b * phi,
      ],
    ]
  )

  F_in_hat = fd.replace(
    F,
    {
      phi: phi_hyb,
      u[0]: u_hyb[0],
      u[1]: u_hyb[1],
    },
  )

# Stability parameter tau_hat = |u_hat * n| + sqrt(phi_hat + phi_b) (Lax-Friedrichs type dissipation)
  tau_hat = abs(fd.dot(u_hyb, n)) + fd.sqrt(phi_hyb + phi_b)

# HDG Numerical flux: F_hat = F(u_hat) + tau*(q-q_hat)*n
  F_hat = F_in_hat + tau_hat * fd.outer(q - q_hyb, n)

# Bilinear form L
  L = (
    fd.inner(F, fd.grad(v)) * fd.dx
    + fd.inner(s, v) * fd.dx
    - fd.inner(fd.dot(F_hat, n), v)("+") * fd.dS
    - fd.inner(fd.dot(F_hat, n), v)("-") * fd.dS
    - fd.inner(fd.dot(F_hat, n), v) * fd.ds
  )
```

LISTING B.2: Python script for the bilinear form L

```
# Flux Jacobian matrices for characteristic boundary treatment
# A1 (flux Jacobian in x-direction)
  A1 = fd.as_matrix(
    [
      [0, 1, 0],
      [
        -u_hyb[0] * u_hyb[0] / ((phi_hyb + phi_b) * (phi_hyb + phi_b))
        + phi_hyb
        + phi_b,
        2 * u_hyb[0] / (phi_hyb + phi_b),
        0,
      ],
      [
                -u_hyb[0] * u_hyb[1] / ((phi_hyb + phi_b) * (phi_hyb + phi_b)),
                u_hyb[1] / (phi_hyb + phi_b),
                u_hyb[0] / (phi_hyb + phi_b),
            ],
        ]
    )
# A2 (flux Jacobian in y-direction)
    A2 = fd.as_matrix(
        [
            [0, 0, 1],
            [
                -u_hyb[1] * u_hyb[0] / ((phi_hyb + phi_b) * (phi_hyb + phi_b)),
                u_hyb[1] / (phi_hyb + phi_b),
                u_hyb[0] / (phi_hyb + phi_b),
            ],
            [
                -u_hyb[1] * u_hyb[1] / ((phi_hyb + phi_b) * (phi_hyb + phi_b))
                + phi_hyb
                + phi_b,
                0,
                2 * u_hyb[1] / (phi_hyb + phi_b),
            ],
        ]
    )

# Boundary normal flux Jacobian
    A = A1 * n[0] + A2 * n[1]

# Decomposition for Riemann solver at the boundary
# |A| approximated with Lax-Friedrichs approach using local wave speed
    abs_A = fd.as_matrix(
        [
            [tau_hat, 0, 0],
            [0, tau_hat, 0],
            [0, 0, tau_hat],
        ],
    )
# Split into incoming and outgoing characteristics (Riemann solver approach)
    A_plus = 0.5 * (A + abs_A)  # Outgoing characteristics
    A_minus = 0.5 * (A - abs_A)  # Incoming characteristics

# Boundary numerical flux using characteristic decomposition (Bh = boundary hybrid flux)
    Bh = A_plus * q - abs_A * q_hyb - A_minus * q_bc

# Bilinear form Flx
    Flx = (
        fd.inner(fd.dot(F_hat, n), mu)("+") * fd.dS
        + fd.inner(fd.dot(F_hat, n), mu)("-") * fd.dS
        + fd.inner(Bh, mu) * fd.ds
    )
```

LISTING B.3: Python script for the bilinear form Ξ

```
# Source term (Bathymetry and Coriolis)
    bath = phi_ini * fd.grad(phi_b)

    coriolis = f_coriolis * fd.perp(u_ini)

    s = fd.as_vector(
        [
            0,
            bath[0] + coriolis[0],
            bath[1] + coriolis[1],
        ]
    )

# Linear flux tensor F
    F_lineal = fd.as_matrix(
        [
            [u_ini[0], u_ini[1]],
            [phi_b * phi_ini, 0],
            [0, phi_b * phi_ini],
        ]
    )

    F_in_hat_lineal= fd.replace(
        F_lineal,
        {
            phi_ini: phi_hyb,
            u_ini[0]: u_hyb[0],
            u_ini[1]: u_hyb[1],
        },
    )

# Stability parameters
    tau_star_l = sqrt_phi_b
    tau_hat = tau_star_l

# HDG Numerical fluxes between elements
    F_hat_lineal = fd.as_vector(
        [
            F_in_hat_lineal[0] + tau_hat * (phi - phi_hyb) * n,
            F_in_hat_lineal[1] + tau_hat * (u[0] - u_hyb[0]) * n
   ,
            F_in_hat_lineal[2] + tau_hat * (u[1] - u_hyb[1]) * n
   ,
        ]
    )

# Bilinear form L
 L = (
        fd.inner(F_lineal, fd.grad(v)) * fd.dx
        + fd.inner(s, v) * fd.dx
        - fd.inner(fd.dot(F_hat_lineal, n("+")), v("+")) * fd.dS
        - fd.inner(fd.dot(F_hat_lineal, n("-")), v("-")) * fd.dS
        - fd.inner(fd.dot(F_hat_lineal_ds, n), v) * fd.ds
    )
```

LISTING B.4: Python script for the bilinear form L_L

```
# Boundary normal flux Jacobian
    A = fd.as_matrix(
        [
            [0, n[0], n[1]],
            [phi_b * n[0], 0, 0],
            [phi_b * n[1], 0, 0],
        ]
    )

# Decomposition for Riemann solver at the boundary
# |A| approximated with Lax-Friedrichs approach using local wave speed
    det_A = fd.as_matrix(
        [
            [tau_hat, 0, 0],
            [0, tau_hat, 0],
            [0, 0, tau_hat],
        ],
    )

# Split into incoming and outgoing characteristics (Riemann solver approach)
    A_plus = 0.5 * (A + det_A)
    A_minus = 0.5 * (A - det_A)

# Boundary numerical flux using characteristic decomposition (Bh = boundary hybrid flux)
    Bh_lineal = A_plus * q - det_A * q_hyb - A_minus * q_bc

# Bilinear form Flx
    Flx = (
        fd.inner(fd.dot(F_hat_lineal, n), mu)("+") * fd.dS
        + fd.inner(fd.dot(F_hat_lineal, n), mu)("-") * fd.dS
        + fd.inner(Bh_lineal, mu) * fd.ds
    )
```

LISTING B.5: Python script for the bilinear form Ξ_L

```
# Nonlinear flux
    F = fd.as_matrix(
        [
            [u_ini[0], u_ini[1]],
            [
                u_ini[0] * u_ini[0] / (phi_ini + phi_b)
                + phi_ini * phi_ini / 2
                + phi_b * phi_ini,
                u_ini[0] * u_ini[1] / (phi_ini + phi_b),
            ],
            [
                u_ini[0] * u_ini[1] / (phi_ini + phi_b),
                u_ini[1] * u_ini[1] / (phi_ini + phi_b)
                + phi_ini * phi_ini / 2
                + phi_b * phi_ini,
            ],
        ]
    )


```

```
# Linear flux
    F_lineal = fd.as_matrix(
        [
            [u_ini[0], u_ini[1]],
            [phi_b * phi_ini, 0],
            [0, phi_b * phi_ini],
        ]
    )

    F_nl = F - F_lineal

# Stability parameters
    tau_star_dS = fd.max_value(
        abs(fd.inner(u_ini("+"), n("+"))) + fd.sqrt(phi_ini("+") + phi_b("+")),
        abs(fd.inner(u_ini("-"), n("-"))) + fd.sqrt(phi_ini("-") + phi_b("-")),
    )
    tau_star_ds = fd.max_value(
        abs(fd.inner(u_ini, n)) + fd.sqrt(phi_ini + phi_b),
        abs(fd.inner(u_bc, -n)) + fd.sqrt(phi_bc + phi_b),
    )
    tau_star_l = sqrt_phi_b
    tau_hat = tau_star_l

# DG Numerical fluxes between elements
    F_star_dS = fd.as_vector(
        [
            fd.avg(F[0]) + tau_star_dS / 2 * fd.jump(phi_ini, n),
            fd.avg(F[1]) + tau_star_dS / 2 * fd.jump(u_ini[0], n),
            fd.avg(F[2]) + tau_star_dS / 2 * fd.jump(u_ini[1], n),
        ]
    )

    F_star_lineal_dS = fd.replace(
        F_star_dS,
        {
            F[0]: F_lineal[0],
            F[1]: F_lineal[1],
            F[2]: F_lineal[2],
            tau_star_dS: tau_star_l,
        },
    )

    F_star_NL_dS = F_star_dS - F_star_lineal_dS

# DG Numerical fluxes on the boundary
    F_star_ds = fd.as_vector(
        [
            F[0] + tau_star_ds / 2 * (phi_ini - phi_bc) * n,
            F[1] + tau_star_ds / 2 * (u_ini[0] - u_bc[0]) * n,
            F[2] + tau_star_ds / 2 * (u_ini[1] - u_bc[1]) * n,
        ]
    )

    F_star_lineal_ds = fd.replace(
        F_star_ds,
        {
            F[0]: F_lineal[0],
            F[1]: F_lineal[1],
            F[2]: F_lineal[2],
            tau_star_ds: tau_star_l,
        },
    )

    F_star_NL_ds = F_star_ds - F_star_lineal_ds

 # Bilinear form N
    NL_rhs = (
        fd.inner(F_nl, fd.grad(v)) * fd.dx
        - fd.inner(fd.dot(F_star_NL_dS, n("+")), v("+")) * fd.dS
        - fd.inner(fd.dot(F_star_NL_dS, n("-")), v("-")) * fd.dS
        - fd.inner(fd.dot(F_star_NL_ds, n), v) * fd.ds
    )
```

LISTING B.6: Python script for the bilinear form N

B.3 Solver parameters for the equation system

The specific parameters for the system resolution are:

```
HDG_AMG_params = {
    "mat_type": "matfree",
    "ksp_type": "preonly",
    "pc_type": "python",
    "pc_python_type": "firedrake.SCPC",
    "pc_sc_eliminate_fields": "0",
    "condensed_field": inner_params_HDG_AMG,
}

# AMG inner parameters
inner_params_HDG_AMG = {
    "mat_type": "aij",
    "ksp_type": "gmres",
    "ksp_rtol": 1e-10,
    "pc_type": "gamg",
    "pc_mg_log": None,
    "mg_levels": {
        "ksp_type": "chebyshev",
        "ksp_max_it": 2,
        "pc_type": "bjacobi",
        "sub_pc_type": "sor",
    },
}
```

LISTING B.7: Python code with the PETSc solver parameters

Appendix C

Error calculation

To quantify the error of the model results in the test cases with analytical solutions, the numerical results ($\mathbf{q}$) are compared against the exact solution ($\mathbf{q}_{\text{exact}}$), using the L_2 norm. The errors for η and $\mathbf{U}$ are calculated as follows:

$$\text{Error}_{L2\,\mathbf{q}} = \left(\sum_K \int_K \left(\mathbf{q}_{\text{exact}} - \mathbf{q} \right)^2 \, dK \right)^{\frac{1}{2}}$$

The energy E is defined as:

$$E = \frac{\left(||\eta||^2_{\Omega_h} + ||\mathbf{U}||^2_{\Omega_h} \right)}{2}$$

and the error of $\sqrt{E}$ can be calculated as:

$$\text{Error}_{L2\,\sqrt{E}} = \sqrt{\frac{\left(\text{Error}^2_{L2\,\eta} + \text{Error}^2_{L2\,\mathbf{U}} \right)}{2}}$$

Bibliography

Aizinger, V., & Dawson, C. (2002). A discontinuous Galerkin method for two-dimensional flow and transport in shallow water. *Advances in Water Resources*, *25*(1), 67–84. https://doi.org/10.1016/S0309-1708(01)00019-7

Allen, J., Cornell, J., Moore, M., Crisp, N., & Dunning, J. (2002). Operational oceanography using the 'new'SeaSoar ocean undulator. *Sea Technology*, *43*(4), 35–40. http://eprints.soton.ac.uk/id/eprint/6059

Alnæs, M. S., Logg, A., Ølgaard, K. B., Rognes, M. E., & Wells, G. N. (2014). Unified form language: A domain-specific language for weak formulations of partial differential equations. *ACM Trans. Math. Softw.*, *40*(2). https://doi.org/10.1145/2566630

Arabshahi, H. (2016). *Space-time hybridized discontinuous Galerkin methods for shallow water equations* [Doctoral dissertation, The University of Texas at Austin]. http://hdl.handle.net/2152/47014

Ascher, U. M., Ruuth, S. J., & Spiteri, R. J. (1997). Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations [Special Issue on Time Integration]. *Applied Numerical Mathematics*, *25*(2), 151–167. https://doi.org/10.1016/S0168-9274(97)00056-1

Bachmayer, R., Leonard, N. E., Graver, J., Fiorelli, E., Bhatta, P., & Paley, D. (2004). Underwater gliders: recent developments and future applications. *Proceedings of the 2004 International Symposium on Underwater Technology (IEEE Cat. No.04EX869)*, 195–200. https://doi.org/10.1109/ut.2004.1405540

Bachmayer, R., Young, B. D., Williams, C., Bishop, C., Knapp, C., & Foley, J. (2006). Development and deployment of ocean gliders on the Newfoundland Shelf. *Proceedings of the Unmanned Vehicle Systems Canada Conference 2006*. https://www.researchgate.net/publication/44070140_Development_and_deployment_of_ocean_gliders_on_the_Newfoundland_Shelf

Balay, S., Abhyankar, S., Adams, M. F., Benson, S., Brown, J., Brune, P., Buschelman, K., Constantinescu, E., Dalcin, L., Dener, A., Eijkhout, V., Faibussowitsch, J., Gropp, W. D., Hapla, V., Isaac, T., Jolivet, P., Karpeev, D., Kaushik, D., Knepley, M. G., … Zhang, J. (2024). *PETSc/TAO Users*

Manual (tech. rep. No. ANL-21/39 - Revision 3.22). Argonne National Laboratory. https://doi.org/10.2172/2205494

Baliunas, S., Frick, P., Sokoloff, D., & Soon, W. (1997). Time scales and trends in the central England temperature data (1659-1990): A wavelet analysis. *Geophysical Research Letters*, *24*(11), 1351–1354. https://doi.org/10.1029/97gl01184

Barton, E. D. (1987). Meanders, eddies and intrusions in the thermohaline front off Northwest Africa. *Oceanologica acta*, *10*(3), 267–283. https://archimer.ifremer.fr/doc/00108/21903/19494.pdf

Bashmachnikov, I., Nascimento, Â., Neves, F., Menezes, T., & Koldunov, N. V. (2015). Distribution of intermediate water masses in the subtropical northeast Atlantic. *Ocean Science*, *11*(5), 803–827. https://doi.org/10.5194/os-11-803-2015

Betteridge, J. (2020). *Efficient elliptic solvers for higher order DG discretisations on modern architectures and applications in atmospheric modelling* [Doctoral dissertation, University of Bath]. https://researchportal.bath.ac.uk/en/studentTheses/efficient-elliptic-solvers-for-higher-order-dg-discretisations-on

Betteridge, J., Gibson, T. H., Graham, I. G., & Müller, E. H. (2021). Multigrid preconditioners for the hybridised discontinuous galerkin discretisation of the shallow water equations. *Journal of Computational Physics*, *426*, 109948. https://doi.org/10.1016/j.jcp.2020.109948

Brenner, S. C., & Scott, L. R. (2008). *The Mathematical Theory of Finite Element Methods*. Springer New York. https://doi.org/10.1007/978-0-387-75934-0

Brooks, A. N., & Hughes, T. J. (1982). Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, *32*(1–3), 199–259. https://doi.org/10.1016/0045-7825(82)90071-8

Bui-Thanh, T. (2015). From Godunov to a unified hybridized discontinuous Galerkin framework for partial differential equations. *Journal of Computational Physics*, *295*, 114–146. https://doi.org/10.1016/j.jcp.2015.04.009

Bui-Thanh, T. (2016). Construction and Analysis of HDG Methods for Linearized Shallow Water Equations. *SIAM Journal on Scientific Computing*, *38*(6), A3696–A3719. https://doi.org/10.1137/16m1057243

Burgoa, N., Machín, F., Rodríguez-Santana, A., Marrero-Díaz, A., Álvarez-Salgado, X. A., Fernández-Castro, B., Gelado-Caballero, D., & Arístegui, J. (2021). Cape Verde Frontal Zone in summer 2017: lateral transports of mass, dissolved oxygen and inorganic nutrients. *Ocean Science*, *17*(3), 769–788. https://doi.org/10.5194/os-17-769-2021

Camayo, R., & Campos, E. J. D. (2006). Application of wavelet transform in the study of coastal trapped waves off the west coast of South America. *Geophysical Research Letters*, *33*(22), L22601. https://doi.org/10.1029/2006gl026395

Campanero, R., Burgoa, N., Fernández-Castro, B., Valiente, S., Nieto-Cid, M., Martínez-Pérez, A. M., Gelado-Caballero, M. D., Hernández-Hernández, N., Marrero-Díaz, A., Machín, F., Rodríguez-Santana, A., Hernández-García, I., Delgado-Huertas, A., Martínez-Marrero, A., Arístegui, J., & Álvarez-Salgado, X. A. (2022). High-resolution variability of dissolved and suspended organic matter in the Cape Verde Frontal Zone. *Frontiers in Marine Science*, *9*, 1006432. https://doi.org/10.3389/fmars.2022.1006432

Cockburn, B. (2004). Discontinuous Galerkin Methods for Computational Fluid Dynamics. In *Encyclopedia of computational mechanics*. John Wiley; Sons, Ltd. https://doi.org/10.1002/0470091355.ecm053

Cockburn, B. (2016). Static Condensation, Hybridization, and the Devising of the HDG Methods. In *Building bridges: Connections and challenges in modern approaches to numerical partial differential equations* (pp. 129–177). Springer International Publishing. https://doi.org/10.1007/978-3-319-41640-3_5

Cockburn, B. (2023). Hybridizable discontinuous Galerkin methods for second-order elliptic problems: overview, a new result and open problems. *Japan Journal of Industrial and Applied Mathematics*, *40*(3), 1637–1676. https://doi.org/10.1007/s13160-023-00603-9

Cockburn, B., Gopalakrishnan, J., & Lazarov, R. (2009). Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems. *SIAM Journal on Numerical Analysis*, *47*(2), 1319–1365. https://doi.org/10.1137/070706616

Codina, R. (1998). Comparison of some finite element methods for solving the diffusion-convection-reaction equation. *Computer Methods in Applied Mechanics and Engineering*, *156*(1–4), 185–210. https://doi.org/10.1016/s0045-7825(97)00206-5

Collineau, S., & Brunet, Y. (1993). Detection of turbulent coherent motions in a forest canopy part I: Wavelet analysis. *Boundary-Layer Meteorology*, *65*(4), 357–379. https://doi.org/10.1007/BF00707033

Combes, J. M., Grossmann, A., & Tchamitchian, P. (1990). *Wavelets: Time-Frequency Methods and Phase Space. Proceedings of the International Conference, Marseille, France, December 14–18, 1987*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-75988-8

Conde, S., Gottlieb, S., Grant, Z. J., & Shadid, J. N. (2017). Implicit and implicit–explicit strong stability preserving runge–kutta methods with high linear order. *Journal of Scientific Computing*, *73*(2–3), 667–690. https://doi.org/10.1007/s10915-017-0560-2

Costa-Solé, A., Ruiz-Gironés, E., & Sarrate, J. (2019). An HDG formulation for incompressible and immiscible two-phase porous media flow problems. *International Journal of Computational Fluid Dynamics*, *33*(4), 137–148. https://doi.org/10.1080/10618562.2019.1617855

Costa-Solé, A. (2020). *High-order hybridizable discontinuous Galerkin formulation and implicit Runge-Kutta schemes for multiphase flow through porous media* [Doctoral dissertation, Universitat Politècnica de Catalunya]. https://doi.org/10.5821/dissertation-2117-328200

Crouzeix, M. (1980). Une méthode multipas implicite-explicite pour l'approximation des équations d'évolution paraboliques. *Numer. Math.*, *35*(3), 257–276. https://doi.org/10.1007/BF01396412

Cushman-Roisin, B. (2011). *Introduction to geophysical fluid dynamics* (Vol. 101) [Includes bibliographical references and index]. Academic Press. https://doi.org/10.1016/B978-0-12-088759-0.00026-2

Danilov, S., Kivman, G., & Schröter, J. (2004). A finite-element ocean model: Principles and evaluation. *Ocean Modelling*, *6*(2), 125–150. https://doi.org/10.1016/s1463-5003(02)00063-x

Dawson, C., Loveland, M., Pachev, B., Proft, J., & Valseth, E. (2024). SWEMniCS: a software toolbox for modeling coastal ocean circulation, storm surges, inland, and compound flooding. *npj Natural Hazards*, *1*(1). https://doi.org/10.1038/s44304-024-00036-5

Dawson, C., Westerink, J. J., Feyen, J. C., & Pothina, D. (2006). Continuous, discontinuous and coupled discontinuous–continuous Galerkin finite element methods for the shallow water equations. *International Journal for Numerical Methods in Fluids*, *52*(1), 63–88. https://doi.org/10.1002/fld.1156

Donea, J. (1984). A Taylor–Galerkin method for convective transport problems. *International Journal for Numerical Methods in Engineering*, *20*(1), 101–119. https://doi.org/10.1002/nme.1620200108

Donea, J., & Huerta, A. (2004). *Finite element methods for flow problems* (Repr.). Wiley. https://doi.org/10.1002/0470013826

Douglas, J., Jr., & Russell, T. F. (1982). Numerical Methods for Convection-Dominated Diffusion Problems Based on Combining the Method of Characteristics with Finite Element or Finite Difference Procedures. *SIAM Journal on Numerical Analysis*, *19*(5), 871–885. https://doi.org/10.1137/0719063

Dumbser, M., & Casulli, V. (2013). A staggered semi-implicit spectral discontinuous Galerkin scheme for the shallow water equations. *Applied Mathematics and Computation*, *219*(15), 8057–8077. https://doi.org/10.1016/j.amc.2013.02.041

Ellmenreich, J., Giacomini, M., Huerta, A., & Lederer, P. L. (2025). Characteristic boundary conditions for Hybridizable Discontinuous Galerkin methods. https://doi.org/10.48550/arXiv.2503.19684

Ern, A., Piperno, S., & Djadel, K. (2007). A well-balanced Runge–Kutta discontinuous Galerkin method for the shallow-water equations with flooding and drying. *International Journal for Numerical Methods in Fluids*, *58*(1), 1–25. https://doi.org/10.1002/fld.1674

Eskilsson, C., & Sherwin, S. J. (2004). A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations. *International Journal for Numerical Methods in Fluids*, *45*(6), 605–623. https://doi.org/10.1002/fld.709

European Union-Copernicus Marine Service. (2016). Global Ocean 1/12° Physics Analysis and Forecast updated Daily. https://doi.org/10.48670/MOI-00016

European Union-Copernicus Marine Service. (2021). GLOBAL OCEAN GRIDDED L4 SEA SURFACE HEIGHTS AND DERIVED VARIABLES REPROCESSED (1993-ONGOING). https://doi.org/10.48670/MOI-00148

Farge, M. (1992). Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics*, *24*(1), 395–457. https://doi.org/10.1146/annurev.fl.24.010192.002143

Felipe, A., Sevilla, R., & Hassan, O. (2024). A conservative degree adaptive HDG method for transient incompressible flows. *International Journal of Numerical Methods for Heat and Fluid Flow*, *35*(1), 300–329. https://doi.org/10.1108/hff-09-2024-0651

Fernández, P. (2019). *Entropy-stable hybridized discontinuous Galerkin methods for large-eddy simulation of transitional and turbulent flows* [Doctoral dissertation, Massachusetts Institute of Technology]. https://hdl.handle.net/1721.1/122496

Ferracina, L., & Spijker, M. N. (2004a). Stepsize Restrictions for the Total-Variation-Diminishing Property in General Runge–Kutta Methods. *SIAM Journal on Numerical Analysis*, *42*(3), 1073–1093. https://doi.org/10.1137/s0036142902415584

Ferracina, L., & Spijker, M. N. (2004b). An extension and analysis of the Shu-Osher representation of Runge-Kutta methods. *Mathematics of Computation*, *74*(249), 201–220. https://doi.org/10.1090/s0025-5718-04-01664-3

Fraeijs de Veubeke, B. (1965). Displacement and equilibrium models in the finite element method. In *Stress analysis*. John Wiley; Sons.

Gamage, N., & Blumen, W. (1993). Comparative analysis of low-level cold fronts: Wavelet, Fourier, and empirical orthogonal function decompositions. *Monthly Weather Review*, *121*(10), 2867–2878. https://doi.org/10.1175/1520-0493(1993)121%3C2867:CAOLLC%3E2.0.CO;2

Gamage, N., & Hagelberg, C. (1993). Detection and analysis of microfronts and associated coherent events using localized transforms. *Journal of Atmospheric Sciences*, *50*(5), 750–756. https://doi.org/10.1175/1520-0469(1993)050%3C0750:DAAOMA%3E2.0.CO;2

Gao, W., & Li, B. L. (1993). Wavelet Analysis of Coherent Structures at the Atmosphere-Forest Interface. *Journal of Applied Meteorology and Climatology*, *32*(11), 1717–1725. https://doi.org/10.1175/1520-0450(1993)032<1717:WAOCSA>2.0.CO;2

Garel, E., Laiz, I., Drago, T., & Relvas, P. (2016). Characterisation of coastal counter-currents on the inner shelf of the Gulf of Cadiz. *Journal of Marine Systems*, *155*, 19–34. https://doi.org/10.1016/j.jmarsys.2015.11.001

Gibson, T. H., Mitchell, L., Ham, D. A., & Cotter, C. J. (2020). Slate: extending Firedrake's domain-specific abstraction to hybridized solvers for geoscience and beyond. *Geoscientific Model Development*, *13*(2), 735–761. https://doi.org/10.5194/gmd-13-735-2020

Giraldo, F. X., Hesthaven, J. S., & Warburton, T. (2002). Nodal High-Order Discontinuous Galerkin Methods for the Spherical Shallow Water Equations. *Journal of Computational Physics*, *181*(2), 499–525. https://doi.org/10.1006/jcph.2002.7139

Giraldo, F. X., & Warburton, T. (2007). A high-order triangular discontinuous Galerkin oceanic shallow water model. *International Journal for Numerical Methods in Fluids*, *56*(7), 899–925. https://doi.org/10.1002/fld.1562

Gollmer, S. M., Harshvardhan, Cahalan, R. F., & Snider, J. B. (1995). Windowed and Wavelet Analysis of Marine Stratocumulus Cloud Inhomogeneity. *Journal of Atmospheric Sciences*, *52*(16), 3013–3030. https://doi.org/10.1175/1520-0469(1995)052<3013:WAWAOM>2.0.CO;2

Golub, G. H., & Varga, R. S. (1961). Chebyshev semi-iterative methods, successive overrelaxation iterative methods, and second order Richardson iterative methods. *Numer. Math.*, *3*(1), 147–156. https://doi.org/10.1007/BF01386013

Gottlieb, S., Ketcheson, D., & Shu, C.-W. (2011). *Strong Stability Preserving Explicit Runge—Kutta Methods*. World Scientific Publishing Co. Pte. Ltd. https://doi.org/10.1142/9789814289276_0002

Gottlieb, S., & Shu, C.-W. (1998). Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation*, *67*(221), 73–85. https://doi.org/10.1090/s0025-5718-98-00913-2

Grinsted, A., Moore, J. C., & Jevrejeva, S. (2004). Application of the cross wavelet transform and wavelet coherence to geophysical time series. *Nonlinear Processes in Geophysics*, *11*(5/6), 561–566. https://doi.org/10.5194/npg-11-561-2004

Grossmann, A., & Morlet, J. (1984). Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape. *SIAM Journal on Mathematical Analysis*, *15*(4), 723–736. https://doi.org/10.1137/0515056

Gu, D., & Philander, S. G. H. (1995). Secular Changes of Annual and Interannual Variability in the Tropics during the Past Century. *Journal of Climate*, *8*(4), 864–876. https://doi.org/10.1175/1520-0442(1995)008<0864:SCOAAI>2.0.CO;2

Hagen, E. (1985). A meandering intermediate front North-West off Cape Verde Islands. *Oceanographie Tropicale*, *20*(1), 71–83. https://www.researchgate.net/publication/32986626_A_meandering_intermediate_front_North-West_off_Cape_Verde_Islands

Hales, B., & Takahashi, T. (2002). The pumping SeaSoar: a high-resolution seawater sampling platform. *Journal of Atmospheric and Oceanic Technology*, *19*(7), 1096–1104. https://doi.org/10.1175/1520-0426(2002)019%3C1096:TPSAHR%3E2.0.CO;2

Ham, D. A., Kelly, P. H. J., Mitchell, L., Cotter, C. J., Kirby, R. C., Sagiyama, K., Bouziani, N., Vorderwuelbecke, S., Gregory, T. J., Betteridge, J., Shapero, D. R., Nixon-Hill, R. W., Ward, C. J., Farrell, P. E., Brubeck, P. D., Marsden, I., Gibson, T. H., Homolya, M., Sun, T., … Markall, G. R. (2023, May). *Firedrake User Manual* (First edition). Imperial College

London, University of Oxford, Baylor University, and University of Washington. https://doi.org/10.25561/104839

Hanert, E., Legat, V., & Deleersnijder, E. (2003). A comparison of three finite elements to solve the linear shallow water equations. *Ocean Modelling*, *5*(1), 17–35. https://doi.org/10.1016/s1463-5003(02)00012-4

Hernández-García, I., Ramos, A., & Rodríguez-Santana, A. (2018). *Validation of ocean forecasting model data with those obtained from the first transoceanic autonomous underwater vehicles (gliders) missions in the North-East Atlantic basin* [Final degree project]. Universidad de Las Palmas de Gran Canaria [published]. http://hdl.handle.net/10553/57935

Hernández-García, I., Coca, J., Ramos, A., Rodríguez-Santana, Á., & Machin, F. (2024). Assessing hydrological sampling approaches in the Cape Verde frontal zone in November 2017. *Scientia Marina*, *88*(4), e090. https://doi.org/10.3989/scimar.05509.090

Hernández-Guerra, A., Fraile-Nuez, E., López-Laatzen, F., Martínez, A., Parrilla, G., & Vélez-Belchí, P. (2005). Canary Current and North Equatorial Current from an inverse box model. *Journal of Geophysical Research: Oceans*, *110*(C12). https://doi.org/10.1029/2005jc003032

Hesthaven, J. S., & Warburton, T. (2008). *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer New York. https://doi.org/10.1007/978-0-387-72067-8

Higueras, I., Happenhofer, N., Koch, O., & Kupka, F. (2014). Optimized strong stability preserving IMEX Runge–Kutta methods. *Journal of Computational and Applied Mathematics*, *272*, 116–140. https://doi.org/10.1016/j.cam.2014.05.011

Hughes, P., & Barton, E. D. (1974). Stratification and water mass structure in the upwelling area off northwest Africa in April/May 1969. *Deep Sea Research and Oceanographic Abstracts*, *21*(8), 611–620. https://doi.org/10.1016/0011-7471(74)90046-1

Hughes, T. J. R., Franca, L. P., & Hulbert, G. M. (1989). A new finite element formulation for computational fluid dynamics: VIII. The galerkin/least-squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, *73*(2), 173–189. https://doi.org/10.1016/0045-7825(89)90111-4

Hughes, T. J. R., Scovazzi, G., & Tezduyar, T. E. (2010). Stabilized Methods for Compressible Flows. *Journal of Scientific Computing*, *43*(3), 343–368. https://doi.org/10.1007/s10915-008-9233-5

Hughes, T. J. (1995). Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Computer Methods in Applied Mechanics and Engineering*, *127*(1–4), 387–401. https://doi.org/10.1016/0045-7825(95)00844-9

Isherwood, L., Grant, Z. J., & Gottlieb, S. (2018). Strong Stability Preserving Integrating Factor Runge–Kutta Methods. *SIAM Journal on Numerical Analysis*, *56*(6), 3276–3307. https://doi.org/10.1137/17m1143290

Iskandarani, M., Haidvogel, D. B., & Boyd, J. P. (1995). A staggered spectral element model with application to the oceanic shallow water equations. *International Journal for Numerical Methods in Fluids*, *20*(5), 393–414. https://doi.org/10.1002/fld.1650200504

Javadzadeh Moghtader, M. (2016). *High-order hybridizable discontinuous Galerkin method for viscous compressible flows* [Doctoral dissertation, Universitat Politècnica de Catalunya]. https://doi.org/10.5821/dissertation-2117-105834

Jevrejeva, S., Moore, J. C., & Grinsted, A. (2003). Influence of the Arctic Oscillation and El Niño-Southern Oscillation (ENSO) on ice conditions in the Baltic Sea: The wavelet approach. *Journal of Geophysical Research: Atmospheres*, *108*(D21), 4677. https://doi.org/10.1029/2003jd003417

Kämpf, J. (2009). *Ocean modelling for beginners*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-00820-7

Kang, S., Giraldo, F. X., & Bui-Thanh, T. (2020). IMEX HDG-DG: A coupled implicit hybridized discontinuous Galerkin and explicit discontinuous Galerkin approach for shallow water systems. *Journal of Computational Physics*, *401*, 109010. https://doi.org/10.1016/j.jcp.2019.109010

Kärnä, T., Kramer, S. C., Mitchell, L., Ham, D. A., Piggott, M. D., & Baptista, A. M. (2018). Thetis coastal ocean model: discontinuous Galerkin discretization for the three-dimensional hydrostatic equations. *Geoscientific Model Development*, *11*(11), 4359–4382. https://doi.org/10.5194/gmd-11-4359-2018

Ketcheson, D. I., Macdonald, C. B., & Gottlieb, S. (2009). Optimal implicit strong stability preserving Runge–Kutta methods. *Applied Numerical Mathematics*, *59*(2), 373–392. https://doi.org/10.1016/j.apnum.2008.03.034

Kirby, R. M., Sherwin, S. J., & Cockburn, B. (2012). To CG or to HDG: A Comparative Study. *Journal of Scientific Computing*, *51*(1), 183–212. https://doi.org/10.1007/s10915-011-9501-7

Kirk, K., Horváth, T., & Rhebergen, S. (2022). Analysis of an exactly mass conserving space-time hybridized discontinuous Galerkin method for the time-dependent Navier–Stokes equations. *Mathematics of Computation*, *92*(340), 525–556. https://doi.org/10.1090/mcom/3796

Kolar, R. L., Gray, W. G., & Westerink, J. J. (1996). Boundary conditions in shallow water models—an alternative implementation for finite element codes. *International Journal for Numerical Methods in Fluids*, *22*(7), 603–618. https://doi.org/10.1002/(sici)1097-0363(19960415)22:7<603::aid-fld369>3.0.co;2-e

Koshlyakov, M. N., & Grachev, Y. M. (1973). Meso-scale currents at a hydrophysical polygon in the tropical Atlantic. *Deep Sea Research and Oceanographic Abstracts*, *20*(6), 507–526. https://doi.org/10.1016/0011-7471(73)90075-2

Kubatko, E. J., Bunya, S., Dawson, C., Westerink, J. J., & Mirabito, C. (2009). A Performance Comparison of Continuous and Discontinuous Finite Element Shallow Water Models. *Journal of Scientific Computing*, *40*(1), 315–339. https://doi.org/10.1007/s10915-009-9268-2

Kubatko, E. J., Westerink, J. J., & Dawson, C. (2006). hp Discontinuous Galerkin methods for advection dominated problems in shallow water flow. *Computer Methods in Applied Mechanics and Engineering*, *196*(1–3), 437–451. https://doi.org/10.1016/j.cma.2006.05.002

Kumar, P., & Foufoula-Georgiou, E. (1993). A New Look at Rainfall Fluctuations and Scaling Properties of Spatial Rainfall Using Orthogonal Wavelets. *Journal of Applied Meteorology and Climatology*, *32*(2), 209–222. https://doi.org/10.1175/1520-0450(1993)032<0209:ANLARF>2.0.CO;2

Kundu, P. K., Cohen, I. M., Dowling, D. R., & Capecelatro, J. (2024, January). *Fluid Mechanics (Seventh Edition)* (7th ed.). Academic Press. https://doi.org/10.1016/C2018-0-02033-2

Lau, K. M., & Weng, H. (1995). Climate Signal Detection Using Wavelet Transform: How to Make a Time Series Sing. *Bulletin of the American Meteorological Society*, *76*(12), 2391–2402. https://doi.org/10.1175/1520-0477(1995)076<2391:CSDUWT>2.0.CO;2

Le Roux, D. Y., Lin, C. A., & Staniforth, A. (2000). A Semi-implicit Semi-Lagrangian Finite-Element Shallow-Water Ocean Model. *Monthly Weather Review*, *128*(5), 1384–1401. https://doi.org/10.1175/1520-0493(2000)128<1384:asislf>2.0.co;2

Le Roux, D. Y., Staniforth, A., & Lin, C. A. (1998). Finite Elements for Shallow-Water Equation Ocean Models. *Monthly Weather Review*, *126*(7), 1931–1951. https://doi.org/10.1175/1520-0493(1998)126<1931:fefswe>2.0.co;2

Lee, J. K., & Froehlich, D. C. (1987). *Review of literature on the finite-element solution of the equations of two-dimensional surface-water flow in the horizontal plane*. https://doi.org/10.3133/cir1009

LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems* (Vol. 31). Cambridge university press. https://doi.org/10.1017/CBO9780511791253

LeVeque, R. J. (2007). *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems*. SIAM. https://doi.org/10.1137/1.9780898717839

Liu, P. C. (1994). Wavelet spectrum analysis and ocean wind waves. In E. Foufoula-Georgiou & P. Kumar (Eds.), *Wavelet analysis and its applications* (pp. 151–166, Vol. 4). Academic Press. https://doi.org/10.1016/B978-0-08-052087-2.50012-8

Logg, A., Mardal, K.-A., & Wells, G. (Eds.). (2012). *Automated Solution of Differential Equations by the Finite Element Method: The fenics book*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-23099-8

Löhner, R., Morgan, K., & Zienkiewicz, O. C. (1984). The solution of non-linear hyperbolic equation systems by the finite element method. *International Journal for Numerical Methods in Fluids*, *4*(11), 1043–1063. https://doi.org/10.1002/fld.1650041105

Machín, F., Hernández-Guerra, A., & Pelegrí, J. L. (2006). Mass fluxes in the Canary Basin. *Progress in Oceanography*, *70*(2–4), 416–447. https://doi.org/10.1016/j.pocean.2006.03.019

Mak, M. (1995). Orthogonal wavelet analysis: interannual variability in the surface temperature. *Bulletin of the American Meteorological Society*, *76*, 2391–2402. https://doi.org/10.1175/1520-0477(1995)076%3C2179:OWAIVI%3E2.0.CO;2

Marras, S., Kelly, J. F., Moragues, M., Müller, A., Kopera, M. A., Vázquez, M., Giraldo, F. X., Houzeaux, G., & Jorba, O. (2015). A Review of Element-Based Galerkin Methods for Numerical Weather Prediction: Finite Elements, Spectral Elements, and Discontinuous Galerkin. *Archives of Computational Methods in Engineering*, *23*(4), 673–722. https://doi.org/10.1007/s11831-015-9152-1

Martínez-Marrero, A., Rodríguez-Santana, A., Hernández-Guerra, A., Fraile-Nuez, E., López-Laatzen, F., Vélez-Belchí, P., & Parrilla, G. (2008). Distribution of water masses and diapycnal mixing in the Cape Verde Frontal Zone. *Geophysical Research Letters*, *35*(7), L07609. https://doi.org/10.1029/2008gl033229

Meyers, S. D., Kelly, B. G., & O'Brien, J. J. (1993). An Introduction to Wavelet Analysis in Oceanography and Meteorology: With Application to the Dispersion of Yanai Waves. *Monthly Weather Review*, *121*(10), 2858–2866. https://doi.org/10.1175/1520-0493(1993)121<2858:AITWAI>2.0.CO;2

Mittelstaedt, E. (1983). The upwelling area off Northwest Africa—A description of phenomena related to coastal upwelling. *Progress in Oceanography*, *12*(3), 307–331. https://doi.org/10.1016/0079-6611(83)90012-5

Morlet, J. (1983). Sampling Theory and Wave Propagation. In C. H. Chen (Ed.), *Issues in acoustic signal — image processing and recognition* (pp. 233–261). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-82002-1_12

Morlet, J., Arens, G., Fourgeau, E., & Giard, D. (1982). Wave propagation and sampling theory—Part I: Complex signal and scattering in multilayered media. *Geophysics*, *47*(2), 203–221. https://doi.org/10.1190/1.1441328

Moron, V., Vautard, R., & Ghil, M. (1998). Trends, interdecadal and interannual oscillations in global sea-surface temperatures. *Climate Dynamics*, *14*, 545–569. https://doi.org/10.1007/s003820050241

Navarro, L., Martínez-Marrero, A., & Rodríguez-Santana, A. (2018). *Hydrographical and dynamical properties of the Cape Verde Frontal Zone during the FLUXES-II survey* [Final degree project]. Universidad de Las Palmas de Gran Canaria [published]. http://hdl.handle.net/10553/75072

Navier, C. L. M. H. (1823). Mémoire sur les lois du mouvement des fluides. *Mémoires de l'Académie Royale des Sciences de l'Institut de France*, *6*(1823), 389–440.

Nguyen, N., Peraire, J., & Cockburn, B. (2011). An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier–Stokes equations. *Journal of Computational Physics*, *230*(4), 1147–1170. https://doi.org/10.1016/j.jcp.2010.10.032

Ortleb, S. From Finite Volumes to Discontinuous Galerkin and fluix reconstruction. In: In *Gofun 2017 openfoam*. 2017. https://www.foamacademy.com/wp-content/uploads/2016/11/ortleb_gofun.pdf

Pachev, B. A. (2024). *Automated GPU acceleration of stabilized shallow water solvers with FEniCSx* [Doctoral dissertation, The University of Texas at Austin]. The University of Texas at Austin. https://doi.org/10.26153/TSW/56005

Pan, W., Kramer, S. C., & Piggott, M. D. (2021). A σ-coordinate non-hydrostatic discontinuous finite element coastal ocean model. *Ocean Modelling*, *157*, 101732. https://doi.org/10.1016/j.ocemod.2020.101732

Pareschi, L., & Russo, G. (2005). Implicit–Explicit Runge–Kutta Schemes and Applications to Hyperbolic Systems with Relaxation. *Journal of Scientific Computing*, *25*(1), 129–155. https://doi.org/10.1007/s10915-004-4636-4

Pelegrí, J. L., Peña-Izquierdo, J., Machín, F., Meiners, C., & Presas-Navarro, C. (2017). Oceanography of the Cape Verde Basin and Mauritanian Slope Waters. In A. Ramos, F. Ramil, & J. Sanz (Eds.), *Deep-Sea Ecosystems Off Mauritania: Research of Marine Biodiversity and Habitats in the Northwest African Margin* (pp. 119–153). Springer Netherlands. https://doi.org/10.1007/978-94-024-1023-5_3

Pérez, F. F., Mintrop, L., Llinás, O., González-Dávila, M., Castro, C. G., Alvarez, M., Körtzinger, A., Santana-Casiano, M., Rueda, M. J., & Ríos, A. F. (2001). Mixing analysis of nutrients, oxygen and inorganic carbon in the Canary Islands region. *Journal of Marine Systems*, *28*(3-4), 183–201. https://doi.org/10.1016/s0924-7963(01)00003-3

Pérez-Rodríguez, P., Pelegrí, J. L., & Marrero-Díaz, A. (2001). Dynamical characteristics of the Cape Verde frontal zone. *Scientia Marina*, *65*(S1), 241–250. https://doi.org/10.3989/scimar.2001.65s1241

Piggott, M. D., Gorman, G. J., Pain, C. C., Allison, P. A., Candy, A. S., Martin, B. T., & Wells, M. R. (2007). A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes. *International Journal for Numerical Methods in Fluids*, *56*(8), 1003–1015. https://doi.org/10.1002/fld.1663

Pollard, R. (1986). Frontal surveys with a towed profiling conductivity/temperature/depth measurement package (SeaSoar). *Nature*, *323*(6087), 433–435. https://doi.org/10.1038/323433a0

Ramos, A. G., García-Garrido, V. J., Mancho, A. M., Wiggins, S., Coca, J., Glenn, S., Schofield, O., Kohut, J., Aragon, D., Kerfoot, J., Haskins, T., Miles, T., Haldeman, C., Strandskov, N., Allsup, B., Jones, C., & Shapiro, J. (2018). Lagrangian coherent structure assisted path planning for transoceanic autonomous underwater vehicle missions. *Scientific Reports*, *8*(1), 4575. https://doi.org/10.1038/s41598-018-23028-8

Reed, W. H., & Hill, T. R. (1973). *Triangular mesh methods for the neutron transport equation* (tech. rep.). Los Alamos Scientific Lab., N. Mex.(USA).

Restelli, M., & Giraldo, F. X. (2009). A Conservative Discontinuous Galerkin Semi-Implicit Formulation for the Navier–Stokes Equations in Nonhydrostatic Mesoscale Modeling. *SIAM Journal on Scientific Computing*, *31*(3), 2231–2257. https://doi.org/10.1137/070708470

Ricardo, K., Lee, D., & Duru, K. (2024). Conservation and stability in a discontinuous Galerkin method for the vector invariant spherical shallow water equations. *Journal of Computational Physics*, *500*, 112763. https://doi.org/10.1016/j.jcp.2024.112763

Ricchiuto, M., & Bollermann, A. (2009). Stabilized residual distribution for shallow water simulations. *Journal of Computational Physics*, *228*(4), 1071–1115. https://doi.org/10.1016/j.jcp.2008.10.020

Rudnick, D. L., Davis, R. E., Eriksen, C. C., Fratantoni, D. M., & Perry, M. J. (2004). Underwater Gliders for Ocean Research. *Marine Technology Society Journal*, *38*(2), 73–84. https://doi.org/10.4031/002533204787522703

Saad, Y., & Schultz, M. H. (1986). GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, *7*(3), 856–869. https://doi.org/10.1137/0907058

Samii, A., Kazhyken, K., Michoski, C., & Dawson, C. (2019). A Comparison of the Explicit and Implicit Hybridizable Discontinuous Galerkin Methods for Nonlinear Shallow Water Equations. *Journal of Scientific Computing*, *80*(3), 1936–1956. https://doi.org/10.1007/s10915-019-01007-z

Sangrà, P., Pascual, A., Rodríguez-Santana, Á., Machín, F., Mason, E., McWilliams, J. C., Pelegrí, J. L., Dong, C., Rubio, A., Arístegui, J., Marrero-Díaz, Á., Hernández-Guerra, A., Martínez-Marrero, A., & Auladell, M. (2009). The Canary Eddy Corridor: A major pathway for long-lived eddies in the subtropical North Atlantic. *Deep Sea Research Part I: Oceanographic Research Papers*, *56*(12), 2100–2114. https://doi.org/10.1016/j.dsr.2009.08.008

Savant, G., McAlpin, T. O., & Trahan, C. J. (2019). Streamline Upwind Petrov-Galerkin–Based Shallow Water Model for Large-Scale Geophysical Flows in Cartesian and Spherical Coordinates. *Journal of Waterway, Port, Coastal, and Ocean Engineering*, *145*(5). https://doi.org/10.1061/(asce)ww.1943-5460.0000521

Sayas, F.-J. (2015). A gentle introduction to the Finite Element Method. https://team-pancho.github.io/documents/anIntro2FEM_2015.pdf

Schwanenberg, D., & Köngeter, J. (2000). A Discontinuous Galerkin Method for the Shallow Water Equations with Source Terms. In B. Cockburn, G. E. Karniadakis, & C.-W. Shu (Eds.), *Discontinuous galerkin methods* (pp. 419–424). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-59721-3_42

Shakib, F., & Hughes, T. J. R. (1991). A new finite element formulation for computational fluid dynamics: IX. Fourier analysis of space-time Galerkin/least-squares algorithms. *Computer Methods in Applied Mechanics and Engineering*, *87*(1), 35–58. https://doi.org/10.1016/0045-7825(91)90145-v

Shu, C.-W. (1988). Total-Variation-Diminishing Time Discretizationss. *SIAM Journal on Scientific and Statistical Computing*, *9*(6), 1073–1084. https://doi.org/10.1137/0909073

Shu, C.-W., & Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, *77*(2), 439–471. https://doi.org/10.1016/0021-9991(88)90177-5

Spijker, M. N. (2007). Stepsize Conditions for General Monotonicity in Numerical Initial Value Problems. *SIAM Journal on Numerical Analysis*, *45*(3), 1226–1245. https://doi.org/10.1137/060661739

Stokes, G. G. (1845). On the Theories of the Internal Friction of Fluids in Motion and of the Equilibrium and Motion of Elastic Solids. *Trans. Cambridge Philos. Soc.*, *8*, 287–319.

Talley, L. D., Pickard, G. L., Emery, W. J., & Swift, J. H. (2011). *Descriptive physical oceanography: an introduction* (6th ed.). Academic press. https://doi.org/10.1016/C2009-0-24322-4

Tomczak, M. (1981). An analysis of mixing in the frontal zone of South and North Atlantic Central Water off North-West Africa. *Progress in Oceanography*, *10*(3), 173–192. https://doi.org/10.1016/0079-6611(81)90011-2

Tomczak, M., & Hughes, P. (1980). Three dimensional variability of water masses and currents in the Canary Current upwelling region. *Meteor Forschungsergebnisse: Reihe A, Allgemeines, Physik und Chemie des Meeres*, *21*, 1–24. https://oceanrep.geomar.de/id/eprint/56408/

Torrence, C., & Compo, G. P. (1998). A Practical Guide to Wavelet Analysis. *Bulletin of the American Meteorological Society*, *79*(1), 61–78. https://doi.org/10.1175/1520-0477(1998)079<0061:APGTWA>2.0.CO;2

Troupin, C., Beltran, J., Heslop, E., Torner, M., Garau, B., Allen, J., Ruiz, S., & Tintoré, J. (2015). A toolbox for glider data processing and management. *Methods in Oceanography*, *13-14*, 13–23. https://doi.org/10.1016/j.mio.2016.01.001

Ueckermann, M. P., & Lermusiaux, P. F. J. (2016). Hybridizable discontinuous Galerkin projection methods for Navier–Stokes and Boussinesq equations. *Journal of Computational Physics*, *306*, 390–421. https://doi.org/10.1016/j.jcp.2015.11.028

Valseth, E., & Dawson, C. (2021). A stable space-time FE method for the shallow water equations. *Computational Geosciences*, *26*(1), 53–70. https://doi.org/10.1007/s10596-021-10108-4

Vélez-Belchí, P., Pérez-Hernández, M. D., Casanova-Masjoan, M., Cana, L., & Hernández-Guerra, A. (2017). On the seasonal variability of the Canary Current and the Atlantic Meridional Overturning Circulation. *Journal of Geophysical Research: Oceans*, *122*(6), 4518–4538. https://doi.org/10.1002/2017jc012774

Venkata-Ramana, R., Krishna, B., Kumar, S. R., & Pandey, N. G. (2013). Monthly Rainfall Prediction Using Wavelet Neural Network Analysis. *Water Resources Management*, *27*(10), 3697–3711. https://doi.org/10.1007/s11269-013-0374-4

Vo, J. H. (2017). *Modeling flow encountering abrupt topography using hybridizable discontinuous Galerkin projection methods* [Doctoral dissertation, Massachusetts Institute of Technology]. http://hdl.handle.net/1721.1/113970

von Schuckmann, K., Le-Traon, P. Y., Alvarez-Fanjul, E., Axell, L., Balmaseda, M., Breivik, L. A., Brewin, R. J. W., Bricaud, C., Drevillon, M., Drillet, Y., Dubois, C., Embury, O., Etienne, H., García-Sotillo, M., Garric, G., Gasparin, F., Gutknecht, E., Guinehut, S., Hernandez, F., . . . Verbrugge, N. (2016). The Copernicus Marine Environment Monitoring Service Ocean State Report. *Journal of Operational Oceanography*, *9*(sup2), s235–s320. https://doi.org/10.1080/1755876x.2016.1273446

Walters, R. A., & Barragy, E. J. (1997). COMPARISON OF *H* AND *P* FINITE ELEMENT APPROXIMATIONS OF THE SHALLOW WATER EQUATIONS. *International Journal for Numerical Methods in Fluids*, *24*(1), 61–79. https://doi.org/10.1002/(sici)1097-0363(19970115)24:1<61::aid-fld479>3.0.co;2-y

Walters, R. A. (2005). Coastal ocean models: Two useful finite element methods. *Continental Shelf Research*, *25*(7–8), 775–793. https://doi.org/10.1016/j.csr.2004.09.020

Walters, R. A., & Carey, G. F. (1983). Analysis of spurious oscillation modes for the shallow water and Navier-Stokes equations. *Computers & Fluids*, *11*(1), 51–68. https://doi.org/10.1016/0045-7930(83)90013-0

Wang, B., & Wang, Y. (1996). Temporal Structure of the Southern Oscillation as Revealed by Waveform and Wavelet Analysis. *Journal of Climate*, *9*(7), 1586–1598. https://doi.org/10.1175/1520-0442(1996)009<1586:TSOTSO>2.0.CO;2

Weng, H., & Lau, K. M. (1994). Wavelets, Period Doubling, and Time–Frequency Localization with Application to Organization of Convection over the Tropical Western Pacific. *Journal of Atmospheric Sciences*, *51*(17), 2523–2541. https://doi.org/10.1175/1520-0469(1994)051<2523:WPDATL>2.0.CO;2

Wichitrnithed, C., Valseth, E., Kubatko, E. J., Kang, Y., Hudson, M., & Dawson, C. (2024). A discontinuous Galerkin finite element model for compound flood simulations. *Computer Methods in Applied Mechanics and Engineering*, *420*, 116707. https://doi.org/10.1016/j.cma.2023.116707

Williams, R. T., & Zienkiewicz, O. C. (1981). Improved finite element forms for the shallow-water wave equations. *International Journal for Numerical Methods in Fluids*, *1*(1), 81–97. https://doi.org/10.1002/fld.1650010107

Wu, X., Trask, N., & Chan, J. (2024). Entropy stable discontinuous Galerkin methods for the shallow water equations with subcell positivity preservation. *Numerical Methods for Partial Differential Equations*, *40*(6), e23129. https://doi.org/10.1002/num.23129

Yiou, P., Sornette, D., & Ghil, M. (2000). Data-adaptive wavelets and multi-scale singular-spectrum analysis. *Physica D: Nonlinear Phenomena*, *142*(3-4), 254–290. https://doi.org/10.1016/S0167-2789(00)00045-2

Zenk, W., Klein, B., & Schroder, M. (1991). Cape Verde Frontal Zone. *Deep Sea Research Part A. Oceanographic Research Papers*, *38*(sup1), S505–S530. https://doi.org/10.1016/s0198-0149(12)80022-7

Zienkiewicz, O. C., Taylor, R. L., & Zhu, J. Z. (2013). *The finite element method: Its basis and fundamentals*. Elsevier. https://doi.org/10.1016/C2009-0-24909-9