

Université catholique de Louvain
Faculté des Sciences Appliquées
Unité de Mécanique Appliquée

Discontinuous Galerkin methods for geophysical flow modeling

Thesis presented for the degree of
Doctor in applied Sciences
by

Paul-Emile Bernard

Members of jury:

Professor Jean-François Remacle, Department of Civil Engineering, UCL (Advisor)
Professor Vincent Legat, Department of Mechanics, UCL (Advisor)
Professor Eric Deleersnijder, Department of Mechanics, UCL (Advisor)
Professor Grégoire Winckelmans, Department of Mechanics, UCL (President)
Professor Sandra Soares Frazão, Department of Civil Engineering, UCL
Professor Ludovic Noels, Université de Liège
Professor Gérard Degrez, Université Libre de Bruxelles
Professor Spencer Sherwin, Imperial College (London, UK)

Louvain-la-Neuve, October 2008

Remerciements

Je voudrais dans ces quelques lignes remercier tous ceux qui ont contribué, de près ou de loin, à la réalisation de cette thèse.

Je remercie tout d'abord mes promoteurs, les Professeurs Eric Deleersnijder, Vincent Legat et Jean-François Remacle, pour m'avoir proposé cet intéressant sujet de recherche, et pour m'avoir aidé par leurs conseils, leur soutien et leurs grandes compétences à réaliser cette thèse. S'il est vrai que l'encadrement conjoint de trois promoteurs peut impliquer de nombreuses discussions, négociations ou compromis, la combinaison de trois points de vues souvent différents et complémentaires implique avant tout une grande richesse, tant sur le plan scientifique que humain.
Je voudrais remercier Eric Deleersnijder dont les nombreuses idées et pistes de réflexions, ainsi que la rigueur scientifique, ont constitué un des moteurs de ce travail. Son dévouement passionné à la recherche amène ses collaborateurs et étudiants à se dépasser, sans parler de ses encouragements répétés à la communication et la publication scientifique.
Je suis également très reconnaissant envers Vincent Legat pour ces années passées au sein de l'unité de mécanique appliquée. Depuis le début du travail de fin d'études jusqu'à la fin de ma thèse, sa rigueur scientifique et sa clairvoyance, mais surtout sa gentillesse, sa simplicité et ses nombreux encouragements ont été une aide précieuse. Ses nombreux conseils avisés quant à la rédaction ou la communication scientifique en particulier m'ont beaucoup appris. Non seulement sur le plan de la rédaction, mais aussi parce que nos discussions m'amenaient immanquablement à porter un autre regard sur mon travail et à le comprendre différemment. Je tiens à l'en remercier.
Finalement, je remercie profondément Jean-François Remacle. C'est probablement à lui et à son grand enthousiasme que je dois la passion-

nante orientation “discontinue” de cette thèse. Sa bonne humeur, sa compétence et ses conseils avisés ont toujours été une source d'inspiration et de motivation, que ce soit en conférence, devant un clavier, ou derrière une Flying V.

Je tiens également à remercier les membres de l'équipe SLIM, mais aussi les chercheurs de “Mema ++” en général, dont la bonne humeur, les compétences et la collaboration tant scientifique qu'informatique m'ont permis de réaliser ce travail. Merci également à Richard pour ses bons conseils à la relecture de ce manuscrit. Je remercie tout particulièrement mon collègue de bureau, Jonathan, pour sa gentillesse, sa disponibilité, et sa précieuse aide informatique. Sans lui, je n'aurais sans doute jamais connu les joies de *vi*, les reconfigurations de *xorg.conf* ou les commandes obscures du changement d'adresse MAC.

Finalement, je remercie du fond du coeur ma famille et mes amis sans lesquels rien n'aurait été possible. Je remercie particulièrement mes parents pour leur soutien inconditionnel, et pour m'avoir supporté, dans tous les sens du terme, durant ces années. Un grand merci, bien évidement, à Sh@rlie, Misha et Alizée, pour leur bonne humeur, les nombreux moments de détente au cinéma ou les soirées interminables au QG. Un grand merci à Stéphane, que ce soit pour les godets à la taverne, les heures passées au Spirit ou devant une Strat ou encore sa précieuse aide pour sortir du Grand Canyon. Merci à cousin Pierre pour ces nombreux échanges philo- voire théologiques autour d'un Moods, ainsi qu'à Julien et Jeff pour ces détentes hebdomadaires autour d'une table de billard. Et merci bien évidement à Tic pour ses raccourcis, les évitements de rennes et tous ces bons moments de rallye Scandinave. Merci à tous.

Cette thèse a été réalisée avec le soutien du Fond pour la formation à la Recherche dans l'Industrie et dans l'Agriculture (FRIA), ainsi que celui de la Communauté Française de Belgique au travers du contrat d'Actions de Recherche Concertées ARC 04/09-316.

Summary

Remerciements **iii**

Chapter 1 Introduction **1**

1.1 Ocean Modeling: from finite differences to unstructured meshes 2
1.2 Motivation for this thesis 6
1.3 Structure of the thesis 10
1.4 Supporting publications 12

Chapter 2 Governing equations: the shallow water equations **15**

2.1 General equations in a rotating framework 16
2.2 The Shallow Water Equations 22
2.3 Linear barotropic shallow water waves 25
2.3.1 Dispersion relations and classical approximations . 27
2.3.2 Exact dispersion relations 32

Chapter 3 The Discontinuous Galerkin Method **37**

3.1 Overview 38
3.2 The Discontinuous Galerkin Method 41
3.3 Accurate advection schemes 44
3.3.1 One-dimensional linear advection equation 44

3.3.2 Riemann solver for a linear system of PDE's . . . 46
3.4 A one-dimensional benchmark 47
3.5 Approximate Riemann solvers for non linear conservation laws 60
3.5.1 Exact Riemann solver for non-linear systems . . . 60
3.5.2 Roe Solver 65
3.5.3 Lax-Friedrich solver 68
3.6 Conclusions 68

Chapter 4 Efficient Adaptive Strategy using DG 71
4.1 Metric, error estimation and mesh adaptation 72
4.1.1 Towards an optimal mesh: creation of a metric . . 73
4.1.2 Usual approaches for error estimation and metric generation 76
4.2 h- and p- adaptivity with a DG approach 79
4.2.1 Jumps as error estimation 80
4.2.2 Optimal h-adaptation strategy 82
4.2.3 p-adaptation to a target error 84
4.2.4 h-adaptation procedures and projection of the solution 85
4.3 p-adaptation on a 1D benchmark 88
4.4 2D h-adaptation 89
4.4.1 Adaptive convergence applied to the Stommel gyre 89
4.4.2 Propagation of a baroclinic anticyclonic eddy at midlatitudes 98
4.5 Conclusions 103

Chapter 5 High-order geometries for the DG method 105
5.1 High-order representation of the boundaries 106

5.1.1 Discrete DG formulation of viscous shallow water equations . . . 108
5.1.2 Cylinder test case and the impact of the boundaries discretization . . . 112
5.1.3 Rattray island in a realistic tidal flow . . . 120
5.2 Local high-order tangent basis approach on curved manifolds . . . 135
5.2.1 Usual approaches for solving PDE's on curved manifolds . . . 137
5.2.2 Local Tangent Basis approach using the Discontinuous Galerkin method . . . 142
5.2.3 Solving the shallow water equations on curved manifolds . . . 154
5.3 Conclusions . . . 166

Chapter 6 Modal analysis of spectral properties on unstructured meshes 171

6.1 Introduction . . . 172
6.2 Modal analysis method on unstructured grids . . . 174
6.2.1 Basics of dispersion error analysis . . . 174
6.2.2 General method on unstructured grids . . . 176
6.2.3 One-dimensional numerical dispersion relation and grid cutoff . . . 179
6.3 DG Spectral convergence for dispersive waves . . . 181
6.3.1 f-plane approximation on structured grids . . . 183
6.3.2 Structured and unstructured meshes with variable Coriolis parameter . . . 187
6.4 Comparison with the mixed $P_1^{NC} - P_1$ approach . . . 196

6.4.1 Discrete formulation using the mixed $P_1^{NC} - P_1$ formulation 196
6.4.2 Dispersion errors for the $P_1^{NC} - P_1$ pair 199
6.4.3 Influence of the subgrid scale viscosity on the $P_1^{NC} - P_1$ pair 207
6.5 Conclusions 208

Chapter 7 Conclusions and Perspectives **211**

Chapter 1

Introduction

"In short, the ocean is a fundamental part of the Earth system and mankind cannot afford to function without an improved knowledge of the ocean in all of its disciplines"

"Descriptive Physical Oceanography", G.L. Pickard and W.J. Emery.

The ocean has a large impact on human activity. First of all, it plays a crucial role on the Earth climate system. Indeed, the ocean covers about 70 % of the Earth's surface and stores huge amounts of energy in the form of heat. The high heat capacity of water induces a large thermic inertia while high salt concentrations generated by the creation of sea-ice contributes to the thermohaline circulation. These processes involve long memory and response time, making of the ocean a large buffer, attenuating important climate variations. The ocean also contains a huge amount of CO_2, about 60 times superior to the atmosphere level. Some argue that changes in the ocean's circulation would turn the ocean into a source of CO_2 rather than a sink, drastically increasing the CO_2 level in the atmosphere.
Moreover, about half of mankind lives in coastal regions, the ocean providing food, oil, recreation, energy, etc. Economic development leads to increasing harmful effects on the marine coastal environment. The continuous modifications of water movements and currents should be taken into account in coastal areas management issues such as the choice of new constructions sites or the study of the impact of wastes or pollutants on the surrounding ecological systems. Understanding the ocean dynamics is thus mandatory in many fields of human activities. How-

ever, its large space and time scales lead to a poor amount of data and a poor knowledge of the ocean system and of its interactions with other climate components as sea-ice or atmosphere.
Theoretical considerations and data collections become nowadays insufficient to achieve a better understanding of the Earth system. Mathematical models are now necessary to understand the Earth climate, to predict its evolution, and to optimize the human impact on our environment.

1.1 Ocean Modeling: from finite differences to unstructured meshes

An ocean model first comprises differential equations of fluid mechanics, describing the hydrodynamics. Assumptions are already made at this point, eliminating some irrelevant processes for ocean global circulation, as the propagation of sound waves. These equations are then discretized to be solved numerically, providing a set of algebraic equations. The relevance and accuracy of the results will be determined in a large part by the choice of the numerical method (finite differences, finite volumes, finite elements, etc.) and the choice of the discretization, e.g. the kind of grid, the choice of vertical coordinates, the discretization of unknown fields, etc. These two components are considered in this work. However, it must be noticed that an ocean model contains many more components. A key component is for instance the data assimilation, accounting in a large part for the relevance and interpretation of the results. Let us also mention the set of parameterizations introduced for modeling the turbulence, the subgrid scale processes or the possible interactions with sea-ice, atmosphere, etc. The calibration of these parameterizations has a significant impact on the final results, and also depends on the numerical techniques and the grid resolution.

The first ocean general circulation model (OGCM) has been developed in the late sixties by Kirk Bryan at the Geophysical Fluid Dynamics Laboratory (Princeton, New-Jersey) [Bryan and Cox, 1967, Bryan, 1969] and was based on structured grids using finite differences. The first models based on this technology were reaching a grid resolution of several

hundreds kilometers. Such coarse grids did not allow one to solve hydrodynamic instability processes and to represent meso-scale eddies. Improvements in computers technology and parallel computing coupled to some simplifications of the physics have led in the seventies to grid resolutions of severals tenths of kilometers, allowing one to observe meso-scale eddies (see for instance the NCAR model [Holland, 1978, McWilliams and Gent, 1978, 1980]) and their large role in the ocean's kinetic energy. Many improvements in the fields of parametrization of subgrid scale processes as vertical turbulence [Mellor and Yamada, 1982] or isopycnal mixing [Gent and McWilliams, 1990], or in hybrid vertical coordinates [Bleck, 1978], data assimilation or parallel computing have yield a few efficient, modular and widely used OGCM [Griffies et al., 2000] such as MICOM (Miami Isopycnic Coordinate Ocean Model), HYCOM (HYbrid Coordinate Ocean Model), OPA (Océan PArallélisé), MITgcm (Massachusetts Institute of Technology Global Circulation Model), POM (Princeton Ocean Model) or MOM (Modular Ocean Model). Although these models are different in terms of parametrization, they share a common basis, the use of structured grids.

Many improvements in the fields of engineering have been achieved since three decades with the developments of new numerical methods based on unstructured meshes. Therefore, some components of those first OGCM's may now seem out of date, and especially the use of structured grids leading to lack of flexibility in terms of spatial resolution.
A first issue when considering structured grids of the world ocean is the coordinate system. Classical transformations as the longitude-latitude coordinates, or the stereographic projection, generate singular points at the poles, leading to instabilities [Williamson, 1979]. Although solutions have been developed to avoid those issues, a more elegant solution is clearly obtained by considering unstructured grids. The other major drawback of structured grids is clearly its rigidness for both controlling the spatial resolution and representing complex geometries. Indeed, the staircase representation of coastlines leads to spurious oscillations and additional stresses in the boundary currents. The rigidness also leads to huge difficulties for refining the spatial resolution in specific regions, and thus to represent accurately equatorial dynamics, western boundary currents, meso-scales eddies or continental slopes with relatively reasonable computational costs.

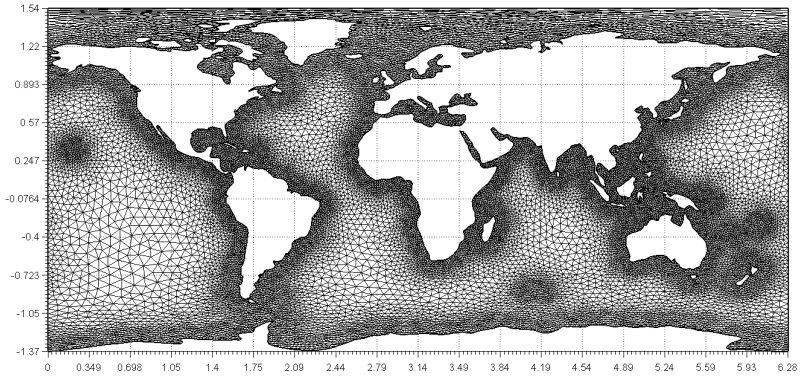


Figure 1.1: Mesh of the world ocean in geographic coordinates. The anisotropic mesh in the stereographic metric leads to isotropic mesh in geometric coordinates. Courtesy of Jonathan Lambrechts [Lambrechts et al., 2008b].

Therefore, new *second generation* OGCM's are now under study, with the aim of taking advantage of the potential of modern numerical techniques such as finite elements. Indeed, the ocean presents a large range of time and space scales, ranging from 1 km and a few hours for internal gravity waves to thousands of kilometers and years for the thermohaline circulation. The small and large scales clearly experience strong interactions. It is thus mandatory to extend the range of resolved scales. The use of unstructured grids thus clearly appears as a natural choice.
The first finite element formulation for ocean modeling has been proposed by Fix in 1975 [Fix, 1975]. Those first developments were based on the derivation of a wave equation to compute the free-surface elevation, coupled to the primitive momentum equation. The same discretization was thus used for both elevation and velocity, without generating any spurious elevation mode. Unfortunately, this approach is known to suffer from advective instabilities. Moreover, it does not satisfy the continuity in a discrete sense, leading to possible conservation issues especially for long time computations as climate applications. Other approaches

were then developed, using mixed finite element formulations to solve the primitive shallow water equations without spurious oscillations [Hua and Thomasset, 1984]. This new trend has known many improvements during the last decade [Le Roux et al., 1998, 2000, Le Roux, 2001, Hanert et al., 2004, Walters, 2006], with new projects of finite element global ocean models as:

- FEOM (Finite Element Ocean circulation Model) developed at the Alfred Wegener Institute for Polar and Marine Research (Bremerhaven, Germany), based on stabilized linear continuous finite elements to avoid oscillations [Danilov et al., 2004, Wang et al., 2008].
- ICOM (Imperial College Ocean Model) at the Imperial College (London, United Kingdom), based on a three-dimensional non-hydrostatic formulation on tetrahedrons [Ford et al., 2004b,a, Pain et al., 2005].
- SLIM (Second-generation Louvain-la-Neuve Ice-ocean Model) at the Université catholique de Louvain (Louvain-la-Neuve, Belgium) [White and Deleersnijder, 2007, White et al., 2008, Lambrechts et al., 2008b, Comblen et al., 2008], based on the idea of the mixed $P_1^{NC} - P_1$ finite element pair.

Besides the finite element (FE) method, finite volumes (FV) have also been recently applied to oceanic computations on unstructured grids [Casulli and Walters, 2000, Stuhne and Peltier, 2006, Adcroft et al., 2004]. Let us cite for instance the SUNTANS model (Stanford Unstructured Nonhydrostatic Terrain-following Adaptive Navier-Stokes Simulator) developed at the Stanford University [Fringer et al., 2006], or the model developed by the research team of Prof. Julie Pietrzak at Delft University of Technology [Ham et al., 2005]. Finite volume methods present interesting properties of local conservation and allow one to introduce accurate advection schemes. Therefore, finite volumes can be seen as better suited than FE for ocean modeling dominated by advection processes, although presenting less flexibility for the choice of the interpolants.
Finally, attempts have been made to use spectral elements [Ma, 1993, Boyd, 1998] as in the SEOM model (Spectral Element Ocean Model) [Iskandarani et al., 1995, 2003, Choi et al., 2004]. Indeed, the high-order representation of the fields turns out to be very efficient in applications involving smooth domains and forcings. Yet, the method seems

to rapidly suffer from Gibbs oscillations at high-order when considering irregular geometries. Although very efficient for atmospheric computations, the attention of the unstructured ocean modeling community is now focused on FV and FE methods.

1.2 Motivation for this thesis

This thesis is carried out in the framework of the SLIM project, with the aim of building a second generation ocean model based on unstructured grids and finite elements. Besides the classical continuous FE methods and its various mixed formulations derived the last decade, another popular new trend in engineering applications is the discontinuous Galerkin (DG) method, i.e. discontinuous finite elements presenting many interesting numerical properties, especially at high polynomial orders. The motivation for this thesis is therefore to investigate the use of the high-order DG method for ocean modeling.

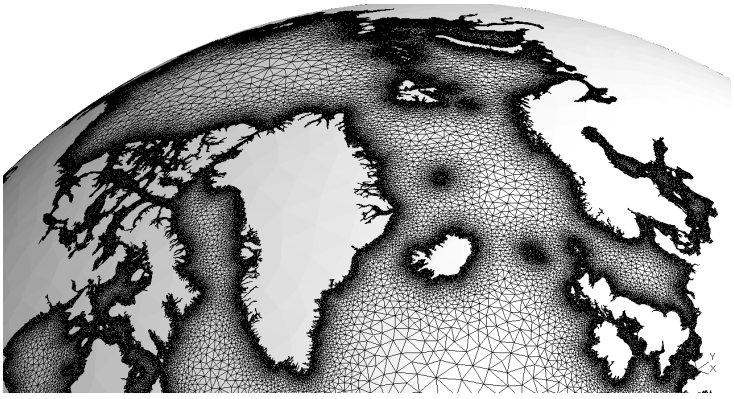


Figure 1.2: High-resolution mesh of the world ocean, close-up view of the Greenland. Courtesy of Jonathan Lambrechts.

The SLIM project started at the beginning of 2000, with the development of first essential components as the generation of unstructured two-dimensional meshes, possibly anisotropic, of the world ocean

[Legrand et al., 2000, 2006, 2007]. In order to solve the primitive equations while avoiding spurious oscillations, a second component has been the choice of a suitable FE pair for a mixed formulation of the shallow water equations [Hanert et al., 2002]. Based on these components, the SLIM project then experienced some expansion. Some key points of the development of the SLIM model are:

- the efficient **generation of unstructured grids of the world ocean**: the mesh generation is now coupled to the freely-available mesh generator GMSH[1] developed by Christophe Geuzaine and Jean-François Remacle [Geuzaine and Remacle, 2008], as described in [Lambrechts et al., 2008a]. A parametric domain is meshed accordingly to a metric based on the stereographic projection (see Figure (1.1)). Several criterions are taken into account such as the coastlines proximity, the gravity waves velocity or the bathymetry variations. However, optimal triangular meshes in terms of elements shapes are not always sufficient when considering really high-order polynomials with the DG method.

- the use of **suitable finite elements**: the $P_1^{NC} - P_1$ FE pair eventually appeared as the more appropriate choice. This pair, first used to build a noise-free FE two-layer shallow water model [Hua and Thomasset, 1984], provides a continuous linear representation of the water elevation while the velocity is a discontinuous piecewise linear field except at the mid-edges. This pair became popular because it clearly exhibits the best numerical properties in terms of dispersion error and computational cost among the set of element pairs which do not suffer from elevation modes (i.e. the $P_2 - P_2$, MINI, $P_1^{NC} - P_1$ or $P_1 iso P_2 - P_1$ pairs) [Le Roux et al., 2007, Le Roux and Pouliot, 2008].
  However, the dispersion analysis of these pairs were performed on structured grids. But in the context of ocean modeling on unstructured meshes, it is mandatory to analyze the impact of unstructured meshes on the dispersion properties. Besides, the DG method does not seem to present the usual issues of spurious oscillations, and preliminary studies on scalar advection equations have revealed promising dispersion properties.

[1] www.geuz.org/gmsh

- the use of **accurate advection schemes**: if FV methods are designed for advection, classical FE were firstly used for elliptic problems. Computing hyperbolic processes with FE clearly requires the introduction of dissipation to stabilize the scheme. This can be efficiently achieved with the $P_1^{NC} - P_1$ by performing upwinding on the semi-discontinuous velocity. However, the DG method seems to be a more natural choice in this framework since the method was designed for advection, allowing one to introduce efficient advection schemes at the discontinuities as in FV methods.

- the development of efficient **adaptation strategies**: dynamic mesh adaptation is mandatory for many multi-scale applications, especially in coastal processes analysis. The DG method also presents a large potential for this last key point, especially as mesh adaptation appears to be much more complicated using mixed formulations as the $P_1^{NC} - P_1$ pair. Indeed, the DG method provides an efficient error estimator, the jumps at inter-element discontinuities. Moreover, the local nature of the method and the use of the same discretization for both elevation and velocity leads to very efficient adaptation strategies in terms of computational costs and conservation properties.

Therefore, according to these key points, the DG method *a priori* seems to present a large potential for ocean modeling.
The method has been applied since a few years to oceanic, hydrologic or atmospheric applications [Schwanenberg and Kongeter, 2000, Aizinger and Dawson, 2002, Giraldo et al., 2002, Dawson and Proft, 2004, Nair et al., 2005, Kubatko et al., 2006, Giraldo, 2006, Remacle et al., 2006, Aizinger and Dawson, 2007]. The DG method can be considered as combining the advantages of the FV methods and the FE methods. Indeed, it provides accurate advection schemes, the possibility to increase the resolution by local modifications of the interpolation order inside each element, and the mathematical framework of Galerkin procedures for building error estimators. Moreover, the DG method presents efficient superconvergence properties of the discretization, dispersion and dissipation errors, and reveals therefore very efficient when using *really*

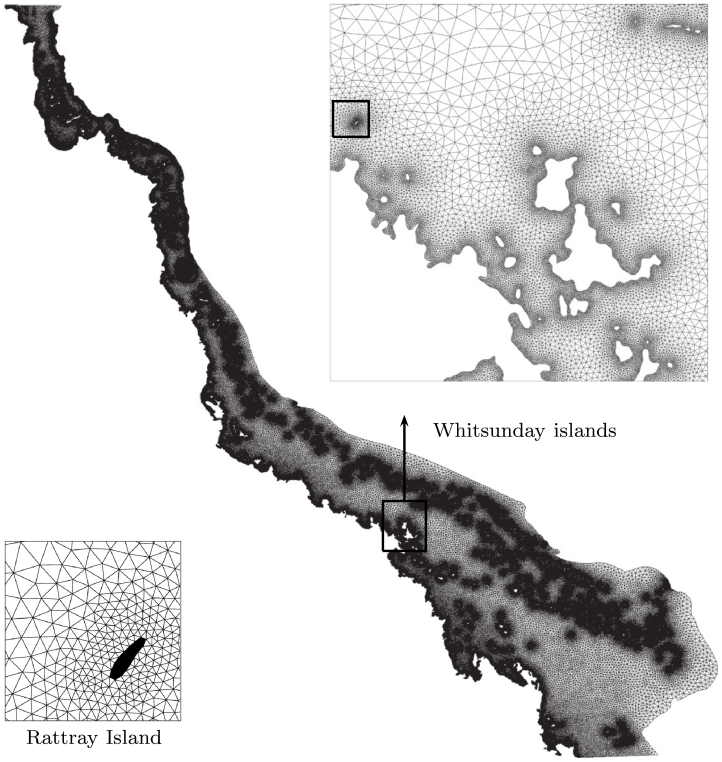


Figure 1.3: Mesh of the Great Barrier Reef with 850843 triangles and embedded close-up view of the Whitsunday islands and Rattray island. The element size is ranging between 150 m and 10 km. Courtesy of Jonathan Lambrechts [Lambrechts et al., 2008b].

high-order polynomials. Yet, the DG method may also present some drawbacks of standard spectral elements, as spurious oscillations in very irregular geometries, when using sufficiently high-order polynomials. Indeed, many works mentioned above are restricted to the use of low orders, typically until second order, when considering the typical irregular geometries and topographies of ocean.

The purpose of this research work is thus to study the potential of the DG method for oceanic applications, and in particular to investigate some key points of ocean modeling as the advection processes, the dispersion and dissipation properties or the adaptation strategies, with polynomial orders as high as possible in order to benefit from the full potential of the DG method.

1.3 Structure of the thesis

In **Chapter 2** are presented the general governing equations for the computation of large-scale oceanic flows in a rotating framework and the corresponding common assumptions. The two-dimensional shallow water equations solved in this work are then derived. The last section of the chapter considers the linearized form of this set of equations in order to obtain the classical approximate dispersion relations corresponding to shallow water waves. A semi-analytical approach is finally proposed to evaluate the exact dispersion relations of the dispersive Poincaré and Rossby waves. These reference relations will be latter compared to the numerical DG and $P_1^{NC} - P_1$ dispersion relations.
The DG method is presented in **Chapter 3** using a simple one-dimensional approach, to illustrate as simply as possible its interesting properties of advection and polynomial convergence. An analytical benchmark including Poincaré waves is performed to compute the convergence rates of discretization errors and to compare the computational times for several polynomial orders. This one-dimensional approach is also used to clearly explain the underlying ideas of the Riemann and Roe solvers for the linear and non-linear shallow water equations, respectively.
Chapter 4 deals with the mesh adaptation strategies. After a brief reminder of the standard approaches for the error estimation and the

creation of a new optimal mesh, the developed approach using the DG method is explained. In particular, the link between discretization error and jumps is considered, as well as the creation of new metric fields for both $p-$ and $h-$adaptivity. The strategy is first validated on the simple analytical Stommel model in a square domain to illustrate the efficiency of the approach using different polynomial orders. Effectivity index and adaptive convergence rates are computed and discussed. A more realistic computation in the Gulf of Mexico is finally performed. The mesh is adapted during the propagation of a typical anticyclonic eddy using a reduced gravity model.

Chapter 5 is devoted to some geometrical aspects and their influence on the high-order polynomials with the DG method.

In the first section a method for the treatment of irregular boundaries is proposed , with high-order polynomials and coarse meshes. Indeed, it has been shown that spurious oscillations may appear with high-order DG method around irregular bluff bodies when considering a linear representation of the obstacles. Ocean computations involve many coasts, reefs and islands, whose geometry cannot be represented as accurately as in classical engineering computations around cylinders or airfoils. A method is then proposed using curved triangles to represent coastlines. This method is applied to the computation of tidal flows around the Rattray island using very coarse meshes. It is shown that the method proposed avoids oscillations, but also eliminates the possible spurious creation of vorticity in the wake of the island. The numerical DG velocity field is finally compared to in-situ current measurements.

The second part of Chapter 5 is devoted to an alternative approach for computing conservation laws on general curved manifolds, using the DG method. The vectorial fields are defined using a local tangent basis, while the flux functions are still written in the classical three-dimensional basis. The DG method is particularly well suited for this approach, combining the advantages of both classical standard approaches described in the literature. The method is validated on some usual atmospheric benchmarks including some Williamson test cases. The robustness and generality of the method are tested on very irregular non-conventional manifolds.

In **Chapter 6** a general method is proposed for analyzing the dispersion and dissipation properties of any numerical scheme on any mesh, possibly unstructured. A fair comparison between numerical methods is

therefore performed. This technique is applied to the DG method for the two-dimensional shallow water equations with dispersive waves. The theoretical superconvergence properties for the one-dimensional advection equation are obtained for the 2D dispersive shallow water waves, on both structured and unstructured meshes. The method is finally applied to the analysis of the $P_1^{NC} - P_1$ pair. It is shown that this pair is very sensitive to the mesh. While theoretical rates are obtained on structured grids, the rates rapidly decrease on slightly distorted meshes. The convergence rate is finally highly reduced on unstructured grids. Conclusions and perspectives are eventually proposed in **Chapter 7**.

1.4 Supporting publications

The work undertook during this doctoral research has led to the following refereed publications:

1. P.-E. Bernard, N. Chevaugeon, V. Legat, E. Deleersnijder, J.-F. Remacle (2007). *High-order h-adaptive discontinuous Galerkin methods for ocean modeling*, Ocean Dynamics, 57, 109–121.

2. P.-E. Bernard, E. Deleersnijder, J.-F. Remacle, V. Legat (2008). *Dispersion analysis of discontinuous Galerkin schemes applied to Poincaré, Kelvin and Rossby waves*, Journal of Scientific Computing, 34, 26–47.

3. P.-E.Bernard, J.-F.Remacle, V.Legat (2008). *Boundary discretization for high order discontinuous Galerkin computations of tidal flows around shallow water islands*, International Journal for Numerical Methods in Fluids (in press), doi:10.1002/fld.1831.

4. P.-E.Bernard, J.-F.Remacle, V.Legat (2008). *Modal analysis on unstructured meshes of the dispersion properties of the $P_1^{NC} - P_1$ pair*, Ocean Modelling, (in press), doi:10.1016/j.ocemod.2008.03.005.

5. P.-E.Bernard, J.-F.Remacle, R. Comblen, V.Legat (2008). *High-order discontinuous Galerkin schemes on general 2D manifolds*, Journal of Computational Physics, (submitted).

6. P.-E.Bernard, J.-F.Remacle, V.Legat (2008). *A high-order discontinuous Galerkin approach for solving the shallow water equations*

on curved domains, Journal of Computational and Applied Mathematics, (submitted).

7. J.Lambrechts, E.Hanert, E.Deleersnijder, P.-E.Bernard, V.Legat, J.-F.Remacle, E.Wolanski (2008). *A multiscale model of the whole Great Barrier Reef Hydrodynamics*, Estuarine, Coastal and Shelf Science (in press), doi:10.1016/j.ecss.2008.03.016.

Chapter 2

Governing equations: the shallow water equations

Contents

2.1 General equations in a rotating framework . 16

2.2 The Shallow Water Equations 22

2.3 Linear barotropic shallow water waves 25

2.3.1 Dispersion relations and classical approximations 27

2.3.2 Exact dispersion relations 32

This chapter presents the general governing equations of large scale oceanic flows in a rotating framework and the main corresponding assumptions. These equations are then derived to obtain the classical shallow water equations, describing the flow of a thin layer of incompressible fluid under the influence of a gravitational force without stratification. These equations can be considered as the long wave limit of Euler's equations of inviscid fluid dynamics where the wave lengths are much larger than the water depth.
Finally, the shallow water equations are linearized for the analysis of the linear barotropic waves, mainly the Poincaré, Rossby and Kelvin waves. Classical approximate analytical dispersion relations are derived as the exact reference dispersion relations mandatory for comparing with numerical dispersion relations from the discontinuous Galerkin method.

2.1 General equations in a rotating framework

Let us consider the general fluid mechanics conservation laws in the rotating framework depicted in Figure (2.1):

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{v}) = 0 \tag{2.1}$$

$$\frac{\partial \rho \boldsymbol{v}}{\partial t} + \nabla \cdot (\rho \boldsymbol{v}\boldsymbol{v}) + \rho \boldsymbol{a} = \nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{g} \tag{2.2}$$

where ρ and $\boldsymbol{v} = (u, v, w)$ denote the unknown fluid density and velocity fields, respectively, expressed in a local cartesian non-inertial basis $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$, t is the time, $\boldsymbol{g}$ represents the gravitational acceleration and $\boldsymbol{\sigma}$ denotes the stress tensor. Computations in a non-inertial rotating framework eventually requires the use of the additional acceleration term $\boldsymbol{a}$, detailed further.
The stress tensor is classically decomposed into an isotropic term due to the pressure and a viscous stress tensor $\boldsymbol{\tau}$:

$$\boldsymbol{\sigma} = -p\boldsymbol{\delta} + \boldsymbol{\tau} \tag{2.3}$$

with p the fluid pressure and $\boldsymbol{\delta}$ the Kronecker tensor. Let us now develop these terms and introduce the usual approximations.

Non-Inertial framework

The acceleration term on a rotating sphere can be written as the sum of a Coriolis and a centrifugal accelerations, $\boldsymbol{a}_c$ and $\boldsymbol{a}_e$ respectively:

$$\boldsymbol{a} = \boldsymbol{a}_c + \boldsymbol{a}_e = 2\boldsymbol{\Omega} \times \boldsymbol{v} + \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{x}) \tag{2.4}$$

with Ω the framework rotation vector and $\boldsymbol{x}$ the position vector on the surface. For the Earth parameters, with a radius $R_E = 6.37122\ 10^6$ km, a rotation rate $\Omega_E = 7.292\ 10^{-5}$ rad s^{-1} and a gravitational acceleration $g = 9.81$ m s^{-2}, the maximum centrifugal acceleration value accounts for about 0.3 percent of the Earth gravity and is directed along the normal direction to the Earth's rotation axis. Because of the slightly flattened shape of the Earth, the gravitational acceleration is usually not aligned with the local vertical direction, since it points to the center of the

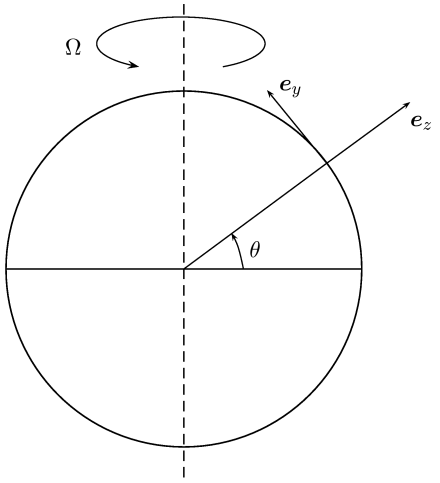


Figure 2.1: Rotating framework of the Earth and its local non-inertial cartesian basis $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$.

Earth. However, the flattened shape of the Earth is precisely due to the centrifugal acceleration. This implies that the sum of the local gravity acceleration and the centrifugal acceleration results in a vector whose direction is precisely the local vertical direction. Therefore, the centrifugal acceleration is commonly neglected while the gravity direction is assumed to be aligned with the local normal to the Earth's surface. The Coriolis acceleration is usually decomposed in two parts:

$$\begin{aligned} 2\boldsymbol{\Omega} \times \boldsymbol{v} &= 2\Omega\cos(\theta)\boldsymbol{e}_y \times \boldsymbol{v} + 2\Omega\sin(\theta)\boldsymbol{e}_z \times \boldsymbol{v} \\ &= f_*\boldsymbol{e}_y \times \boldsymbol{v} + f\boldsymbol{e}_z \times \boldsymbol{v} \end{aligned}$$

where f is called the Coriolis parameter while f_* is the reciprocal Coriolis parameter. The Coriolis parameter is positive in the northern hemisphere and negative in the southern hemisphere with extremum values at the poles while the reciprocal Coriolis parameter is defined positive everywhere but vanishes at the poles. The influence of f_* is generally neglected compared to f. Indeed, the vectorial product

$$2\boldsymbol{\Omega} \times \boldsymbol{v} = \begin{pmatrix} f_* w - f v \\ f u \\ -f_* u \end{pmatrix}$$

yields two terms in the local longitudinal direction, $f_* w - f v$, where the horizontal velocity w is clearly several orders of magnitude slower than the vertical components. In the latitude direction, only f remains in the term fu. Considering the momentum equation in the local vertical direction, an analysis of the amplitude of the different terms reveals that the Coriolis term $f_* u$ is about 10^{-3} times smaller than the vertical pressure term $\frac{1}{\rho}\frac{\partial p}{\partial z}$ [Cushman-Roisin, 1994]. The impact of the reciprocal Coriolis parameter is therefore usually neglected.

Newtonian fluid assumption

The deformation rate tensor $\boldsymbol{d}$ is defined as:

$$\boldsymbol{d} = \frac{1}{2}\left(\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T\right) \tag{2.5}$$

and can be decomposed into so-called spherical and deviatory parts, $\boldsymbol{d}^s$ and $\boldsymbol{d}^d$ respectively:

$$\begin{aligned} \boldsymbol{d} &= \boldsymbol{d}^s + \boldsymbol{d}^d \ , \\ \boldsymbol{d}^s &= (\boldsymbol{\delta} : \boldsymbol{d})\,\frac{\boldsymbol{\delta}}{3} \ , \\ \boldsymbol{d}^d &= \boldsymbol{d} - \boldsymbol{d}^s \ . \end{aligned}$$

The spherical part is a measure of the dilatation rate while the deviatory part measures the shear rate. Notice that the spherical contribution only depends on the fluid compressibility, i.e. on the trace of the deformation tensor, since

$$(\boldsymbol{\delta} : \boldsymbol{d}) = \mathrm{tr}(\boldsymbol{d}) = \nabla \cdot \boldsymbol{v}.$$

The Newtonian fluid assumption expresses the tensor of viscous stresses as proportional to the dilatation and shear rates:

$$\boldsymbol{\tau} = 3\kappa(p,T)\boldsymbol{d}^s + 2\mu(p,T)\boldsymbol{d}^d \ , \tag{2.6}$$

with T the fluid temperature, κ and μ denoting the so-called volume and molecular dynamic viscosities respectively.

Boussinesq approximation and hydrostatic pressure

The Boussinesq approximation consists in neglecting the fluid density variations $\hat{\rho}$ in all terms,

$$\rho(\boldsymbol{x},t) = \rho_0 + \hat{\rho}(\boldsymbol{x},t) \cong \rho_0 \ ,$$

except in the gravitational term. This approximation is somehow equivalent to consider a variable gravity depending on the fluid density. This assumption leads to consider the incompressibility of the flow since the continuity equation is reduced to a zero divergence velocity. Therefore, every process associated to the fluid compressibility is eliminated from the system as for instance the propagation of sound waves. Finally, the zero divergence velocity involves further simplifications in the Newtonian fluid model of the viscous stresses (2.6) which read:

$$\boldsymbol{\tau} = 2\mu(p,T)\boldsymbol{d} \ . \tag{2.7}$$

The continuity and momentum equations (2.1) and (2.2) then read:

$$\nabla \cdot \boldsymbol{v} = 0 \tag{2.8}$$

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} + f\boldsymbol{e}_z \times \boldsymbol{v} = -\frac{1}{\rho_0}\nabla p + \nabla \cdot (2\nu \boldsymbol{d}) + \frac{\rho}{\rho_0}\boldsymbol{g} \tag{2.9}$$

Further simplifications can be obtained by analyzing the orders of magnitude of the terms in the $z-$ component of momentum equation (2.9) [Cushman-Roisin, 1994]. The ocean is characterized by its large horizontal scale compared to its small vertical one. The typical horizontal length is about $L = 1000$ km $= 10^6$ m, while the typical vertical depth is $H = 10^3$ m. Introducing the typical horizontal velocity $U \simeq 0.1$ ms^{-1} into the continuity equation (2.8), we obtain the following order of magnitude for the vertical velocity:

$$W \simeq \frac{H}{L}U \simeq 10^{-4} \text{ ms}^{-1} \ll U \ .$$

Considering large scale flows, the time scale is long enough to experience the Earth's rotation. Let us chose a typical time scale of about two weeks, $T = 10^7$ s. Finally, the horizontal viscosity is neglected for sake

of simplicity while the typical maximum vertical turbulent viscosity is chosen as $\nu_v \simeq 10^{-1}$ m^{2}s^{-1}. Therefore, the orders of magnitude of the terms for the vertical momentum equation are computed as:

$$\underbrace{\frac{\partial w}{\partial t}}_{\frac{W}{T}} + \underbrace{u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y}}_{\frac{UW}{L}} + \underbrace{w\frac{\partial w}{\partial z}}_{\frac{WW}{H}} = -\frac{1}{\rho_0}\frac{\partial p}{\partial z} - \underbrace{\frac{\rho}{\rho_0}g}_{g} + \underbrace{\frac{\partial}{\partial z}\left(\nu_v \frac{\partial w}{\partial z}\right)}_{\frac{\nu_v W}{H^2}} .$$

For the typical values of the ocean, the viscous term and the left hand side terms are of the order of $\mathcal{O}(10^{-11})$, while the gravity term is clearly of the order $\mathcal{O}(1)$. Therefore, the vertical momentum equation is usually simplified into the so-called hydrostatic approximation:

$$\frac{\partial p}{\partial z} = -\rho g \ . \tag{2.10}$$

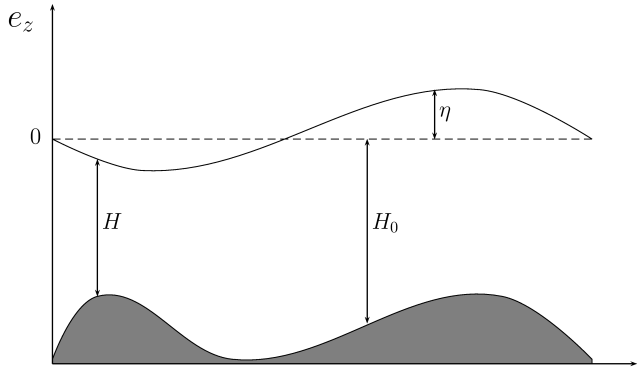


Figure 2.2: Definition of the water depth $H(x, y, t)$, bathymetry $H_0(x, y)$ and elevation $\eta(x, y, t)$

The horizontal pressure gradient is then obtained by solving the hydrostatic equation (2.10). Let us consider the relative elevation of the free surface of the ocean $\eta(x, y)$ depicted in Figure (2.2) defined as:

$$H(x, y, t) = H_0(x, y) + \eta(x, y, t) \tag{2.11}$$

with $H_0(x, y)$ the depth of the ocean and $H(x, y, t)$ the total depth of the water column. The integration of the pressure along the vertical dimension leads to the pressure:

$$p = p_{atm} + \int_z^\eta \rho g dz \tag{2.12}$$

$$\cong p_{atm} + \rho g \eta + \int_z^0 \rho g dz \tag{2.13}$$

with p_{atm} the atmospheric pressure at the ocean surface.
Considering eventually a constant atmospheric pressure and a constant water density, the horizontal pressure gradient reads:

$$\nabla_h p = \rho_0 g \nabla_h \eta \ , \tag{2.14}$$

where ∇_h denotes the horizontal gradient operator.

Resulting governing equations

With the previous assumptions, the continuity and momentum equations read:

$$\frac{\partial w}{\partial z} + \nabla_h \cdot \boldsymbol{u} = 0 \ , \tag{2.15}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla_h \boldsymbol{u} + w \frac{\partial \boldsymbol{u}}{\partial z} + f \boldsymbol{e}_z \times \boldsymbol{u} = -g \nabla_h \eta + \nabla_h \cdot (\nu_h \nabla_h \boldsymbol{u}) + \frac{\partial}{\partial z} \left(\nu_v \frac{\partial \boldsymbol{u}}{\partial z} \right) \tag{2.16}$$

with $\boldsymbol{u} = (u, v)$ the horizontal velocity.
Impermeability boundary conditions, also called kinematic boundary conditions, at the ocean bottom Γ_b and ocean surface Γ_s read:

$$w = \frac{DH_0}{Dt} = -\boldsymbol{u}_b \cdot \nabla_h H_0 \qquad \text{on } \Gamma_b \ , \tag{2.17}$$

$$w = \frac{D\eta}{Dt} = \frac{\partial \eta}{\partial t} + \boldsymbol{u}_s \cdot \nabla_h \eta \qquad \text{on } \Gamma_s \ , \tag{2.18}$$

where $\boldsymbol{u}_s$ and $\boldsymbol{u}_b$ denotes the surface and bottom velocities respectively. This set of equations possibly coupled to turbulence closure schemes constitutes the basic set of equations for solving large scale oceanic flows.

2.2 The Shallow Water Equations

The shallow water equations are two-dimensional equations derived from the governing equations (2.15) and (2.16) by integrating over the vertical dimension. Let us define the depth-averaged two-dimensional velocity $\bar{\mathbf{u}} \in \mathcal{R}^2$:

$$H\bar{\mathbf{u}} = \int_{-H_0}^{\eta} \boldsymbol{u}(x, y, z)\ dz\ . \tag{2.19}$$

Integrating over the depth

The shallow water continuity equation is obtained by integrating (2.15) over the fluid depth with the impermeability boundary conditions (2.17) and (2.18) and using the Leibniz theorem:

$$\begin{aligned}
\int_{-H_0}^{\eta} \left(\frac{\partial w}{\partial z} + \nabla_h \cdot \boldsymbol{u} \right)\ dz &= [w]_{-H_0}^{\eta} + \int_{-H_0}^{\eta} \nabla_h \cdot \boldsymbol{u}\ dz \\
&= \frac{\partial \eta}{\partial t} + \boldsymbol{u}_s \cdot \nabla_h \eta + \boldsymbol{u}_b \cdot \nabla_h H_0 \\
&\quad + \nabla_h \cdot (H\bar{\mathbf{u}}) - \nabla_h \eta \cdot \boldsymbol{u}_s - \nabla_h H_0 \cdot \boldsymbol{u}_b \\
&= \frac{\partial \eta}{\partial t} + \nabla_h \cdot (H\bar{\mathbf{u}})\ .
\end{aligned}$$

The convective terms of the momentum equation are rewritten by introducing the continuity equation (2.15):

$$\boldsymbol{u} \cdot \nabla_h \boldsymbol{u} + w \frac{\partial \boldsymbol{u}}{\partial z} = \nabla_h \cdot (\boldsymbol{u}\boldsymbol{u}) + \frac{\partial w \boldsymbol{u}}{\partial z}\ .$$

Integrating the Lagrangian derivative of the momentum equation with impermeability leads to:

$$\begin{aligned}
\int_{-H_0}^{\eta} \left(\frac{\partial \boldsymbol{u}}{\partial t} + \nabla_h \cdot (\boldsymbol{u}\boldsymbol{u}) + \frac{\partial w \boldsymbol{u}}{\partial z} \right)\ dz& \\
= \frac{\partial H\bar{\mathbf{u}}}{\partial t} - \frac{\partial \eta}{\partial t} \boldsymbol{u}_s + \nabla_h \cdot \int_{-H_0}^{\eta} \boldsymbol{u}\boldsymbol{u}\ dz& \\
- \boldsymbol{u}_s \boldsymbol{u}_s \cdot \nabla_h \eta - \boldsymbol{u}_b \boldsymbol{u}_b \cdot \nabla_h H_0 + [w\boldsymbol{u}]_{-H_0}^{\eta}& \\
= \frac{\partial H\bar{\mathbf{u}}}{\partial t} + \nabla_h \cdot \int_{-H_0}^{\eta} \boldsymbol{u}\boldsymbol{u}\ dz \qquad .&
\end{aligned}$$

The so-called *differential advection term* $\nabla_h \cdot \int_{-H_0}^{\eta} \boldsymbol{u}\boldsymbol{u}\, dz$ describes the lateral momentum transfer due to the variations of the velocity $\boldsymbol{u}$ compared to the depth-average $\bar{\mathbf{u}}$ [Vreugdenhill, 1994a]. This term is commonly added to the turbulent and molecular friction terms and modelized into a dissipation term.
The integration of the Coriolis, pressure gradient and vertical viscous terms yields:

$$\begin{aligned}
\int_{-H_0}^{\eta} f\boldsymbol{e}_z \times \boldsymbol{u}\, dz &= f\boldsymbol{e}_z \times H\bar{\mathbf{u}} \\
\int_{-H_0}^{\eta} g\nabla_h \eta\, dz &= gH\nabla_h \eta \\
\int_{-H_0}^{\eta} \frac{\partial}{\partial z}\left(\nu_v \frac{\partial \boldsymbol{u}}{\partial z}\right) dz &= \left[\nu_v \frac{\partial \boldsymbol{u}}{\partial z}\right]_{-H_0}^{\eta} = \frac{1}{\rho_0}\left(\boldsymbol{\tau}_s - \boldsymbol{\tau}_b\right)
\end{aligned}$$

with $\boldsymbol{\tau}_b$ the bottom friction and $\boldsymbol{\tau}_s$ the wind friction at the ocean surface.
The horizontal diffusion terms are eventually parametrized as:

$$\nabla_h \cdot (\nu_h H \nabla_h \bar{\mathbf{u}})$$

where the diffusive effect of the convective terms has been included into the horizontal diffusivity coefficient ν_h.

Conservative and non-conservative shallow water formulations

The resulting variables are depth-averaged and every spatial operators is now applied to the horizontal dimension. Therefore, notations in the following considerations will be simplified by dropping the average symbol and the h subscript.
Integration over the depth yields the so-called *conservative form* of the shallow water equations:

$$\frac{\partial H}{\partial t} + \nabla \cdot (H\boldsymbol{u}) = 0 \tag{2.20}$$

$$\frac{\partial H\boldsymbol{u}}{\partial t} + \nabla \cdot (H\boldsymbol{u}\boldsymbol{u}) + gH\nabla\eta = -f\boldsymbol{e}_z \times H\boldsymbol{u} + \nu\nabla \cdot (H\nabla\boldsymbol{u}) + \frac{\boldsymbol{\tau}_s - \boldsymbol{\tau}_b}{\rho_0} \tag{2.21}$$

where the unknown fields are usually the transport ($H\boldsymbol{u}$) and the water column depth H.
Dividing this formulation by the water depth H yields a second classical formulation called the *non-conservative form* of the shallow water equations:

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (H\boldsymbol{u}) = 0 \tag{2.22}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + g\nabla\eta = -f\boldsymbol{e}_z \times \boldsymbol{u} + \frac{\nu}{H}\nabla \cdot (H\nabla\boldsymbol{u}) + \frac{\boldsymbol{\tau}_s - \boldsymbol{\tau}_b}{\rho_0 H} \tag{2.23}$$

where the classical unknown fields are the depth-average mean velocity $\boldsymbol{u}$ and the relative water elevation η.

Although both formulations are strictly equivalent in the continuous world, their discretizations may lead to quite different discrete algebraic system unless an appropriate treatment is applied to the spatial operators. Indeed, the continuity equation (2.20) in the conservative formulation is linear considering the transport as unknown field, which is not the case in the non-conservative formulation unless linearization is applied. The advective term in (2.21) is expressed as a divergence: this formulation is then well suited for a classical discontinuous Galerkin discretization while the non-conservative expression requires special treatment. On the other hand, the pressure term remains linear in the non-conservative formulation, while it requires to be split in two parts $gH\nabla\eta = g\nabla H^2/2 - gH\nabla H_0$ in the conservative formulation, considering the water depth as unknown field.

Despite the many simplifications brought to the governing equations, those simplified shallow water equations still contain two of the main driving forces applied to oceanic flows: the Coriolis effect and the wind forcing. The shallow water equations provide good approximations of the hydrodynamic processes of the upper layer in wind-driven oceanic flows. Indeed, such flows are essentially made of horizontal water movements, as for instance the wind-driven Gulf-Stream which consists in a western boundary current transferring huge amounts of heat and energy to higher latitudes.

The shallow water equations have been used since the last decades to represent complex processes as atmospheric flows [Behrens, 1998, Thomas et al., 2001, Giraldo, 2006], tidal flows [Davies and Jones, 1991, Lambrechts et al., 2008b], river flows [Vreugdenhill, 1994b, Aizinger and Dawson, 2002], dam break and flooding problems [Soares Frazão and Zech, 2002b], coastal flows [Casulli and Zanolli, 2000, Dawson and Proft, 2004], tsunamis and planetary flows [Pietrzak et al., 2007], lake hydrodynamic processes [Gourgue et al., 2007, Naithani et al., 2003] or storm surges if a variable atmospheric pressure is taken into account.

2.3 Linear barotropic shallow water waves

This section is devoted to the analysis of the different kinds of waves associated to the linear shallow water equations and in particular to their dispersion relations which provide the wave frequency in terms of the associated wavenumbers and physical parameters.
The linear shallow water equations exhibit three kinds of waves, namely the Poincaré, Kelvin and Rossby waves. Those waves play a crucial role in ocean dynamics since even minor errors in those waves frequencies may lead to large lacks of accuracy in climate and weather predictions. It is thus mandatory for an ocean model to propagate those waves in an accurate way.

We consider the simplified square domain of size L at midlatitudes as depicted in Figure (2.3). The following developments assume that the non-linear transport terms and all dissipation mechanisms are neglected and that the bathymetry H_0 is constant. The shallow water equations (2.22) and (2.23) then read:

$$\frac{\partial \eta}{\partial t} + H_0 \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = 0, \tag{2.24}$$

$$\frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} - fv = 0, \tag{2.25}$$

$$\frac{\partial v}{\partial t} + g \frac{\partial \eta}{\partial y} + fu = 0 \tag{2.26}$$

where u and v denotes the velocity components in the $-x$ and $-y$ directions respectively. In order to simplify the following algebra, we consider

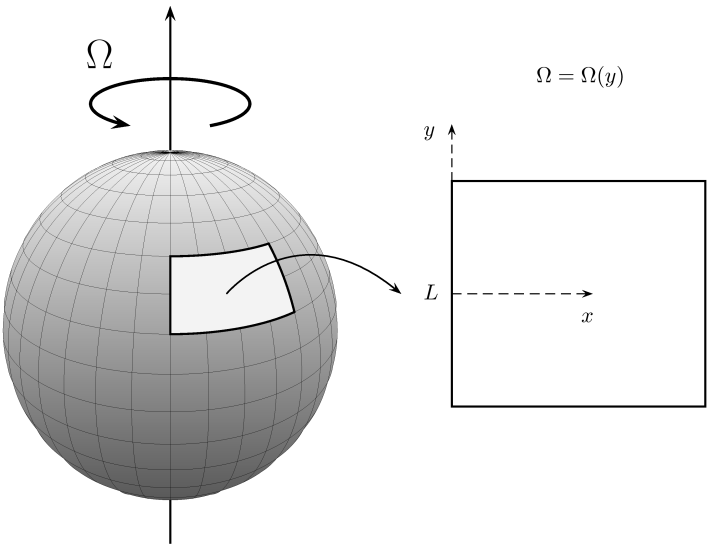


Figure 2.3: Definition of the local cartesian domain of size L at midlatitudes.

the classical β-plane approximation which consists in approaching the Coriolis parameter with a linear relationship, $f = f_0 + \beta y$ [Cushman-Roisin, 1994]. It must be noticed that such an assumption is not a limitation to the generality of the further developments and extension to the general Coriolis expression is straightforward.

2.3.1 Dispersion relations and classical approximations

The following paragraphs present the classical approximations widely used in the literature [Majda, 2003, Cushman-Roisin, 1994, Pedlosky, 1979] to obtain the approximate dispersion relations of the Poincaré, Kelvin and Rossby waves.

Poincaré or inertia-gravity waves

Poincaré waves, also called inertia-gravity waves, are relatively fast waves whose frequency is modified by the Coriolis parameter and the gravitational acceleration. Those waves are called *dispersive* waves since they experience the Earth rotation and exhibit a non-linear relationship between frequency and wavenumber. Inertia-gravity waves are believed to have only a minor influence on the large scale motions and require a less accurate computation than Kelvin and Rossby waves for modeling ocean global circulation.

The classical analysis leading to approximate Poincaré dispersion relations is based on the assumption that the unknown fields can be expressed as a solution of the Fourier type, i.e. as the real part of imaginary exponentials:

$$(\eta, u, v)\,(x, y, t) = (\tilde{\eta}, \tilde{u}, \tilde{v})\;\; e^{i(k_x x + k_y y + \omega t)} \tag{2.27}$$

where k_x and k_y are the wavenumbers in the $-x$ and $-y$ dimensions respectively, and ω is the wave frequency. Notice that this strong assumption is only valid when considering a constant Coriolis parameter, and does not allow one to represent Rossby waves. Indeed, assuming the solution to be the product of the constants $(\tilde{\eta}, \tilde{u}, \tilde{v})$ and sine functions

annihilates the dependence with the latitude, which is present in the β-plane assumption. The following results are then valid for the so-called f-plane approximation. Substituting expression (2.27) in the governing equations (2.24), (2.25) and (2.26) leads to the following system:

$$\begin{pmatrix} i\omega & iH_0k_x & iH_0k_y \\ igk_x & i\omega & -f_0 \\ igk_y & f_0 & i\omega \end{pmatrix} \begin{pmatrix} \tilde{\eta} \\ \tilde{u} \\ \tilde{v} \end{pmatrix} = \mathbf{0} \ . \tag{2.28}$$

Obtaining a non-zero solution requires the determinant of the matrix in (2.28) to be zero, which leads to the classical dispersion relations:

$$\omega = 0 \ , \tag{2.29}$$

$$\omega = \pm\sqrt{gH_0k^2 + f_0^2} \ , \tag{2.30}$$

with k the norm of the wavenumber vector $\boldsymbol{k} = (k_x, k_y)$, i.e. $k^2 = k_x^2 + k_y^2$. The first zero frequency corresponds to geostrophic modes while the two other values of ω correspond to the positive and negative Poincaré frequencies.
Notice that, considering the no rotation limit $f = 0$, the dispersion relation becomes $\omega = \sqrt{gH_0}k$ which corresponds to gravity waves. The same limit is obtained by considering very large wavenumbers, which means small wavelengths compared to the deformation radius $R = \sqrt{gH_0}/f$.

Kelvin waves

The Kelvin waves are generated by the tides and the winds and their existence requires the support of lateral boundaries. They propagate with the coasts on their right (left) in the northern (southern) hemisphere and exhibit an exponential decay away from the coastlines scaling as the deformation radius. Indeed, Kelvin waves propagating with the coast on their left cannot exist in the northern hemisphere because this solution violates finite energy principle with an exponential growth in y. The most popular example of Kelvin waves is the propagation of tidal waves in the North Sea. Indeed, a Kelvin wave coming from the Channel will propagate northeastward along the coasts of Belgium and Holland while the same wave coming from the North Sea will propagate

southward along the English coast.
In the context of general ocean circulation models, Kelvin waves are sometimes considered as having a minor impact since they transport small fractions of the total energy. However, their impact cannot be neglected when considering regional applications like coastal seas. Indeed, they are believed to carry large amounts of energy and play therefore a crucial role in mixing and dissipation processes in coastal dynamics.

The analysis of the Kelvin waves requires us to consider the existence of coasts. Let us consider for instance a coastline oriented along the east-west direction, and assume the velocity component v to be zero. The shallow water equations (2.24), (2.25) and (2.26) reads:

$$\frac{\partial \eta}{\partial t} + H_0 \frac{\partial u}{\partial x} = 0, \tag{2.31}$$

$$\frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} = 0, \tag{2.32}$$

$$g \frac{\partial \eta}{\partial y} = -fu. \tag{2.33}$$

The $-y$ component of the momentum equation (2.33) yields the geostrophic balance in the y direction. By cross differentiation, the first two equations can be written as a simple wave equation:

$$\frac{\partial^2 u}{\partial t^2} - gH_0 \frac{\partial^2 u}{\partial x^2} = 0 \ .$$

The solution is the real part of:

$$u(x, y, t) = Y(y) e^{(i(k_x x - \omega t))}$$

with the general y-dependence $Y(y)$ and the dispersion relation for the Kelvin waves reads:

$$\omega = \pm \sqrt{gH_0} k_x \ . \tag{2.34}$$

Frequencies are independent of the function $Y(y)$: this expression corresponds to a non-dispersive wave and is therefore a valid analytical relation for any value of the Coriolis parameter f. Moreover, the general solution can be written as:

$$u(x, y, t) = \exp\left(-\frac{y}{\sqrt{gH_0}} \left(f_0 + \frac{\beta}{2} y \right) \right) \exp\left(i(k_x x - \omega t) \right) \tag{2.35}$$

According to [Majda, 2003], in mid-latitudes, a good approximation is to take $f = f_0$. In this case, the structure of u is of the form:

$$u(x, y, t) = \exp\left(-y f_0/\sqrt{gH_0}\right) \exp\left(i(k_x x - \omega t)\right) \tag{2.36}$$

characterized by the Rossby radius $\sqrt{gH_0}/f_0$. Near the equator, the Coriolis force may be approximated by $f = \beta y$ and the equatorial Kelvin modes are of the form:

$$u(x, y, t) = \exp\left(-y^2 \frac{\beta}{2\sqrt{gH_0}})\right) \exp\left(i(k_x x - \omega t)\right) \tag{2.37}$$

characterized by the equatorial deformation radius $L_E^2 = \sqrt{gH_0}/\beta$.

Rossby or planetary waves

The Rossby waves, or planetary waves, may be considered as small perturbations of the geostrophic balance due to the Earth's curvature. More precisely, they are very slow waves compared to the Poincaré and Kelvin waves, propagating only westward and generated by the variability of the Coriolis parameter with the latitude. These slow waves (with a typical wave speed of a few centimeters per second) exhibit very large wavelengths, typically hundreds of kilometers, while the corresponding water elevation is of a few centimeters. They are known to intensify the western boundary currents which are responsible for the transfer of huge amount of heat and energy. It is thus mandatory for an ocean model to represent those waves accurately since minor variations in those waves propagation may affect the climate and weather over large areas. The Rossby waves may take several years to cross an entire ocean basin before affecting ocean processes located thousands of kilometers away. Such a process might have already happen in 1993 [Jacobs et al., 1994], the 1982-83 El-Niño being responsible for the creation of Rossby waves pushing the Kurishio Current northward and affecting weather ten years after.

The dispersion relation (2.29) corresponding to the geostrophic modes has been obtained by assuming a solution of the Fourier type and, as

a consequence, a f-plane approximation for the Coriolis factor. In fact, those zero-frequency modes may develop a slow evolution by introducing less restrictive assumptions and then become planetary waves.
The linearized shallow water equations under the β-plane assumption clearly present a set of large terms (the f_0, g and H_0 terms) and a set of very small terms (the time derivatives and the β terms) which can be considered as small perturbations modifying the wave evolution. As proposed in [Cushman-Roisin, 1994], an analytical approximate dispersion relation can be found by assuming that the large terms dominate in the momentum equations (2.25) and (2.26). The u and v components of the velocity then yield in first approximation the geostrophic balance:

$$\begin{aligned} u &\cong -\frac{g}{f_0}\frac{\partial \eta}{\partial y} \ , \\ v &\cong \frac{g}{f_0}\frac{\partial \eta}{\partial x} \ . \end{aligned}$$

These expressions can then be re-introduced into the *small* terms of the momentum equations, which yields:

$$\begin{aligned} -\frac{g}{f_0}\frac{\partial^2 \eta}{\partial y \partial t} - f_0 v + g\left(1 - \frac{\beta}{f_0}y\right)\frac{\partial \eta}{\partial x} = 0 \ , \\ +\frac{g}{f_0}\frac{\partial^2 \eta}{\partial x \partial t} + f_0 u + g\left(1 - \frac{\beta}{f_0}y\right)\frac{\partial \eta}{\partial y} = 0 \ , \end{aligned}$$

and the corresponding solutions read:

$$\begin{aligned} u &= -\frac{g}{f_0}\left(1 - \frac{\beta}{f_0}y\right)\frac{\partial \eta}{\partial y} - \frac{g}{f_0^2}\frac{\partial^2 \eta}{\partial x \partial t} \ , \\ v &= +\frac{g}{f_0}\left(1 - \frac{\beta}{f_0}y\right)\frac{\partial \eta}{\partial x} - \frac{g}{f_0^2}\frac{\partial^2 \eta}{\partial y \partial t} \ . \end{aligned}$$

Introducing eventually these expressions in the continuity equation (2.24) yields:

$$\frac{\partial \eta}{\partial t} - R^2 \frac{\partial}{\partial t}\nabla^2 \eta - \beta R^2 \frac{\partial \eta}{\partial x} = 0$$

with R the deformation radius $\sqrt{gH_0}/f_0$. Solving this equation with constant coefficients leads to the approximate analytical dispersion relation for the Rossby waves:

$$\omega = -\beta R^2 \frac{k_x}{1 + R^2\left(k_x^2 + k_y^2\right)} \ . \tag{2.38}$$

2.3.2 Exact dispersion relations

In the framework of this thesis, in order to analyze the ability of the discontinuous Galerkin method to represent the ocean waves in an accurate way, we will have to compare the numerical dispersion relations to the exact continuous dispersion relations for the linear shallow water equations. Although sufficiently accurate for traditional analysis, the analytical dispersion relations for the dispersive Poincaré and Rossby waves, (2.30) and (2.38) respectively, have been obtained using many approximations and are not sufficiently accurate to be compared to DG numerical dispersion relations. The purpose of the following developments is then to obtain sufficiently accurate reference dispersion relations for the dispersive waves, without any assumption.

Let us consider the simplified square domain of size L at midlatitudes as depicted in Figure (2.4). Reflecting boundary conditions are assumed on the northern and southern parts of this local domain while periodic boundary conditions are used in the east-west direction:

$$\begin{aligned} \eta(0,y) &= \eta(L,y), \\ u(0,y) &= u(L,y), \\ v(0,y) &= v(L,y), \\ v(x,-L/2) &= 0 = v(x,L/2). \end{aligned}$$

As proposed by [Longuet-Higgins, 1965], we derive a third order equation in v for the shallow water equations. By substituting mass equation (2.24) into the momentum equations (2.25) and (2.26), we obtain:

$$\begin{aligned} \frac{\partial^2 u}{\partial t^2} - gH_0\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 v}{\partial x \partial y}\right) - f\frac{\partial v}{\partial t} &= 0 \ , \\ \frac{\partial^2 v}{\partial t^2} - gH_0\left(\frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 u}{\partial x \partial y}\right) + f\frac{\partial u}{\partial t} &= 0 \ . \end{aligned}$$

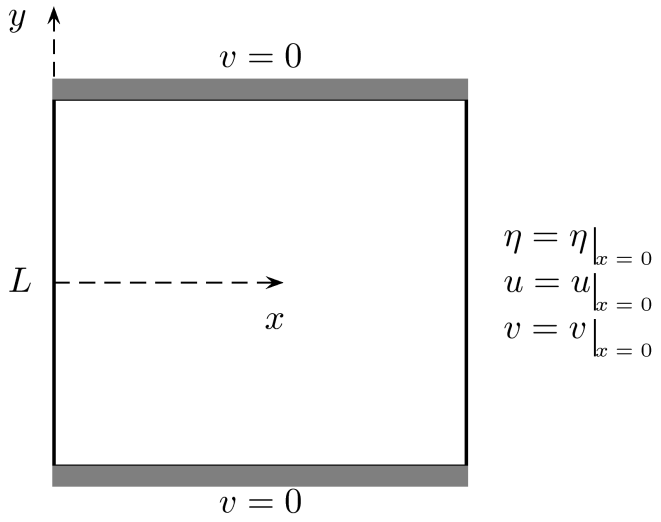


Figure 2.4: Definition of the square domain of size L with solid walls on the northern and southern boundaries and periodic conditions on the eastern and western boundaries.

Then we differentiate these equations and deduce the third order equation:

$$\begin{aligned} &+\left(\frac{\partial^2}{\partial t^2}-gH_0\frac{\partial^2}{\partial x^2}\right)\left(\frac{\partial^2 v}{\partial t^2}-gH_0\left(\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 u}{\partial x\partial y}\right)+f\frac{\partial u}{\partial t}\right)\\ &-\left(f\frac{\partial}{\partial t}-gH_0\frac{\partial^2}{\partial x\partial y}\right)\left(\frac{\partial^2 u}{\partial t^2}-gH_0\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 v}{\partial x\partial y}\right)-f\frac{\partial v}{\partial t}\right) &= 0\\ &\Big| \\ &\left(\frac{\partial^3}{\partial t^3}-gH_0\frac{\partial}{\partial t}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)+f^2\frac{\partial}{\partial t}-gH_0\frac{\partial f}{\partial y}\frac{\partial}{\partial x}\right)\frac{\partial v}{\partial t} &= 0\ . \end{aligned} \tag{2.39}$$

Finally, recombining the terms, we obtain:

$$\left(\frac{\partial}{\partial t}\left(-\frac{\partial^2}{\partial y^2}+\frac{1}{gH_0}\left(\frac{\partial^2}{\partial t^2}+f^2\right)\right)-\frac{\partial}{\partial x}\left(\beta+\frac{\partial}{\partial x}\frac{\partial}{\partial t}\right)\right)\frac{\partial v}{\partial t}=0\ . \tag{2.40}$$

Since the Coriolis parameter is represented by a linear function of y, we cannot assume a plane wave solution. The solution exhibits a general y-dependence and has to be written as the real part of the following expression:

$$v(x,y,t)=Y(y)\exp\left(i(k_x x-\omega t)\right) \tag{2.41}$$

with the unknown function $Y(y)$ and the wave number in the x direction k_x.
By substituting equation (2.41) in equation (2.40), we then obtain a typical Sturm-Liouville eigenproblem:

$$\frac{d^2Y}{dy^2}-\underbrace{\left[\frac{2f_0\beta}{gH_0}y+\frac{\beta^2}{gH_0}y^2\right]}_{g(y)}Y+\underbrace{\left[\frac{1}{gH_0}\left(\omega^2-f_0^2\right)-\frac{\beta k_x}{\omega}-k_x^2\right]}_{k_y^2}Y=0\ . \tag{2.42}$$

For each value of k_x, we obtain a one-dimensional eigenvalue problem which can be numerically solved with $Y(-L/2)=0=Y(L/2)$ as boundary conditions. The eigenvalues are k_y^2 while the eigenvectors are the modes in the y-direction $Y(y)$. For each given couple (k_x,k_y), the frequencies are obtained as the roots of the third order equation:

$$\omega^3-\left(gH_0k^2+f_0^2\right)\omega-gH_0\beta k_x=0\ . \tag{2.43}$$

The smallest root in norm is the Rossby frequency associated to the given couple (k_x, k_y), while the two others are the Poincaré frequencies representing the westward and the eastward Poincaré waves. This approach yields exact dispersion relations with a variable Coriolis parameter for the Poincaré and Rossby waves, as illustrated in Figure (2.5).

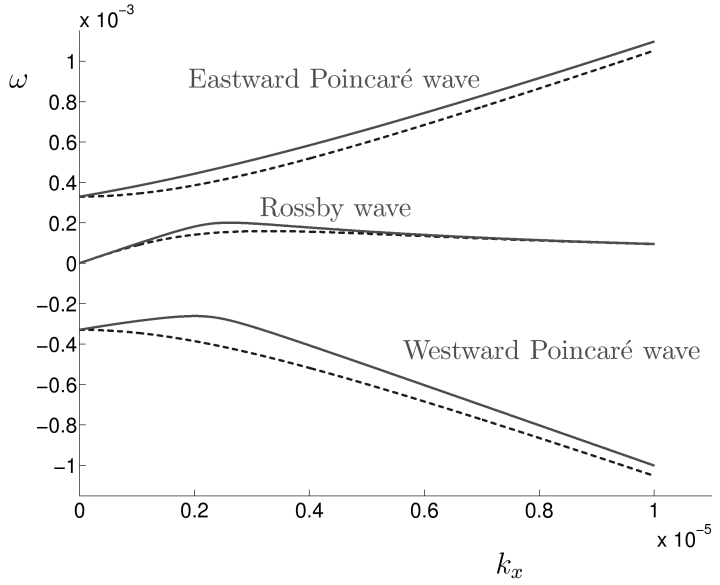


Figure 2.5: Dispersion relations of the Poincaré and Rossby waves computed with $H_0 = 1000\ m$, $g = 10\ ms^{-2}$, $f_0 = 10^{-4}\ s^{-1}$ and $\beta = 1.05\ 10^{-9}\ m^{-1}s^{-1}$. Continuous lines are the exact frequencies from the Sturm-Liouville approach while the dashed lines are the classical approximations of a constant Coriolis parameter for Poincaré waves and the WKB approximation for Rossby waves. These relations are computed on a large domain of size $L = 10^6\ m$. Only the first mode in the y direction was considered: $k_y = \pi/L$.

The Wentzel-Kramers-Brillouin approximation

The Wentzel-Kramers-Brillouin (WKB) approximation [Wentzel, 1926]

consists in neglecting small terms under the assumption of slowly varying coefficients, as the β-terms have been neglected in developments to obtain (2.38).
By neglecting the third order time derivative in equation (2.39), we obtain the classical approximate dispersion relation (2.38). Note that the terms depending on y are usually neglected as well with the WKB approximation, which leads to a different definition of the k_y's in (2.38) than in (2.43) : the eigenfunctions become sines, $Y_n = \sin(n\pi(x - L/2))$, and the wave numbers read $k_{yn}^2 = n\pi/L$ in the simplified cartesian domain.
In Figure (2.5), we clearly observe that such an approximation is not accurate enough to perform a convergence study, since the errors are beyond the numerical scheme accuracy. We then need to use the Sturm-Liouville formulation (2.42) and (2.43) to obtain reference solutions for such a study.

Constant Coriolis coefficient $\mathbf{f = f_0}$

Finally, let us now assume the Coriolis coefficient to be constant. The Sturm-Liouville relation (2.42) with $\beta = 0$ reads:

$$\frac{d^2Y}{dy^2} + \underbrace{\left[\frac{1}{gH_0}\left(\omega^2 - f_0^2\right) - k_x^2\right]}_{k_y^2} Y = 0 \ , \tag{2.44}$$

which is a classical wave equation, the modes $Y(y)$ becoming sines functions. The corresponding dispersion relation is the approximate relation obtained for the Poincaré waves (2.30).

Chapter 3

The Discontinuous Galerkin Method

Contents

3.1 Overview 38

3.2 The Discontinuous Galerkin Method 41

3.3 Accurate advection schemes 44

3.3.1 One-dimensional linear advection equation . . 44

3.3.2 Riemann solver for a linear system of PDE's 46

3.4 A one-dimensional benchmark 47

3.5 Approximate Riemann solvers for non linear conservation laws 60

3.5.1 Exact Riemann solver for non-linear systems 60

3.5.2 Roe Solver 65

3.5.3 Lax-Friedrich solver 68

3.6 Conclusions 68

This chapter presents the basics of the discontinuous Galerkin (DG) method and its main advantages using in a first step a simple one-dimensional advection equation. Accurate advection schemes are then derived as Riemann or Roe solvers, for linear and non-linear systems of PDE's, respectively. The method is applied to the one-dimensional benchmark of the shallow water equations in a rotating framework, to

illustrate the influence of both the advection scheme and the use of high-order polynomial shape functions.

3.1 Overview

The discontinuous Galerkin (DG) method consists in a discontinuous finite element method combining the advantaging features of both the classical continuous Galerkin formulations and the finite volumes methods. Indeed, the classical interpolation theory is still valid and the resolution can be locally increased by increasing the polynomial order of the function space, while the discontinuities allow to build accurate high-order advection schemes.
The original DG method was first introduced by Reed and Hill in [Reed and Hill, 1973] for solving the neutron transport equation. The success of this method has led a few years later to the development of the Runge-Kutta discontinuous Galerkin (RKDG) method for solving systems of non-linear hyperbolic equations with a spatial DG discretization and a classical explicit Runge-Kutta time discretization [Cockburn and Shu, 1989, 1991, 1998a]. Advection-diffusion problems have then been addressed by introducing the local discontinuous Galerkin (LDG) method to handle diffusion operators, based on the work of Bassi and Rebay [Bassi and Rebay, 1997b, Cockburn and Shu, 1998b]. Other techniques for computing diffusion operators have been developed, based on the interior penalty (IP) methods. The origin of penalty formulation is found in the late 1960s where it was introduced for enforcing Dirichlet boundary conditions [Lions, 1968]. IP methods have then been used to weakly impose $\mathcal{C}^1$ continuity for fourth-order problems [Babuska and Zlamal, 1973], and to compute consistent and stable diffusive DG operators [Douglas and Dupont, 1976, Arnold et al., 2002].

The increasing popularity of the DG method is due to several interesting numerical properties:

- *Handling complex geometries.* As the classical continuous finite element method, the DG method is better suited than finite differences for addressing multi-scale processes since it provides a great flexibility for handling complex geometries and a natural and easy

way of imposing natural boundary conditions. Indeed, no special treatment is required for achieving uniform high-order accuracy as in finite differences.

- *High-order accuracy.* Since inherited from the classical Galerkin procedures, the interpolation theory is still valid for the DG method and predicts a convergence of the interpolation error of the order $\mathcal{O}\left(h^{p+1}\right)$, where p is the polynomial order of the function space. The DG method provides then an easy way to increase the accuracy of the scheme without enlarging the computation stencil. Indeed, the polynomial order of the function space can be easily increased inside an element since there is no continuity requirements between elements.

- *Designed for advection* Classical finite element methods are particularly well suited for solving diffusion processes, while advection-dominated processes require to introduce a sufficient amount of numerical diffusion to stabilize the continuous methods. Using the DG method, the discontinuities between elements allow one to use high order advection schemes as for instance the approximate Riemann solvers, which makes the DG method very accurate for solving advection-dominated processes.

- *Efficient computation properties.* The discontinuities lead to a block-diagonal mass matrix, whose blocks can be easily inverted. The DG method becomes then highly parallelizable. Moreover, only the degrees of freedom sharing a face are involved in the convective fluxes computations at the interfaces between elements. The number of communications between processors at each explicit Runge-Kutta inner step is then minimum. This block-diagonal property also tends to reduce the bandwidth of the global matrices and the computational cost for the system resolution, compared to other techniques presenting the same number of degrees of freedom. Other efficient techniques must be mentioned at this point, as the quadrature-free techniques reducing the computational costs for some particular applications [Atkins and Shu, 1998, Marchan-

dise et al., 2006c].

- *Adaptive strategies.* Mesh adaptivity is an important matter, especially for hyperbolic problems, given the complexity of the solution structure. The DG method provides an easy and efficient way to deal with mesh adaptation. First, an accurate *a posteriori* error-estimator is contained in the inter-element discontinuities, also called jumps. The discretization error is proportional to the jumps, which are also proportional to h^{p+1}. In the framework of h-adaptivity, a simple evaluation of the jumps amplitudes provides then a new size field for the creation of a new better suited mesh reducing the discretization error in strong gradients regions. Besides, the adaptation procedure itself is highly efficient due to the discontinuous character of the DG method. Indeed, the mesh adaptation can be performed locally, by considering small cavities, and the projections from the original mesh to the new one are then local procedures. The solution projection does not require the construction of large global matrices and the memory footprint is reduced to only one mesh instead of two. Moreover, this technique is highly parallelizable since every operation is performed locally [Biswas et al., 1994, deCougny et al., 1994, Devine et al., 1993, 1994, Ozturan et al., 1994, Remacle et al., 2001, 2003]. Finally, performing efficient p-adaptivity is also straightforward since the interpolation order may differ between elements.

- *Slope-limiting.* Although shocks are not present in ocean dynamics, the ocean may exhibit strong spatial variations and quasi-discontinuities which lead to spurious oscillations of the solution. This issue is also present in small scales hydrodynamics models as river flows or dam break problems. The DG method offers, at low orders, very natural slope limiting techniques to damp out the spurious oscillations generated around discontinuities or strong gradients of the solution [van Leer, 1974, 1979, Cockburn and Shu, 1991].

These properties have made of the DG method a very popular new trend for solving engineering problems where advection plays an impor-

tant role [Cockburn et al., 2000]. The method is now widely used for solving a large variety of problems in many fields of practical engineering such as computational fluid dynamics [Remacle et al., 2005, Hillewaert et al., 2006, Krivodonova et al., 2004, Bassi and Rebay, 1997a, Bey and Oden, 1991, Lomtev et al., 1998, Lomtev and Karniadakis, 1999, Warburton et al., 1997], aeroacoustics problems [Chevaugeon et al., 2005b,a], levelset methods [Marchandise et al., 2007, Marchandise and Remacle, 2006], magnetohydrodynamics [Warburton and Karniadakis, 1999] or hydrodynamics and ocean modeling [Remacle et al., 2006, Bernard et al., 2007, Dawson and Proft, 2004, Kubatko et al., 2006, Aizinger and Dawson, 2002, 2007, Giraldo et al., 2002, Giraldo, 2006].
However, there is still some controversy about the use of the DG method for solving diffusion operators. Indeed, since designed for computing advection equations, solving diffusion processes with the DG method is not as natural as in classical continuous finite element methods. Nevertheless, many techniques have been studied so far and good results have been obtained using methods as LDG or IP [Arnold et al., 2002]. Another controversy concerns the computational cost of the DG method. The method is clearly competitive when performing computations at high polynomial orders. But the question is still open for the comparison of the low-order DG method to finite volumes or linear finite elements.

3.2 The Discontinuous Galerkin Method

For sake of simplicity, we introduce in this section the basics of the DG method using a one-dimensional example in order to illustrate the main ideas and properties of the method. We consider the following hyperbolic conservation law for the quantity u in one dimension:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \tag{3.1}$$

with t the time, f the *flux* function and x belonging to the finite domain Ω.
The domain of computation Ω is divided into a set of $\mathcal{N}_e$ non overlapping elements Ω_e:

$$\Omega = \bigcup_{e=1}^{\mathcal{N}_e} \Omega_e \ .$$

The standard discontinuous Galerkin weak formulation is obtained by multiplying equation (3.1) by a smooth function $\hat{u}$ and integrating over each element Ω_e:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \hat{u}u \; d\Omega + \int_{\Omega_e} \hat{u}\frac{\partial f(u)}{\partial x} \; d\Omega = 0 \qquad \forall e \in [1, \mathcal{N}_c]. \tag{3.2}$$

We seek an approximation u^h of u such that $u^h \in \mathcal{U}_p^h$ with $\mathcal{U}_p^h$ the finite dimensional space

$$\mathcal{U}_p^h = \left\{ v \in L^2 : v|_{\Omega_e} \in P^p(\Omega_e), e = 1, ..., \mathcal{N}_e \right\}$$

where $P^p(\Omega_e)$ denotes the space of polynomials on element Ω_e of degree at most p and L^2 the space of functions whose square can be integrated on Ω. We define on each element a discrete representation u_e^h of u as

$$u_e^h = \sum_{i=1}^{\mathcal{N}_p} \phi_i U_i^e \quad \in \mathcal{U}_p^h$$

where U_i^e are the *nodal values* on element e, ϕ_i are a set of independent shape functions and $\mathcal{N}_p$ is the number of degrees of freedom on one element of order p. Although there are many possible choices for the shape functions, we will consider Lagrange shape functions in the following developments. Notice that this particular choice does not affect the further conclusions. The number of degrees of freedom depends on both the polynomial degree of interpolation and the element type and its evolution is described in Table (3.1). The discontinuous interpolation u^h of u is then the set of u_e^h over each element, as depicted in Figure (3.1).

The weak formulation (3.2) is then discretized as:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \hat{u}^h u^h \; d\Omega + \int_{\Omega_e} \hat{u}^h \frac{\partial f(u^h)}{\partial x} \; d\Omega = 0 \tag{3.3}$$

with the test function $\hat{u}^h \in \mathcal{U}_p^h$.
Two neighboring elements in continuous finite element method share common nodes that ensure the continuity of the finite element approximation and that allows information transfer between elements. It is

Element Type	$\mathcal{N}_p$
Edge	$p+1$
Triangle	$(p+1)(p+2)/2$
Quadrangle	$(p+1)^2$
Hexahedra	$(p+1)^3$
Tetrahedron	$(p+1)(p+2)(p+3)/6$
Prism	$(p+1)^2(p+2)/2$

Table 3.1: Evolution of the number of degrees of freedom $\mathcal{N}_p$ with the element type for the (complete) polynomial order p.

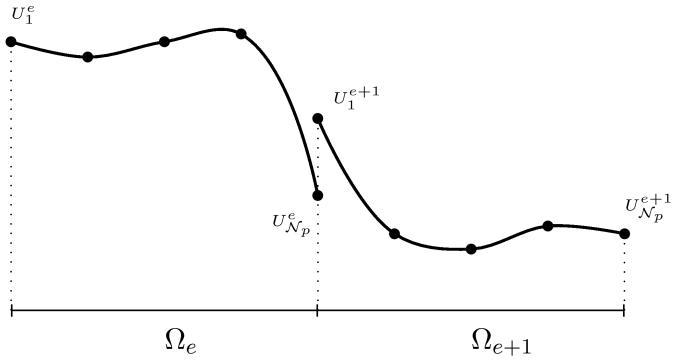


Figure 3.1: Discontinuous approximation u^h on elements Ω_e and Ω_{e+1}

mandatory to ensure an equivalent information transfer between elements with the DG method, by the following integration by part:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \hat{u}^h u^h \; d\Omega - \int_{\Omega_e} f(u^h)\frac{\partial \hat{u}^h}{\partial x} \; d\Omega + \int_{\partial\Omega_e} f_n(u^h)\hat{u}^h \; d\Gamma = 0 \qquad (3.4)$$

with f_n the normal flux to the element edge. The main difference with the continuous finite element method lies in the fact that the discrete solution presents different values $U^e_{\mathcal{N}_p}$ and U^{e+1}_1 at the discontinuity between elements Ω_e and Ω_{e+1} as in Figure (3.1). The formulation (3.4) is thus somehow ambiguous and incomplete since the discretization leads to a discrete normal flux function $f_n(u^h)$ depending on both right and left values at the discontinuity:

$$f_n(u^h) = f_n(U^e_{\mathcal{N}_p}, U^{e+1}_1). \qquad (3.5)$$

This flux function has to be properly defined since it will be responsible in a large part for the accuracy and robustness of the scheme. This discontinuity thus provides a way of introducing a suitable advection scheme.

3.3 Accurate advection schemes

Hyperbolic equations are governed by the theory of characteristics, stating that the information is transferred by some invariant quantities, the *characteristic variables*, traveling along curves in the space-time domain, called *characteristic curves*. This theory can be used to find relevant values $u^*(U^e_{\mathcal{N}_p}, U^{e+1}_1)$ for computing the numerical flux $f_n(u^*)$ at the discontinuities.

3.3.1 One-dimensional linear advection equation

Let us consider in a first step the simple linear advection equation of the quantity u:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, \qquad (3.6)$$

with c a constant advection velocity. The flux function is thus the simple linear function

$$f(u) = cu \ .$$

Let us now consider the discontinuous initial condition depicted in Figure (3.2):

$$\begin{aligned} u(x, t = 0) &= u_L \quad \text{for } x < 0 \\ u(x, t = 0) &= u_R \quad \text{for } x > 0. \end{aligned}$$

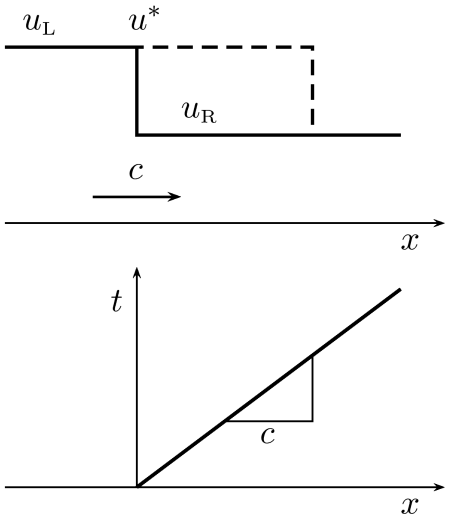


Figure 3.2: Initial discontinuous solution u advected at the speed c along the characteristic $x = ct$.

The invariant quantity u is advected along the characteristic curve $x = ct$ (see Figure (3.2)) and the solution reads:

$$\begin{aligned} u(x, t) &= u_L \quad \text{if } x < ct, \\ u(x, t) &= u_R \quad \text{if } x > ct. \end{aligned}$$

With a positive advection speed c, the information clearly propagates from the left to the right. This direction of propagation has to be taken

into account in the numerical flux computation. Indeed, the flux has to be computed as:

$$f(u^*) = f(u_L) \quad \text{if } c > 0, \tag{3.7}$$
$$f(u^*) = f(u_R) \quad \text{if } c < 0, \tag{3.8}$$

to be in agreement with the theory of characteristics. It can be demonstrated [LeVeque, 1992] that this *upwinding* on the characteristic quantity u for low order schemes is mathematically equivalent to introduce the minimum amount of dissipation in the numerical system to avoid unresolved oscillations of the solution. More precisely, a different amount of dissipation is applied to every mode of the solution, so that the unresolved high-wavenumber modes experience a relatively strong numerical dissipation compared to the well resolved low-wavenumber modes. On the other hand, using a centered scheme, i.e. taking u^* as the mean of u_L and u_R, is equivalent to a zero numerical diffusion. The non physical high wavenumbers produced will not be damped out, leading to non physical unresolved oscillations. Using finally the *downwind* value of u will correspond to negative numerical diffusion leading to an unstable scheme.

3.3.2 Riemann solver for a linear system of PDE's

The idea of the Riemann solver is to decouple the system of equations by expressing it as a set of independent scalar advection equations.
Let us consider the following system for the vector of variables $\boldsymbol{U}$ of size N:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}(\boldsymbol{U})}{\partial x} = 0, \tag{3.9}$$

with $\boldsymbol{F}$ the vector of flux functions.
We seek the set of *Riemann invariants*, i.e. the characteristics variables $\boldsymbol{U}^c$ travelling along the characteristic curves, which are thus the variables of the uncoupled system. Equation (3.9) can be written as:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}\frac{\partial \boldsymbol{U}}{\partial x} = 0, \tag{3.10}$$

where $\boldsymbol{A} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$ is the constant Jacobian matrix. The Jacobian matrix can be diagonalized as:

$$\boldsymbol{A} = \boldsymbol{R}\boldsymbol{\Lambda}\boldsymbol{R}^{-1} \tag{3.11}$$

with $\boldsymbol{R}$ the matrix containing the eigenvectors of $\boldsymbol{A}$ and $\boldsymbol{\Lambda}$ the diagonal matrix of eigenvalues λ_i, $i = 1, \ldots, N$. Multiplying (3.10) on the left by $\boldsymbol{R}^{-1}$ leads to the uncoupled system:

$$\frac{\partial \boldsymbol{U}^c}{\partial t} + \boldsymbol{\Lambda}\frac{\partial \boldsymbol{U}^c}{\partial x} = 0, \tag{3.12}$$

where $\boldsymbol{U}^c = \boldsymbol{R}^{-1}\boldsymbol{U}$ are the Riemann invariants. Each scalar invariant U_i^c is advected at the speed λ_i. According to the theory of characteristics, we obtain the following solution for an initial discontinuous state:

$$U_i^c(x) = U_{iL}^c \quad \text{if } \lambda_i > 0, \tag{3.13}$$
$$U_i^c(x) = U_{iR}^c \quad \text{if } \lambda_i < 0. \tag{3.14}$$

The above system can be solved to express $\boldsymbol{U}$ in terms of $\boldsymbol{U}_L$ and $\boldsymbol{U}_R$, which yields consistent values of $\boldsymbol{U}$ at the discontinuity according to the characteristics.

3.4 A one-dimensional benchmark

This section is devoted to the analysis of a particular one-dimensional benchmark for the linearized shallow water equations in a rotating framework. This benchmark contains the Coriolis effect and involves the two components of the velocity while remaining 1D and presenting an analytical solution. We will then be able to perform an error analysis and to observe the impact of both the Riemann solver and the use of high order shape functions.

We consider here the linearized non-conservative form of the inviscid shallow water equations (2.22) and (2.23), with the assumption of an infinite domain in the y-direction, which implies that the derivatives

with respect to y vanish while the v-component of the velocity remains:

$$\frac{\partial \eta}{\partial t} + H_0 \frac{\partial u}{\partial x} = 0, \tag{3.15}$$

$$\frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} - fv = 0, \tag{3.16}$$

$$\frac{\partial v}{\partial t} + fu = 0, \tag{3.17}$$

with $x \in [-L/2, L/2]$ and L the domain length. We finally assume a constant Coriolis parameter.

The problem can be non-dimensionalized by considering the following dimensionless quantities: tf, x/L, η/η_0 and $uH_0/(Lf\eta_0)$. These simplified equations present the following useful non-dimensional analytical solution:

$$\frac{u(x,t)H_0}{Lf\eta_0} = \sum_{n=1}^{\infty} H_n \frac{gH_0 k_n}{f^2 L^2 \omega_n} \sin(\omega_n t) \sin(k_n x), \tag{3.18}$$

$$\frac{v(x,t)H_0}{Lf\eta_0} = \sum_{n=1}^{\infty} H_n \frac{gH_0 k_n}{f^2 L^2 \omega_n} (\cos(\omega_n t) - 1) \sin(k_n x), \tag{3.19}$$

$$\frac{\eta(x,t)}{\eta_0} = \sum_{n=1}^{\infty} H_n \left(1 - \frac{gH_0 k_n^2}{f^2 L^2 \omega_n^2} (1 - \cos(\omega_n t))\right) \cos(k_n x), \tag{3.20}$$

where the coefficients H_n can be computed as:

$$H_n = 2 \int_0^1 \eta(x,0) \cos(k_n x) \; dx \; .$$

The complete derivation of this analytical solution may be found in [White et al., 2006]. We consider the following initial condition:

$$u(x,0) = v(x,0) = 0, \tag{3.21}$$

$$\eta(x,0) = \eta_0 \tanh\left(C \frac{x}{L}\right). \tag{3.22}$$

with $C = 100L$ to obtain the initial elevation field depicted in Figure (3.3). The strong gradients will act as discontinuities and produce oscillations if the solution is not sufficiently well resolved. A completely discontinuous initial field would require the use of slope limiters to avoid

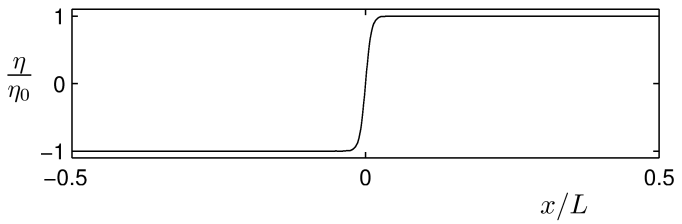


Figure 3.3: Initial quasi-discontinuous non-dimensional elevation field.

oscillations. Such limiters are not considered here.

Let us now derive the values from the Riemann solver to obtain an accurate advection scheme. The linear fluxes and the corresponding Jacobian matrix read:

$$\boldsymbol{F} = \begin{bmatrix} H_0 u \\ g\eta \\ 0 \end{bmatrix} \quad , \quad \boldsymbol{A} = \begin{bmatrix} 0 & H_0 & 0 \\ g & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{3.23}$$

The Jacobian matrix is then decomposed into eigenvectors $\boldsymbol{R}$ and eigenvalues $\boldsymbol{\Lambda}$:

$$\boldsymbol{R} = \begin{bmatrix} \sqrt{g/H_0} & -\sqrt{g/H_0} & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad , \quad \boldsymbol{\Lambda} = \begin{bmatrix} \sqrt{gH_0} & 0 & 0 \\ 0 & -\sqrt{gH_0} & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{3.24}$$

The Riemann invariants are then computed as the product of the inverse matrix of the eigenvectors and the vector of variables:

$$\boldsymbol{U}_c = \left[\frac{u}{2} + \sqrt{\frac{g}{H_0}} \frac{\eta}{2} \quad , \quad \frac{u}{2} - \sqrt{\frac{g}{h}} \frac{\eta}{2} \quad , \quad v \right]^T . \tag{3.25}$$

Each Riemann invariant is advected along the linear characteristics as depicted in Figure (3.4), whose slopes are given by the eigenvalues. The resolution of the characteristic relations yields the following values to

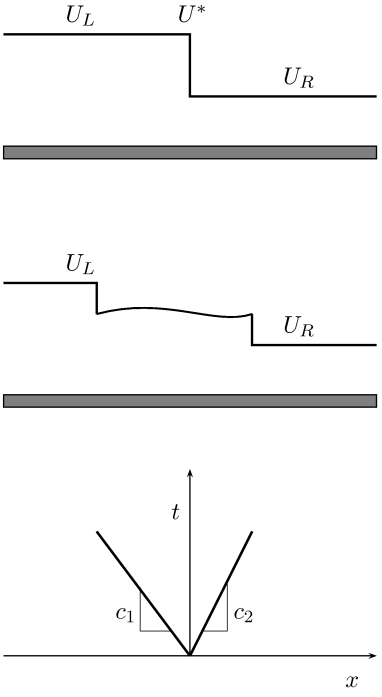


Figure 3.4: The linear hyperbolic system of the shallow water equations corresponds to three Riemann invariants propagating along the characteristics determined by $c_2^1 = \lambda_2^1 = \pm\sqrt{gH_0},\ \lambda_3 = 0$.

use at the interfaces:

$$\eta = \{\eta\} + \sqrt{\frac{H_0}{g}}[\![u]\!] \tag{3.26}$$

$$u = \{u\} + \sqrt{\frac{g}{H_0}}[\![\eta]\!] \tag{3.27}$$

$$v = \{v\} \tag{3.28}$$

where $\{a\} = \frac{a_L + a_R}{2}$ denotes the mean value of a and $[\![a]\!] = \frac{a_L - a_R}{2}$ denotes the jump of a. For this one-dimensional linear problem, the flux function using a Riemann solver can be written in the following more convenient form:

$$\boldsymbol{F} = \boldsymbol{A}\left(\{\boldsymbol{U}\} + \boldsymbol{D}[\![\boldsymbol{U}]\!]\right) \tag{3.29}$$

with $\boldsymbol{D}$ the matrix responsible for the introduction of numerical dissipation in the scheme:

$$\boldsymbol{D} = \begin{bmatrix} 0 & \sqrt{H_0/g} & 0 \\ \sqrt{g/H_0} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{3.30}$$

The values of the physical parameters are $g = 9.81 < ms^{-2}$, $H_0 = 100\ m$, $L = 10^6\ m$, $f = 10^{-4}\ s^{-1}$ and $\eta_0 = 1\ m$. The system is solved using a fourth-order Runge-Kutta (RK44) time scheme. In Figure (3.5) are depicted the analytical elevation (thin red lines) and its DG approximation (dotted blue lines) with the dots corresponding to the nodal values, for a polynomial order $p = 2$ and 40 elements at the non-dimensional times $tf = 1, 2, 20$. The figure illustrates the influence of the advection scheme on the solution. Although stable, the use of a centered scheme does not introduce any numerical dissipation, the non physical high wavenumbers thus remain in the system, leading to non-physical oscillations. On the other hand, the use of accurate advection schemes (the exact Riemann solver in this case) yields a small but sufficient amount of diffusion to damp out all the unresolved high-wavenumber modes.

Indeed, in order to observe the dissipative behaviour of the Riemann solver on the discrete system, let us write the DG weak formulation $b(\boldsymbol{U}^k, \hat{\boldsymbol{U}})$ for the element Ω_k depicted in Figure (3.6) in a similar way

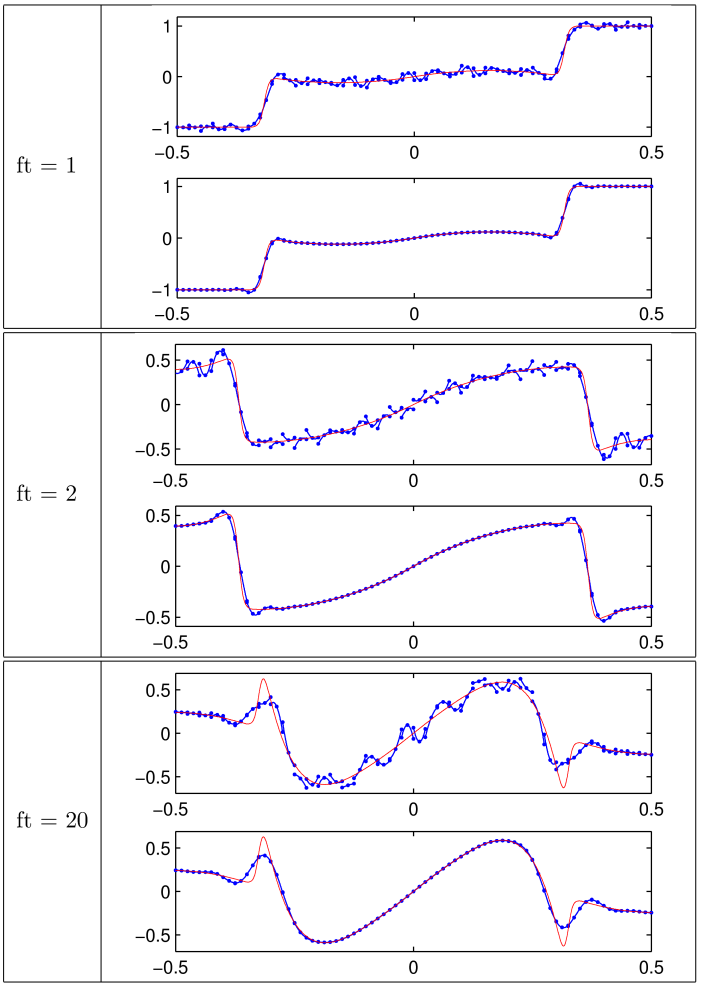


Figure 3.5: The use of a centered advection scheme (top row for each time) produces oscillations while high wavenumbers are damped out with a Riemann solver (bottom row for each time). On the figure are depicted the non-dimensional DG elevation field with nodal values (blue dots) and the analytical solution (continuous red line) for the non-dimensional times $ft = 1, 2$ and 20 (top to bottom).

than the scalar weak formulation (3.4):

$$b(\boldsymbol{U}^k, \hat{\boldsymbol{U}}) = \int_{x_k}^{x_{k+1}} \frac{\partial \boldsymbol{U}^k}{\partial t} \hat{\boldsymbol{U}} \; dx - \int_{x_k}^{x_{k+1}} \boldsymbol{F} \frac{\partial \hat{\boldsymbol{U}}}{\partial x} \; dx + \left[\widehat{\boldsymbol{F}_n} \hat{\boldsymbol{U}} \right]_{x_k}^{x_{k+1}} \quad (3.31)$$

with $\widehat{\boldsymbol{F}_n}$ the flux function (3.29) derived from the Riemann solver. Let us now integrate by part the second term of (3.31) in order to obtain the so-called *LeSaint-Raviart* formulation:

$$\begin{aligned} b(\boldsymbol{U}^k, \hat{\boldsymbol{U}}) = \int_{x_k}^{x_{k+1}} \frac{\partial \boldsymbol{U}^k}{\partial t} \hat{\boldsymbol{U}} \; dx + \int_{x_k}^{x_{k+1}} \frac{\partial \boldsymbol{F}}{\partial x} \hat{\boldsymbol{U}} \; dx \\ - \left[\boldsymbol{F}_n \hat{\boldsymbol{U}} \right]_{x_k}^{x_{k+1}} + \left[\widehat{\boldsymbol{F}_n} \hat{\boldsymbol{U}} \right]_{x_k}^{x_{k+1}} \; . \end{aligned} \quad (3.32)$$

Notice that this second integration by part leads to an equivalent discrete system only if the new integration on the boundaries involves the inner values of $\boldsymbol{U}$, instead of another Riemann flux function. The linear flux $\boldsymbol{F}_n$ is therefore computed as $\boldsymbol{A}\boldsymbol{U}$. In order to prove the $\mathcal{L}_2$ stability of the DG scheme [Cockburn et al., 2000, Marchandise et al., 2006a], let us integrate (3.32) in time from $t = 0$ to $t = T$ and sum on the elements:

$$\mathbb{B}(\boldsymbol{U}^k, \boldsymbol{U}^k) = \int_0^T \sum_{k=1}^{\mathcal{N}_e} b(\boldsymbol{U}^k, \boldsymbol{U}^k) \; dt = 0 \; .$$

It can be shown easily that this sum of the terms

$$\begin{aligned} \int_{x_k}^{x_{k+1}} \frac{\partial \boldsymbol{F}}{\partial x} \boldsymbol{U}^k \; dx &= \int_{x_k}^{x_{k+1}} \boldsymbol{A} \frac{\partial \boldsymbol{U}^k}{\partial x} \boldsymbol{U}^k \; dx \\ &= \int_{x_k}^{x_{k+1}} \frac{1}{2} \boldsymbol{A} \frac{\partial \left(\boldsymbol{U}^k \right)^2}{\partial x} \; dx \\ &= \frac{1}{2} \boldsymbol{A} \left[\boldsymbol{U}^{k2} \right]_{x_k}^{x_{k+1}} \end{aligned}$$

combined to the terms from $-\left[\boldsymbol{F}_n \boldsymbol{U}^k \right]_{x_k}^{x_{k+1}}$ exactly compensate the centered part of the Riemann flux in the possibly dissipative terms from $\left[\widehat{\boldsymbol{F}_n} \boldsymbol{U}^k \right]_{x_k}^{x_{k+1}}$. Therefore, it comes from the time derivative and the dis-

sipative part of the flux:

$$\begin{aligned}
\mathbb{B}(\boldsymbol{U}^k, \boldsymbol{U}^k) = & \int_0^T \sum_{k=1}^{\mathcal{N}_e} \int_{x_k}^{x_{k+1}} \frac{1}{2} \frac{\partial \boldsymbol{U}^{k\,2}}{\partial t} \, dx \, dt \\
& + \int_0^T \sum_{k=1}^{\mathcal{N}_e} \Big[\boldsymbol{AD} [\![\boldsymbol{U}^k]\!] \boldsymbol{U}^k \Big]_{x_k}^{x_{k+1}} \, dt \\
= & \frac{1}{2} \sum_{k=1}^{\mathcal{N}_e} \int_{x_k}^{x_{k+1}} \boldsymbol{U}^k(x,T)^2 - \boldsymbol{U}^k(x,0)^2 \, dx \\
& + \int_0^T 2 \boldsymbol{AD} \sum_{k=1}^{\mathcal{N}_e} [\![\boldsymbol{U}^k]\!]^2 \, dt \\
= & \frac{1}{2} \| \boldsymbol{U}(T) \|_{\mathcal{L}_2}^2 - \frac{1}{2} \| \boldsymbol{U}(0) \|_{\mathcal{L}_2}^2 + \int_0^T 2 \boldsymbol{AD} \sum_{k=1}^{\mathcal{N}_e} [\![\boldsymbol{U}^k]\!]^2 \, dt \\
= & \; 0
\end{aligned}$$

We clearly observe that the use of a centered scheme instead of a Riemann solver for the computation of the flux function leads to an exact conservation of the $\mathcal{L}_2$ norm of the solution. The high non-physical wavenumbers are therefore conserved in the system. The scheme is clearly $\mathcal{L}_2$ stable for both the centered scheme and the Riemann solver, while the inversion of the dissipative part of the Riemann solver, corresponding to a downwinding scheme, leads to instabilities. Moreover, the dissipation of the DG method with a Riemann solver depends on the amplitude of the jumps of the solution. The jumps are known to be proportional to the discretization error, i.e. proportional to h^{p+1}. The numerical dissipation introduced therefore decreases rapidly as the polynomial order increases.

There are many techniques for introducing numerical diffusion in the scheme, but controlling the amount of diffusion is sometimes a tedious task and low-wavenumber modes can be affected as well as unresolved modes, i.e. modes presenting too high wavenumbers for the polynomial order and the element size used to be represented without aliasing on the grid. The use of Riemann-like solvers in the flux computation is clearly a simple and efficient technique which guarantee the introduction of

p	1	2	3	4	5	6	7	8
r	2.33	3.65	4.75	4.90	6.38	6.86	8.03	8.45

Table 3.2: Convergence rates r obtained in Figure (3.9) for the $\mathcal{L}_2$ norm of the error for polynomial orders $p = 1$ to 8.

diffusion only for the unresolved modes, while keeping a quasi-centered scheme for the low-wavenumber modes.

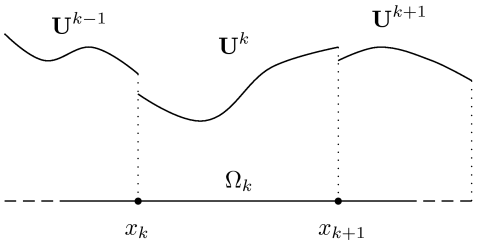


Figure 3.6: Solution $\boldsymbol{U}^k$ on element Ω_k bounded by x_k and x_{k+1}.

In Figure (3.7) and Figure (3.8) are compared the solutions at order $p = 1, 4$ and 8 at the time $tf = 20$ for a constant number of degrees of freedom, 90 and 180 respectively. The accuracy of the solution is clearly increased with higher polynomial orders for an equal number of degrees of freedom. Indeed, the steep gradients are better resolved with higher orders, while the use of $p = 1$ tends to smooth the solution. We also observe the creation of small oscillations with high order function spaces as the gradients are not sufficiently well resolved, since no limiter is considered here. These oscillations tend to vanish as the resolution increases.

A more quantitative analysis can be performed by studying the evolution of the $\mathcal{L}_2$ norm of the discretization error, defined as

$$\| e \|^2_{\mathcal{L}_2} = \int_\Omega (u^h - u)^2 \, d\Omega \, , \tag{3.33}$$

with the polynomial order and the element size. The mesh refinement convergence is presented in Figure (3.9) for polynomial orders from 1 to 8 and meshes of 10 2^n elements. The corresponding convergence rates

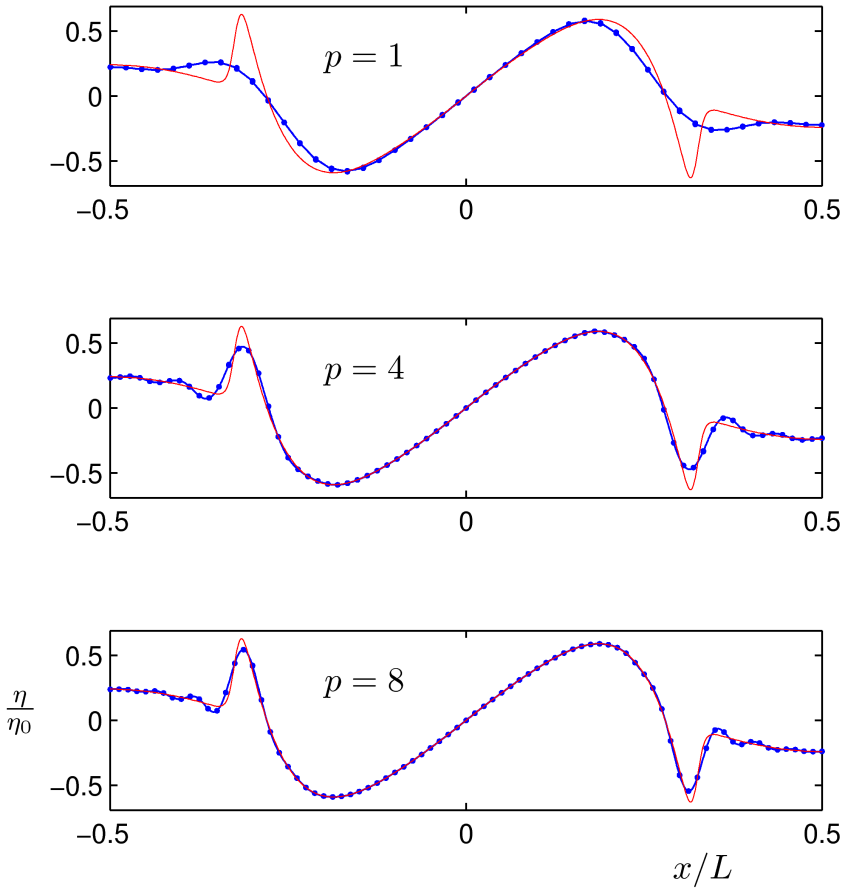


Figure 3.7: Non-dimensional elevation field at non-dimensional time $ft = 20$ for polynomial orders p of 1 (top), 4 (middle) and 8 (bottom) for a number of elements of 45, 18 and 10 respectively, leading to an equivalent number of degrees of freedom of $\mathcal{N}_d = 90$. The DG solution is the blue line, the dots denoting the nodal values, while the red continuous line denotes the analytical solution.

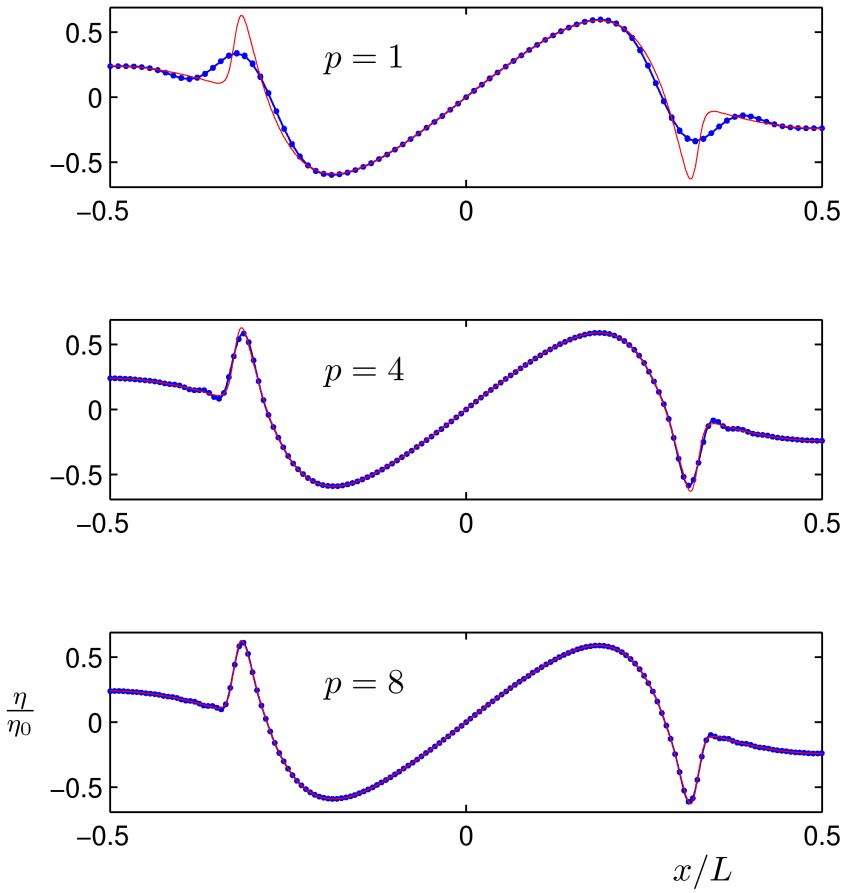


Figure 3.8: Non-dimensional elevation field at non-dimensional time $ft = 20$ for polynomial orders p of 1 (top), 4 (middle) and 8 (bottom) for a number of elements of 90, 36 and 20 respectively, leading to an equivalent number of degrees of freedom of $\mathcal{N}_d = 180$. The DG solution is the blue line, the dots denoting the nodal values, while the red continuous line denotes the analytical solution.

are given in Table (3.2) where the theoretical convergence of $\mathcal{O}(h^{p+1})$ is obtained. This convergence property illustrates the efficiency of high order function spaces. The use of the first polynomial order requires 640 elements, i.e. 1280 degrees of freedom, to reach the asymptotic convergence region, and reaches then the relative error of about 10^{-6}, while the same error is obtained for instance with only 40 elements at order $p = 5$, i.e. 240 degrees of freedom.

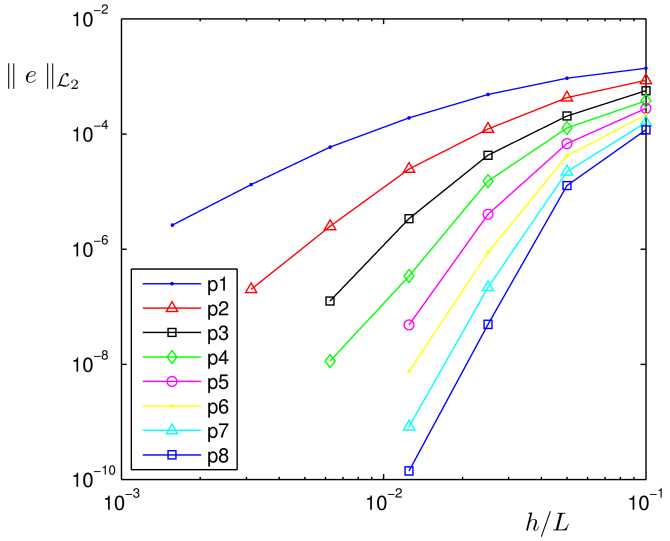


Figure 3.9: The $\mathcal{L}_2$ norm of the error converges for different element sizes h as $\mathcal{O}(h^{p+1})$ for polynomial orders $p = 1$ to 8.

The respective computational times for the different values in Figure (3.9) are given in Table (3.3). We observe that the most accurate $p = 1$ computation required $74s$, while the same accuracy is reached with $p = 2$ in about $12s$ and in $5s$ with $p = 5$. It has to be mentioned that the computation times for the higher order are far from the optimal. Indeed, the errors introduced by the fourth order accurate RK4 time scheme become higher than the discretization errors for the high orders. The time steps in these computations have then been divided by factors of 2 for $p = 6$ or 3 for $p = 7, 8$ compared to the CFL condition to improve the

$\mathcal{N}_e$	20	40	80	160	320	640
p_1	4.12	4.30	4.57	8.00	21.54	73.98
p_2	3.31	3.76	5.48	12.03	37.75	
p_3	3.40	4.12	6.84	17.26		
p_4	3.53	4.55	8.40	23.47		
p_5	3.66	4.98	10.15			
p_6	4.33	7.59	21.85			
p_7	5.27	11.99	37.08			
p_8	6.06	13.86	44.51			

Table 3.3: Evolution of the computational time in seconds (corresponding to Figure (3.9)) with the order p and the number of elements $\mathcal{N}_e$.

convergence rate, i.e. to obtain smaller time errors than spatial errors. The computational times could then be reduced by using appropriate higher order time discretizations for the high polynomial orders. Of course, the use of the $p = 1$ could probably also be increased by using lower order time schemes. Moreover, the code is also far from optimal for the use of low order polynomials. But these considerations will not modify the several orders of magnitude between the computational times or degrees of freedom. A $45s$ computation using $p = 8$ provides for instance an error which is four orders of magnitude lower than a $74s$ computation with linear elements.
It can be shown for smooth solutions [Karniadakis and Sherwin, 2005] that the computational time c_t is related to the polynomial order by the following relation:

$$c_t = p^a$$

where the power a depends on the dimension of the problem and the complexity of the solution. On the other hand, the discretization error e^h is related to the element size h as $e^h \propto h^{p+1}$. Therefore, for a given sufficient accuracy, the computational time will always be decreased as the polynomial order increases, as illustrated by the computational times of Table (3.3).

3.5 Approximate Riemann solvers for non linear conservation laws

Non linear flux functions lead to non constant Jacobian matrices. Therefore, more complex Riemann invariants are advected along curved characteristics resulting in increased computational costs for computing the exact Riemann solver. This section briefly describes the main ideas of the Riemann solver for non linear systems and some usual approximate Riemann solvers. We consider in particular the developments leading to the so-called *Roe solver*.

3.5.1 Exact Riemann solver for non-linear systems

Let us consider the Riemann problem for a general one-dimensional non-linear hyperbolic system of m equations and m unknown fields with the initial discontinuous solution composed of the left and right values $\boldsymbol{U}_R$ and $\boldsymbol{U}_L$:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) = 0 \tag{3.34}$$

$$\boldsymbol{U}(x,0) = \begin{cases} \boldsymbol{U}_L & \text{if} \quad x < 0 \\ \boldsymbol{U}_R & \text{if} \quad x > 0 \end{cases} \tag{3.35}$$

The solution $\boldsymbol{U}(x,t)$ can be decomposed in $m+1$ constant states separated by m waves [Toro, 1997, LeVeque, 2002]. Each eigenvalue $\lambda_i,\ i = 1 \dots m$ of the system corresponds to a wave family. In the case of linear systems with constant coefficients as considered previously, each wave consists in a discontinuity advected at the speed $S_i = \lambda_i$. For non-linear systems, these waves may be composed of discontinuities as shock waves or contact waves, or of smooth transition waves as rarefaction waves. The nature of each wave will depend on the initial condition of the Riemann problem.
Solving the Riemann problem thus requires to

1. determine the natures of the waves,
2. solve the states associated to each wave,
3. recombine the solutions to obtain the discontinuities values.

Shock Waves and the Rankine-Hugoniot Condition

Physical shock waves are thin transition layers where take place very rapid modifications of the physical quantities as the velocity, the pressure or the temperature. We illustrate this phenomenon by considering the scalar non-linear advection equation of the quantity $u(x,t)$:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \tag{3.36}$$

$$f(u) = \frac{u^2}{2} \tag{3.37}$$

$$u(x,0) = \begin{cases} u_L & \text{if} \quad x < 0 \\ u_R & \text{if} \quad x > 0 \end{cases} \tag{3.38}$$

known as the *Burgers* equation, with $u_L > u_R$.

Those transition layers for strong shocks are about 10^{-7} m wide in

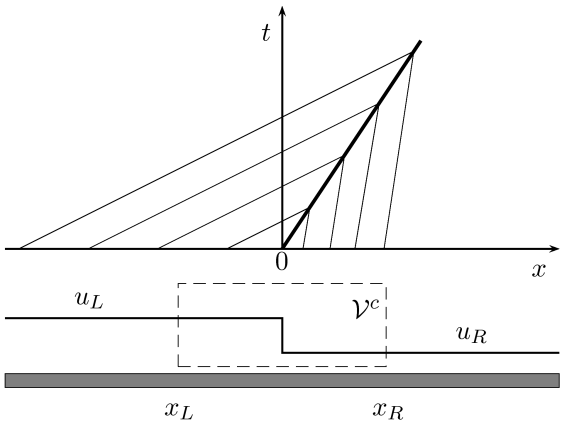


Figure 3.10: Shock propagating at the speed S with $\lambda(u_L) > S > \lambda(u_R)$.

the air, which corresponds to the mean free path of the particles. The mathematical modelization of the shocks as discontinuities seems thus appropriate in most of cases. We consider however the integral form of (3.36) on a control volume $\mathcal{V}^c \in [x_L, x_R]$ where a shock wave takes place,

as depicted in Figure (3.10):

$$f(u(x_L,t)) - f(u(x_R,t)) = \frac{\partial}{\partial t}\int_{x_L}^{x_R} u\ dx\ .$$

The solution $u(x,t)$ presents a jump discontinuity across the shock whose path is given by the curve $s(t)$, with a shock speed $S = \frac{\partial s}{\partial t}$. The integral form has thus to be rewritten as:

$$\begin{aligned} f(u(x_L,t)) - f(u(x_R,t)) &= \frac{\partial}{\partial t}\int_{x_L}^{s(t)} u\ dx + \frac{\partial}{\partial t}\int_{s(t)}^{x_R} u\ dx \\ &= [u(s_L,t) - u(s_R,t)]\, S \\ &\quad + \int_{x_L}^{s(t)} \frac{\partial u}{\partial t}\ dx + \int_{s(t)}^{x_R} \frac{\partial u}{\partial t}\ dx \end{aligned}$$

where we took advantage of the Leibniz theorem and with

$$\begin{aligned} u(s_L,t) &= \lim_{\substack{x\to s(t)\\ x<s(t)}} u(x,t)\ , \\ u(s_R,t) &= \lim_{\substack{x\to s(t)\\ x>s(t)}} u(x,t)\ . \end{aligned}$$

The integral terms of the time derivatives vanish since $\frac{\partial u}{\partial t}$ is bounded, which yields the following relationship:

$$\Delta f = S\ \Delta u \tag{3.39}$$

well known as the *Rankine-Hugoniot (RH) condition*, governing the shock speed S.
The solution of the shock wave is thus clearly given by the theory of characteristics and the RH condition:

$$u(x,t) = \begin{cases} u_L & \text{if} \quad x < St \\ u_R & \text{if} \quad x > St \end{cases} \tag{3.40}$$

$$S = \frac{\Delta f}{\Delta u}\ . \tag{3.41}$$

In can be finally shown that the shock waves satisfy the following *entropy condition*:

$$\lambda(u_L) > S > \lambda(u_R) \tag{3.42}$$

clearly satisfied in Figure (3.10).

Rarefaction Waves and the Entropy Condition

Let us now consider the opposite initial condition $u_L < u_R$ for the Burgers equation. One possible mathematical solution is depicted in the lower part of Figure (3.11). This solution involves a "rarefaction shock" also called "entropy violating shock": this configuration does not satisfy the entropy condition (3.42) and has to be rejected. The alternative solution, the "rarefaction wave", is depicted in the upper part of Figure (3.11). The solution corresponding to a rarefaction waves is thus :

$$u(x,t) = \begin{cases} u_L & \text{if} \quad x < \lambda_L t \\ u_R & \text{if} \quad x \geq \lambda_R t \end{cases} \tag{3.43}$$

$$\lambda = \frac{x}{t} \quad \text{if} \quad \lambda_L t < x < \lambda_R t \ . \tag{3.44}$$

where we use the fact that the Riemann invariants remain constant when advected through the discontinuity.

An intermediate solution between rarefaction and shock waves is the *contact wave*, which involves a shock with characteristic curves parallel to the shock direction. This correspond to the relation $\lambda(u_L) = \lambda(u_R) = S$. Solving this kind of wave requires then the use of the RH condition, and the Riemann invariants advection theory.

Exact Riemann solver for a non-linear hyperbolic system

The exact solution to the non-linear Riemann problem requires to first determine the nature of each wave for each eigenvalue of the system, i.e. a shock wave, rarefaction wave, or contact wave. Each wave is then solved as proposed in the previous scalar cases.
In can be shown [LeVeque, 2002] that the solution for the non-linear

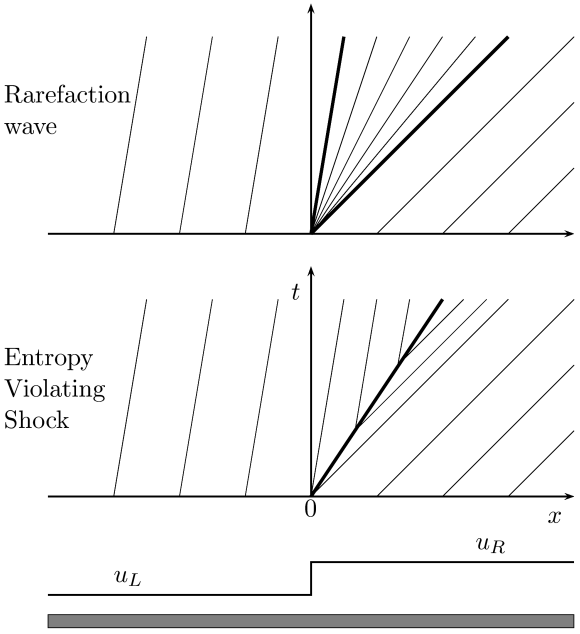


Figure 3.11: The rarefaction shock (middle row) violates the entropy condition and has to be rejected as a physical solution. The entropy satisfying solution is the rarefaction wave depicted in the upper row.

one-dimensional shallow water equations is given by:

$$\gamma_L(H) = \begin{cases} u_L + 2(\sqrt{gH_L} - \sqrt{gH}) & \text{if } H < H_L \ , \\ u_L - (H - H_L)\sqrt{\frac{g}{2}\left(\frac{1}{H} + \frac{1}{H_L}\right)} & \text{if } H > H_L \ , \end{cases} \tag{3.45}$$

$$\gamma_R(H) = \begin{cases} u_R - 2(\sqrt{gH_R} - \sqrt{gH}) & \text{if } H < H_R \ , \\ u_R + (H - H_R)\sqrt{\frac{g}{2}\left(\frac{1}{H} + \frac{1}{H_R}\right)} & \text{if } H > H_R \ . \end{cases} \tag{3.46}$$

$$\tag{3.47}$$

Indeed, the functions γ_L and γ_R will connect the solution H to the left and right states, respectively, by physically correct waves, and the solution thus satisfies $\gamma_L(H) = \gamma_R(H)$. This computation can be done by means of a non-linear root finder applied to $\gamma(H) = \gamma_L(H) - \gamma_R(H)$.

Solving the exact non-linear Riemann problem is much more expensive than solving the linear one, since iterative methods are usually required. Yet, this effort is not always justified. Indeed, we are clearly not interested in solving accurately the Riemann problem, but in introducing an appropriate amount of dissipation into the numerical scheme with the correct sign. Moreover, *approximate Riemann solvers* reveal cheaper with no significative reduction of the solution accuracy, especially for increased polynomial orders.

3.5.2 Roe Solver

One of the most popular advection schemes is the *Roe solver* proposed in [Roe, 1981]. The idea is to locally linearize the Jacobian matrix $\boldsymbol{A}$, which means performing a linearization of the characteristic curves and speeds, and to compute then the exact Riemann solver of the resulting linearized system. Roe proposed an appropriate linearization of $\boldsymbol{A}(\boldsymbol{U}_L, \boldsymbol{U}_R)$, say $\tilde{\boldsymbol{A}}(\boldsymbol{U}_L, \boldsymbol{U}_R)$, by introducing the following requirements on the linearized Jacobian matrix:

1. Obviously, the linearized system has to preserve the hyperbolic nature of the system, which implies to present a set of real eigenvalues $\tilde{\lambda}_i(\boldsymbol{U_L}, \boldsymbol{U_R})$ and a complete set of linearly independent eigenvectors $\tilde{\boldsymbol{R}}^i$.

2. The linearized Jacobian has to be consistent with the original Jacobian:
$$\tilde{\boldsymbol{A}}(\boldsymbol{U},\boldsymbol{U}) = \boldsymbol{A}(\boldsymbol{U}) \ .$$
This relation also implies that, at the refinement limit, the jumps tend to vanish and the impact of the linearization tends to disappear. The Roe linearization thus tends to the exact Riemann solver for a sufficiently refined grid.

3. The linearization has eventually to guarantee the respect of the Rankine-Hugoniot condition, i.e. the conservation across discontinuities:
$$\boldsymbol{F}(\boldsymbol{U}_L) - \boldsymbol{F}(\boldsymbol{U}_R) = \tilde{\boldsymbol{A}}\left(\boldsymbol{U}_L - \boldsymbol{U}_R\right) \ .$$

It comes from the first requirements that the linearized Jacobian matrix can be expressed as

$$\tilde{\boldsymbol{A}}(\boldsymbol{U}_L,\boldsymbol{U}_R) = \boldsymbol{A}(\tilde{\boldsymbol{U}})$$

where $\tilde{\boldsymbol{U}}(\boldsymbol{U}_L,\boldsymbol{U}_R)$ are the *Roe averages*, i.e. appropriate variable averages such that $\tilde{\boldsymbol{U}}(\boldsymbol{U},\boldsymbol{U}) = \boldsymbol{U}$.

Let us compute this linearization for the conservative form of the non-linear shallow water equations (2.20) and (2.21). The non-linear flux matrix reads

$$\boldsymbol{F} = \begin{bmatrix} Hu_n & \dfrac{(Hu_n)^2}{H} + \dfrac{g}{2}H^2 & \dfrac{Hu_nHv_t}{H} \end{bmatrix}^T$$

and yields the following linearized Jacobian matrix:

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} 0 & 1 & 0 \\ g\tilde{H} - \tilde{u_n}^2 & 2\tilde{u_n} & 0 \\ -\tilde{u_n}\tilde{v_t} & \tilde{v_t} & \tilde{u_n} \end{bmatrix}$$

where u_n and v_t denote the normal and tangential components of the velocity to the discontinuity. We now introduce the third requirement and apply the Rankine-Hugoniot condition to find the Roe averages $(\tilde{H}, \tilde{u_n}, \tilde{v_t})$. Since the first row of the Jacobian matrix is linear, the system is not well posed, its first line is trivial. We thus modify the first line to define the first Roe average value of H as a simple arithmetic

mean value $\tilde{H} = \{H\}$. The resolution of the Rankine-Hugoniot system thus yields the Roe averages:

$$\tilde{H} = \{H\}, \tag{3.48}$$

$$\tilde{u_n} = \frac{u_{nL}\sqrt{H_L} + u_{nR}\sqrt{H_R}}{\sqrt{H_L} + \sqrt{H_R}}, \tag{3.49}$$

$$\tilde{v_t} = \frac{v_{tL}\sqrt{H_L} + v_{tR}\sqrt{H_R}}{\sqrt{H_L} + \sqrt{H_R}}. \tag{3.50}$$

The linearized eigenvalues $\tilde{\boldsymbol{\Lambda}}(\boldsymbol{U}) = \boldsymbol{\Lambda}(\tilde{\boldsymbol{U}})$ and eigenvectors $\tilde{\boldsymbol{R}}(\boldsymbol{U}) = \boldsymbol{R}(\tilde{\boldsymbol{U}})$ then yield the Riemann invariants of the linearized system, $\tilde{\boldsymbol{U}}^* = \tilde{\boldsymbol{R}}^{-1}\boldsymbol{U}$. The exact Riemann solver of the linearized system provides the following variable values:

$$H = \{H\} + \frac{1}{\tilde{c}}\left([\![Hu]\!] - \tilde{u}[\![H]\!]\right), \tag{3.51}$$

$$Hu = \{Hu\} + \frac{1}{\tilde{c}}\left(\tilde{u}[\![Hu]\!] - \tilde{u}^2[\![H]\!]\right), \tag{3.52}$$

$$Hv = (Hv)_{\text{up}} + \tilde{v}(\{H\} - H_{\text{up}}) + \frac{1}{\tilde{c}}\left(\tilde{v}[\![Hu]\!] - \tilde{u}\tilde{v}[\![H]\!]\right). \tag{3.53}$$

with $\tilde{c} = \sqrt{g\tilde{H}}$. The upwinded values $(Hv)_{\text{up}}$ and $(H)_{\text{up}}$ are defined as the left or right values, depending on the sign of the corresponding third eigenvalue $\tilde{\lambda}_3 = \tilde{u}_n$.

In the case of the bi-dimensional non-linear shallow water equations in their conservative form, the Roe solver can be approximated by the following simple expression of the fluxes [Remacle et al., 2006, Soares Frazão and Zech, 2002a]:

$$\boldsymbol{F}(\boldsymbol{U}_L, \boldsymbol{U}_R)\cdot\boldsymbol{n} = \{\boldsymbol{F}\}\cdot\boldsymbol{n} + Fr[\![\boldsymbol{F}]\!]\cdot\boldsymbol{n} + \tilde{c}(1-Fr^2)[\![\boldsymbol{U}]\!]\ . \tag{3.54}$$

The first term clearly corresponds to a centered scheme flux, while others are dissipation terms depending on the jumps amplitudes and on the averaged Froude number Fr defined as:

$$Fr = \frac{\tilde{u_n}}{\tilde{c}}\ . \tag{3.55}$$

3.5.3 Lax-Friedrich solver

Besides the Roe solver, many other Riemann-like solvers exist with different computational costs and accuracy, as for instance the Roe, Osher, Godunov, HLL, HLLC or Lax-Friedrich solvers [Godunov, 1959, Roe, 1981, Toro, 1997, LeVeque, 2002]. However, the use of high order methods with high order accuracy reduces the importance of the Riemann solver choice. Indeed, the amplitude of the jumps varies as h^{p+1}. Using high polynomial orders decreases the amplitude of the jumps and therefore the amplitude of the diffusive part in the advection scheme. In practical DG computations, no significative difference is found in terms of accuracy between these usual solvers. One of the most easily implemented and cheaper solvers is the Lax-Friedrich solver which reads:

$$\boldsymbol{F}(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n} = \{\boldsymbol{F}\} \cdot \boldsymbol{n} + |\lambda|_{max} [\![\boldsymbol{U}]\!] \tag{3.56}$$

where $|\lambda|_{max}$ denotes the maximum absolute value of the eigenvalues of the system, i.e. the propagation speed of the information.

3.6 Conclusions

The DG method and some of its underlying key points have been introduced using simple one-dimensional equations. In particular, the one-dimensional analytical benchmark considered, involving Poincaré waves, is very illustrative of the efficiency of the DG method, combining the best of both finite volume and finite element worlds.
The efficiency of the use of Riemann solvers at the discontinuities has been shown and proved. This technique clearly appears to be a natural and efficient choice for solving the shallow water equations. The DG method is therefore highly appealing since perfectly suited for solving the advection processes dominating the ocean dynamics. Considering non-linear systems of equations, the same efficiency is obtained using approximate Riemann solvers. Yet, the choice of the approximate Riemann solver tends to be less and less crucial as the polynomial order increases, i.e. as the method differs from finite volume methods.
The efficiency of the high-order polynomials has finally been observed on the one-dimensional benchmark, illustrating a second key aspect of the DG method. The use of higher polynomial orders always increases the efficiency of the scheme, and therefore decreases the computational costs,

at least for such a set of conservation laws without complex parametrizations or modelizations in a simple domain. The aim of the following work is therefore to reduce the limitations of the high-order polynomials, which are mainly controlled by the possible irregularities of the geometry.

Chapter 4

Efficient Adaptive Strategy using the Discontinuous Galerkin Method

Contents

4.1 Metric, error estimation and mesh adaptation 72
4.1.1 Towards an optimal mesh: creation of a metric 73
4.1.2 Usual approaches for error estimation and metric generation 76
4.2 *h*- and *p*- adaptivity with a DG approach . . 79
4.2.1 Jumps as error estimation 80
4.2.2 Optimal h-adaptation strategy 82
4.2.3 p-adaptation to a target error 84
4.2.4 h-adaptation procedures and projection of the solution 85
4.3 *p*-adaptation on a 1D benchmark 88
4.4 2D *h*-adaptation 89
4.4.1 Adaptive convergence applied to the Stommel gyre 89
4.4.2 Propagation of a baroclinic anticyclonic eddy at midlatitudes 98
4.5 Conclusions 103

The accuracy and convergence of a numerical scheme depends on the mesh quality [Ciarlet, 1991], i.e. on its shape and size. This is one of the advantages of the finite element method. Indeed, the underlying mesh is usually *a priori* adapted to the simulated processes on a given domain, which increases the flexibility and accuracy of the method for a given computational cost. For very transient processes and fields presenting local strong gradients, there might be no *a priori* refinement criterion. Therefore, the mesh adaptation becomes very useful in order to dynamically increase the resolution in those large gradients regions. In particular, ocean modeling exhibits quasi discontinuities or important dynamics features like meso-scale eddies or polar fronts. Those meso-scale processes contain for instance a large part of the ocean energy and require thus to be followed in time and solved accurately. The ocean also exhibits many different length scales in time and space, with very unsteady behavior and almost discontinuous fields. Dynamic mesh adaptation strategies following those structures has a great potential in the field of ocean modeling [e.g. Behrens, 1998, Heinze and Hense, 2002, Nair et al., 2005, Giraldo et al., 2002, St-Cyr et al., 2006, Piggott et al., 2008].

In this chapter are presented some usual components of the adaptation strategies, as the creation of a metric, a new mesh size field and the error estimation. The approach based on the DG method is then presented, in particular the error estimation based on the jumps and the creation of a new size field considering h- and p-adaptation. The method is eventually applied to a one-dimensional benchmark to illustrate the DG p-adaptation strategy, and to two-dimensional examples to show the efficiency of h-adaptation for ocean modeling.

4.1 Metric, error estimation and mesh adaptation

The mesh adaptation strategies can be decomposed in three stages:

1. An efficient *error estimator* is required to locate the large gradients regions presenting larger errors.

2. Once the error has been estimated, a new mesh size field has to be created to perform the mesh adaptation. A typical procedure is to create a metric based on the error such that the optimal new mesh presents unit length edges in the new metric.

3. The actual adaptation of the mesh is then performed according to the new size field, and the solution has to be projected from the first mesh to the new optimal mesh.

There are basically three main strategies for the mesh adaptation procedure, called r-, p- and h-adaptation. The r-adaptation consists in modifying the mesh using the only operation of moving nodes. The elements' shapes and sizes are thus modified while keeping the same number of nodes during the process. This adaptation strategy is usually associated with Arbitrary Lagrangian Eulerian methods (ALE), taking into account the nodes movements into the equations as numerical advection velocity. The main advantage of such methods is the simple and accurate way of transferring the information between meshes. The p-adaptation consists in modifying the polynomial interpolation order of the solution from one element to another, while keeping unchanged the underlying mesh. This technique can be particularly complicated when considering continuous finite elements, since continuity requirements have to be addressed on the edges between elements of different orders, but the technique becomes very simple when considering discontinuous finite elements. The h-adaptation technique consists eventually in modifying the mesh itself by splitting edges, merging elements or swapping edges.

4.1.1 Towards an optimal mesh: creation of a metric

The p- adaptivity does not require modifying the mesh, but to modify directly the polynomial order based on the error estimation. But considering r- or h- adaptivity, a new size field of the domain $\Omega \in \mathcal{R}^3$ has to be generated. This can be achieved by means of a metric tensor $\mathcal{M}(\boldsymbol{x}), \boldsymbol{x} \in \Omega$, defined as a symmetric positive-definite 3×3 matrix, i.e. with the entries $m_{ii} > 0$ for $i \in [1, 3]$ and $det(\mathcal{M}) > 0$.
The scalar product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ and the norm in this new

metric thus read:

$$\begin{aligned} < \boldsymbol{a}, \boldsymbol{b} >_{\mathcal{M}} &= \boldsymbol{a}^T \mathcal{M} \boldsymbol{b} \ , \\ \| \boldsymbol{a} \|_{\mathcal{M}} &= \sqrt{< \boldsymbol{a}, \boldsymbol{a} >_{\mathcal{M}}} \ . \end{aligned}$$

Let us consider a point $P \in \mathcal{R}^3$. The ball of P of size a in the metric $\mathcal{M}$ is then defined as the set of points M equidistant of a from P in the new metric,

$$\| \overrightarrow{PM} \|_{\mathcal{M}} = \sqrt{\overrightarrow{PM}^T \mathcal{M} \overrightarrow{PM}} = a \ .$$

This ball in the new metric is clearly represented by an ellipsoid in the classical unit metric. Indeed, the metric can be decomposed as:

$$\mathcal{M} = \boldsymbol{R}^T \boldsymbol{\Lambda} \boldsymbol{R} \tag{4.1}$$

with $\boldsymbol{R}$ the matrix of eigenvectors $(\boldsymbol{E}_1, \boldsymbol{E}_2, \boldsymbol{E}_3)$ and $\boldsymbol{\Lambda}$ the diagonal matrix of eigenvalues λ_i. Following a classical approach (see for instance

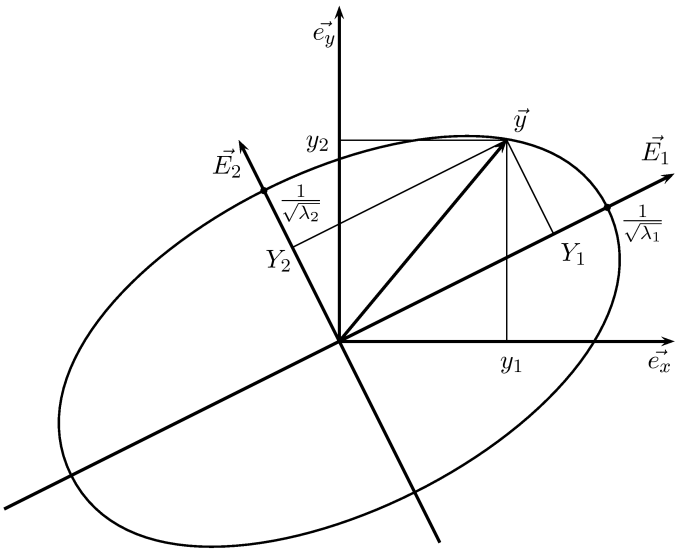


Figure 4.1: Representation of the metric's ellipse in $\mathcal{R}^2$. The edge $\vec{y}$ is optimal since lying on the ellipse defined by $\mathcal{M}$.

[Remacle et al., 2006]), the optimal mesh is defined as the mesh presenting edges of unit length in this new metric field. Let $\boldsymbol{y}$ denote an edge passing through vertex P. The optimal size of this edge is thus given by:

$$\begin{aligned} 1 &= \boldsymbol{y}^T \mathcal{M} \boldsymbol{y} \\ &= \boldsymbol{y}^T \boldsymbol{R}^T \boldsymbol{\Lambda} \boldsymbol{R} \boldsymbol{y} \end{aligned}$$

and $\boldsymbol{Ry}$ is thus the expression of the edge in the principal axis of $\mathcal{M}$ as depicted in Figure (4.1) while the eigenvalues functions $1/\sqrt{\lambda_i}$ represent the lengths of the ellipsoid in its main directions. Let us denote by y_i and Y_i the coordinates of the edge in the standard cartesian axes and in the eigenvectors axis respectively. The optimal edge size is thus given by:

$$1 = Y_1^2 \lambda_1 + Y_2^2 \lambda_2 + Y_3^2 \lambda_3 \tag{4.2}$$

which means that the new edge is optimal if lying on the ellipsoid defined by (4.2). This definition of an optimal mesh will then be used as a constraint for the creation of the new mesh.

There are basically two main approaches for the generation of a new mesh. The first approach consists in adapting the existing mesh by refining or coarsening the existing elements, according to the new metric, without moving nodes. The main advantage of this method lies in its simplicity in terms of implementation and in the fact that it keeps unchanged the original nodes, which reduces the possible numerical diffusion or Gibbs oscillations, i.e. overshoots, when using high order polynomials. But the drawback is that the adapted mesh may present badly shaped elements and is certainly not optimal in the sense that it does not satisfy exactly the optimality constraint. This technique has been used for instance in [Löhner and Baum, 1992, Biswas and Strawn, 1994, Dindar et al., 2000].

The second approach considers the local or global creation of a different mesh, by allowing operations as edges or faces swapping, nodes movements or deletion or insertion of vertices. The new mesh can be created using classical methods as frontal [Löhner, 1990, Peraire et al., 1992], Delaunay [George, 1999] or *octree* methods [Baehmann et al., 1992]. This second approach clearly provides a better optimization of the mesh, while increasing the risk of generating oscillations or numerical diffusion. More details on this approach may be found in [Coupez, 2000, Pain et al., 2001,

Remacle et al., 2002].

4.1.2 Usual approaches for error estimation and metric generation

Let us consider the finite element approximation $u^h(\boldsymbol{x})$ of an unknown field $u(\boldsymbol{x}), \boldsymbol{x} \in \Omega$. The *discretization error* is defined as:

$$e^h(\boldsymbol{x}) = u(\boldsymbol{x}) - u^h(\boldsymbol{x}) \ .$$

The approximation u^h is usually not an interpolation, which implies that the nodal values of u^h do not match the solution u at any given point. The discretization error is thus more complicated to evaluate. Therefore, we also consider the *interpolation error* defined as:

$$\tilde{e}^h(\boldsymbol{x}) = u(\boldsymbol{x}) - \tilde{u}^h(\boldsymbol{x}) \ .$$

where $\tilde{u}^h(\boldsymbol{x})$ represents an interpolation of u on each element. A widely used result in the literature is the Lemma of Cea, stating that the discretization error is bounded by the interpolation error:

$$\| u - u^h \| \leq C \, \| u - \tilde{u}^h \| \tag{4.3}$$

with C a constant depending on the dimension of the problem. This inequality and the strong link between both errors allows one to quantify the discretization error by only considering the interpolation error. Therefore, many interesting results can be obtained from this Lemma, as the classical estimation of the finite element discretization error. The standard Galerkin procedure for finite element methods considers the integration of the equations on elements after multiplying by *smooth functions*. This smoothness is rigorously defined using Sobolev spaces. In particular, finite element methods usually consider the Hilbert spaces, i.e. a typical example of Sobolev spaces, defined as the family of spaces $\mathcal{H}^m$ of dimension m:

$$\mathcal{H}^m = \left\{ v | D^{\boldsymbol{\alpha}} v \in L_2(\Omega), | \, \boldsymbol{\alpha} \, | \leq m \right\} \tag{4.4}$$

with the notation:

$$D^{\boldsymbol{\alpha}} = \frac{\partial^{|\boldsymbol{\alpha}|}}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \dots \partial x_n^{\alpha_n}} \quad \text{with} \quad \forall \boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_n)$$

$$|\boldsymbol{\alpha}| = \sum_{i=1}^{n} \alpha_i$$

for a domain $\Omega \subset \mathcal{R}^n$, and with L_2 the space of the functions whose square integral is finite:

$$L_2(\Omega) = \left\{ v | \int_\Omega v^2 \, d\Omega < \infty \right\} .$$

Those Hilbert spaces $\mathcal{H}^m$ thus contain the set of functions whose square of the derivatives up to order m can be integrated. For instance, the space of dimension 0 corresponds to $\mathcal{H}^0(\Omega) = L_2(\Omega)$.
With those definitions, the classical estimation of the finite element discretization error reads:

$$\| e^h \| \leq C h^{min(p+1,r)} \, \| u \|_{min(p+1,r)} \tag{4.5}$$

where p is the polynomial order of interpolation, h is the element size and r is the higher dimension of the Sobolev space to which the exact solution u belongs. Clearly, the norm of the exact solution is *a priori* unknown, but this inequality yields the asymptotic convergence rate of the error of $p+1$, except if the solution does not belong to the Sobolev space $\mathcal{H}^{r+1}$ with $r < p+1$. This reduction of convergence may happen for instance when using high-order polynomial shape function with a low-order forcing defined on Ω (e.g. a piecewise constant wind forcing in the shallow water equations), or with poor representation of the domain boundaries introducing discontinuities.
Therefore, for a given error e^h, the relationship (4.5) allows one to compute a new metric field for performing mesh adaptation. The last step consists in estimating this error. A posteriori error estimations for standard Galerkin procedures are widely described in the literature [Ainsworth and Oden, 1992, 1997, 2000] and are therefore not discussed further in this work.

Another simpler and efficient approach consists in assuming that the error of a given field $u^h \in \mathcal{R}^n$ is proportional to the second derivatives

of this field, and consists more precisely to evaluate its Hessian matrix $\boldsymbol{H}$ on each vertex [Dobrzynski, 2005, Remacle et al., 2006]:

$$H_{ij} = \frac{\partial^2 u^h}{\partial x_i \partial x_j} \quad i,j = 1,\ldots,\dim(\Omega). \tag{4.6}$$

This Hessian matrix is then decomposed as the metric tensor into:

$$\boldsymbol{H} = \boldsymbol{R}^T \tilde{\boldsymbol{\Lambda}} \boldsymbol{R} \tag{4.7}$$

with $\tilde{\boldsymbol{\Lambda}}$ the diagonal matrix of eigenvalues $\tilde{\lambda}_i$ and $\boldsymbol{R}$ the eigenvectors corresponding to those of the metric tensor. The eigenvalues of the metric tensor, i.e. the size of the new mesh edges, is then computed as

$$\lambda_i = \frac{c\tilde{\lambda}}{\epsilon}$$

with c a constant and ϵ a user defined target error.

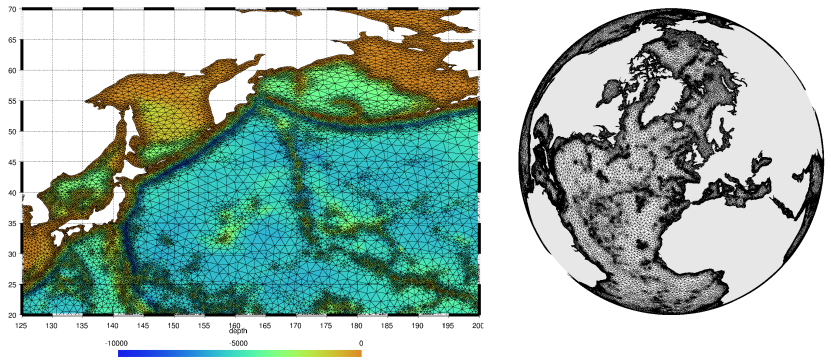


Figure 4.2: Meshes of the global ocean with the Hessian of the bathymetry used as a refinement criterion, capturing large bathymetry gradients around fault lines between tectonic plates. (Courtesy of Jonathan Lambrechts, [Lambrechts et al., 2008a])

This approach is used for instance in [Remacle et al., 2006] for performing anisotropic mesh adaptation for solving the shallow water equations with the error estimation based on the Hessian matrix of the elevation field. A similar approach may also be found in [Lambrechts et al.,

2008a] for mesh generation of the Earth's oceans, using three refinement criterions: the distance to the coasts, the gravity waves speed, i.e. $\sqrt{gH}$ with H the bathymetry, and the Hessian matrix of the bathymetry to capture strong gradients of topography. Those three criterions are clearly illustrated on Figure (4.2), presenting meshes of the Earth's oceans. In the left part of Figure (4.2) is depicted the mesh obtained for the North Pacific Ocean between Corea and Alaska, in longitude-latitude coordinates. The refinement taking the bathymetry gradients into account clearly allows one to observe the fault lines with a bathymetry of about 10000 m corresponding to the separation of the Eurasian plate, the Philippine plate (South of Japan), and the Pacific plate. The right part of the figure illustrates the mesh of the North Atlantic Ocean, where the separation between the North America plate, African plate and Eurasian plate may be observed.

4.2 *h*- and *p*- adaptivity with a DG approach

Although the three basic steps of the adaptation strategy as explained above remain unchanged, the use of the DG method for adaptation reveals very efficient compared to other methods when taking advantage of the discontinuities.
Indeed, on the one hand, the jumps of the solution between elements are a very cheap and efficient error estimator compared to classical estimations or to the Hessian matrix computation. On the other hand, the discontinuous nature of the method allows one to perform only local operations, either for the mesh modifications or for the projection of the solution. This property represents a substantial gain in terms of memory footprint and computational time. Indeed, the use of continuous finite elements, or semi-continuous as non conformal P_1^{NC} elements, requires the projections of the solution to be done as a global operation, since the nodal values are coupled between elements. This also implies that the two meshes have to be kept in memory, while local operations allow one to consider only one mesh during the adaptation procedures.

The following section deals with the use of the DG method for adaptive strategies, and in particular with the error estimation, the creation

of the new mesh metric and the adaptation itself. However, although the use of anisotropic meshes is obviously the optimal choice especially where the flow is itself anisotropic [Remacle et al., 2005], we restrict ourselves to the creation of isotropic meshes in order to simplify the mesh metric definition.

4.2.1 Jumps as error estimation

We consider here the use of an explicit Runge-Kutta time stepping of order $p+1$ in time together with a DG method of order p in space. Under those assumptions, it has been shown that the spatial error is at least one order of magnitude higher than the error in time [Chevaugeon et al., 2007]. This allows one to only consider the spatial error of discretization, converging as $\mathcal{O}(h^{p+1})$ according to (4.5). The main idea of the error estimator used in this work is that the inter-element jumps of the solution are converging at this same rate [Marchandise et al., 2006b]. The jumps provide therefore an efficient and very cheap error estimator.

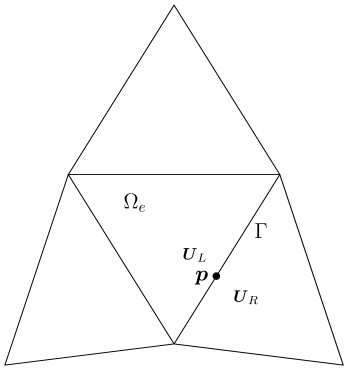


Figure 4.3: Element Ω_e and its three surrounding elements for the jumps computation at point $\boldsymbol{p}$ on edge Γ.

Indeed, let us consider the DG approximation $u^h(\boldsymbol{x})$ of a field $u(\boldsymbol{x})$ on the element Ω_e depicted in Figure (4.3). The jump of u^h at the point

$\boldsymbol{p}$ on the edge Γ converges at the same rate than the spatial error:

$$(U_L - U_R)(\boldsymbol{p}) = \mathcal{O}\left(h^{p+1}\right) \ ,$$

where U_L and U_R denotes the left and right values, respectively, of u^h on both sides of the edge Γ and h is the local mesh size. Another interesting property of the DG method is the superconvergent behaviour of the approximation u^h at downwind faces [Marchandise et al., 2006b]. Indeed, the discretization error on u^h converges at the rate $\mathcal{O}(h^{2p+1})$ on downwind faces, which means that either U_L or U_R is a particularly good approximation of the exact solution on Γ. Let us choose the mean value of u^h as an approximation of the exact solution u:

$$u(\boldsymbol{p}) \simeq \frac{1}{2}(U_L + U_R)(\boldsymbol{p}).$$

We thus obtain the following error estimator E:

$$E(\boldsymbol{p}) = |\, u^h - u \,|\, (\boldsymbol{p}) \simeq \frac{1}{2}(U_L - U_R)(\boldsymbol{p}) = \mathcal{O}(h^{p+1}).$$

The jumps may therefore be considered as a good image of the discretization error.

Let us now develop this relation to use it as an error indicator. The average error along each edge e_Γ reads:

$$\begin{aligned} e_\Gamma^2 &= \frac{1}{\|\Gamma\|}\int_\Gamma E^2 \, ds \\ &\simeq \frac{1}{4\,\|\Gamma\|}\int_\Gamma (U_L - U_R)^2 \, ds \end{aligned}$$

An *a posteriori* error estimator e_{Ω_e} can be computed on each element Ω_e using the mean of the local errors on the surrounding edges Γ_k:

$$e_{\Omega_e}^2 = \frac{1}{3}\triangle \sum_{k=1}^{3} e_{\Gamma_k}^2 \tag{4.8}$$

where $\triangle$ denotes the area of Ω_e. This simple approach clearly yields a constant error per element and leads consequently to a constant resulting size field into each high order triangle. Notice that more complex rules

could be investigated for computing e_{Ω_e}, especially when considering anisotropic meshes with high-order polynomials.
An expression of the total error is provided by summing all elementary errors,

$$e^2 = \int_\Omega E^2 \, d\Omega = \sum_e e_{\Omega_e}^2 \ , \tag{4.9}$$

The relatives global and local errors, ϵ and ϵ_{Ω_e} respectively, read:

$$\begin{aligned} \epsilon^2 &= \frac{\int_\Omega E^2 \, d\Omega}{2 \int_\Omega (u^h)^2 \, d\Omega} = \frac{e^2}{2 \parallel u^h \parallel_{\mathcal{L}^2}^2} = \mathcal{O}(h^{2(p+1)}) \qquad (4.10) \\ &= \sum_e \epsilon_{\Omega_e}^2 < 1 \ , \\ \epsilon_{\Omega_e}^2 &= \frac{e_{\Omega_e}^2}{2 \parallel u^h \parallel_{\mathcal{L}^2}^2} \ . \qquad (4.11) \end{aligned}$$

We consider in the following developments the element Ω_e and the new *optimal* element Ω_e^* after adaptation, with the element sizes h_e and h_e^* (computed as the corresponding circumscribed radius for instance) and the polynomial orders p_e and p_e^*, respectively. Since the relative error is proportional to h^{p+1}, those parameters satisfy the relation:

$$\frac{\epsilon_{\Omega_e}}{\epsilon_{\Omega_e}^*} = \frac{h_e^{(p+1)}}{h_e^{*(p^*+1)}} \tag{4.12}$$

where $\epsilon_{\Omega_e}^*$ is the *target* error on the adapted mesh.

4.2.2 Optimal *h*-adaptation strategy

An h-adaptation implies a variation of the element size h while the polynomial order remains unchanged. Therefore, the relation (4.12) reads:

$$\frac{\epsilon_{\Omega_e}}{\epsilon_{\Omega_e}^*} = \left(\frac{h_e}{h_e^*} \right)^{p+1} \ . \tag{4.13}$$

The first simple strategy consists in using a user-defined target error for $\epsilon_{\Omega_e}^*$. The relation (4.13) then provides the new elementary size field:

$$h_e^* = h_e \left(\frac{\epsilon_{\Omega_e}^*}{\epsilon_{\Omega_e}} \right)^{\frac{1}{p+1}}$$

However, this strategy presents the drawback that the user-defined error has to be calibrated for every kind of problem and domain, which may reveals somehow tedious. More elegant techniques consists either in minimizing the number of element while keeping the relative error below a given threshold, or in minimizing the discretization error while keeping a constant number of elements.
The total error in the optimal mesh reads:

$$\epsilon^{*2} = \sum_e \epsilon^{*2}_{\Omega_e} = \sum_e \epsilon^2_{\Omega_e} \left(\frac{h^*_e}{h_e} \right)^{2(p+1)} . \tag{4.14}$$

Clearly, the number of elements in the optimal mesh $\mathcal{N}^*_e$ can be expressed as:

$$\mathcal{N}^*_e = \sum_e \left(\frac{h_e}{h^*_e} \right)^d ,$$

where d denotes the dimension of the problem.

Let us first consider the problem of minimizing $\mathcal{N}^*_e$ while controlling the error ϵ^*. This problem yields the following saddle point optimization problem:

$$\min_{r_e} \max_{\lambda} \left[\sum_e r_e^2 + \lambda \left(\sum_e \epsilon^2_{\Omega_e} r_e^{-2(p+1)} - \epsilon^{*2} \right) \right] ,$$

where λ is a Lagrange multiplier and r_e denotes the element sizes ratio $\frac{h_e}{h^*_e}$. It can be shown that

$$r_e^2 = \left(\lambda (p+1) \epsilon^2_{\Omega_e} \right)^{1/(p+2)} \tag{4.15}$$

and the solution of the problem can be computed in a closed form if p is constant:

$$r_e = {\epsilon^*_{\Omega_e}}^{-\frac{1}{(p+1)}} \epsilon^{\frac{1}{(p+2)}}_{\Omega_e} \left(\sum_e \epsilon^{\frac{2}{(p+2)}}_{\Omega_e} \right)^{\frac{1}{(2p+2)}} . \tag{4.16}$$

The second problem tends to control the number of elements $\mathcal{N}^*_e$ while minimizing the error ϵ^*:

$$\min_{r_e} \max_{\lambda} \left[\sum_e \epsilon^2_{\Omega_e} r_e^{-2(p+1)} + \lambda \left(\sum_e r_e^2 - \mathcal{N}^*_e \right) \right] ,$$

with λ another Lagrange multiplier. Considering a constant value of p, the solution reads:

$$r_e = \sqrt{\mathcal{N}_e^*}\epsilon_{\Omega_e}^{\frac{1}{(p+2)}} \left(\sum_e \epsilon_{\Omega_e}^{\frac{2}{(p+2)}} \right)^{-1/2} . \tag{4.17}$$

The mesh adaptation criterion is then defined for both problems by the r_e's.

4.2.3 *p*-adaptation to a target error

Considering p-adaptivity, the relation (4.12) can be simplified by assuming a constant h for a variable p:

$$\frac{\epsilon_{\Omega_e}}{\epsilon_{\Omega_e}^*} = (h_e)^{p-p^*} . \tag{4.18}$$

The polynomial order is typically restricted to a given range, say $[p_{\min}, p_{\max}]$. For a given target error $\epsilon_{\Omega_e}^*$, the new polynomial order on each element is thus given by

$$p^* = \max\left(\min\left(p - \frac{\ln\left(\epsilon_{\Omega_e}/\epsilon_{\Omega_e}^*\right)}{\ln h_e}, p_{\max} \right), p_{\min} \right) . \tag{4.19}$$

Clearly, the projection from $\boldsymbol{U}$ to $\boldsymbol{U}^*$ between elements of different polynomial orders reads:

$$\boldsymbol{U}^* = \boldsymbol{M}_{p^*}^{p^*-1}\boldsymbol{M}_{p^*}^{p}\boldsymbol{U}$$

where the matrix $\boldsymbol{M}_{\boldsymbol{p}}^{\boldsymbol{q}}$ denotes the projection from polynomial order q into order p and reads:

$$\boldsymbol{M}_{\boldsymbol{p}}^{\boldsymbol{q}} = \int_{\Omega_e} L_i^p L_j^q \, d\Omega ,$$

with L_i^p the ith Lagrange shape function of order p.
The refinement process, corresponding to increase the order p, obviously does not modify the solution. Indeed, the new polynomial space is higher, but contains the lower space. The second operation of coarsening, i.e. reducing the polynomial order, clearly leads to some loss of

information. Indeed, the new space present a lower number of degrees of freedom, leading to the smoothing of the solution. Clearly, this projection may be seen as the projection on a coarser mesh, filtering the higher frequencies. One could take advantage of this filtering process for performing for instance p-multigrid methods [Hillewaert et al., 2006]. This loss of information however takes place into the low error regions, limiting thus the numerical dissipations. Notice finally that this projection clearly leads to a perfect conservation of the fields during the adaptation process.

4.2.4 *h*-adaptation procedures and projection of the solution

For a given metric tensor $\mathcal{M}(\boldsymbol{x})$ on the domain Ω, the aim of the adaptation procedure is to modify the existing mesh to obtain a mesh presenting unit length edges in the given metric. At one given time step, a metric field is computed at every node of the mesh using the *a posteriori* error estimation. Three major questions usually arise when dealing with mesh adaptation and have to be carefully addressed:

- Is there any loss of conservation introduced by the adaptation strategy?
- Is there any significant additional numerical diffusion introduced by adaptation?
- Is there any production of spurious oscillations?

We consider here a mesh adaptation algorithm that modifies locally the current mesh by allowing only the three operations depicted in Figure (4.4):

- Splitting the long edges,
- Collapsing the short edges,
- Swapping edges between badly shaped elements.

The long edges Γ are defined as edges such that $\| \Gamma \| > 1.4$ in the new metric field, while short edges are edges of size $\| \Gamma \| < 0.5$. The local

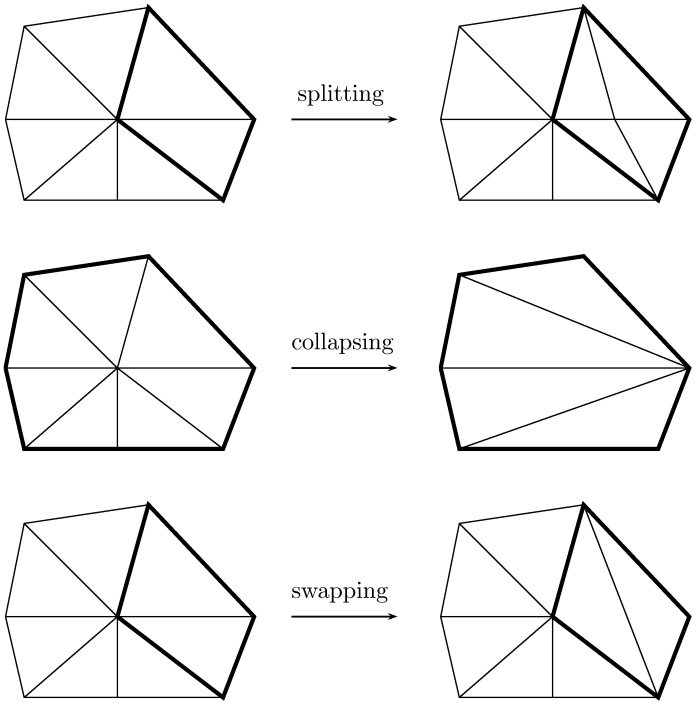


Figure 4.4: Three local mesh modifications of splitting, collapsing and swapping. The adaptation procedure is performed locally, considering only the cavities depicted in bold.

mesh adaptation procedure is applied iteratively until the convergence criterion is satisfied. The algorithm typically stops when every edge of the domain has a dimensionless size in the range $\| \Gamma \| \in [0.5, 1.4]$. Using this interval for short and long edges ensures that the two new edges created by a bisection will not be short edges, preventing therefore possible oscillations between refinements and coarsenings. Notice that the range $[0.5, 1.4]$, sometimes slightly extended, is mainly based on empirical considerations and not on rigourous mathematical justifications. More details about this mesh adaptation procedure may be found for instance in [Remacle et al., 2005, Li, 2003].

As the mesh adaptation itself, the projection of the solution from the original mesh to the adapted one is performed as a local operation, for each adapted cavity, by means of a local $\mathcal{L}_2$ projection. This projection implies that the adaptation procedure does not generate any loss of conservation since the fields are conserved locally. Notice that this projection has to be performed carefully since it requires the integration of discontinuous fields.
The splitting operation for refining the mesh clearly keeps unchanged the DG solution, without generation of oscillation or numerical diffusion. Although this property may seem obvious considering *classical* finite elements, it must be noticed that it is not valid for any kind of finite element. Indeed, the non-conformal P_1^{NC} elements for instance require the field continuity on the mid-edge points. Therefore, refining the mesh with such elements presents serious issues since it may generate strong modifications of the solution.
The coarsening operation, i.e. the collapsing procedure, clearly causes some loss of information, since reducing the number of degrees of freedom. This may lead to numerical dissipation and spurious oscillations during the projection, especially for high-order polynomials. However, this collapsing operation is performed in regions presenting low errors. The modifications of the solution due to the coarsening have thus no significant impact.
The largest issues are certainly due to the last operation of edges swapping. These are not confined in lower error regions, but are performed in the entire domain as the elements become badly shaped, and may thus introduce numerical dissipation especially when using insufficiently accurate integration schemes during the $\mathcal{L}^2$ projection. However, no excessive

amount of diffusion has been observed in the numerical tests. Notice finally that another operation of nodes repositioning is sometimes used, typically using Laplacian smoothing, to improve the elements shape factor. This has not been considered in this work to avoid excessive dissipations.

4.3 *p*-adaptation on a 1D benchmark

We consider here the resolution of the one-dimensional benchmark of Section (3.4):

$$\begin{aligned} \frac{\partial \eta}{\partial t} + H_0 \frac{\partial u}{\partial x} &= 0, \\ \frac{\partial u}{\partial t} + g \frac{\partial \eta}{\partial x} - fv &= 0, \\ \frac{\partial v}{\partial t} + fu &= 0, \end{aligned}$$

using p-adaptivity and a RK44 time scheme. The physical parameters are $g = 9.81 < ms^{-2}$, $H_0 = 100\ m$, $L = 10^6\ m$, $f = 10^{-4}\ s^{-1}$ and $\eta_0 = 1\ m$, while the adaptation parameters are a polynomial order ranging in $[1, 8]$ and a relative target error of 0.03 on a 40 elements grid. The frequency of p-adaptation is $f\Delta t = 0.025$.

We observe the results in Figure (4.5) and Figure (4.6) for non-dimensional times $ft = 0.05, 0.5, 1.2, 2, 5$ and 20. For each of these time steps, the upper plot presents the elevation field, the dots denoting every degree of freedom. The lower plot illustrates both the polynomial order in abscissa and the estimated relative error given by the colored bars, i.e. the ratio between the local error and the total error on the whole domain.
The computation starts with a polynomial order of $p = 4$ everywhere. During the earlier stages of the computation, we observe in Figure (4.5) that the error estimator, i.e. the jumps of elevation, clearly identifies the central region as a large gradients region. The solution is thus directly p-adapted, leading to first order functions everywhere except in the center where $p = 8$ is reached, after only two adaptations. The target error is not reached in this region, obviously because of the restriction of a

maximum polynomial order of 8.
The error estimator and the polynomial order then follows perfectly the quasi-discontinuities, leaving first order elements in the center of the domain. We finally observe that as the waves propagates, the error estimator predicts larger errors in the downwind regions of discontinuities, since the error is increasing and the mesh has not been adapted yet, while the target error constraint is satisfied in the upwind region of decreasing error.

In Figure (4.7) is depicted the time evolution of the number of degrees of freedom (blue continuous line) and the mean number of degrees of freedom. Reaching an equivalent accuracy requires 360 degrees of freedom on the 40 elements grid, instead of a mean of 137.

4.4 2D *h*-adaptation

This section first deals with a validation step on a benchmark problem: the Stommel model. An adaptive convergence experiment is performed in order to test both the convergent behavior of the DG scheme and the efficiency of the adaptive strategy. A second experiment is then performed to simulate the propagation of a typical anticyclonic baroclinic eddy in the realistic domain of the Gulf of Mexico.

4.4.1 Adaptive convergence applied to the Stommel gyre

Interesting simplifications can be done to the shallow water equations in order to obtain the Stommel model, considered as the simplest set of equations to modelize the ocean circulation in the North Atlantic. This simplified problem presents an analytical solution and is used here to perform the following convergence study.

Let us consider the non-conservative form of the shallow water equations (2.22) and (2.23). The non-linear transport terms $\boldsymbol{u} \cdot \nabla(H\boldsymbol{u})$ are neglected. We assume a constant bathymetry H_0 and use a β-plane approximation for the Coriolis parameter, i.e. $f = f_0 + \beta_0 y$, where $f_0 \approx 10^{-4}\ s^{-1}$ and $\beta_0 \approx 10^{-11}\ m^{-1}s^{-1}$. A simple linear bottom friction is used, i.e. $\boldsymbol{\tau}^b = \gamma \boldsymbol{u}$, with γ a constant friction coefficient. The typical

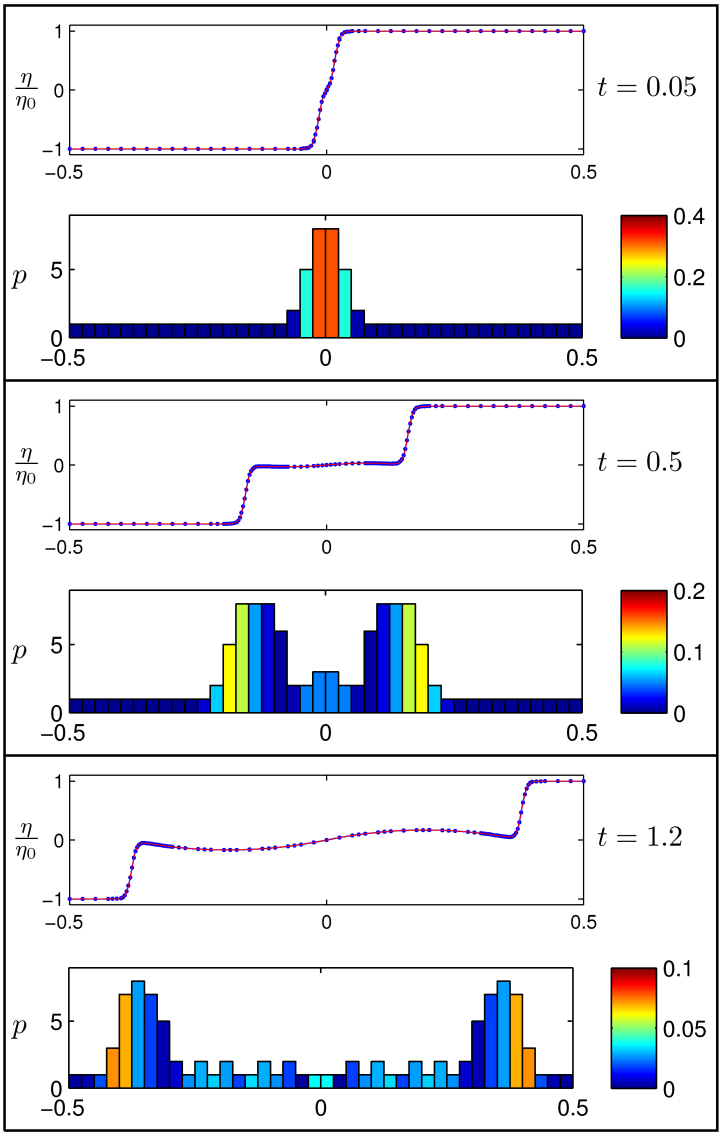


Figure 4.5: Evolution of the polynomial order $p \in [1 \dots 8]$ for non-dimensional times $ft = 0.05, 0.5$ and 1.2 on a 40 elements grid. Colors represent the percentages of estimated error per element, based on the jumps of the elevation field.

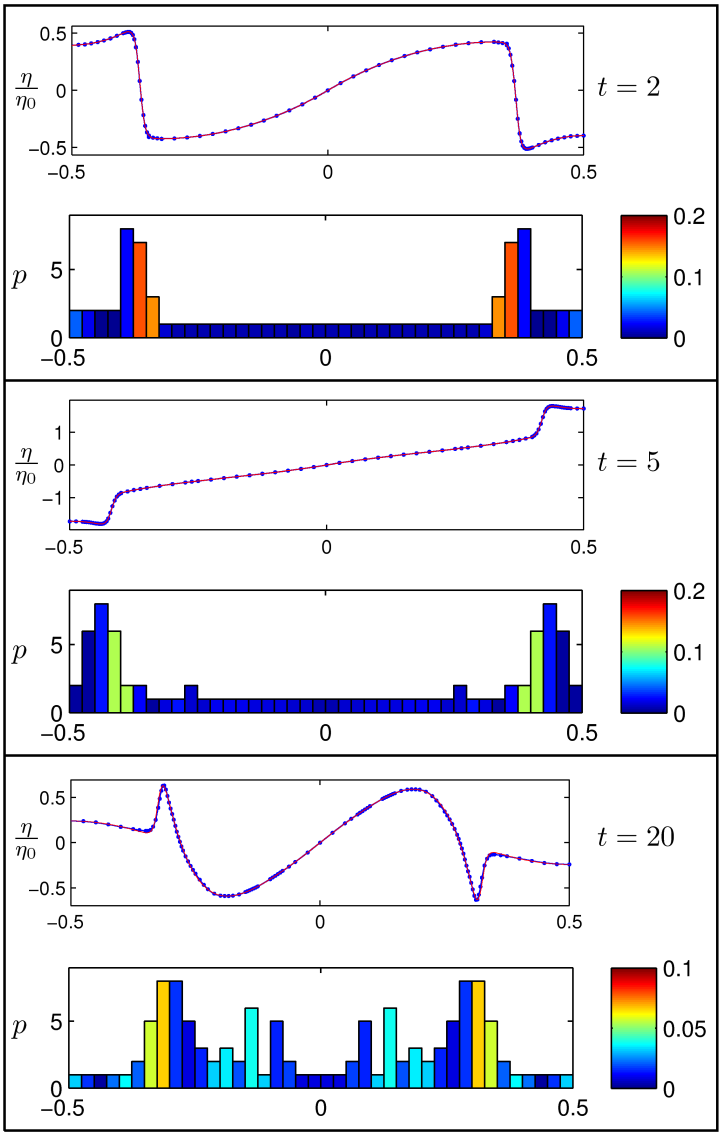


Figure 4.6: Evolution of the polynomial order $p \in [1 \dots 8]$ for non-dimensional times $ft = 2, 5$ and 20 on a 40 elements grid. Colors represent the percentages of estimated error per element, based on the jumps of the elevation field.

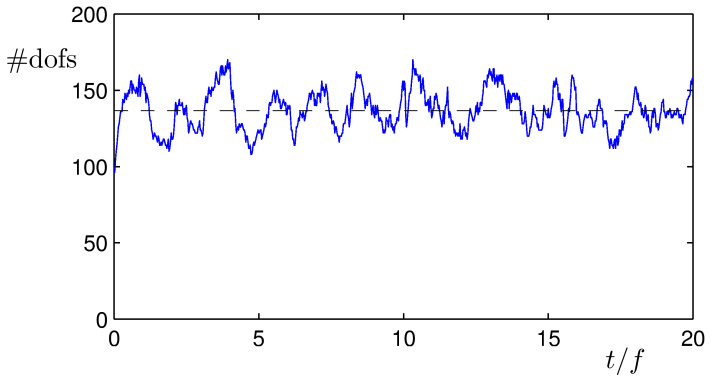


Figure 4.7: Evolution of the total number of degrees of freedom (continuous blue line) with non-dimensional time. The mean number of degrees of freedom is 137 and corresponds to the dashed black line. Ensuring an equivalent accuracy without adaptivity, i.e. using $p = 8$ on 40 elements, would require 360 degrees of freedom instead of 137.

surface stress is given by $\boldsymbol{\tau}^s = \tau_0 \sin(\pi y')$ with $y' = \frac{y}{L_y} \in [-0.5; 0.5]$ the non-dimensional coordinate with L_y the typical size of the domain along the y-dimension. The relative elevation is finally neglected compared to the bathymetry in the continuity equation, leading to the following classical linearization: $H = H_0 + \eta \cong H_0$. With those approximations, the linearized form of the shallow water equations becomes:

$$\frac{\partial \eta}{\partial t} + \nabla \cdot (H_0 \boldsymbol{u}) = 0 \ , \tag{4.20}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + g\nabla\eta + (f_0 + \beta_0 y)\mathbf{e}_z \times \boldsymbol{u} = -\gamma \boldsymbol{u} + \frac{\boldsymbol{\tau}^s}{H_0 \rho}. \tag{4.21}$$

Equations (4.20) and (4.21) form the so-called Stommel model and are solved on a 1000 km square which leads to the typical "Stommel gyre" depicted in Figure (4.8) [Stommel, 1948]. The Coriolis effect leads to a geostrophic balance, creating a recirculation cell. The linear part βy of the Coriolis factor f tends to move the eddy westward (eastward) in the northern (southern) hemisphere, leading to a boundary layer at the western boundary of the domain. The size of this boundary layer is determined by the ratio $\frac{\gamma}{\beta L_x}$. The adaptive method is therefore very

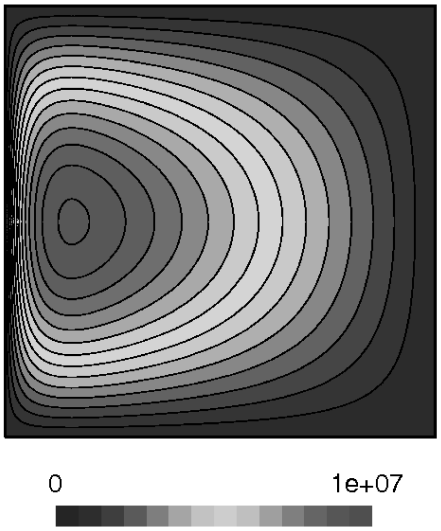


Figure 4.8: Isolines of the stream function $\Psi(x, y)$ obtained for the Stommel model. Notice that Ψ is the stream function of the transport so the units read $m^3 s^{-1}$. The following standard parameters are used : $f_0 = 10^{-4}\ s^{-1}$, $\beta = 2\ 10^{-11}\ m^{-1}s^{-1}$, $\tau_0 = 10^{-1}\ Nm^{-2}$, $\gamma = 10^{-6}\ s^{-1}$, $g = 10\ ms^{-2}$, $h = 10^3\ m$, $\rho = 10^3\ kgm^{-3}$ and $L_x = L_y = 10^6\ m$ the length of the domain along the x and y dimensions.

useful in order to capture the large gradients on this western boundary, while the eastern part of the domain does not require such a fine discretization.
The analytical Stommel gyre solution reads :

$$\begin{aligned}
\Psi(x,y) &= \frac{\mathcal{D}\tau_0 L_y}{\pi^2\gamma\rho} f_1(x)\cos(\pi y) \\
U(x,y) &= \frac{\mathcal{D}\tau_0}{\pi\gamma\rho} f_1(x)\sin(\pi y) \\
V(x,y) &= \frac{\mathcal{D}\tau_0}{\pi\gamma\rho\delta} f_2(x)\cos(\pi y) \\
\eta(x,y) &= \frac{\mathcal{D}\tau_0 f_0 L_x}{\pi\gamma\rho\delta g h}\left[-\frac{\gamma}{f_0\delta\pi} f_2(x)\sin(\pi y)\right. \\
&\quad \left. + \frac{1}{\pi} f_1(x)\left(\cos(\pi y)(1+\beta y) - \frac{\beta}{\pi}\sin(\pi y)\right)\right]
\end{aligned}$$

with the following functions :

$$\begin{aligned}
f_1(x) &= \frac{\pi}{\mathcal{D}}\left(1 + \frac{(e^{z_2}-1)e^{xz_1} + (1-e^{z_1})e^{xz_2}}{e^{z_1}-e^{z_2}}\right) \\
f_2(x) &= \frac{1}{\mathcal{D}}\frac{(e^{z_2}-1)z_1 e^{xz_1} + (1-e^{z_1})z_2 e^{xz_2}}{e^{z_1}-e^{z_2}} \\
\mathcal{D} &= \frac{(e^{z_2}-1)z_1 + (1-e^{z_1})z_2}{e^{z_1}-e^{z_2}} \\
z_2^1 &= \frac{-1 \pm \sqrt{1+(2\pi\delta\epsilon)^2}}{2\epsilon}
\end{aligned}$$

and with x and y the non dimensional axis $x \in [0\ 1]$ and $y \in [-0.5\ 0.5]$. The dimensionless parameters used are the aspect ratio of the domain $\delta = \frac{L_x}{L_y}$ and $\epsilon = \frac{\gamma}{L_x\beta}$ the parameter defining the boundary layer width.

Two different polynomial orders (P1 and P4 elements) have been used in the adapted computations, giving two different theoretical convergence rates, according to (4.13). In the last part of Figure (4.9) is depicted the evolution of the $\mathcal{L}^2$ norm of the discretization error with the characteristic element size on uniformly refined meshes. The theoretical asymptotic behaviour is reached on dense uniform meshes.
The evolution of the $\mathcal{L}^2$ norm of the error with the characteristic element size for a given target error is illustrated in the right part of Figure (4.9),

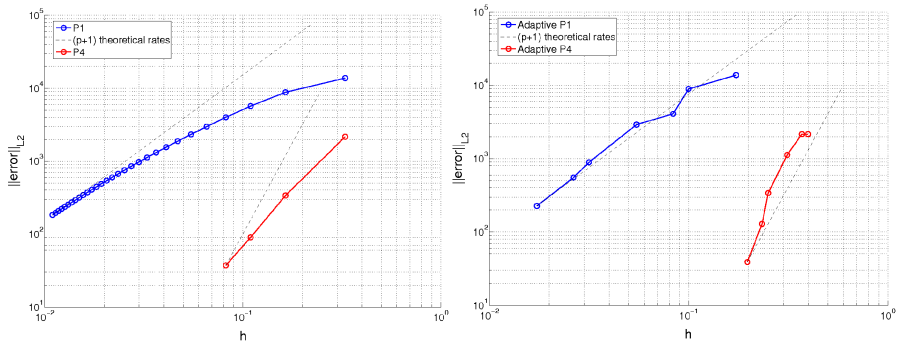


Figure 4.9: Convergence of the $\mathcal{L}^2$ norm of the error ϵ with the local mesh size h using uniformly refined meshes (left) and adaptively refined meshes (right). Theoretical convergence rates (dashed lines) are reached for both polynomial orders 1 (blue) and 4 (red).

with the lines dashed lines corresponding to the theoretical convergence rates.

The numerical results clearly fit the theoretical rates, both with fixed regular mesh and adapted mesh. The kind of oscillatory behaviour on the convergence plot when using adaptive meshes can be explained by the use of the interval $[0.5; 1.4]$ to transform the error estimation into a new edge size field. To reach the same error norm of approximately 5×10^{-3} with linear shape functions, the non-adaptive structured regular mesh required 648 elements while the adaptive method needs only 79 elements. On the same mesh, the error is more than a hundred times smaller with $P4$ element than with $P1$.
The maximum error is also compared to the number of degrees of freedom in Figure (4.10). Let us consider the same target for the maximum error, say about 10^{-2}. We observe that the adaptive strategy with first order elements reduces of a factor 5 the number of degrees of freedom, compared to the non-adaptive computation. Increasing the polynomial order to four leads to about 100 times less degrees of freedom for this same maximum error. The meshes in Figure (4.10) correspond to the first (top right) and fourth (bottom right) order adaptive meshes for approximately the same maximum error of about 10^{-2}, presenting re-

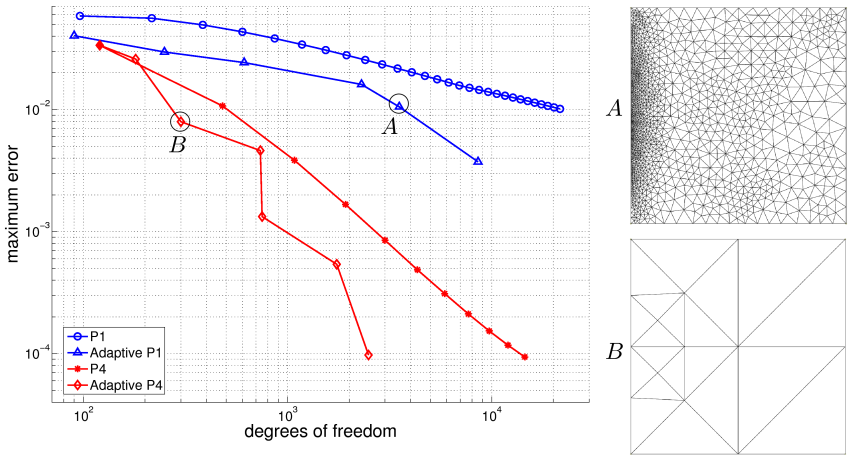


Figure 4.10: The evolution of the maximum error located in the western boundary layer illustrates the efficiency of coupling both adaptivity and high order elements. For the same maximum target error of about 10^{-2}, about 20000 elements are required at $p = 1$ on an uniform mesh, compared to 3000 elements at $p = 1$ with mesh adaptivity, and 20 elements at $p = 4$ with adapted mesh.

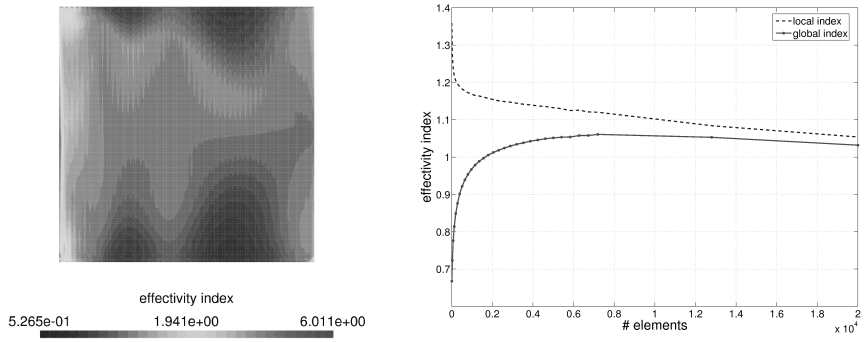


Figure 4.11: The global (dashed) and local (continuous) effectivity indexes computed with first order elements both tend to one, confirming the efficiency of the jumps as error estimators (right). The local effectivity field (left), computed on the uniform 20000 elements mesh, presents the same "boundary layer structure" than the solution and the error.

spectively 20 and more than 1000 elements.

Finally, a classical way to quantify the quality of an error estimator is the effectivity index [Ainsworth and Oden, 2000], defined as the ratio between the norm of the error estimator ϵ and the norm of the exact error e^h. The *global effectivity index* reads:

$$\theta = \frac{\| \epsilon \|_{\mathcal{L}^2}}{\| e^h \|_{\mathcal{L}^2}} \ , \tag{4.22}$$

while the local effectivity index is defined in the same way on each element. With the first order elements, mean jumps have been defined on each node as the mean of the jumps on adjacent elements for the computation of $\| \epsilon \|_{\mathcal{L}^2}$. The evolution of this index with the number of elements is depicted in Figure (4.11). The local index depicted here, computed as the mean of all locals indexes in the domain, reach a value of 1.03. A value so close to one confirms the jumps to be relevant as error estimator. The local effectivity field depicted in Figure (4.11) reveals as expected an error concentration in the boundary layer.

The choice of a mesh metric based on the inter-elements jumps leads to the expected theoretical convergence rate for any polynomial order. If both adaptivity and high-order polynomials have been shown to be efficient separately, we also observed that combining the two approaches is quite simple using the DG method, providing an extremely efficient scheme especially for that applications presenting regions of large gradients.

4.4.2 Propagation of a baroclinic anticyclonic eddy at midlatitudes

The following results concern the propagation of a typical baroclinic anticyclonic eddy at midlatitudes [Hanert et al., 2005] in the Gulf of Mexico. The basin is assumed closed, the Yucatan Channel and the Florida Straits with their inflow and outflow are neglected. Although this experiment is highly idealized, it is expected to represent some of the features of the life cycle of anticyclonic eddies with the adaptive capture of eddies propagation.

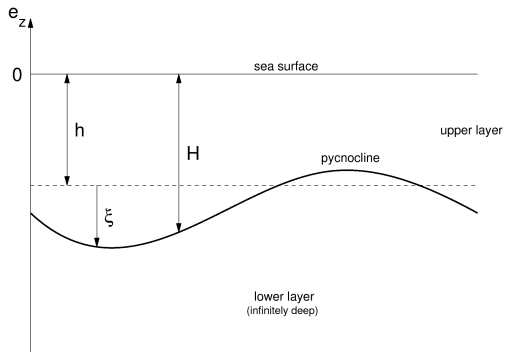


Figure 4.12: Notations for the shallow water reduced gravity model, with the time-independent height h and ξ the downward displacement of the interface between the two layers.

To model the flow in the Gulf of Mexico, it is essential to take into account its baroclinic aspects. In this respect, the simplest approach consists in assuming that the domain is divided into two constant density layers separated by a free, impermeable interface: the pycnocline. As the lower layer is much deeper than the top layer, it is possible to further simplify the model to a reduced-gravity one. Such a model is widely used in oceanography and limnology [Naithani et al., 2003]. To establish reduced-gravity equations, a closure hypothesis on the pressure force is needed, that eventually leads to a closed set of equations governing the motion in the surface layer. Accordingly, the downward displacement of the pycnocline, ξ, and the depth-averaged velocity in the surface layer, $\boldsymbol{u}$, obey equations that are similar to the classical shallow-water equations:

$$\frac{\partial \xi}{\partial t} + \nabla \cdot (H\boldsymbol{u}) = 0 \ , \tag{4.23}$$

$$\frac{\partial H\boldsymbol{u}}{\partial t} + \nabla \cdot (H\boldsymbol{u}\boldsymbol{u}) + f\boldsymbol{e}_z \times H\boldsymbol{u} = -g'H\nabla\xi \ , \tag{4.24}$$

where $H = h+\xi$ is the actual depth of the upper layer (see Figure (4.12)), given that h is the unperturbed equilibrium height of the surface layer. The so-called reduced gravity g' is defined as $g' = g\delta$, with g the gravity and $\delta = \frac{\Delta\rho}{\rho}$ the relative density difference between the bottom and the surface layers.

A Gaussian distribution of water elevation η is assumed at initial time:

$$\eta(x, y, t = 0) = C \exp[-D(x^2 + y^2)] \ , \tag{4.25}$$

with $C = 68.2\ m$ and $D = 5.92 \times 10^{-11}\ m^{-2}$. The β-plane assumption is made, i.e. $f = f_0 + \beta y$, with the Coriolis parameters taken at $25N$: $f_0 = 6.1635\ 10^{-5}\ s^{-1}$ and $\beta_0 = 2.0746\ 10^{-11} m^{-1} s^{-1}$. The initial velocity field is computed to be in geostrophic balance and reads:

$$u(\boldsymbol{x}, t = 0) = -\frac{g}{f}\frac{\partial \eta}{\partial y} = 2\frac{g'}{f_0 + \beta_0 y} CDy \exp[-D(x^2 + y^2)],$$

$$v(\boldsymbol{x}, t = 0) = \frac{g}{f}\frac{\partial \eta}{\partial x} = -2\frac{g'}{f_0 + \beta_0 y} CDx \exp[-D(x^2 + y^2)],$$

leading to a maximum initial flow speed of 1 ms^{-1} with $g' = 0.137\ ms^{-2}$ and $h = 100\ m$. The maximum Froude number reached during the simulation is 0.5, which is very high compared to classical shallow water

flows in the ocean, and the Rossby number is $Ro = \frac{\sqrt{gH_0}}{Lf_0} = \cong 9 \times 10^{-3}$.

There is no wind forcing or bottom friction. The Coriolis effect is thus the only source term responsible for moving the eddy westward. This propagation of slow Rossby waves, represented with the shallow water equations, is due to the β-effect. Such waves have a major effect on the large scale circulation, and thus on weather and climate. For instance, Rossby waves can intensify western boundary currents, which transport huge quantities of heat. Even a minor shift in the position of the current can thus affect weather over large areas of the globe.

In Figure (4.13) is represented the time evolution of the eddy for $p = 1$. Its westward propagation is captured by the mesh evolution (right column): a minimum size field of ten kilometers has been applied to keep a low number of elements.
After one week of physical time simulation, the mesh has been adapted three times to capture the regions with larger variations, i.e. in the vicinity of the eddy and along the coast lines. At this time, the mesh presents approximately 3000 elements. The eddy keeps moving westward until approximately week 11, when it reaches the coast. Its shape is then modified when a second eddy appears, spinning in the opposite direction. The number of elements grows up to 7000 in order to resolve this large variation region. On week 14, one can see the slow creation of a western boundary current, flowing southward. As the eddy keeps moving southward, the mesh seems to perfectly capture the evolution of this current and the eddy generated at the southern extremity of the gulf on week 23. The initial eddy then collapses to generate smaller eddies, spinning and mixing on the western boundary. The number of elements decreases eventually to reach the initial value of about 3000.
The mesh has been well adapted to eddies and currents, but large field variations and mesh refinement must also be noticed on sharp irregular coasts, as on the north and east-north of the Gulf. A restriction of ten kilometers has been applied on the size of elements on adaptive mesh. To reach the same accuracy with a non-adaptive mesh, about 34000 elements would be required with the first order polynomial shape functions.

Examples of mesh size fields are eventually depicted in Figure (4.14) with their corresponding adapted mesh. The upper mesh captures the

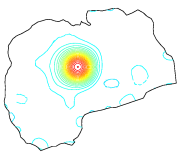
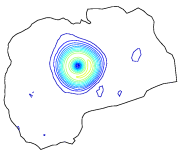
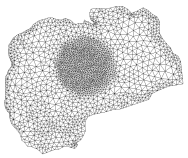
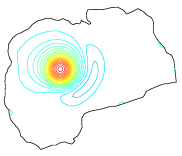
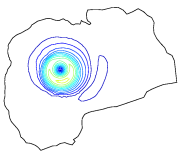
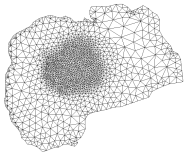
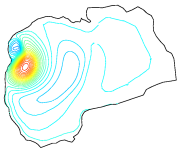
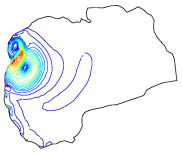
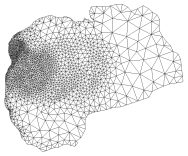
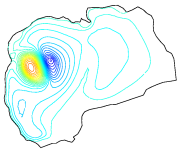
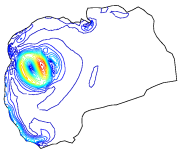
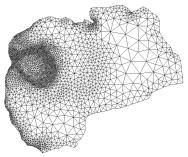
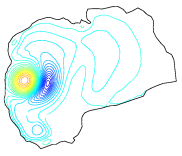
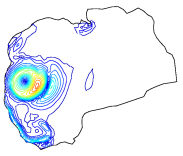
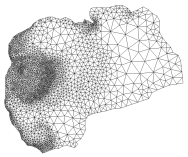
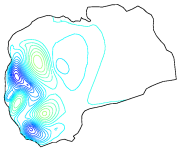
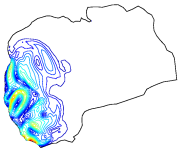
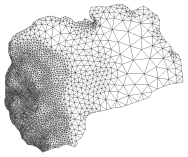
Week	η	velocity	adaptive mesh
1			
3			
14			
20			
23			
37			

Figure 4.13: Isolines of elevation and velocity norm and the corresponding adaptive mesh at different times of the eddy propagation for $p = 1$. On each plot, the 40 iso-η lines extend from -35 m (blue lines) to 65 m (red lines), while the 20 isovelocity lines extend from 0 m/s to 1.5 m/s.

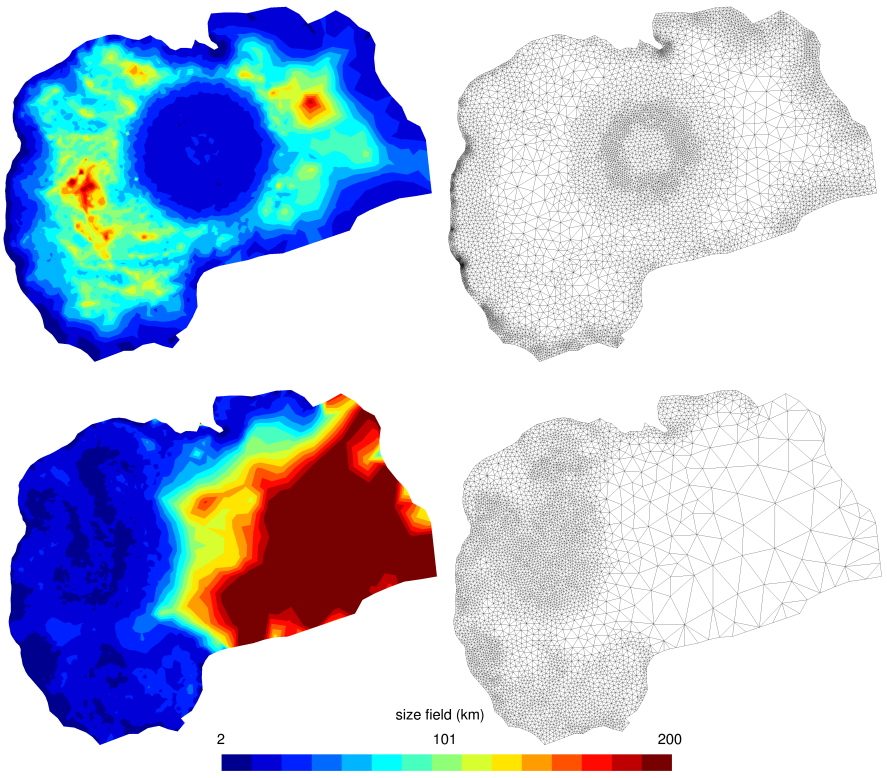


Figure 4.14: Examples of mesh size fields (left) and corresponding adapted meshes (right) obtained at different time steps. The mesh on top presents 14545 triangles to capture the eddy in the middle of the Gulf while the bottom mesh presents 9618 triangles capturing the western boundary current.

eddy during its westward propagation, using 14545 elements, while the mesh in lower part of the figure exhibits 9613 elements capturing the western boundary layer.

4.5 Conclusions

An adaptation strategy using the DG method has then been proposed. The DG method has been shown to be really efficient for adaptivity because of the cheap error estimation contained in the inter-element jumps. Moreover, the proposed strategy involves only local mesh and field modifications. Local modifications ensure local conservation of the fields, drastically reduces the computational costs and especially the memory footprint, and provides simple adaptive strategies in parallel. Finally, the strategy proposed here without nodes movements or mesh optimization introduces a very small amount of numerical diffusion compared to standard approaches. The DG method is therefore very well suited for adaptive strategies in ocean applications. Other techniques require global modifications and complex error estimations, but moreover, mixed formulations as the $P_1^{NC} - P_1$ pair lead to very complicated mesh adaptations. Indeed, even the mesh refinement procedures modify the solution and introduce significant errors, while the refined fields remain unchanged with the DG approach.

The method has been successfully applied to $h-$adaptive computations as the Stommel benchmark or the propagation of an anticylonic eddy in the Gulf of Mexico. A one-dimensional benchmark has also enlightened the efficiency of $p-$adaptation strategies. Both the use of high-order shape functions and adaptive strategies with the DG method lead to efficient computational techniques. But the computational costs may be drastically reduced when combining those two techniques, which is easily achieved with the DG method.
Notice finally that adaptive computations performed on simple tidal flows have provided mesh size fields proportional to the gravity waves speed $\sqrt{gH}$. This criteria, usually taken into account in the mesh generation for non-adaptive computations, is also obtained using our *a posteriori* error estimator based on the jumps of the solution.

Two restrictions have however to be mentioned at this point concerning mesh adaptation for oceanic computations. The first restriction concerns $h-$adaptivity in general when combined to large variations of the bathymetry. Some preliminary results have been obtained by intersecting the metric corresponding to the optimal mesh with a metric based on the bathymetry variations, in order to minimize the modifications of the bathymetry during the adaptation process. This strategy is particularly efficient if a preliminary smoothing of the bathymetry is performed according to the initial mesh resolution. Indeed, using rough irregular bathymetry data in adaptive computations is somehow contrary to common sense. However, even small modifications of the bathymetry may lead to conservation issues for long time computations as climate studies. Further investigations on the subject should be considered, but if $h-$adaptation is very efficient and recommended for local coastal oceanic computations, it is not yet well suited for long time computations with large bathymetry variations at this point. On the other hand, $p-$adaptation does not present the same drawback when considering an initial bathymetry belonging to the lower polynomial space. This clearly seems, at this point, the optimal solution for performing adaptations with the DG method for simulating realistic ocean processes.
A second remark must be formulated concerning the use of $h-$adaptation with high-order polynomials in complex geometries. Indeed, irregular coastlines leads to large gradients, and therefore to large error estimations. A simple naive refinement of the mesh boundaries may lead to very sharp coastlines. When combined to very accurate high-order polynomials, some irrelevant physical processes may appear as the creation of Von Karman streets. A more realistic solution would clearly be obtained by adapting the coastlines according to some embedded CAD representing the coasts as high-order curves rather than by simply adapting according to the boundary elements edges. Notice that the representation of coasts using NURBS (Non-Uniform Rational Basis Splines) presents a large potential for such applications since coarsening operations on the boundaries keep the geometry unchanged. Finally, low-order representations of the coasts with high-order polynomials may lead under certain circumstances to spurious oscillations, typical of high-order methods. This latter point is the subject of the next chapter.

Chapter 5

High-order geometries for the Discontinuous Galerkin method

Contents

5.1 High-order representation of the boundaries 106

5.1.1 Discrete DG formulation of viscous shallow water equations 108

5.1.2 Cylinder test case and the impact of the boundaries discretization 112

5.1.3 Rattray island in a realistic tidal flow 120

5.2 Local high-order tangent basis approach on curved manifolds 135

5.2.1 Usual approaches for solving PDE's on curved manifolds 137

5.2.2 Local Tangent Basis approach using the Discontinuous Galerkin method 142

5.2.3 Solving the shallow water equations on curved manifolds 154

5.3 Conclusions 166

The first part of this chapter is devoted to the treatment of the boundaries with the high-order DG method. Indeed, it has been shown

that using high-order polynomials to represent the unknown fields coupled to a poor representation of the boundaries may lead under certain circumstances to oscillations and instabilities of the method. A method is proposed here using high-order mappings representing curved triangles to represent the coasts. This approach is applied to the realistic computation of tidal flows around the Rattray island, including a modelization of the subgrid scale processes, real tidal forcings and an irregular bathymetry. The results are then compared to in-situ measurements.
The second part of the chapter deals with another aspect of curved geometries, i.e. computations of vectorial conservation laws on curved manifolds. An alternative approach is proposed using the DG method, combining the advantages of both standard methods described in the literature. This robust, general and efficient method is applied to the computation of atmospheric benchmarks on the sphere, discretized using high-order curvilinear triangles.

5.1 High-order representation of the boundaries

The discontinuous Galerkin method is known to be very efficient at high polynomial orders, because of its superconvergence properties on the discretization and dispersion errors. High-order function spaces are commonly used with the DG method in classical engineering problems, involving relatively smooth geometries compared to the mesh resolution, as for instance computations around an airfoil. Indeed, the DG method reveals much more efficient than low order methods in simple geometries as the Stommel model in a square.
Yet the application of high-order DG method to realistic processes of ocean modeling is still a challenging issue, especially because of the complexity of the geometrical features as coastlines or bathymetry. As the complexity of the geometry or bathymetry increases, the interpolation order is commonly decreased [Aizinger and Dawson, 2002] to avoid the classical issues due to high-order spaces.
The aim of this section is to compute coastal processes, involving realistic complex geometries and bathymetry, with high-order polynomials, in order to benefit from its convergence properties.

In the first subsection are presented the governing equations and their classical DG discretization including the derivation of the weak formulation with source and diffusion terms to take into account both a complex bathymetry and the subgrid scale processes, respectively. The next subsection deals with the issue of the boundaries discretization in the context of quasi steady flows at low Froude numbers. Bassi and Rebay [Bassi and Rebay, 1997a] have shown that a poor representation of the boundaries with piecewise linear edges does not lead to a steady state solution for the benchmark of a cylinder in a mean flow using the Euler equations. The high order DG method produces oscillations and spurious eddies in the wake of smooth bluff bodies with steady flows computations. We observe the same behavior with the shallow water equations. A converged solution is only obtained with a more accurate boundary representation as a high order mapping between the reference element and the real one or with techniques as the smoothing of the normals as proposed by Krivodonova and Berger [Krivodonova and Berger, 2006]. But in the presence of thousands of islands and coasts, we are restricted to the use of coarse meshes which means a relatively low mesh resolution close to the boundaries. With too coarse meshes and especially irregular bluff bodies as islands, we observe that spurious oscillations are still generated with the reconstruction of a continuous linear representation of the normals, while the use of high order mappings still avoids oscillations. Moreover, the continuous linear normals does not allow an accurate representation of the shape of boundaries for very irregular bluff bodies.
Finally, the last results deal with the computation of tidal flows around the Rattray island located in the Great Barrier Reef. The study of a hydrodynamic model of the Great Barrier Reef is particularly interesting since the coastal processes in this region exhibit the same kind of scale ranges than the world ocean in a smaller domain of computation [Lambrechts et al., 2008b]. The main large scale forcing is the oceanic inflow of the East Australian Current coming from the Coral Sea. This current flows westward into the shelf region and splits in two branches, creating a northward and a southward flowing current [Wolanski, 1994]. Those large scale long shore currents may be strongly deviated by the wind, the tides and the topography. Thousands of islands and reefs generate small scale eddies and tidal jets [Wolanski et al., 1996]. It is mandatory to simulate all scales simultaneously since small and large scale processes

experience strong interactions. Moreover, increasing the resolution in a small area allows to study this area without the drawback of imposing arbitrary boundary conditions around it.
In this framework of a large hydrodynamic model or world ocean model, we wish to perform flow computations around shallow water islands with meshes as coarse as possible. The realistic test case of the Rattray island is particularly interesting because many in situ measurements of elevation, velocity, tidal forcings, topology and bathymetry are available to validate and calibrate the model. It is clear that we are not interested in solving the boundary layer of the island. All we expect is to reach the correct rate of vorticity and the correct dispersion relation for wave propagation.

5.1.1 Discrete DG formulation of viscous shallow water equations

In the general mathematical framework of the discontinuous Galerkin methods, we consider a system of conservation laws written in the *conservative* form:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}) + \nabla \cdot \boldsymbol{D}(\boldsymbol{U}) \tag{5.1}$$

where $\boldsymbol{U}$ is the vector of unknown fields, $\boldsymbol{F}$ is the convective flux matrix containing the pressure and transport terms, $\boldsymbol{S}$ is the vector containing the so called source terms which do not involve any derivative of the unknown fields, and $\boldsymbol{D}$ is the diffusive flux matrix.

The domain of computation Ω is then divided into a set of $\mathcal{N}_e$ non overlapping elements Ω_e:

$$\Omega = \bigcup_{e=1}^{\mathcal{N}_e} \Omega_e \ .$$

The standard discontinuous Galerkin weak formulation is obtained by multiplying equation (5.1) by a smooth function $\hat{\boldsymbol{U}}$ and integrating on each element Ω_e:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{U}\hat{\boldsymbol{U}}\, d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{F}(\boldsymbol{U})\hat{\boldsymbol{U}}\, d\Omega = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U})\hat{\boldsymbol{U}}\, d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{D}(\boldsymbol{U})\hat{\boldsymbol{U}}\, d\Omega \tag{5.2}$$

We seek an approximation $\boldsymbol{U}^h$ of $\boldsymbol{U}$ such that $\boldsymbol{U}^h \in \mathcal{U}_p^h$ with $\mathcal{U}_p^h$ the finite dimensional space composed of polynomial functions of order at most p on Ω_e. We define on each element a discrete representation $\boldsymbol{U}^h$ of $\boldsymbol{U}$ as

$$\boldsymbol{U}^h = \sum_{i=1}^{\mathcal{N}_d} \phi_i \boldsymbol{U}_i \quad \in \mathcal{U}_p^h$$

where $\boldsymbol{U}_i$ are the nodal values, ϕ_i are a set of independent shape functions and $\mathcal{N}_d$ is the number of degrees of freedom on one element. Although there are many possible choices for the shape functions, we will consider Lagrange shape functions in the following developments. Therefore, the number of degrees of freedom on each element is related to the polynomial order p by the relation $\mathcal{N}_d = (p+1)(p+2)/2$. The weak formulation is then discretized as:

$$\begin{aligned}\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{U}^h \hat{\boldsymbol{U}^h}\, d\Omega + \int_{\Omega_e} \nabla \cdot \boldsymbol{F}(\boldsymbol{U}^h)\hat{\boldsymbol{U}^h}\, d\Omega = \\ \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\hat{\boldsymbol{U}^h}\, d\Omega + \int_{\Omega_e} \nabla \cdot \boldsymbol{D}(\boldsymbol{U}^h)\hat{\boldsymbol{U}^h}\, d\Omega \end{aligned} \tag{5.3}$$

with the test function $\hat{\boldsymbol{U}}^h \in \mathcal{U}_p^h$.
Two neighboring elements in continuous finite element method share common nodes that ensure the continuity of the finite element approximation and that allows an information transfer between elements. As in the one-dimensional example of Chapter (3), it is mandatory to ensure the same coupling between elements with the DG method, by the following integration by part on the convective and diffusive terms:

$$\begin{aligned}\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{U}^h \hat{\boldsymbol{U}^h}\, d\Omega - \int_{\Omega_e} \boldsymbol{F}(\boldsymbol{U}^h)\cdot\nabla\hat{\boldsymbol{U}^h}\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{F}(\boldsymbol{U}^h)\cdot\boldsymbol{n}\hat{\boldsymbol{U}}^h\, d\Gamma = \\ \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\hat{\boldsymbol{U}^h}\, d\Omega - \int_{\Omega_e} \boldsymbol{D}(\boldsymbol{U}^h)\cdot\nabla\hat{\boldsymbol{U}^h}\, d\Omega + \int_{\partial\Omega_e} \boldsymbol{D}(\boldsymbol{U}^h)\cdot\boldsymbol{n}\hat{\boldsymbol{U}}^h\, d\Gamma \end{aligned} \tag{5.4}$$

where $\boldsymbol{n}$ denotes the outgoing unit normal to element Ω_e and Γ_e the edges of this element.
The flux function used in this work is the simple Lax-Friedrich flux (3.56):

$$\boldsymbol{F}(\boldsymbol{U})\cdot\boldsymbol{n} = 0.5\left(\boldsymbol{F}(\boldsymbol{U}_L) + \boldsymbol{F}(\boldsymbol{U}_R)\right)\cdot\boldsymbol{n} + 0.5|\lambda|_{max}\left(\boldsymbol{U}_L - \boldsymbol{U}_R\right)$$

where $|\lambda|_{max}$ denotes the maximum absolute value of the eigenvalues of the system, i.e. the propagation speed of the information. Other more complex Riemann-like solvers can be used as for instance a Godunov solver or a Roe solver [Godunov, 1959, Roe, 1981, Toro, 1997, LeVeque, 2002], whose expression for the shallow water equations may be found in [Remacle et al., 2006, Bernard et al., 2007]. However, the results obtained for the shallow water equations around irregular bluff bodies do not present significant differences for any of those fluxes.

The discontinuities between elements also lead to many different ways of treating the diffusive flux [Arnold et al., 2002]. Indeed, the solution gradients are not well defined at the discontinuities, as the solution itself. A possible simple method is the classical *Local Discontinuous Galerkin* method (LDG) proposed by Bassi and Rebay [Bassi and Rebay, 1997b, Cockburn and Shu, 1998b]. This technique consists basically in computing the solution gradients at the interfaces as a function of the gradients on both surrounding elements, and of the jumps of the solution. This method is well suited for explicit time schemes as used in this work. However, we considered here the *Interior Penalty* method [Douglas and Dupont, 1976, Riviere et al., 1999] for computing diffusion operators, which reveals more easy and efficient for handling implicit time schemes.

Application to the shallow water equations

Let us first write the viscous conservation equations (2.20) and (2.21) in the generic form (5.1). Notice that the diffusion term does not represent the viscous terms of the Navier-Stokes equations, which are neglected, but represents the subgrid scale processes. It is then mandatory to modelize the subgrid scale viscosity ν as a function of the element size, especially in multi-scale computations as finite element ocean modeling. At the bottom, the friction is parametrized with a quadratic relationship where $\| \boldsymbol{v} \|$ is the amplitude of the velocity and γ is a friction coefficient. Both equations can be injected in the generic form (5.1), by defining $\boldsymbol{U}$,

$\boldsymbol{S}$, $\boldsymbol{D}$ and $\boldsymbol{F}$ as follows:

$$\boldsymbol{U} = \begin{bmatrix} \eta \\ Hu \\ Hv \end{bmatrix},\ \boldsymbol{F}(\boldsymbol{U}) = \begin{bmatrix} Hu & Hv \\ Huu + g\frac{H^2-H_0^2}{2} & Huv \\ Hvu & Hvv + g\frac{H^2-H_0^2}{2} \end{bmatrix},$$
$$\boldsymbol{S}(\boldsymbol{U}) = \begin{bmatrix} 0 \\ g\eta\partial_x H_0 + fHv - \gamma u \parallel \boldsymbol{v} \parallel \\ g\eta\partial_y H_0 - fHu - \gamma v \parallel \boldsymbol{v} \parallel \end{bmatrix},\ \boldsymbol{D}(\boldsymbol{U}) = \begin{bmatrix} 0 & 0 \\ \nu H\partial_x u & \nu H\partial_y u \\ \nu H\partial_x v & \nu H\partial_y v \end{bmatrix} \quad (5.5)$$

Large scale ocean processes as basin gyres or anticyclonic eddies are usually dominated by the geostrophic balance, i.e. the balance between the pressure term and the Coriolis term. The aim is to compute the perturbations of this equilibrium as the Rossby waves or the advection terms, but the pressure term $gH\nabla\eta$ is the dominating term in the system for that kind of processes. To express it as flux with a divergence form, we observe that it must be written as $g\nabla(H^2 - H_0^2) - g\eta\nabla H_0$. With a non constant bathymetry, the dominating term of the system is thus split in two parts: one computed through the fluxes with an approximate Riemann solver and the other one considered as a *source* term. It is then mandatory to integrate accurately those two terms to avoid spurious gravity waves. Indeed, strong oscillations and spurious waves may be observed, especially with a strongly varying bathymetry, when using quadrature free techniques or very distorted curved elements which may involve lack of accuracy in the integration process. Besides the exact integration, to be consistent, it is mandatory to consider the *discrete* gradients of the bathymetry for computing the terms $g\eta\nabla H_0$. We seek to determine the discrete bathymetry gradients $b = \nabla H_0$. Considering a continuous representation of the bathymetry H_0^h at order p, the discontinuous discrete gradients $\boldsymbol{b}^h = (b_x^h, b_y^h)$ at order q are computed on each element as:

$$b_{kj}^h = \left(\int_{\Omega_e} \phi_i^q \phi_j^q \, d\Omega \right)^{-1} \int_{\Omega_e} \frac{\partial H_0^h}{\partial x_k} \phi_i^q \, d\Omega, \text{ with } i,j \in [1, \mathcal{N}_d] \ . \quad (5.6)$$

We must finally notice at this point that, depending on the application, it could be useful to consider the alternative non-conservative formulation of the shallow water equations (2.22) and (2.23). Although both formulations are strictly equivalent in the continuous world, their

discretizations may lead to quite different discrete algebraic system. Indeed, the advection terms have to be split in two terms to be written in the divergence form, while the pressure term $g\nabla\eta$ is no more split. Therefore, where the pressure term appears to be the main driving term, this second formulation could appear to be more robust and accurate since a lack of accuracy in the integration will not generate spurious waves as in the first formulation.

5.1.2 Cylinder test case and the impact of the boundaries discretization

A first simple benchmark consists in a cylinder in a constant mean flow. Without diffusion process, we should observe the convergence to a steady well known solution, as for the Euler equations. The second more realistic benchmark of the Rattray island will then be investigated and the results compared to in-situ measurements. The following computations have been performed using an explicit classical Runge-Kutta time scheme.

Let us consider a cylinder in a mean flow with a Froude number of $Fr = 0.2$. This non dimensional number defined as the ratio between the fluid velocity and the gravity waves speed $\sqrt{gH}$ is similar to the Mach number in the Euler equations. All the dissipation processes and source terms are neglected, i.e. there is no Coriolis force, no bathymetry gradient and no bottom friction nor wind forcing. We use a bathymetry of 1 m which corresponds for a gravitational acceleration of $g = 9.81\ ms^{-2}$ to the imposition of a zero elevation and a constant velocity of $0.2\sqrt{9.81} \cong 0.63\ ms^{-1}$ on the inflow and outflow boundaries. A slip boundary condition on the cylinder is also mandatory to obtain a steady converged solution.

With the use of the DG method and a piecewise linear representation of the cylinder boundary, the solution does not reach the expected steady state unless a special treatment of the boundaries is applied, as shown by Bassi and Rebay for the 2D Euler equations [Bassi and Rebay, 1997a]. Oscillations appear in the vicinity of the obstacle. They grow and propagate in constant or quasi-constant mean flows as tidal flows, leading

to the creation of vorticity and of a wake behind the bluff body. Those oscillations are enforced when using higher polynomial orders. This issue is illustrated in the left part of Figure (5.1): strong oscillations are observed with second order Lagrange shape functions and a piecewise linear representation of the boundaries, generating non physical eddies in the wake of the cylinder and polluting the solution far away from the boundaries.

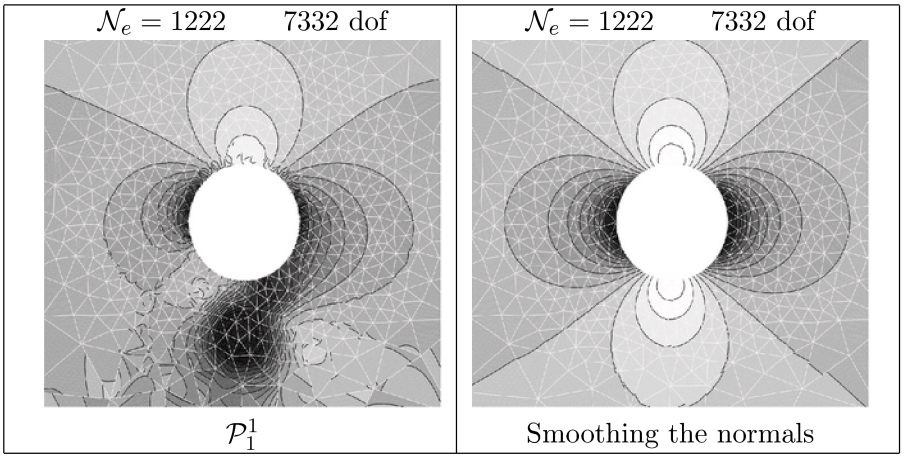


Figure 5.1: Water elevation field for a mean flow around a cylinder with second order Lagrange shape functions and a Froude number of Fr = 0.2. Oscillations appear with a classical piecewise linear representation of the boundaries (left) and prevents convergence to the steady solution. The smoothing of the normals proposed by Krivodonova and Berger suppresses those oscillations and lead to the converged solution (right).

Many techniques allow us to obtain the expected solution. A very cheap and efficient technique proposed by Krivodonova and Berger [Krivodonova and Berger, 2006] consists in building a continuous representation of the normals, and to use those smooth normals in the Riemann solver computation, while keeping the discontinuous representation in the integration on the edges. This continuous representation is depicted in the left column of Figure (5.2). This technique yields the expected steady solution as depicted in the right part of Figure (5.1).

However, the drawback of this technique lies in the reduction of the convergence rates when using sufficiently high order shape functions. Moreover, although this technique works perfectly for sufficiently refined meshes and smooth bluff bodies, oscillations may rapidly appear when using very irregular bluff bodies. Finally, the linear reconstruction of the normals does not represent accurately very irregular obstacles, as depicted in the left part of Figure (5.3).

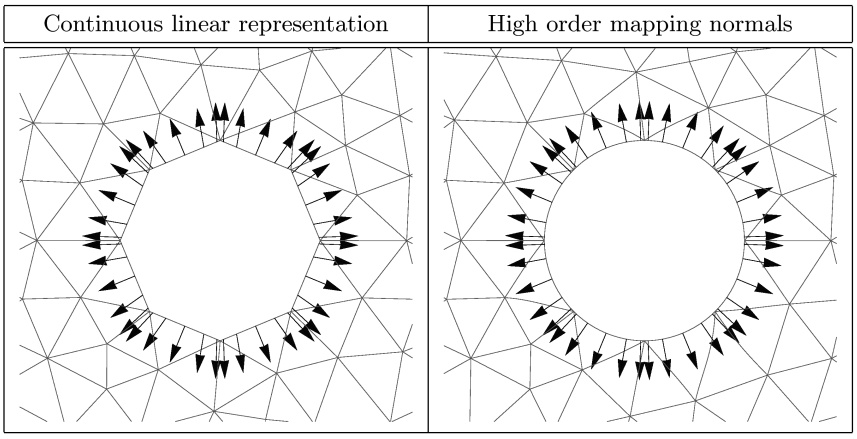


Figure 5.2: The continuous representation of the normals according to Krivodonova and Berger (left) is quite similar to the normals computed from a high order mapping of the triangles (right).

Another approach used in this work consists in building high order meshes, i.e. using curved triangles with high order polynomial mappings, to obtain a better representation of irregular bluff bodies. We consider a triangle composed of a set of $\mathcal{N}_c$ coordinates $\boldsymbol{X}$. A possible continuous representation $\boldsymbol{x} \in \mathcal{R}^2$ of the triangle is given by the simple polynomial mapping

$$\boldsymbol{x}_j = \sum_{i=1}^{\mathcal{N}_c} \psi_i(\boldsymbol{\xi}) \boldsymbol{X}_{i,j}, \qquad j = 1, 2 \text{ in 2D} \tag{5.7}$$

where $\psi(\boldsymbol{\xi})$ denotes the shape functions corresponding to the coordinates $\boldsymbol{X}$ in the bi-dimensional parametric space $\boldsymbol{\xi} = (\xi, \eta)$.
A possible representation consists in using Lagrange shape functions as

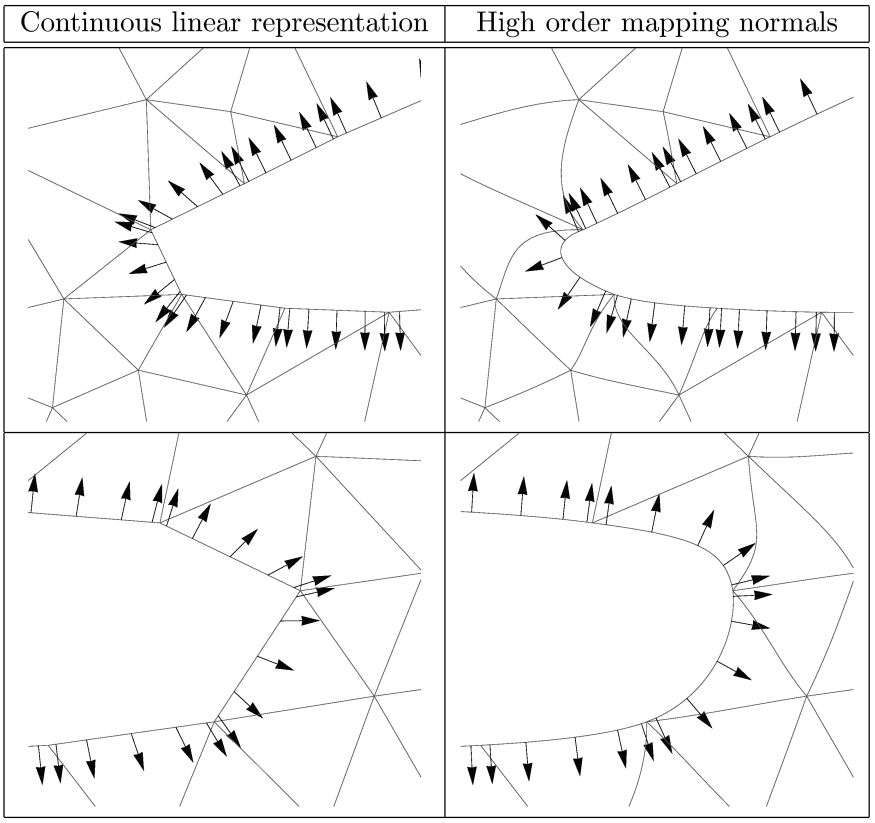


Figure 5.3: Different normals representations on the west (upper row) and east (bottom row) parts of the Rattray island. The continuous representation of the normals (left) becomes quite different from the normals computed from a high order mapping of the triangles (right) for very irregular bluff bodies.

for the polynomial representation of the solution. A mapping of order q is then obtained for a number of coordinates $\mathcal{N}_c$ such that

$$\mathcal{N}_c = \frac{(q+1)(q+2)}{2} \ .$$

However, the high order Lagrange shape functions present nodes located inside the triangle. The position of those nodes is somehow arbitrary and does not bring more informations on the boundary representation, while the nodes coordinates on the edges are computed for instance as equidistant on the parametric curve representing the boundary. To avoid this problem we use in this work Serendip shape functions, i.e. incomplete shape functions at order q, with nodes located only on the edges of the elements. We thus specify for a q-order *Serendip mapping* a number of coordinates:

$$\mathcal{N}_c = 3q \ .$$

Let us introduce the following notation $\mathcal{P}_p^q$, denoting a p-order representation of the solution, using Lagrange shape function, and a q-order incomplete Serendip mapping for the representation of the curved mesh.

Considering smooth bluff bodies as the cylinder, the continuous linear representation of the normals as proposed by Krivodonova and Berger is quite similar to the normals based on a high order polynomial mapping, as depicted in Figure (5.2) where a third order mapping is used. But the difference becomes significative on very irregular boundaries and coarse meshes as depicted in Figure (5.3). For this kind of irregular bluff bodies, the use of high order mappings provides a better representation of the boundaries and seems to reduce the generation of the spurious oscillations. One can observe in Figure (5.3) that the mesh has been optimized to decrease the elements distortion: not only the elements that have a common edge with the domain boundaries are curved. Indeed, it is often necessary to curve some internal edges in order to maintain a valid mesh [Shephard et al., 2005]. The high-order curved meshes used in this work have been generated by the mesh generator GMSH freely available at *http://geuz.org/gmsh/* and described in [Geuzaine and Remacle, 2008].

The results obtained with high order mappings are depicted in Figure (5.4). We observe the influence of both the mesh resolution and the polynomial order with third order mappings. We consider two different

meshes (white triangles) of 674 (upper row) and 245 (lower row) elements. The first computation on the finer mesh (left part of the figure) uses a second order polynomial representation of the solution, $\mathcal{P}_2^3$. This representation produces a relatively smooth solution, but the expected converged solution is not obtained because of the high sensitivity of the method to the asymmetry of the mesh. The pressure becomes higher on the right side of the cylinder, leading to an oscillating solution with non symmetric stagnation points. One can thus increase the polynomial order, or the mesh resolution. The fourth order representation, $\mathcal{P}_4^3$, on the same mesh, leads to the expected solution. Decreasing the mesh resolution (bottom row of the figure), the polynomial order has then to be increased up to order 6 to obtain the converged solution.
We observe that only 8 triangles are sufficient with a third order mapping to represent accurately the cylinder boundaries. But the convergence of the solution to the exact steady state also requires a sufficient resolution, which can be reached by a sufficiently high polynomial order and/or a sufficiently high mesh resolution. We also observe in Figure (5.5) that the mesh resolution does not replace the accurate representation of the boundaries: the solution is obtained with a very fine grid of 20016 triangles and a $\mathcal{P}_1^1$ representation, for which the converged solution is not reached. The solution clearly present strong oscillations in the vicinity of the cylinder and the amplitude of the jumps in the wake of the bluff body suggests a high error compared to the high order mappings solutions. The comparison between the high order $\mathcal{P}_6^3$ and the linear $\mathcal{P}_1^1$ representations is depicted in the close-up views of Figure (5.6). We observe that the solution is clearly more accurate with the $\mathcal{P}_6^3$ representation which presents about nine times less degrees of freedom than the $\mathcal{P}_1^1$ representation. The relative computational costs τ of the different computations given in Figure (5.4) have been normalized with the maximum computational time, i.e. the one of the $\mathcal{P}_1^1$ computation. Considering that the converged steady solutions are only reached for the $\mathcal{P}_4^3$ and $\mathcal{P}_6^3$ representations with meshes of 674 and 245 elements respectively, the lower computational time corresponds clearly to the use of the higher order, i.e. $p = 6$, as expected.

Such curved meshes represent a powerful tool for handling *hp* adaptivity on complex geometries and boundaries [Shephard et al., 2005]. However, the creation of curved meshes with really high order mappings

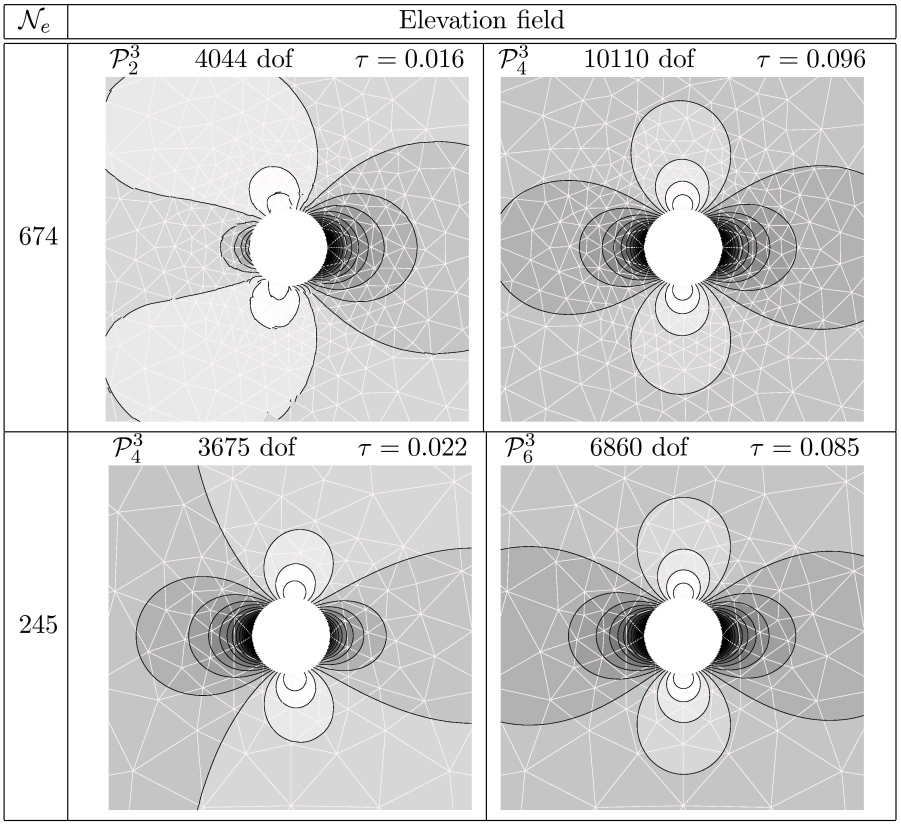


Figure 5.4: Water elevation fields for a mean flow around a cylinder with third order mapping meshes of 674 (upper row) and 245 (lower row) elements. Obtaining the converged steady solution on the coarser grid requires a polynomial order up to order 6, while the order 4 is sufficient on the finer mesh. Only 8 triangles are sufficient to represent accurately the boundaries with a third order mapping. The relative computational times τ have been normalized with the maximum computation time obtained for the $\mathcal{P}_1^1$ representation in Figure (5.5).

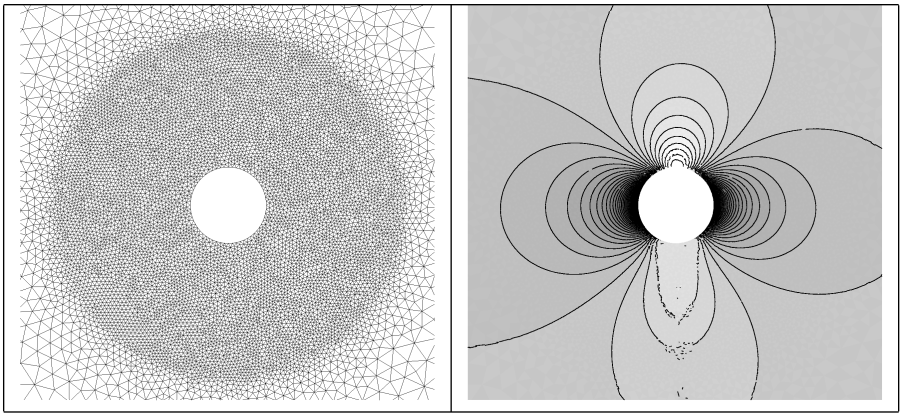


Figure 5.5: Elevation field (right) computed on the fine mesh (left) with a $\mathcal{P}_1^1$ representation. Although using a relatively fine mesh (20016 elements), the poor representation of the boundaries generates strong oscillations on the boundaries and in the wake of the cylinder and does not allow the obtention of the expected converged solution.

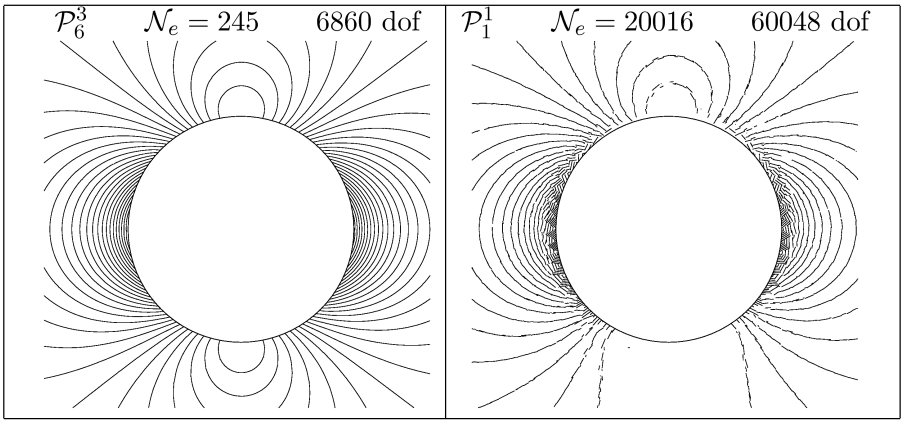


Figure 5.6: Close-up views of the cylinder vicinity for comparison of the solutions obtained with the $\mathcal{P}_6^3$ representation on the 245 elements mesh (left) and a $\mathcal{P}_1^1$ representation on a 20016 elements mesh (right). The $\mathcal{P}_6^3$ computation presents about nine times less degrees of freedom for a more accurate solution.

is sometimes quite difficult without generating Jacobian matrices with negative determinants, or simply bad shaped elements. Indeed, even if presenting strictly positive determinants of the Jacobian matrix, very distorted elements cannot be used since the integration errors becomes very significative. Mesh generation procedures that adapt the element size and mapping order to the boundary curvature are essential tools for meshing really complex domains as the ocean with a low number of curved elements, but it is mandatory to keep the elements distortion as low as possible.

We observe with this first test case that the DG method with the shallow water equations presents the same behavior than with the Euler equations. With unstructured meshes around bluff bodies, an accurate representation of the boundaries and a sufficiently high resolution are mandatory to reach the expected converged solution and to avoid spurious eddies production when working with coarse meshes.
Techniques as the smoothing of the normals are cheaper than the use of high order mappings in term of computational costs and are working perfectly well for flows around smooth bluff bodies. But this method does not suppress oscillations for very irregular bluff bodies with sharp coasts and do not represent irregular boundaries in an accurate way, while the high order mapping still produces smooth solutions.

5.1.3 Rattray island in a realistic tidal flow

The Rattray island is located at the coordinates $19°59' \ S, 148°33' \ E$ in the Great Barrier Reef near Australia. The island is 1.5 km long, 300 m wide and lies in well-mixed water. The surrounding bathymetry is approximately of 25 m, but presents some strong varying gradients as depicted in Figure (5.13). This site has been chosen for many studies and measurements. Indeed, the surrounding waters are turbid which allows aerial observations of the circulation and the scales are the typical coastal water scales [Wolanski et al., 1984].
This is thus a particularly interesting test case for numerical computations since it provides realistic and complex bathymetries and boundaries as shown in Figure (5.13), but also many measurements performed during several days, providing useful data for the tidal forcings (see Figure

(5.12)) and for comparison in the wake of the island.

The tides are almost perpendicular to the main island direction, creating eddies in its wake. Numerically speaking, those eddies are generated through the subgrid scale processes, which means that a diffusion term has to be introduced and parametrized to reproduce those processes, with a larger diffusivity than the molecular diffusivity. The curl operator applied to the momentum equation yields an advection equation of the vorticity. In a constant mean flow with a flat bathymetry, no source term or diffusion term and slip boundary conditions, there should not be any vorticity generation inside the flow even in the presence of very irregular bluff bodies.

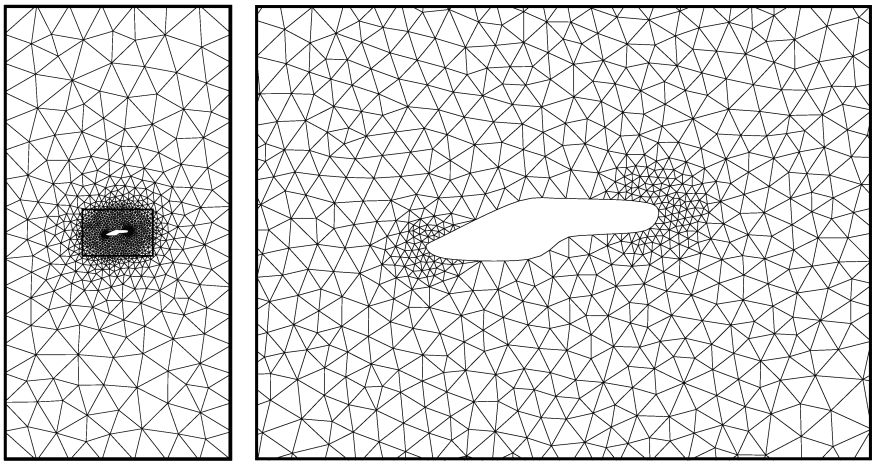


Figure 5.7: Mesh of the idealized Rattray island used for the Rattray experiments in a constant mean flow, with fourth order polynomial function spaces and both first or second order mappings.

Rattray island in a constant mean flow

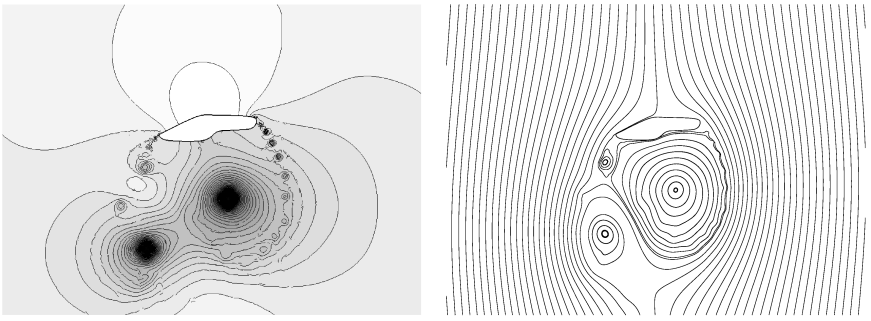
In the first numerical experiment we consider the Rattray island in

a constant mean flow with a Froude number of Fr = 0.1 with a flat bathymetry of $25m$. No diffusion, Coriolis or bottom stress terms were considered while a slip boundary condition is imposed on the island's coast. The flow is then supposed to remain steady without vorticity production.
This experiment illustrates the issue of boundary discretization in ocean processes. The mesh of the idealized Rattray island depicted in Figure (5.7) is used with both $\mathcal{P}_4^1$ and $\mathcal{P}_4^2$ representations. We see in Figure (5.8) the non steady solution obtained with the first order mapping: the linear discretization of the boundaries combined to a high order DG scheme produces oscillations and numerical diffusion, responsible for the creation of non physical eddies even with slip boundary conditions. Their size and amplitude depends on the boundary representation and on the elements size which means that we have no control on those spurious eddies. The smallest eddies ejected from the island edges merge in the wake of the island to create larger eddies. Those larger eddies always grow until they reach a length of the order of the island length. They finally reach the right sizes and amplitudes but not for the right reason. It can be shown that with a no slip boundary condition, the amount of vorticity expelled from the island coastlines does not depend on the boundary layer details, but only depends on the mean flow velocity, and the eddy size depends mainly on the length of the obstacle.

The second experiment in Figure (5.9) is obtained with exactly the same parameters, mesh and solution shape functions, but with a second order mapping. The sufficiently accurate boundary representation leads to the expected steady solution without any spurious eddy. The DG scheme with appropriate boundary representation provides an advection scheme that does not require the use of additional dissipation term for its stabilization and avoid spurious oscillations. The vorticity and eddies generated are then only depending on the subgrid scale parametrization and can then be properly controlled.

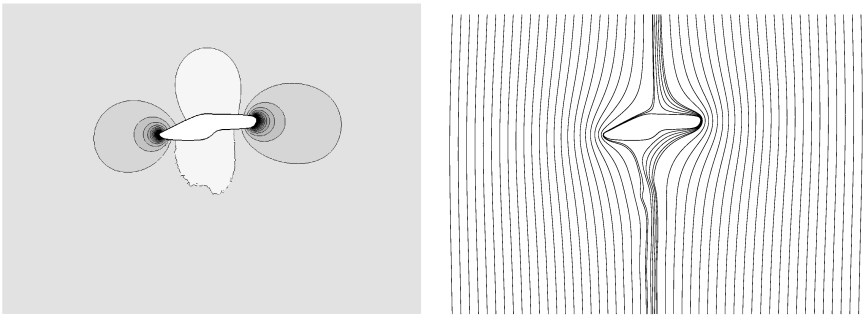
Realistic tidal flows around the Rattray island

The previous section has shown that the DG scheme with appropriate boundary condition does not produce any *spurious* eddy and the numer-



Elevation field Particles path

Figure 5.8: Without variable bathymetry, diffusion, Coriolis or bottom stress term, the expected steady state solution in a constant mean flow is not obtained with a $\mathcal{P}_4^1$ representation. Spurious eddies are created on the poorly discretized boundaries, their size is the smallest elements size.



Elevation field Particles path

Figure 5.9: Steady state solution for a constant mean flow with Fr = 0.1 around the Rattray island without bathymetry, diffusion, Coriolis or bottom stress. Increasing the representation to $\mathcal{P}_4^2$ leads to a numerical scheme creating no eddies in the wake of the island.

ical diffusion introduced by the scheme is not significative compared to the subgrid scale viscosity. We may now introduce the horizontal diffusion terms to modelize those subgrid scale processes with an appropriate boundary condition on the island.

We use a variable viscosity as proposed by Smagorinsky [Smagorinsky, 1963, White et al., 2006], depending on the element size and on the velocity field gradients:

$$\nu(x, y, t) = c \triangle (\boldsymbol{d} : \boldsymbol{d})^{1/2} \tag{5.8}$$

where $\triangle$ denotes the element area and $\boldsymbol{d}$ is the infinitesimal deformation tensor defined as $\boldsymbol{d} = 0.5\left(\nabla \boldsymbol{v} + \nabla \boldsymbol{v}^T\right)$. The parameter c, called Smagorinsky coefficient in the following, is a constant which has to be correctly chosen to obtain the right diffusivity and velocity amplitudes around the island.
To modelize the unresolved boundary layer, the slip boundary condition is maintained on the convective terms while a partial slip condition is used for the diffusion terms. This condition is mandatory to generate the appropriate vorticity on the island boundaries. It consists in an empirical parametrization [Deleersnijder, 1996, Lambrechts et al., 2008b] given by:

$$\nu \frac{\partial \boldsymbol{u}}{\partial n} = \alpha \boldsymbol{u} \tag{5.9}$$

where n is the normal direction to the coast. The value of the partial slip constant α is set to 2.5 $10^{-3} m s^{-1}$. Since the normal velocity equals zero on the boundaries, this condition also reads:

$$\frac{\partial u_n}{\partial n} = 0 \ ,$$
$$\nu \frac{\partial u_t}{\partial n} = \alpha u_t \ ,$$

with t the tangential direction to the coast, and u_n and u_t the normal and tangential components of the velocity respectively.

The physical parameters are $\rho = 1000\ kg\ m^{-3}$, $g = 9.81\ ms^{-2}$ and $\gamma = 2\ 10^{-3}$; the Coriolis parameter is approached with the f-plane approximation, $f = -5\ 10^{-5}\ s^{-1}$, corresponding to the latitude of $20°S$

and no wind forcing is considered. A second order polynomial representation of the bathymetry is used with a restriction on the minimum bathymetry value set to 3 m to avoid any wetting/drying process.

Using the realistic tidal forcings and bathymetry field, a first experiment is performed to illustrate the influence of the parameters and of the boundary condition on the solution. Three computations depicted in Figures (5.10) and (5.11) are performed using the same mesh and coast representation. The first computation (upper rows) correspond to a full slip condition on both the convective and diffusive operators. We observe that there is no vorticity produced by the numerical aspects of the model, since the boundaries are correctly represented. Therefore, the model does not reproduce the large eddies of the observations. The other computations (middle and bottom rows) are performed with the partial slip boundary condition (5.9) and Smagorinsky coefficients values of 0.05 and 0.025. Introducing vorticity with an appropriate boundary condition, we obtain the expected eddies in the wake of the island, for the *right* reasons. We also observe the influence of the Smagorinsky coefficient c. A larger diffusivity reduces the amplitude of the eddies, creating smaller maximum velocities and a larger minimum elevation.

We may now compare the results to in situ observations and measurements, using the partial slip condition on the diffusion terms and the value $c = 0.025$ for this mesh and polynomial order. Notice that this value has not been optimized to obtain the best possible fit with the experimental data.
We see in Figure (5.14) the typical elevation and velocity fields obtained around the Rattray island with a curved coarse high order mesh. Those results are in very good agreement with the observations. The same computation with a piecewise linear representation of the boundaries produces spurious oscillations which eventually lead to an unstable scheme. Indeed, with the use of the realistic bathymetry, the elevation oscillations becomes even larger than the water elevation itself. The use of accurate representation of the boundaries is thus mandatory.
At the rising tide, a large clockwise rotating eddy is created on the south-east side of the island after about one hour. It grows and reaches approximately twice the width of the island [Falconer et al., 1986]. A second eddy is created in the western part of the island, but its size

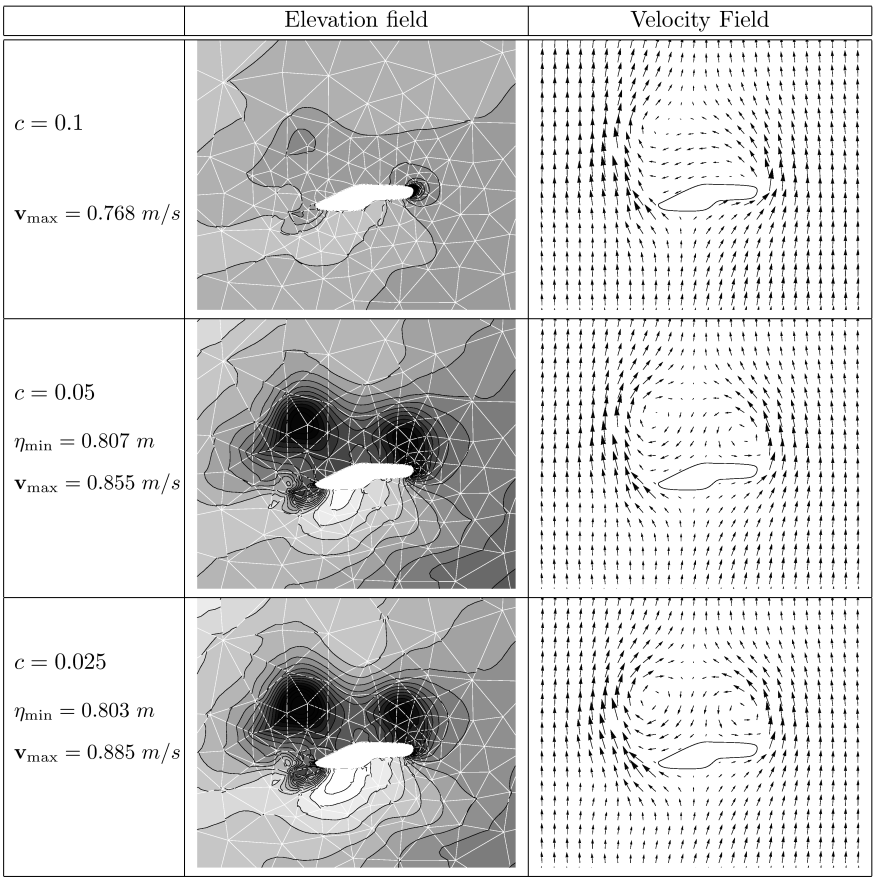


Figure 5.10: Elevation and velocity fields at $t = 35\ h$ for a full slip boundary condition and $c = 0.1$ (upper row), and a partial slip condition with $c = 0.05$ and $c = 0.025$ (middle and bottom rows respectively).

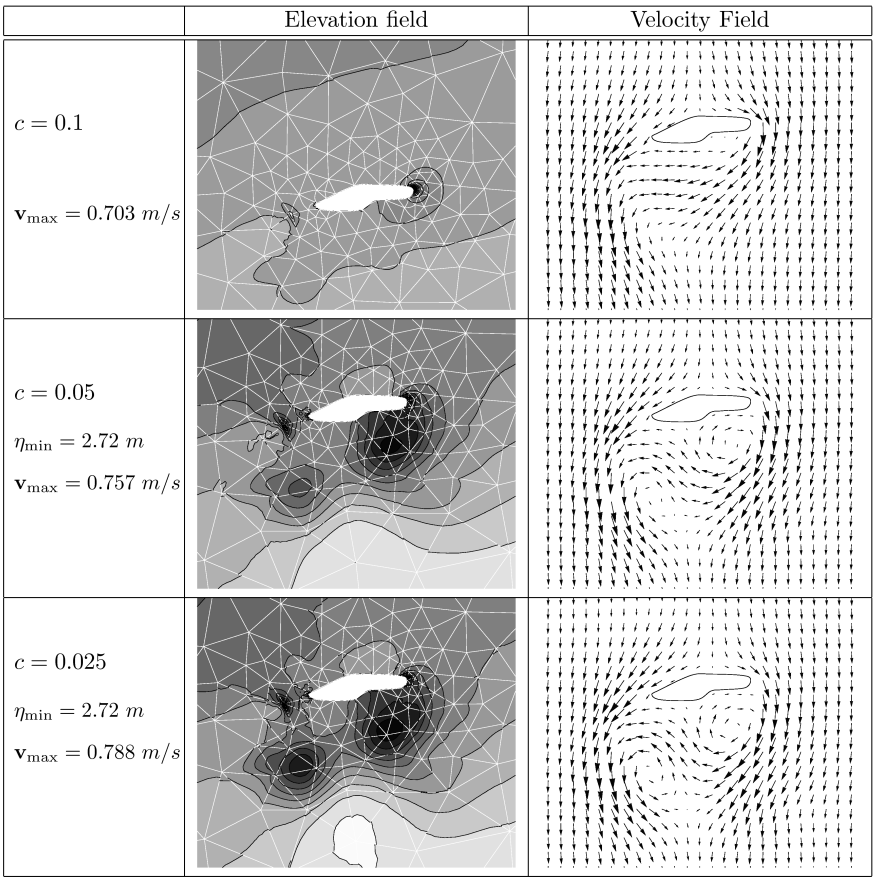


Figure 5.11: Elevation and velocity fields at $t = 78\ h\ 20\ m$ for a full slip boundary condition and $c = 0.1$ (upper row), and a partial slip condition with $c = 0.05$ and $c = 0.025$ (middle and bottom rows respectively).

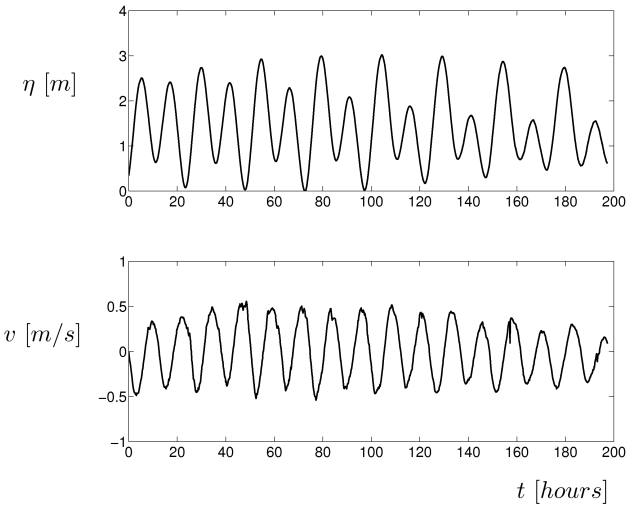


Figure 5.12: Time evolution of elevation and velocity (v-component) tide forcings measured in November 1982 near Rattray island.

remains very small compared to the eastern eddy and compared to the first computation with a flat bathymetry. It is due to the very shallow bathymetry and the resulting high bottom stress. This smaller eddy has not been observed in situ but has been already reproduced by many numerical schemes and models [Deleersnijder et al., 1992, White et al., 2006]. As the tide reverses, two counter-rotating eddies are generated on the northern side of the island. The maximum variable diffusivity is located on elements close to the island, and reaches approximately 5 $m^2 s^{-1}$, which corresponds to the maximum velocities of about 1 ms^{-1} observed in situ.

Currents meters depicted in Figure (5.13) were deployed on November 23th, 1982, during almost 14 days. We observe in Figure (5.16) the good agreement of the v-component of the velocity through time. The evolution of the u-component, usually several orders of magnitude lower than v, does not fit as well the data since the numerical results are highly sensitive to topological differences with the real configuration. We observe however in Figure (5.15) a good agreement in velocity am-

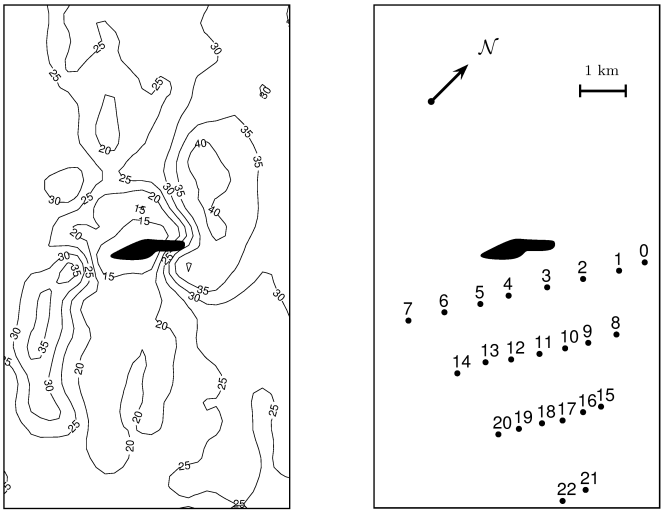


Figure 5.13: Bathymetry isolevels (left) and currentmeters position (right) around the Rattray island.

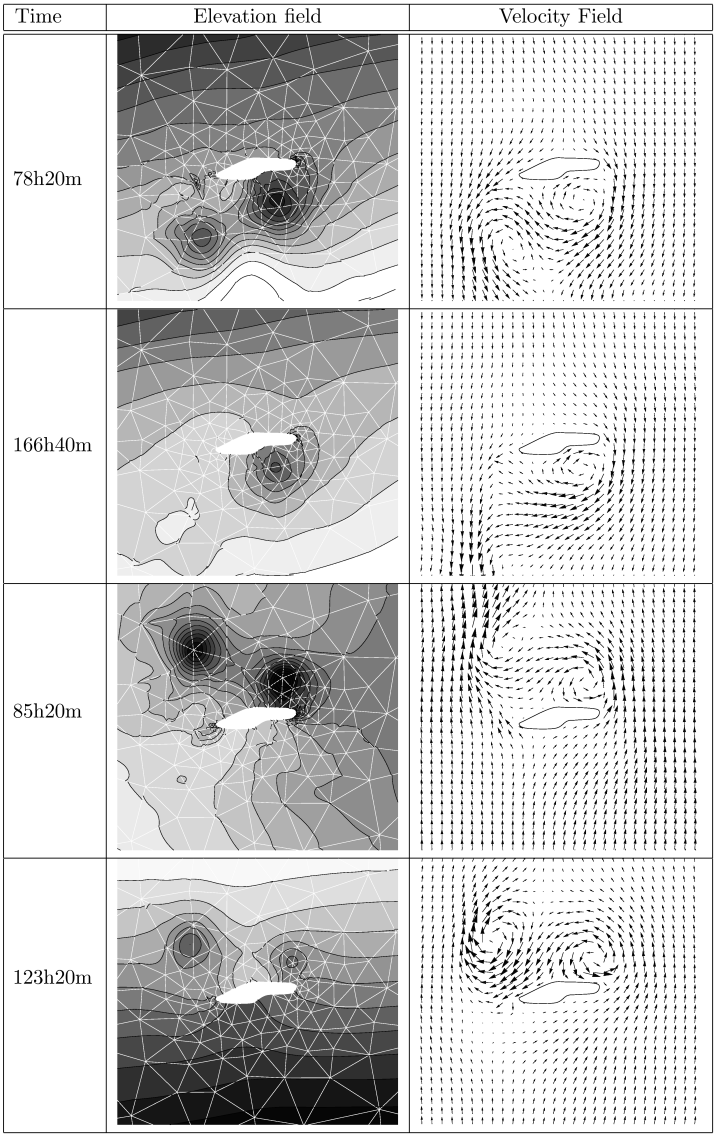


Figure 5.14: Typical elevation and velocity fields with tidal flows around the Rattray island on a coarse mesh using a $\mathcal{P}_4^3$ representation. The very shallow bathymetry on the south-east part of the island tends to reduce the size of the eastern eddy at rising tides (two first upper rows). The falling tide produces the two observed eddies on the northern side (two bottom rows).

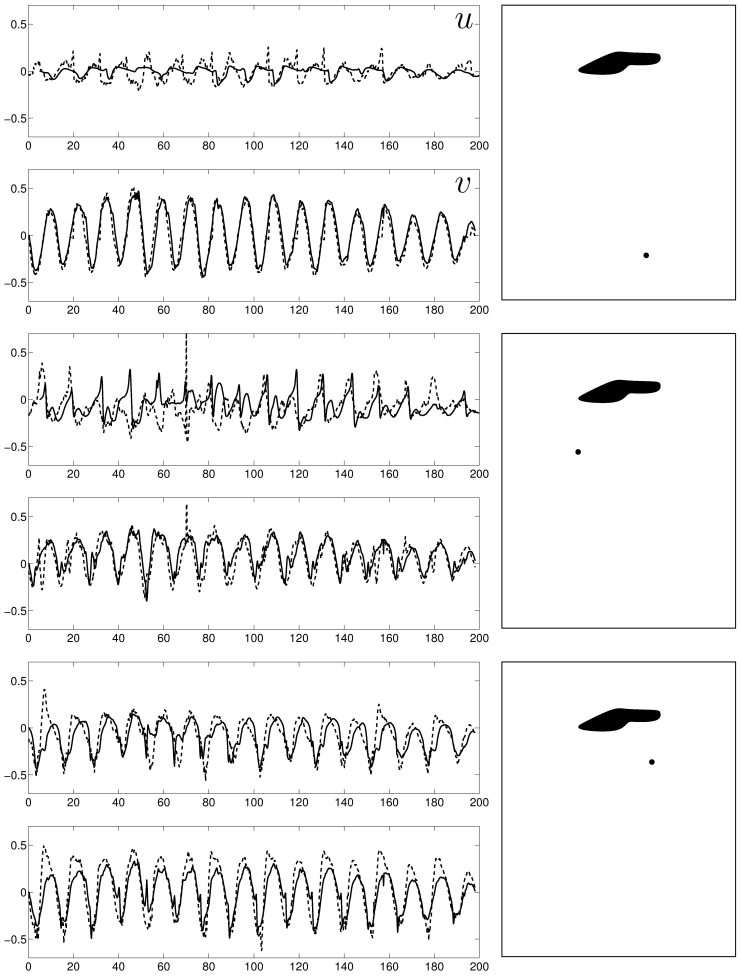


Figure 5.15: Comparison of the evolution in time (in hours) of the u and v-components of the velocity (in $[ms^{-1}]$) for some currentmeters. The DG results (continuous lines) and the measurements (dashed lines) are in good agreements for the v component. The u component is in better agreement for currentmeters close to the island, in the eddy region with larger amplitudes.

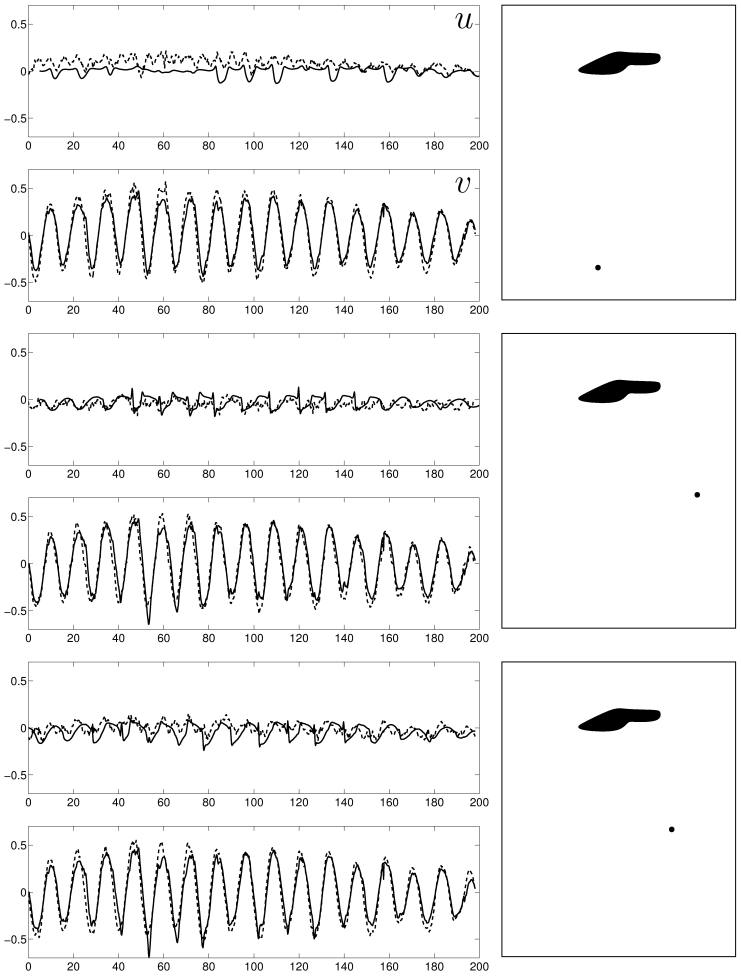


Figure 5.16: Comparison of the evolution in time (in hours) of the u and v-components of the velocity (in $[ms^{-1}]$) for some currentmeters. The DG results (continuous lines) and the measurements (dashed lines) are in good agreements for the v-component. The u-component is in better agreement for currentmeters close to the island, in the eddy region with larger amplitudes.

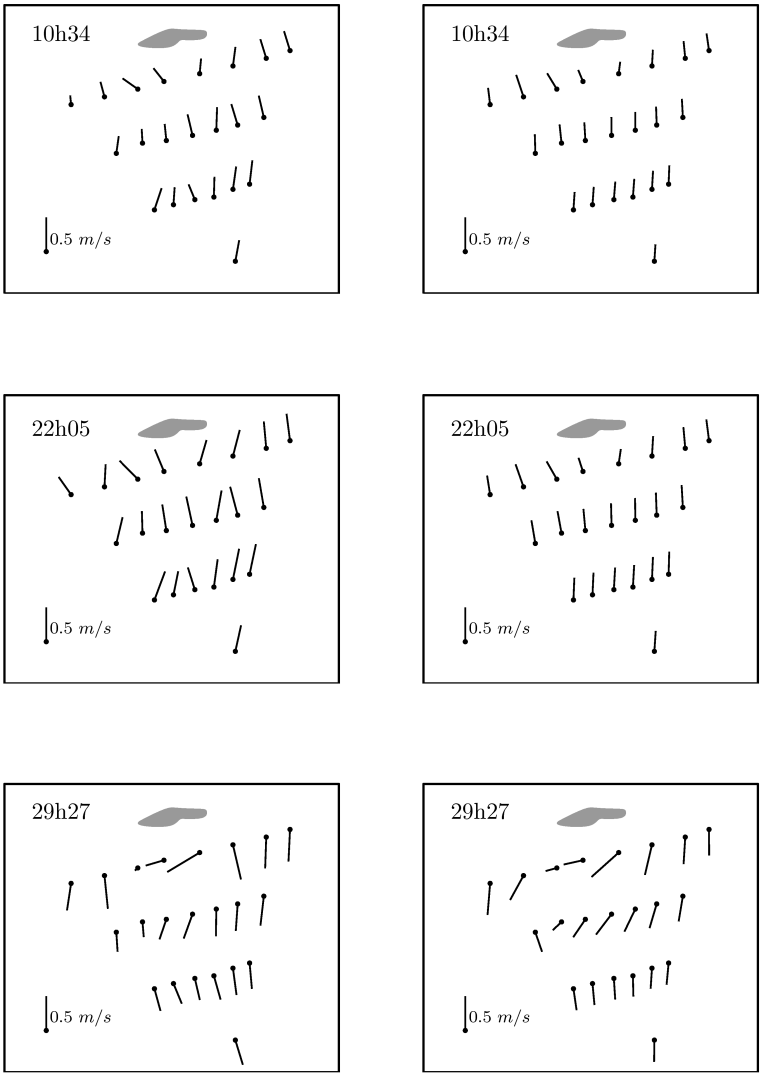


Figure 5.17: Comparison between the experimental data (left) and the DG results (right). The numerical results seem to fit the data with velocities of the right amplitudes and directions.

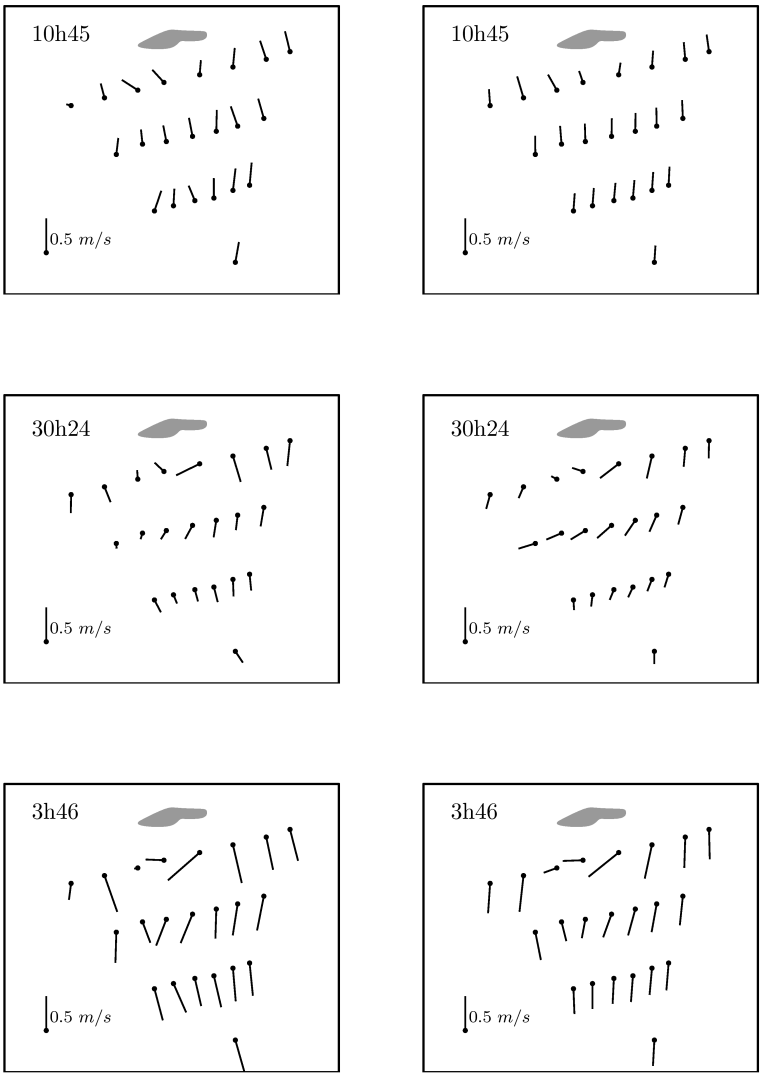


Figure 5.18: Comparison between the experimental data (left) and the DG results (right). The numerical results seem to fit the data with velocities of the right amplitudes and directions.

plitude and time evolution for the currentmeters with a sufficiently high u-component, i.e. in the eddy behind the island. In Figure (5.17) and Figure (5.18) are depicted the velocity fields measured in 1982 and the velocity field resulting from the DG computation at the same locations for several times. We observe that the numerical results are in good agreement with the measurements, considering the uncertainty on the bathymetry shape (measured on a 200m $\times$ 200m grid), the island position, the currentmeters locations and considering the simplified tidal flow forcing assumed to propagate following a straight line parallel to the left and right boundaries of the domain. We obtain eddies of the same shapes and intensities and the velocity evolution seems to fit the data.

5.2 Local high-order tangent basis approach on curved manifolds

Many fields of engineering requires to solve partial differential equations (PDE's) on curved manifolds, as for instance the computation of solid mechanics conservation laws on shells and membranes, or geophysical fluids conservation laws for oceanic or atmospheric predictions. Those computations on curved manifolds are essentially bi-dimensional since occurring on the surface of a curved body. However, a third dimension has to be taken into account since the body curvature introduces important acceleration processes. In the context of ocean modeling, computation on curved manifolds is mandatory to obtain an ocean circulation at the scale of the planet. Indeed, the curvature of the Earth introduces important acceleration term modifying the fluid dynamics.

Past studies have seen the emergence of basically two main families of methods. A first method consists in writing directly the PDE's on the curved manifold. The conservation laws and vector fields are thus written in a bi-dimensional (considering a manifold $\in \mathcal{R}^3$) parametric space [Rossmanith et al., 2004, Rossmanith, 2006]. This approach has been successfully used to compute the shallow water equations on the surface of the sphere for simple atmospheric models [Cho and Polvani, 1996, Iacono et al., 1999, Rancic et al., 1996], to compute relativistic

gas dynamics on space-time manifolds curved under the influence of gravitational forces from neutron stars or black holes [Font, 2000, Pons et al., 1998], or for the propagation of magneto-gravity waves in the solar tachocline [Gilman, 2000]. This approach requires to rewrite the PDE's in order to account for the curvature terms. The vectors are described using the covariant coordinates and the expression of spatial operators depends on the metric of the manifold's surface. The main advantage of this approach is that vectors contain the minimum number of components and the tangency constraint to the manifold is ensured. Yet, it requires an expression of the metric tensor at any point of the space. This restriction somehow decreases the generality of the method. An example of such a transformation may be found in atmospheric computations using the so-called *cubed-sphere* [Ronchi et al., 1996, Taylor et al., 1997, Nair et al., 2005], mapping the six faces of a cube onto a sphere.
Another approach considers the expression of the vector field and the conservation laws in three dimensions with a single global cartesian coordinate system [Swarztrauber et al., 1997, Stuhne and Peltier, 1999, Giraldo, 2000, Giraldo and Warburton, 2005]. This method allows one to keep a general expression of the PDE, without adding explicit acceleration terms as the Christoffel symbols since the geometry remains hidden in some source terms. The conservation laws remain the same as in the case of a 3D domain computation. The drawback of this approach is the use of a larger number of unknowns, since three components are used instead of two. Moreover, a constraint has to be added for the vector fields to remain tangent to the manifold. When considering explicit time steppers, one basic possible technique consists in projecting the residual and the solution on the local tangent plane for instance at every inner Runge-Kutta time step. This technique is thus not well suited for implicit time schemes. One could also consider the constraint imposition through Lagrange multipliers, with the drawback of increasing the number of equations and the computational cost [Côté, 1988, Giraldo, 2006].

In this section is proposed an alternative approach to solve the shallow water equations on the sphere using the discontinuous Galerkin methods. It must be noticed that the method is valid for any vectorial conservation law, any manifold and any finite element method. However,

the DG method is particularly well suited for this approach since there is no continuity requirement between elements for defining a tangent basis to the manifold.
The alternative approach developed here combines the advantages of those two classical methods. It is shown that the DG method is particularly well suited to compute conservation laws on curved manifolds. Indeed, the definition of a local tangent basis on each element is highly simplified since there are no inter-element continuity requirement, while continuous discretizations lead to slightly more complex basis definitions [Comblen et al., 2008]. The alternative approach developed here is applied in particular to the shallow water equations for the computation of oceanic or atmospheric flows. The approach is validated on the sphere with standard test cases proposed by [Williamson et al., 1992, Smith and Dritschel, 2006], and the robustness and generality of the method are eventually tested on some non-conventional irregular manifolds.

5.2.1 Usual approaches for solving PDE's on curved manifolds

The aim is to solve conservation laws in a curved bi-dimensional domain. The partial differential equations are thus essentially bi-dimensional in the sense that the vectorial unknowns are defined as tangent to the manifold and present thus two independent components. However, the manifold is defined in three dimensions and its curvature generates acceleration processes which have to be taken into account.

Let us consider the following set of hyperbolic conservation laws in the general divergence form of hyperbolic systems:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U}) \tag{5.10}$$

with $\boldsymbol{U}(\boldsymbol{x}, t)$ the vector of unknown fields, $\boldsymbol{F}$ the flux matrix containing for instance the advection terms or the pressure gradient terms, and the vector $\boldsymbol{S}$ containing the so-called source terms which do not involve any derivative of the solution. We restrict ourselves to the mass and momentum conservation laws. In order to identify the scalar and vectorial

components of the system, let us define the following quantities:

$$\boldsymbol{U} = \begin{pmatrix} s \\ \boldsymbol{v} \end{pmatrix} , \quad \boldsymbol{F}(\boldsymbol{U}) = \begin{pmatrix} \boldsymbol{G}(\boldsymbol{U}) \\ \boldsymbol{H}(\boldsymbol{U}) \end{pmatrix} , \tag{5.11}$$

with s the scalar unknown and $\boldsymbol{v}$ the vectorial unknown field, i.e. typically the pressure and velocity for mass and momentum conservation laws in fluid dynamics, while $\boldsymbol{G}$ and $\boldsymbol{H}$ denote the sub-vector and sub-matrix of the flux matrix corresponding to the mass and momentum equations, respectively.
This set of equations has to be solved on a general curved manifold $\boldsymbol{x} \in \mathcal{R}^3$ as depicted in the right part of Figure (5.19).

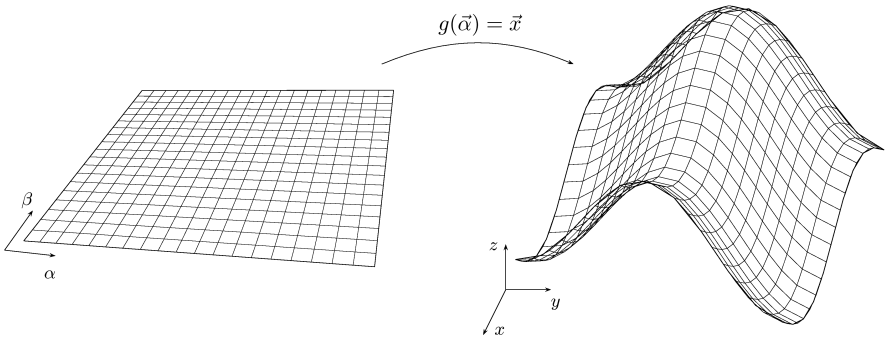


Figure 5.19: Transformation g from the parametric space $\boldsymbol{r} \in \mathcal{R}^2$ to the curved manifold in $\boldsymbol{x} \in \mathcal{R}^3$

The mass equation is a scalar equation describing the time evolution of a scalar unknown field. The main difference between the classical methods for solving PDE's on the manifold lies in the manner to write the gradient operator ∇ and the corresponding fluxes $\boldsymbol{G}$:

- *2D* approach: a first classical method consists in writing the equations directly on the manifold, in two dimensions, by considering a parametric bi-dimensional space $\boldsymbol{\alpha} = (\alpha, \beta) \in \mathcal{R}^2$ and a transformation $g(\boldsymbol{\alpha}) = \boldsymbol{x}$ mapping the parametric coordinates on the curved manifold. The gradient operator and fluxes are thus written

in the contravariant space as

$$\begin{aligned} \nabla &= (\partial/\partial\alpha, \partial/\partial\beta) \ , \\ \boldsymbol{G} &= G_\alpha \hat{\boldsymbol{\alpha}} + G_\beta \hat{\boldsymbol{\beta}} \ , \end{aligned}$$

with $(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}})$ an orthogonal constant unit basis of $\mathcal{R}^2$. Therefore, this approach requires the use of the metric tensor $\mathcal{G}$ based on the transformation g. The divergence operator for instance becomes:

$$\nabla \cdot \boldsymbol{v} = \frac{1}{\sqrt{\det(\mathcal{G})}} \left[\frac{\partial}{\partial\alpha}(\sqrt{\det(\mathcal{G})} u_\alpha) + \frac{\partial}{\partial\beta}(\sqrt{\det(\mathcal{G})} u_\beta) \right]$$

with (u_α, u_β) the components of the vectorial field in the parametric space.

- *3D* approach: another classical approach consists in writing the equations directly in three dimensions, by defining

$$\begin{aligned} \nabla &= (\partial/\partial x, \partial/\partial y, \partial/\partial z) \ , \\ \boldsymbol{G} &= G_x \hat{\boldsymbol{e}}_x + G_y \hat{\boldsymbol{e}}_y + G_z \hat{\boldsymbol{e}}_z \ , \end{aligned}$$

with $(\hat{\boldsymbol{e}}_x, \hat{\boldsymbol{e}}_y, \hat{\boldsymbol{e}}_z)$ an orthogonal constant unit basis of $\mathcal{R}^3$.

Considering the momentum equation, the difference between those methods still lies in the definition of the operators ∇ and the fluxes $\boldsymbol{H}$, but also in the definition of the vectorial unknown field itself.
With the classical *3D* approach, the vectorial field $\boldsymbol{v}$ is written with three components in $\mathcal{R}^3$:

$$\boldsymbol{u} = u\hat{\boldsymbol{e}}_x + v\hat{\boldsymbol{e}}_y + w\hat{\boldsymbol{e}}_z \ .$$

The vectorial equation thus reads:

$$\frac{\partial}{\partial t}(u\hat{\boldsymbol{e}}_x + v\hat{\boldsymbol{e}}_y + w\hat{\boldsymbol{e}}_z) + \frac{\partial}{\partial x}\boldsymbol{H}_x + \frac{\partial}{\partial y}\boldsymbol{H}_y + \frac{\partial}{\partial z}\boldsymbol{H}_z = \boldsymbol{0} \ . \tag{5.12}$$

Each flux vector is then also defined in the same basis as

$$\boldsymbol{H_a} = H_{ax}\hat{\boldsymbol{e}}_x + H_{ay}\hat{\boldsymbol{e}}_y + H_{az}\hat{\boldsymbol{e}}_z \ , \quad \text{for } a = x, y, z. \tag{5.13}$$

Taking advantage of the orthogonality of the basis, the scalar product of this vectorial equation (5.12) with the three basis vectors $\hat{\boldsymbol{e}}_{x,y,z}$ yields the

classical system of "decoupled" equations in u, v and w, each equation describing the time evolution of one component:

$$\begin{cases} \frac{\partial u}{\partial t} + \frac{\partial}{\partial x} H_{xx} + \frac{\partial}{\partial y} H_{yx} + \frac{\partial}{\partial z} H_{zx} \\ \frac{\partial v}{\partial t} + \frac{\partial}{\partial x} H_{xy} + \frac{\partial}{\partial y} H_{yy} + \frac{\partial}{\partial z} H_{zy} \\ \frac{\partial w}{\partial t} + \frac{\partial}{\partial x} H_{xz} + \frac{\partial}{\partial y} H_{yz} + \frac{\partial}{\partial z} H_{zz} \end{cases} \tag{5.14}$$

The main advantage of this approach is the simplicity of the fluxes and the gradient operator. Indeed, the spatial derivatives do not require to be rewritten in the parametric space using the metric tensor. Therefore, no analytical expression of the manifold is required so far, and the expression of the flux function remains the same than in the plane computation. The main drawback is the computational cost: this approach requires the use of three unknowns to represent the vectorial field, i.e. u, v and w, instead of two for the first method. Moreover, there is no guarantee for the vectorial field to remain tangent to the surface. Therefore, the method also requires to ensure the tangency by, for instance, projecting the solution on the local tangent plane at each explicit Runge-Kutta iteration. The method is thus less well suited for implicit time schemes. One can also ensure the tangency by imposing it as a constraint by Lagrange multipliers, which adds one more equation and increases the computational costs.
On the other hand, the *2D* approach considers a vectorial field $\boldsymbol{u}$ with two components in the parametric space:

$$\boldsymbol{u} = u_\alpha \hat{\boldsymbol{\alpha}} + u_\beta \hat{\boldsymbol{\beta}} \ ,$$

as the gradients and fluxes:

$$\frac{\partial}{\partial t} \left(u_\alpha \hat{\boldsymbol{\alpha}} + u_\beta \hat{\boldsymbol{\beta}} \right) + \frac{\partial}{\partial \alpha} \boldsymbol{H}_\alpha + \frac{\partial}{\partial \beta} \boldsymbol{H}_\beta = \boldsymbol{0} \ . \tag{5.15}$$

Applying the same procedure as in the 3D formulation, one thus obtain a system with decoupled time derivatives. The advantage is the use of a minimum number of unknowns and the guarantee for the vectorial field to remain tangent. However, the fluxes have to be rewritten to take into account the curvature of the manifold. This will lead to the creation of acceleration terms with the usual Christoffel symbols. Although efficient, the method appears to be less general since it requires an analytical expression describing the surface and some efforts for computing

a metric tensor and modifying the flux function.

The purpose of method developed here is precisely to combine the advantages of both methods described above while leaving the drawbacks.
Let us define at every point on the curved manifold a bi-dimensional unit tangent basis $(\boldsymbol{t}_\xi, \boldsymbol{t}_\eta)$ without orthogonality requirement. The idea of this alternative approach is to write the vectorial field into this local tangent basis, while the fluxes and spatial operators are still expressed in the three-dimensional basis $\hat{\boldsymbol{e}}_{x,y,z}$. The vectorial equation thus reads:

$$\frac{\partial}{\partial t}\left(u_\xi \boldsymbol{t}_\xi + u_\eta \boldsymbol{t}_\eta\right) + \frac{\partial}{\partial x}\boldsymbol{H}_x + \frac{\partial}{\partial y}\boldsymbol{H}_y + \frac{\partial}{\partial z}\boldsymbol{H}_z = \boldsymbol{0}\ , \tag{5.16}$$

with the flux vectors $\boldsymbol{H}$ expressed following (5.13). The scalar product of (5.16) by $\boldsymbol{t}_\xi$ and $\boldsymbol{t}_\eta$ yields the following system:

$$\left\{\begin{array}{lll}
\frac{\partial}{\partial t}(u_\xi \underbrace{\boldsymbol{t}_\xi \cdot \boldsymbol{t}_\xi}_{1} + u_\eta \boldsymbol{t}_\eta \cdot \boldsymbol{t}_\xi) & + & t_{\xi x}\left(\frac{\partial}{\partial x}H_{xx} + \frac{\partial}{\partial y}H_{yx} + \frac{\partial}{\partial z}H_{zx}\right) \\
 & + & t_{\xi y}\left(\frac{\partial}{\partial x}H_{xy} + \frac{\partial}{\partial y}H_{yy} + \frac{\partial}{\partial z}H_{zy}\right) \\
 & + & t_{\xi z}\left(\frac{\partial}{\partial x}H_{xz} + \frac{\partial}{\partial y}H_{yz} + \frac{\partial}{\partial z}H_{zz}\right) = 0 \\
\frac{\partial}{\partial t}(u_\xi \boldsymbol{t}_\xi \cdot \boldsymbol{t}_\eta + u_\eta \underbrace{\boldsymbol{t}_\eta \cdot \boldsymbol{t}_\eta}_{1}) & + & t_{\eta x}\left(\frac{\partial}{\partial x}H_{xx} + \frac{\partial}{\partial y}H_{yx} + \frac{\partial}{\partial z}H_{zx}\right) \\
 & + & t_{\eta y}\left(\frac{\partial}{\partial x}H_{xy} + \frac{\partial}{\partial y}H_{yy} + \frac{\partial}{\partial z}H_{zy}\right) \\
 & + & t_{\eta z}\left(\frac{\partial}{\partial x}H_{xz} + \frac{\partial}{\partial y}H_{yz} + \frac{\partial}{\partial z}H_{zz}\right) = 0
\end{array}\right. \tag{5.17}$$

Since the local tangent basis is usually not orthogonal, the time derivatives are no more decoupled into a set of scalar equations as in (5.14). The PDE remains fully vectorial and the whole system has thus to be solved at once. Moreover, the scalar product with the tangent basis leads to nine flux terms in each equation, while only three terms were not vanishing in each equation of (5.14), because of the orthogonality of $\hat{\boldsymbol{e}}_{x,y,z}$. But this formulation has the advantage of providing very simple expressions of the fluxes and the spatial derivatives, as in the fully 3D approach, while presenting only two unknowns, u_1 and u_2, and ensuring the conservation of the tangency of the vectorial fields as in the 2D approach. If no analytical expression of a metric tensor is required for

writing the fluxes, the complexity is however reported in the expression of the local tangent basis. Even though this formulation may seem quite complex, it turns out that this issue can be easily handled with the finite element method. In particular, the discontinuous finite element method is well suited for this formulation since it leads to a very simple definition of a local high-order tangent basis with no continuity requirement.

5.2.2 Local Tangent Basis approach using the Discontinuous Galerkin method

This subsection section is devoted to the discretization and resolution of vectorial PDE's in the form (5.17) using the Discontinuous Galerkin (DG) method. After the derivation of a weak formulation, we define the discretization of the vectorial fields and their local tangent basis. An approximate Riemann solver is then developed to take into account the normals discontinuities between elements. The discrete spatial operators are finally derived for solving the vectorial equation.

Discontinuous Galerkin discretization

We consider the set of conservation laws in divergence form as written in (5.10):

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$

with $\boldsymbol{U}(\boldsymbol{x}, t)$ the vector of unknown fields described in (5.11), $\boldsymbol{F}$ the flux matrix and $\boldsymbol{S}$ containing the source terms. The domain of computation Ω is divided into a set of $\mathcal{N}_e$ non-overlapping elements Ω_e:

$$\Omega = \bigcup_{e=1}^{\mathcal{N}_e} \Omega_e \ .$$

Following a classical Galerkin procedure, we obtain the weak formulation by multiplying the scalar and vectorial PDE's (5.10) by smooth test functions $\hat{\eta}$ and $\hat{\boldsymbol{u}}$ respectively and by integrating it on each element Ω_e. For sake of simplicity, we only consider the vectorial conservation law

for deriving the weak formulation. The extension to the scalar PDE is trivial. The vectorial PDE thus reads:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}\cdot\hat{\boldsymbol{u}}\ d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{H}(\boldsymbol{U})\cdot\hat{\boldsymbol{u}}\ d\Omega = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U})\cdot\hat{\boldsymbol{u}}\ d\Omega\ . \quad (5.18)$$

We seek an approximation $\boldsymbol{U}^h = (s^h, \boldsymbol{u}^h)$ of the scalar and vectorial fields $\boldsymbol{U}$ such that $\boldsymbol{U}^h \in \mathcal{U}_p^h$ with $\mathcal{U}_p^h$ the finite dimensional space of polynomials of order at most p. The discretized weak formulation thus read:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}^h\cdot\hat{\boldsymbol{u}}^h\ d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{H}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\ d\Omega = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\ d\Omega \quad (5.19)$$

with the test function $\hat{\boldsymbol{u}}^h \in \mathcal{U}_p^h$.

The transfer information between elements in classical continuous finite element methods is ensured by the fact that two adjacent elements share common nodes, which is not the case in discontinuous formulations. Therefore, this information transfer requires the advection term to be integrated by part:

$$\begin{aligned}\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}^h\cdot\hat{\boldsymbol{u}}^h\ d\Omega - \int_{\Omega_e} \boldsymbol{H}(\boldsymbol{U}^h)\cdot\nabla\hat{\boldsymbol{u}}^h\ d\Omega \\ + \int_{\partial\Omega_e} \boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n}\cdot\hat{\boldsymbol{u}}^h\ d\Gamma = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\ d\Omega \end{aligned} \quad (5.20)$$

where $\boldsymbol{n}$ denotes the outgoing unit normals to the element Ω_e and Γ_e denotes its corresponding edges. This integration on the element edges ensure the information transfer through the normal flux $\boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n}$. The formulation (5.20) remains however incomplete. Indeed, the unknown fields $\boldsymbol{U}^h$ present different values, $\boldsymbol{U}_L$ and $\boldsymbol{U}_R$, on left and right sides of each discontinuity, respectively. An appropriate definition of the normal flux has then to be provided:

$$\boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n} = \boldsymbol{H}(\boldsymbol{U}_L, \boldsymbol{U}_R)\cdot\boldsymbol{n}\ . \quad (5.21)$$

The accuracy of the method will depend in a large part on this normal flux definition. Those discontinuities provide an opportunity to introduce accurate advection schemes. Indeed, Riemann-like solvers can be used to compute the normal flux, introducing the right amount of numerical diffusion in the scheme to damp out the unresolved high frequency

modes. There are many possible choices for this normal flux function (see for instance [Toro, 1997, LeVeque, 2002]). However, no significant difference is observed between the solutions computed with solvers as the approximate Roe solver, Lax-Friedrich solver, or exact Riemann solver. We thus apply in this work the simple Lax-Friedrich solver on the flux matrix:

$$\boldsymbol{F}(\boldsymbol{U}^h)\cdot\boldsymbol{n} = \frac{\boldsymbol{F}(\boldsymbol{U}_L)+\boldsymbol{F}(\boldsymbol{U}_R)}{2}\cdot\boldsymbol{n} + |\lambda|_{max}\frac{\boldsymbol{U}_L-\boldsymbol{U}_R}{2} \tag{5.22}$$

where $|\lambda|_{max}$ denotes the maximum absolute value of the eigenvalues of the system, which corresponds to the advection speed of the Riemann invariants.

Although complete in a mathematical sense, formulation (5.20) is common to many different methods for computing PDE's on manifolds. The efficiency and robustness of the method will depend on the definition and discretization of the quantities $\boldsymbol{u}^h$ and $\boldsymbol{F}$. Let us now consider the definition of those quantities as expressed in (5.17) and proceed to their discretization.

Interpolation of a tangent vector field to a parabola

Let us consider in a first time the following one-dimensional simplified problem depicted in Figure (5.20). We consider a one-dimensional manifold composed of the unique parabola $\boldsymbol{x}(\boldsymbol{\xi}) \in \mathcal{R}^2$ that passes through the three points $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$. We seek to express the vectorial field $\boldsymbol{u}(\xi)$ as the product of an amplitude $u(\xi)$ and a tangent direction $\boldsymbol{t}(\xi)$:

$$\boldsymbol{u}(\xi) = u(\xi)\boldsymbol{t}(\xi)$$

One possible representation of the parabola can be written as

$$\begin{aligned}\boldsymbol{x}(\xi) = \boldsymbol{x}^h(\xi) &= \sum_{i=1}^{3}\boldsymbol{x}_i L_i^2(\xi) \\ &= \underbrace{\frac{(\xi-1)\xi}{2}}_{L_1^2(\xi)}\boldsymbol{x}_1 + \underbrace{(1-\xi^2)}_{L_2^2(\xi)}\boldsymbol{x}_2 + \underbrace{\frac{(\xi+1)\xi}{2}}_{L_3^2(\xi)}\boldsymbol{x}_3\ ,\end{aligned}$$

where L_i^p denotes the Lagrange function space of order p corresponding to the interpolation node $\boldsymbol{x}_i$. It maps the interval $\xi \in [-1, 1]$ onto the parabola using the so called quadratic Lagrange polynomials.

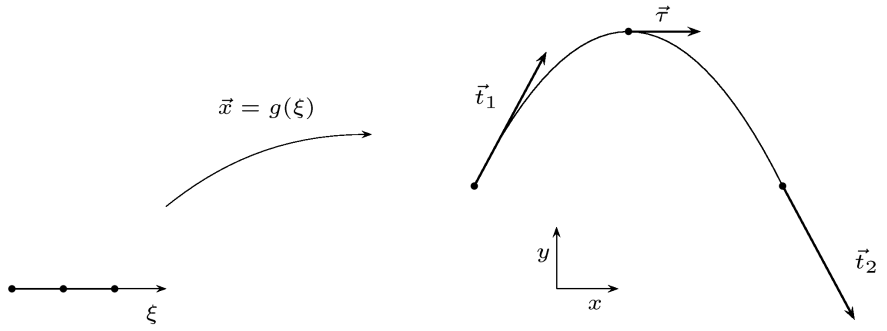


Figure 5.20: Mapping from the reference parametric space $\xi \in [-1, 1]$ onto the parabola $\boldsymbol{x} \in \mathcal{R}^2$ with $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ the unit tangents corresponding to $\xi = \pm 1$.

The tangent vector

$$\begin{aligned} \boldsymbol{\tau}^h(\xi) &= \frac{\partial \boldsymbol{x}}{\partial \xi} = \sum_{i=1}^{3} \boldsymbol{x}_i \frac{\partial L_i^2(\xi)}{\partial \xi} \\ &= \frac{(2\xi - 1)}{2}\, \boldsymbol{x}_1 - 2\xi\, \boldsymbol{x}_2 + \frac{(2\xi + 1)}{2}\, \boldsymbol{x}_3 \end{aligned}$$

can be written in a more convenient form

$$\begin{aligned} \boldsymbol{\tau}^h(\xi) &= \sum_{i=1}^{2} \boldsymbol{\tau}_i L_i^1(\xi) \\ &= \underbrace{\frac{(1-\xi)}{2}}_{L_1^1(\xi)} \underbrace{\left(-\frac{3}{2}\boldsymbol{x}_1 + 2\boldsymbol{x}_2 - \frac{1}{2}\boldsymbol{x}_3\right)}_{\boldsymbol{\tau}_1} + \underbrace{\frac{(1+\xi)}{2}}_{L_2^1(\xi)} \underbrace{\left(\frac{1}{2}\boldsymbol{x}_1 - 2\boldsymbol{x}_2 + \frac{3}{2}\boldsymbol{x}_3\right)}_{\boldsymbol{\tau}_2} \end{aligned}$$

that involves the two tangent vectors $\boldsymbol{\tau}_1 = \boldsymbol{\tau}(-1)$ and $\boldsymbol{\tau}_2 = \boldsymbol{\tau}(1)$ at the end points of the parabola that are interpolated using Lagrange shape functions of one order lower than the parametrization, i.e. of order one.

Now, our aim is to build a vector-valued interpolation $\boldsymbol{v}(\xi)$ that is always tangent to the parabola. Let us consider the following polynomial expansion at order p:

$$u^h(\xi) = \sum_{i=1}^{p+1} u_i L_i^p(\xi).$$

Multiplying the expressions of $\boldsymbol{\tau}^h$ and u^h allows one to build a parametrized tangent vector to the parabola:

$$\begin{aligned} \boldsymbol{u}^h(\xi) &= u^h(\xi)\boldsymbol{\tau}^h(\xi) \\ &= \sum_{i=1}^{p+1} u_i L_i^p(\xi) \left(L_1^1(\xi)\boldsymbol{\tau}_1 + L_2^1(\xi)\boldsymbol{\tau}_2 \right). \end{aligned}$$

For sake of simplicity, let us expand this last expression for $p = 1$:

$$\boldsymbol{u}^h(\xi) = u_1 \underbrace{\left[L_1^1(\xi)^2\boldsymbol{\tau}_1 + L_1^1(\xi)L_2^1(\xi)\boldsymbol{\tau}_2 \right]}_{\boldsymbol{T}_1(\xi)} + u_2 \underbrace{\left[L_1^1(\xi)L_2^1(\xi)\boldsymbol{\tau}_1 + L_2^1(\xi)^2\boldsymbol{\tau}_2 \right]}_{\boldsymbol{T}_2(\xi)}. \tag{5.23}$$

Equation (5.23) is a vector valued finite element approximation that make use of quadratic vector valued shape functions $\boldsymbol{T}_1(\xi)$ and $\boldsymbol{T}_2(\xi)$ and of scalar coefficients u_1 and u_2. Clearly, we have $\boldsymbol{T}_1(-1) = \boldsymbol{\tau}_1$ and $\boldsymbol{T}_2(1) = \boldsymbol{\tau}_2$ so that our approximation looks very much like a classical finite element approximation: u_1 and u_2 can be interpreted as vector amplitudes at element end points. Disappointingly, this way of approximating fields cannot be used in a finite element framework because it does not fulfill the simplest patch test: it is easy to see that (5.23) does not allow one to interpolate a vector of constant amplitude on the parabola. Even if using unit tangent vector $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ on the parabola's edges, the combination does not lead to unit tangents inside the domain as depicted in Figure (5.20).

It is possible to remedy to that problem defining unit tangent vectors $\boldsymbol{t}^h$:

$$\boldsymbol{u}^h(\xi) = u^h(\xi)\boldsymbol{t}^h(\xi) = u^h(\xi)\frac{\boldsymbol{\tau}^h(\xi)}{\|\boldsymbol{\tau}^h(\xi)\|},$$

with their derivative with respect to ξ given by

$$\frac{\partial \boldsymbol{t}^h}{\partial \xi} = \frac{\partial \boldsymbol{\tau}^h}{\partial \xi}\frac{1}{\| \boldsymbol{\tau}^h \|} - \frac{\boldsymbol{\tau}^h}{\| \boldsymbol{\tau}^h \|^3}\left(\boldsymbol{\tau}^h \cdot \frac{\partial \boldsymbol{\tau}^h}{\partial \xi} \right). \tag{5.24}$$

This approximation allows one to make the amplitude of $\boldsymbol{u}^h$ to vary polynomially while remaining tangent to the parabola. Of course, since vector shape functions are not polynomials, the finite element assembly process may appear as more expensive. However, the inverse of the Jacobian matrix used in the usual discrete convective terms is not polynomial neither. Moreover, those non-polynomial expressions may be precomputed, leading to an equivalent computational cost compared to conventional approaches.

Interpolation of tangent vector fields on surface meshes

Let us consider a curved manifold $\mathcal{S} \in \mathcal{R}^3$:

$$\boldsymbol{x} : (\xi, \eta) \in \mathcal{R}^2 \rightarrow \boldsymbol{x}(\xi, \eta) \in \mathcal{S} \ .$$

If $\mathcal{S}$ is sufficiently regular, it is possible to define a tangent plane at any point of its surface. Therefore, this allows one to consider the alternative approach (5.17), where the vectorial field $\boldsymbol{u}$ is expressed as:

$$\boldsymbol{u}(\xi, \eta) = u_\xi(\xi, \eta)\boldsymbol{t}_\xi(\xi, \eta) + u_\eta(\xi, \eta)\boldsymbol{t}_\eta(\xi, \eta) \tag{5.25}$$

with $(\boldsymbol{t}_\xi, \boldsymbol{t}_\eta)$ a unit local tangent basis to $\mathcal{S}$. This basis clearly varies in space and its orthogonality is not required. The aim of this section is to define a finite element approximation of (5.25) that is *a priori* tangent to any manifold $\mathcal{S}$.

In the classical framework of finite elements, the manifold $\mathcal{S}$ is discretized using triangles, possibly curvilinear. Let us consider a curvilinear triangle $\mathcal{T}$ with its mapping

$$\boldsymbol{x}^h : (\xi, \eta) \in \mathcal{R}^2 \rightarrow \boldsymbol{x}^h(\xi, \eta) \in \mathcal{T}$$

that maps the unit triangle $0 < \xi < 1, 0 < \eta < 1-\xi$ onto $\mathcal{T}$ as depicted in Figure (5.21). We assume here the mapping to be polynomial, at order q. The manifold's geometry is thus approximated by the polynomial mapping:

$$\boldsymbol{x}^h(\xi, \eta) = \sum_{i=1}^{\mathcal{N}_q} \boldsymbol{x}_i L_i^q(\xi, \eta)$$

where the $\boldsymbol{x}_i$ are the $\mathcal{N}_q$ nodal coordinates of each triangle of order q ($\mathcal{N}_q = (q+1)(q+2)/2$ for Lagrange shape functions) and L_i^q is the Lagrange shape function of order q relative to node i. The local tangent basis $(\boldsymbol{t}_\xi, \boldsymbol{t}_\eta)$ will thus clearly be approximated using the triangle's tangents, which can be computed by using the derivatives of the polynomial mapping:

$$\boldsymbol{\tau}_\xi^h(\xi,\eta) = \frac{\partial \boldsymbol{x}^h}{\partial \xi} = \sum_{i=1}^{\mathcal{N}_q} \boldsymbol{x}_i \frac{\partial L_i^q}{\partial \xi} \ , \tag{5.26}$$

$$= \sum_{i=1}^{\mathcal{N}_{q-1}} \boldsymbol{\tau}_{\xi i} L_i^{q-1}(\xi,\eta) \ . \tag{5.27}$$

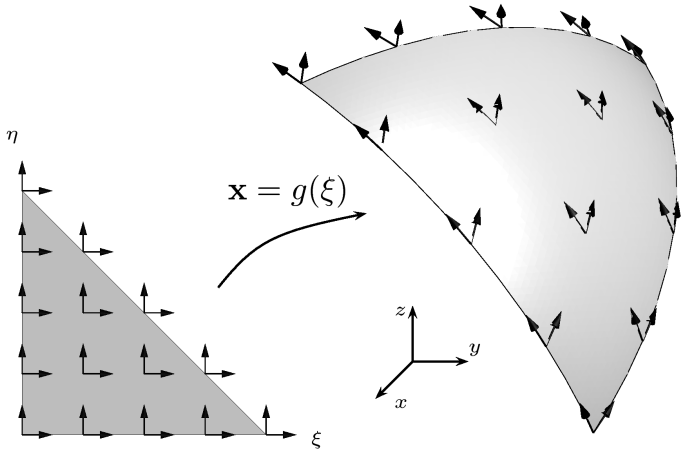


Figure 5.21: Mapping of the reference unit triangle $0 < \xi < 1$, $0 < \eta < 1 - \xi$ onto the curvilinear triangle $\mathcal{T}$.

where $\boldsymbol{\tau}_{\xi i} = \boldsymbol{\tau}_\xi^h(\xi_i, \eta_i)$ is a tangent vector evaluated at the ith Lagrange point (ξ_i, η_i) of order $q-1$, and with the second basis vector $\boldsymbol{\tau}_\eta^h$ expressed in the same way. The tangents $\boldsymbol{\tau}_\xi^h$ and $\boldsymbol{\tau}_\eta^h$ are eventually normalized to obtain the following approximation of the unit tangent basis $(\boldsymbol{t}_\xi^h, \boldsymbol{t}_\eta^h)$:

$$\boldsymbol{t}_\xi \simeq \boldsymbol{t}_\xi^h = \frac{\boldsymbol{\tau}_\xi}{\| \boldsymbol{\tau}_\xi \|} \tag{5.28}$$

and $\boldsymbol{t}_\eta^h$ written in the same way. Any tangent vector at point $\boldsymbol{x}^h(\xi,\eta)$ is therefore a linear combination of $\boldsymbol{t}_\xi^h$ and $\boldsymbol{t}_\eta^h$.

For the formulation (5.25) to be complete, the components u_ξ and u_η still have to be discretized. This can be achieved by discretizing the vector amplitudes as classical scalar fields at any order p:

$$u_\xi(\xi,\eta) \simeq u_\xi^h(\xi,\eta) = \sum_{j=1}^{\mathcal{N}_p} L_j^p(\xi,\eta) U_{\xi j} \ , \tag{5.29}$$

which yields the following discretized tangent vector field

$$\begin{aligned} \boldsymbol{u}^h(\xi,\eta) &= u_\xi^h(\xi,\eta)\boldsymbol{t}_\xi^h(\xi,\eta) + u_\eta^h(\xi,\eta)\boldsymbol{t}_\eta^h(\xi,\eta) \\ &= \sum_{j=1}^{\mathcal{N}_p} \left(L_j^p(\xi,\eta) U_{\xi j} \right) \boldsymbol{t}_\xi^h(\xi,\eta) \\ &\qquad + \sum_{j=1}^{\mathcal{N}_p} \left(L_j^p(\xi,\eta) U_{\eta j} \right) \boldsymbol{t}_\eta^h(\xi,\eta) \ . \end{aligned} \tag{5.30}$$

The fact that $\mathcal{N}_p$ is not necessary equal to $\mathcal{N}_q$ allows one to compute higher-order representation at order p of the vector field on low order mappings of order q. Notice finally that this approach may thus be considered as using vectorial shape functions multiplying the amplitude nodal values $U_{\boldsymbol{\xi}_j}$ to represent vectorial fields. Those shape and test functions are however non-polynomial functions since the tangent basis has to be normalized.

Approximate Riemann solver with discontinuous normals

The computation of the weak formulation (5.20), and in particular the integration on the edges, still requires the definition of an approximate Riemann solver taking into account the possible discontinuities of the normals in the manifold representation. We consider the expression of the vector field (5.30) in the local tangent basis and we use the simple Lax-Friedrich solver (5.22):

$$\boldsymbol{F}(\boldsymbol{U}^h)\cdot\boldsymbol{n} = \frac{\boldsymbol{F}(\boldsymbol{U}_L)+\boldsymbol{F}(\boldsymbol{U}_R)}{2}\cdot\boldsymbol{n} + |\lambda|_{max}\frac{\boldsymbol{U}_L-\boldsymbol{U}_R}{2} \ .$$

The fluxes are written in the global 3D basis $\hat{\boldsymbol{e}}_{x,y,z}$. Therefore, the vectorial field (5.30) has to be expressed in three components as well for the flux computation.

Let us consider the configuration depicted in Figure (5.22): we wish to compute the normal fluxes on the edge Γ between two triangles Ω_L and Ω_R. To simplify the illustration, those triangles are considered flat, belonging to the two planes P_L and P_R. At each edge point, a local basis is obtained by considering the tangent component $\boldsymbol{m}$, common to both triangles, and the left and right normals $\boldsymbol{n}_{L,R}$. On each triangle, a vector field is defined in both basis, say $\boldsymbol{v}_L \in (\boldsymbol{n}_L, \boldsymbol{m})$ and $\boldsymbol{v}_R \in (\boldsymbol{n}_R, \boldsymbol{m})$.

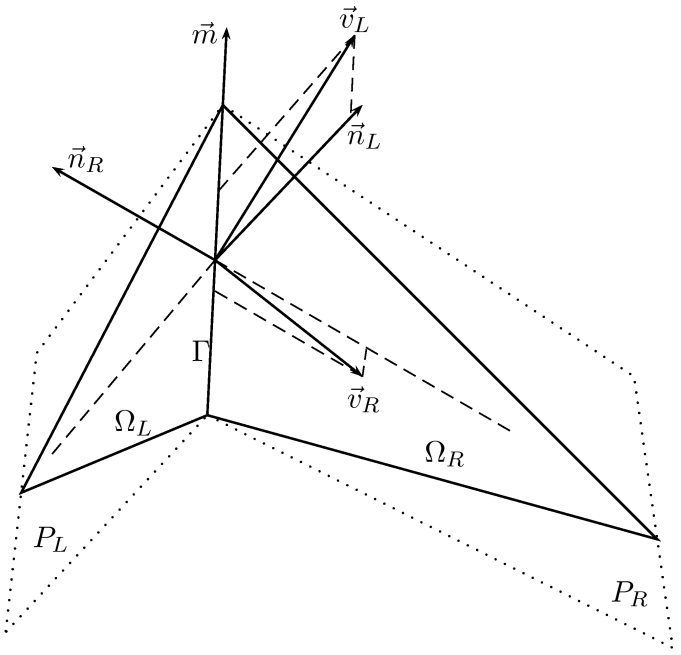


Figure 5.22: Illustration of the discontinuous normals on the edge Γ between two flat triangles Ω_L and Ω_R, lying in planes P_L and P_R with corresponding vectorial fields $\boldsymbol{v}_L$ and $\boldsymbol{v}_R$, respectively.

Let $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$ denote the fluxes computed on the left and right sides of the edge, respectively. For classical computations on a plane, the flux has to be evaluated on only one element. Indeed, the continuity of the normals leads to the same flux value on both sides, $\boldsymbol{F} \cdot \boldsymbol{n} = \boldsymbol{F}^L \cdot \boldsymbol{n}_L = \boldsymbol{F}^R \cdot \boldsymbol{n}_R$, since $\boldsymbol{n}_L = \boldsymbol{n}_R$. But the introduction of discontinuities in the normal directions requires us to compute different fluxes $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$ on both sides of the edges.
We consider in this flux computation that the normal discontinuities have no impact on the physical processes computed. Therefore, the normal flow to an edge should not slow down or be annihilated by a variation of direction, even in the case of orthogonal normals $\boldsymbol{n}_L \cdot \boldsymbol{n}_R = \boldsymbol{0}$. We thus assume a conservation of the vectorial fields amplitudes across the edge, only the directions have to be modified.
Considering for instance the computation of the left flux, $\boldsymbol{F}^L(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n}_L$, the left value of the vector field $\boldsymbol{U}_L$ already lies in the appropriate plane P_L, i.e. in the appropriate basis $(\boldsymbol{m}, \boldsymbol{n}_L)$, while the direction of the right value $\boldsymbol{v}_R$ has to be modified as

$$\boldsymbol{v}_R^L = (\boldsymbol{v}_R \cdot \boldsymbol{m})\, \boldsymbol{m} + (\boldsymbol{v}_R \cdot \boldsymbol{n}_R)\, \boldsymbol{n}_L \tag{5.31}$$

where the notation $\boldsymbol{v}_R^L$ denotes the right vector $\boldsymbol{v}_R$ expressed in the left basis $(\boldsymbol{m}, \boldsymbol{n}_L)$, and with $\boldsymbol{v}_R \cdot \boldsymbol{m}$ and $\boldsymbol{v}_R \cdot \boldsymbol{n}_R$ the tangential and normal components of $\boldsymbol{v}_R$ respectively.
This operation simply corresponds to a rotation of the vectorial field from the basis $(\boldsymbol{m}, \boldsymbol{n}_R)$ to the basis $(\boldsymbol{m}, \boldsymbol{n}_L)$, the amplitude being conserved. The same operation is then applied to $\boldsymbol{v}_L$ to obtain $\boldsymbol{v}_L^R$ for the right flux computation $\boldsymbol{F}^R(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n}_R$. The Lax-Friedrich relation eventually yields two expressions for the fluxes on the left and right sides of the edge, $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$:

$$\begin{aligned}
\boldsymbol{F}^L(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n}_L &= 0.5 \left(\boldsymbol{F}(\boldsymbol{U}_L^L) + \boldsymbol{F}(\boldsymbol{U}_R^L) \right) \cdot \boldsymbol{n}_L + |\lambda|_{max} \left(\boldsymbol{U}_L^L - \boldsymbol{U}_R^L \right) \\
\boldsymbol{F}^R(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n}_R &= 0.5 \left(\boldsymbol{F}(\boldsymbol{U}_L^R) + \boldsymbol{F}(\boldsymbol{U}_R^R) \right) \cdot \boldsymbol{n}_R + |\lambda|_{max} \left(\boldsymbol{U}_L^R - \boldsymbol{U}_R^R \right) \quad .
\end{aligned}$$

Mass matrix and convective operators for vectorial test functions

We may now compute the weak formulation on curved manifolds since all the fields and fluxes have been defined and discretized in the

local tangent and 3D basis respectively.
Let us consider an element at order p with $\mathcal{N}_p$ nodal values to represent a scalar field. According to the discretization (5.30), the vectorial weak equation (5.20) presents then $2\mathcal{N}_p$ unknown values. The time derivative term from the scalar mass conservation equation is computed using the classical local mass matrix of size $\mathcal{N}_p \times \mathcal{N}_p$, while the time derivative terms from the vectorial momentum equation leads to a local *vectorial* mass matrix of $2\mathcal{N}_p \times 2\mathcal{N}_p$:

$$\begin{aligned}\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}^h \cdot \hat{\boldsymbol{u}}^h \, d\Omega &= \frac{\partial}{\partial t}\int_{\Omega_e} U_{\alpha j} L_j^p \boldsymbol{t}_\alpha^h \cdot \boldsymbol{t}_\beta^h L_i^p \, d\Omega \\ &= \underbrace{\int_{\Omega_e} L_j^p \boldsymbol{t}_\alpha^h \cdot \boldsymbol{t}_\beta^h L_i^p \, d\Omega}_{\text{Local mass matrix of } 2\mathcal{N}_p \times 2\mathcal{N}_p} \frac{\partial U_{\alpha j}}{\partial t}\end{aligned}$$

with the summation on the repeated indices j and α, i and j denoting the degrees of freedom numbering ranging from 1 to $\mathcal{N}_p$, and the greek indices $\alpha, \beta = \xi, \eta$ denoting the numbering of components in the tangent dimension. The indices i and β without summation lead thus to the expected $2\mathcal{N}_p$ equations on each element.

The convective terms are quite more complicated since they involve the spatial derivatives of the tangent basis. Considering the expression of the convective fluxes in 3D as in (5.17) with three components $(\boldsymbol{H}_x, \boldsymbol{H}_y, \boldsymbol{H}_z)$, each component being expressed in 3D as well, $\boldsymbol{H}_a = H_{ax}\hat{\boldsymbol{e}}_x + H_{ay}\hat{\boldsymbol{e}}_y + H_{az}\hat{\boldsymbol{e}}_z$, the discrete convective terms read:

$$\begin{aligned}\int_{\Omega_e} \boldsymbol{H} \cdot \nabla \hat{\boldsymbol{u}}^h \, d\Omega &= \int_{\Omega_e} \boldsymbol{H} \cdot \nabla \left(L_i^p \boldsymbol{t}_\beta^h \right) \, d\Omega \\ &= \int_{\Omega_e} H_{ab} \frac{\partial}{\partial a}\left(L_i^p t_{\beta b}^h \right) \, d\Omega \quad \text{for} \quad i \in [1, \mathcal{N}_p], \beta \in [\xi, \eta] \\ &= \int_{\Omega_e} H_{ab} \left(t_{\beta b} \frac{\partial L_i^p}{\partial a} + L_i^p \frac{\partial t_{\beta b}^h}{\partial a} \right) \, d\Omega \, . \end{aligned} \tag{5.32}$$

with implicit summation on indices $a, b = x, y, z$ denoting the components in the three-dimensional basis.
The derivation of the tangent basis $\frac{\partial t_{\beta b}^h}{\partial a}$ involves the inverse of the Jacobian matrix of the mapping and the derivatives with respect to the

reference triangle basis $(\xi, \eta) \in \mathcal{R}^2$ according to (5.24). This tangent derivation clearly yields second derivatives of the mapping, which corresponds to the curvature of the manifold. Therefore, the tangent derivative term in (5.32) corresponds to the curvature terms usually introduced in the classical 2D approach, including the Christoffel symbols. The difference lies in the fact that the current approach is more general since those acceleration terms are contained in the numerical discretization, i.e. in the derivation of the element mappings. No analytical computation is required for describing the manifold. We also observe in (5.32) that the use of a constant orthogonal three-dimensional basis as $\hat{\boldsymbol{e}}_{x,y,z}$ for defining the vectorial fields suppresses the curvature terms for $a \neq b$ and introduce a third equation, yielding the classical form of the 3D approach $\int_{\Omega_e} F_{ab} \frac{\partial L_i^p}{\partial a} \, d\Omega$ where the summation on b disappeared, providing the three usual equations for the three components u_b.

The expressions of the vectorial mass matrix and convective terms may be clarified by developing the expression for both amplitudes and tangents $U_{\xi j}, \boldsymbol{t}_\xi^h$ and $U_{\eta j}, \boldsymbol{t}_\eta^h$. This part of the system thus reads:

$$\begin{bmatrix} \int_{\Omega_e} L_j^p L_i^p \, d\Omega & \int_{\Omega_e} L_j^p L_i^p \boldsymbol{t}_\xi^h \cdot \boldsymbol{t}_\eta^h \, d\Omega \\ \int_{\Omega_e} L_j^p L_i^p \boldsymbol{t}_\xi^h \cdot \boldsymbol{t}_\eta^h \, d\Omega & \int_{\Omega_e} L_j^p L_i^p \, d\Omega \end{bmatrix} \begin{bmatrix} \partial U_{\xi j}/\partial t \\ \partial U_{\eta j}/\partial t \end{bmatrix} = \begin{bmatrix} \int_{\Omega_e} H_{ab} \left(t_{\xi b} \dfrac{\partial L_i^p}{\partial a} + L_i^p \dfrac{\partial t_{\xi b}^h}{\partial a} \right) \, d\Omega \\ \int_{\Omega_e} H_{ab} \left(t_{\eta b} \dfrac{\partial L_i^p}{\partial a} + L_i^p \dfrac{\partial t_{\eta b}^h}{\partial a} \right) \, d\Omega \end{bmatrix}$$

where we used the relation $\boldsymbol{t}_\alpha^h \cdot \boldsymbol{t}_\alpha^h = 1$. We clearly observe on this vectorial mass matrix the coupling between the components and the vectorial nature of the equation. Using an orthogonal basis for $\boldsymbol{t}^h$ (for instance $\hat{\boldsymbol{e}}_{x,y,z}$) yields $\boldsymbol{t}_\alpha^h \cdot \boldsymbol{t}_\beta^h = 0$ for $\alpha \neq \beta$ which decouples the system and provides the simple block-diagonal repetition of the classical scalar mass matrix as in the classical approaches. The integration of the fluxes on the edges and of the source terms are not written here since they do not present any particular difficulty compared to the volume convective term.

Practically, the convective term (5.32) has been implemented to minimize the CPU cost, instead of minimizing the memory footprint, i.e. by pre-computing on each element the mass matrix and the nine convective matrices of size $2\mathcal{N}_p \times \mathcal{N}_k$ with $\mathcal{N}_k$ the number of integration points per element. The same implementation for the classical 3D approach requires to store only three matrices of size $\mathcal{N}_p \times \mathcal{N}_k$. Although requiring to store a larger number of matrices, the local tangent basis approach remains more efficient for sufficiently large computations since using only two unknowns for each vectorial field instead of three plus a tangency constraint.

5.2.3 Solving the shallow water equations on curved manifolds

This subsection deals with the validation of the local tangent basis approach with the shallow water equations. In particular, some standard benchmarks of oceanic/atmospheric processes are performed on the sphere. More irregular complex curved manifolds are eventually considered for testing the robustness and generality of the method.

Four of the seven classical Williamson test cases (W92) are considered [Williamson et al., 1992] in the following, as the perturbed Rossby-Haurwitz waves test case [Thuburn and Li, 2000, Smith and Dritschel, 2006]. Some W92 are not considered here since they do not involve vectorial conservation laws, as the advection of the cosine bell, or because of the complicated source terms involved. The Williamson test cases are considered in every paper dealing with computation of oceanic or atmospheric processes on the sphere, and have become *de facto* standard benchmarks. For the second and third W92 are provided analytical solutions. A refinement convergence study is thus performed by comparison with the analytical solution, while the validation for other test cases requires a comparison to results from other numerical techniques.

High-order polynomial mapping

Since using the high-order discontinuous Galerkin method, we may take advantage of the high-order polynomials for both the solution approximation and for the geometry, i.e. the sphere discretization. Some of the meshes used for the computations are depicted in Figure (5.23). They are obtained from the recursive divisions of icosahedrons, each triangle being divided in four new triangles projected on the sphere. The left column of Figure (5.23) illustrates the classical meshes made of piecewise linear faces, while the right column illustrates the meshes build using a fourth-order mapping, i.e. piecewise fourth order polynomial faces.

High-order meshes made of curved three-dimensional triangles with high order polynomial mappings are used to obtain a better representation of the curved manifolds geometry. Considering a curvilinear triangle at order q composed of a set of $\mathcal{N}_q$ coordinates $\boldsymbol{X} \in \mathcal{R}^3$, the approximate position vector $\boldsymbol{x}^h \in \mathcal{R}^3$ on each triangle reads:

$$\boldsymbol{x}_j^h = \sum_{i=1}^{\mathcal{N}_q} \psi_i(\boldsymbol{\xi}) \boldsymbol{X}_{i,j}, \qquad j = 1, \ldots, 3 \tag{5.33}$$

where $\psi(\boldsymbol{\xi})$ denotes the shape functions corresponding to the coordinates $\boldsymbol{X}$ in the bi-dimensional parametric space $\boldsymbol{\xi} = (\xi, \eta)$. Those shape function still need to be defined. Two classical representations are considered in the following:

- using Lagrange shape function, as for the polynomial approximation of the solution. A mapping of order q requires a number of coordinates $\mathcal{N}_q = \frac{(q+1)(q+2)}{2}$ to define the curved triangle, as depicted in the right part of Figure (5.24).

- using incomplete Lagrange functions, also known as Serendip space functions. This representation does not involve any nodal value inside the elements as illustrated in the left part of Figure (5.24). A mapping of order q provides thus an incomplete set of monomes for order q, but requires a lower number of coordinates since $\mathcal{N}_q = 3q$.

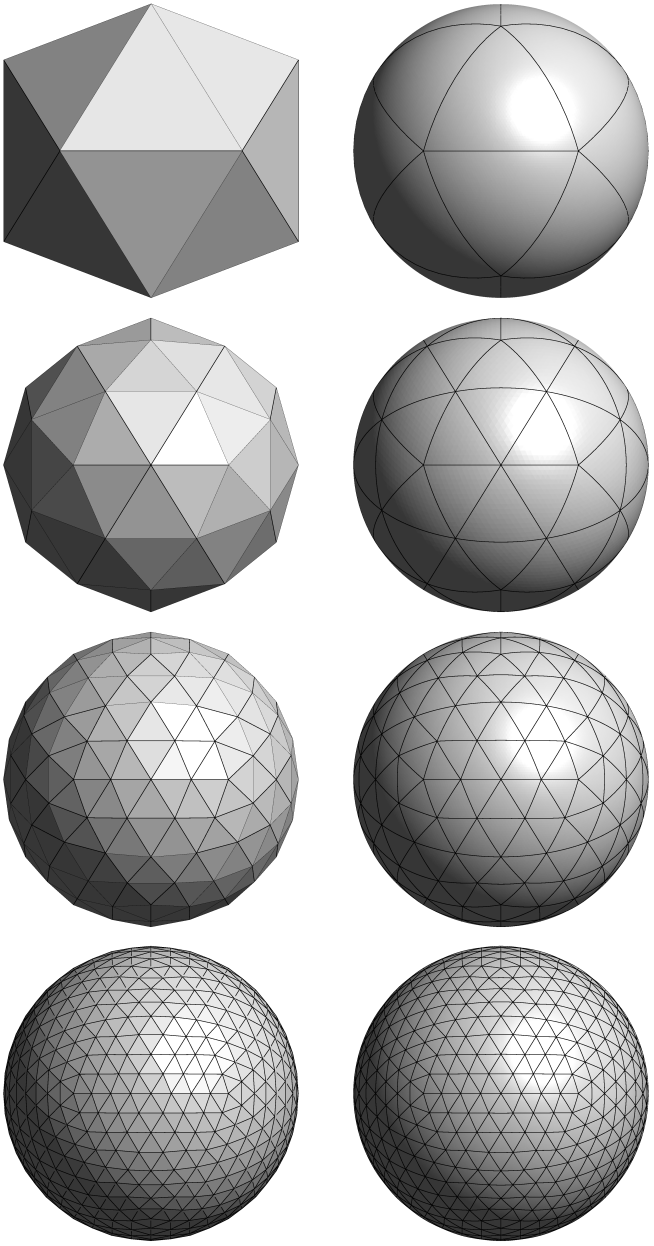


Figure 5.23: Icosahedron-deduced meshes of the sphere used for the refinement convergence study. Meshes present 20, 80, 320 and 1280 elements (top to bottom) with $\mathcal{P}^1$ and $\mathcal{P}^4$ polynomial mappings on the left and right columns respectively.

When considering curved bi-dimensional triangles to represent bluff bodies as islands, the incomplete Serendip is quite sufficient since the boundaries representation is actually complete. Moreover, the high order Lagrange shape functions present nodes located inside the triangle. The position of those nodes may appear arbitrary since they do not add informations on the boundary representation. However, considering curved manifolds, the whole triangle is describing the surface. The use of an incomplete order may thus have some impact on the solution itself.

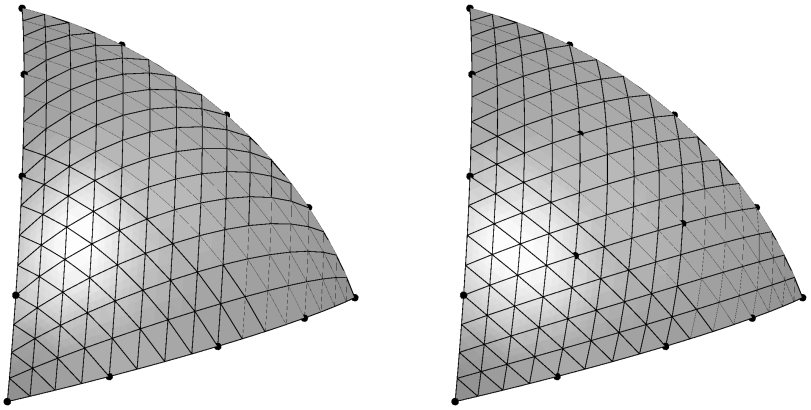


Figure 5.24: The same triangle corresponding to a fourth-order mapping, using the incomplete polynomial Serendip mapping (left triangle) or the complete Lagrange mapping (right triangle) presenting additional nodes inside the element. The inner lines represent the mapped iso-ξ, iso-η and iso-$(1-\eta-\xi)$ values in the reference parametric domain (ξ, η), and are therefore an illustration of the triangle's mapping deformation.

The inner lines into the triangle depicted in Figure (5.24) are resulting from the mapping of the iso-ξ, iso-η and iso-$(1-\xi-\eta)$ lines in the reference bi-dimensional parametric space $\boldsymbol{\xi} = (\xi, \eta)$. Therefore, the space distortion between those lines represents the distortion introduced by the mapping. The incomplete Serendip representation clearly leads to larger distortions, i.e. larger spatial variations of the Jacobian matrix and its determinant. The Lagrange function spaces seem thus better suited for such mappings. Indeed, strongly distorted elements will lead to high quadrature errors, and then to less accurate solutions. Moreover,

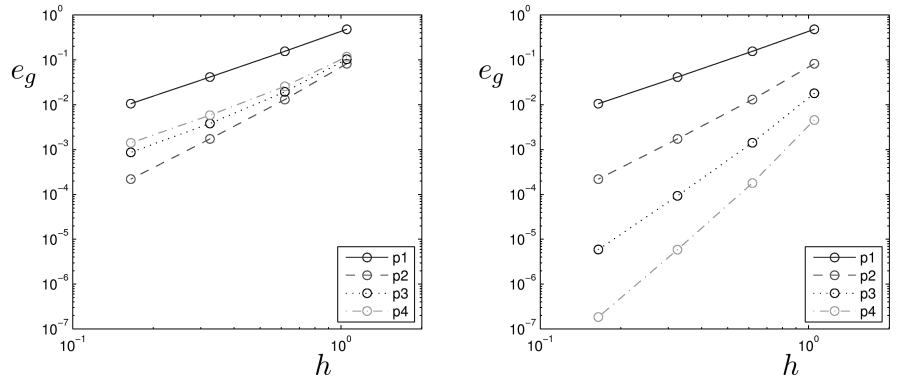


Figure 5.25: The expected convergence rate of $\mathcal{O}(p+1)$ for the geometric error is not reached with high order incomplete Serendip mappings (left) while rates of 2.01, 3.04, 4.06 and 5.08 are obtained with the Lagrange polynomials mappings (right) for orders $q = 1, \ldots, 4$ respectively.

Figure (5.25) illustrates the evolution of the $\mathcal{L}_2$ norm of the geometric error e_g, computed as

$$e_g^2 = \int_\Omega \left(\boldsymbol{x}^h - \boldsymbol{x} \right)^2 \, d\Omega$$

on a sphere of unit radius.
The incompleteness of the Serendip polynomial basis clearly prevents from the expected convergence rate of $\mathcal{O}(p+1)$. The use of a complete Lagrange function space at order q is thus mandatory to keep the geometrical errors lower than the discretization errors of the solution itself, and to obtain the expected theoretical convergence rate for the DG solution.

W92 test cases 2 and 3: steady zonal geostrophic flows

Both test cases 2 and 3 consists of steady state solutions of the nonlinear inviscid shallow water equations on the rotating sphere. W92 2 considers a solid body rotation along the Earth's rotation axis while W92

p	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η
1	2.1	2.8	2.4	2.8	2.3	2.8	2.3	2.8
2	1.9	1.9	3	3.9	3	3.9	3	3.9
3	1.9	1.9	4.2	4.1	4.1	4.1	4.1	4.9
4	1.9	1.9	4.2	4.1	4.4	3.9	5.2	5.8
	$q=1$		$q=2$		$q=3$		$q=4$	

Table 5.1: Convergence rates of the relative $\mathcal{L}_2$ error on elevation and velocity of W92 2, for solutions of order $p = 1, \ldots, 4$ and mapping orders $q = 1, \ldots, 4$ to represent the sphere.

p	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η	$\boldsymbol{v}$	η
1	2.2	2.5	2.2	2.5	2.2	2.5	2.2	2.5
2	2.3	2	2.5	3.8	2.4	3.8	2.4	3.8
3	2.3	2	4.9	5.3	5	5.5	5	5.5
4	2	1.9	4.9	4.5	4.6	4.6	4.6	5.7
	$q=1$		$q=2$		$q=3$		$q=4$	

Table 5.2: Convergence rates of the relative $\mathcal{L}_2$ error on elevation and velocity of W92 3, for solutions of order $p = 1, \ldots, 4$ and mapping orders $q = 1, \ldots, 4$ to represent the sphere.

3 considers this rotation only for the latitudes ranging between 30 S and 90 N, i.e. the North pole.

The $\mathcal{L}^2$ norm of the solution error is computed after 5 days as suggested in [Williamson et al., 1992], on the four meshes depicted in Figure (5.23) for mappings of order 1 to 4. The error norm is computed as

$$e^2 = \frac{\int_\Omega \left(\boldsymbol{u}^h - \boldsymbol{u}\right)^2 \, d\Omega}{\int_\Omega 1 \, d\Omega}$$

and a relative error is finally obtained by dividing by the mean value of the field considered.
In Figure (5.26) are depicted the initial conditions for both test cases: the elevation is defined to compensate the Coriolis effect and the advection terms. The figure also illustrates the evolution of the error with

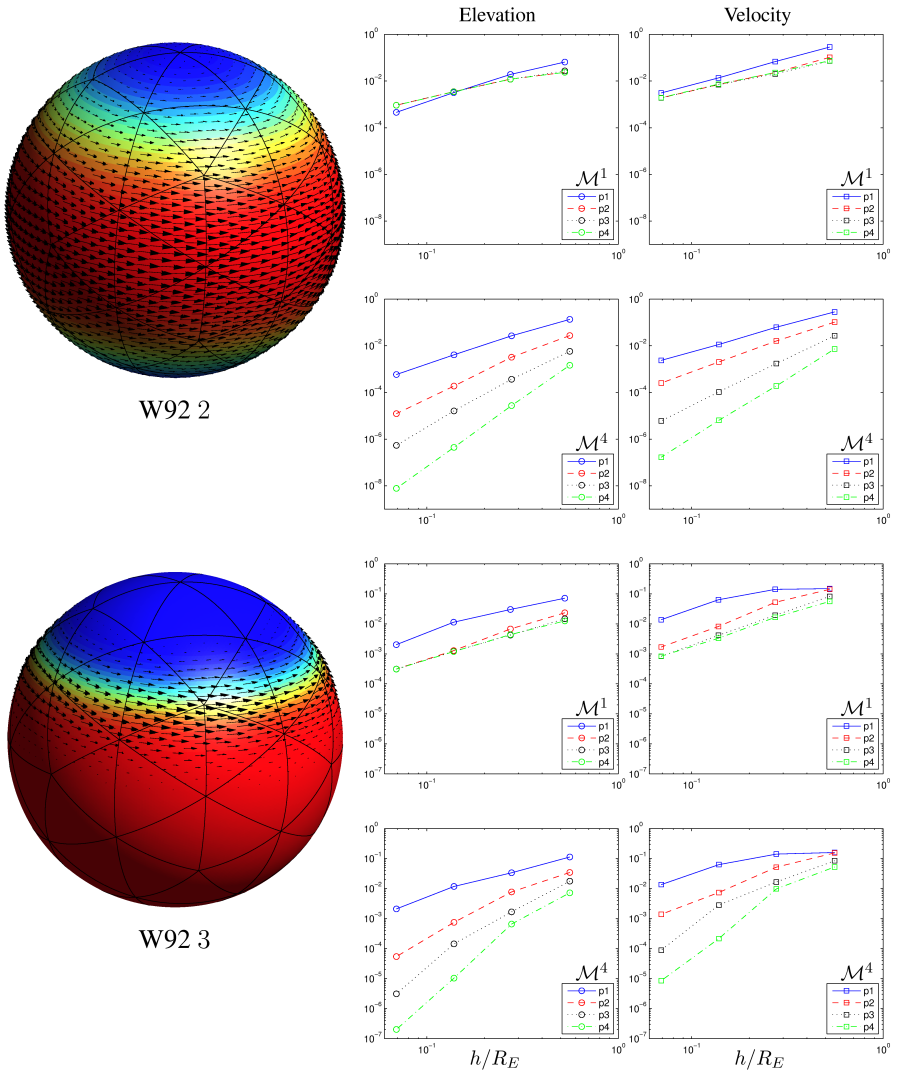


Figure 5.26: Steady solution for W92 2 and 3 and the $\mathcal{L}_2$ norm of the relative error computed after five days. The left column presents the error based on the elevation field, while the right column is based on the velocity. The expected convergence rate is not reached for a high-order solution using a low (first) order mapping $\mathcal{M}^1$ to represent the sphere, while an iso-parametric representation (fourth order mapping $\mathcal{M}^4$) ensures the theoretical rate.

the polynomial solution order, mapping order, and with the relative element size h/R_E with R_E the Earth radius. The plots are related to both mapping orders 1 and 4. The corresponding convergence rates are given in tables Table (5.1) and Table (5.2), also including the results for mapping orders 2 and 3.
We observe for the piecewise linear representation of the sphere, i.e. $q = 1$, the expected convergence rate of $\mathcal{O}(p+1)$ for the piecewise linear solution $p = 1$, while this rate is not reached for higher order solutions. This lack of convergence is clearly due to the fact that the error is dominated by the geometric errors. Indeed, we observe that the theoretical convergence rate is recovered when increasing the mapping order. Notice also that a second order mapping seems sufficient to obtain the expected rate with $p = 3$, while the use of $p = 4$ requires a fourth order mapping. We finally observe that the test case 3 seems less sensitive to the mapping order, the expected rate for $p = 4$ is almost reached with a second order mapping. This is probably due to the higher interpolation order for W92 3, more than one order of magnitude, while the geometrical errors remains the same.

W92 test case 6: Rossby-Haurwitz waves

Although providing no analytical solution, this test case has been widely used for model inter-comparison. It involves the Coriolis force, the elevation gradients and advection terms, and a time evolving solution. The solutions of the non divergent barotropic vorticity equations experience a slow solid rotation around the Earth under the influence of slow waves. The initial solution presents the 4-periodic pattern depicted in Figure (5.27). This solution has been shown to be very unstable, and in particular very sensitive to asymmetry in truncation errors [Thuburn and Li, 2000]. Spectral methods for instance will keep the initial 4-periodic pattern unchanged because of their high degree of symmetry, only the rounding errors are responsible for triggering the instability. On the other hand, the use of icosahedral meshes will excite the 2-periodic component of the unstable mode since the icosahedral mesh and the corresponding truncation errors are 2-periodic [Comblen et al., 2008].

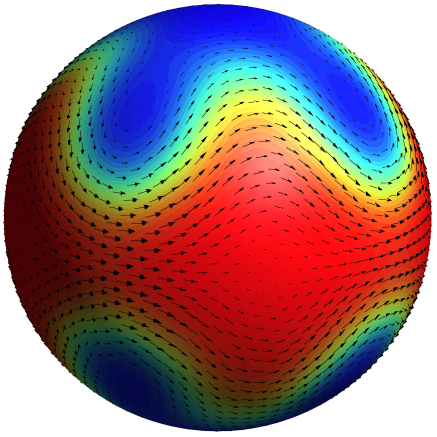


Figure 5.27: 4-periodic initial solution of W92 test case 6.

Figure (5.28) provides a comparison between the DG solution with the local tangent basis approach (left column) and the solution obtained by the German Weather Service using a spectral method. The DG solution used the 320 elements mesh with fourth order mapping and solution, leading to 4800 degrees of freedom, while the spectral method was computed on a 1536×768 grid. The comparison reveals a good agreement between the two approaches, especially in the earlier days of the computation when the 2-periodic component of the unstable modes has not been much excited.

Perturbed Rossby-Haurwitz test case

Since the W92 test case 6 is very sensitive to the methods and the corresponding truncation errors, it does not facilitate the comparison of different methods. A variation has been proposed by [Smith and Dritschel, 2006], which consists in perturbing the initial pattern. The solution thus becomes more deterministic, especially during the early days, and makes comparisons more relevant.

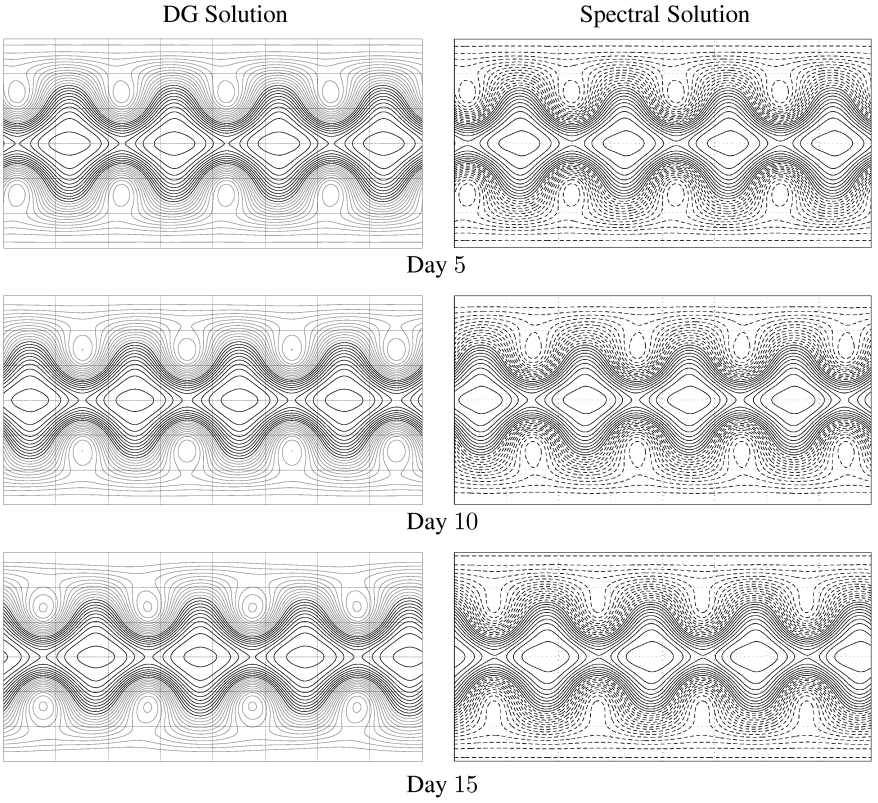


Figure 5.28: Comparison between the DG method with the local tangent basis approach (left column) and the reference solution from the German Weather Service using a spectral method for W92 test case 6. The reference solution was obtained on a 1536×768 grid and a 90 s time step, while the DG method considered a 320 elements mesh with $(p, q) = (4, 4)$ and a 100 s time step. The interval between contour lines is 100 m for both plots.

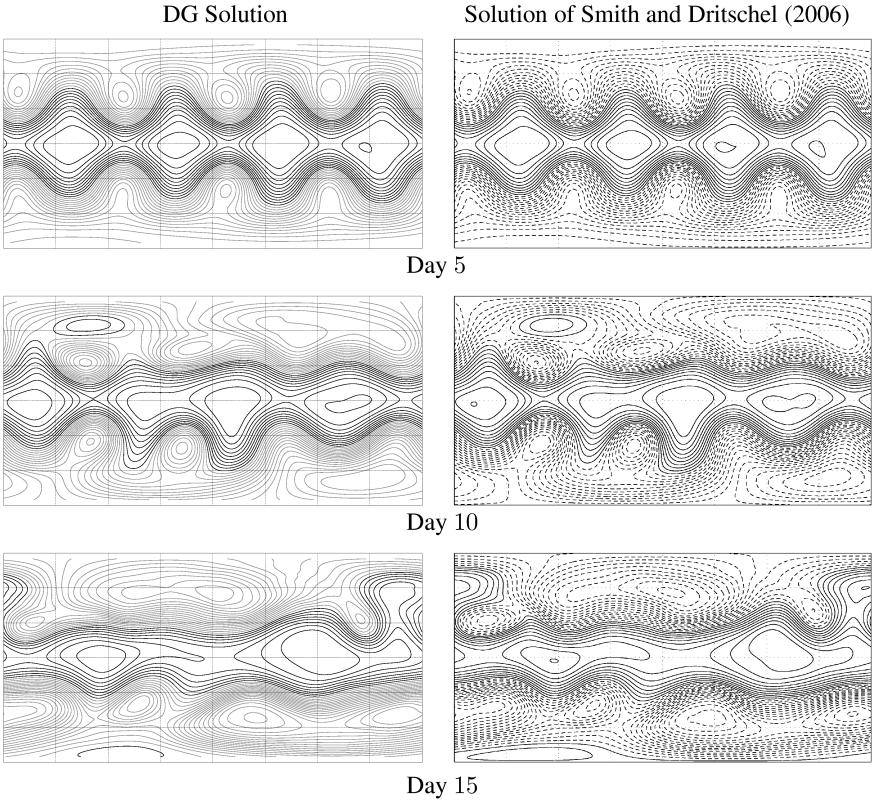


Figure 5.29: A good agreement is observed between the DG solution with local tangent basis approach (left column) and the advective-contour semi-Lagrangian method by Smith and Dritschel (right column) after 5, 10 and 15 days for the perturbed Rossby-Haurwitz test case.

In Figure (5.29) are compared the results obtained with the DG method and the results from [Smith and Dritschel, 2006]. The latter used a 256×256 grid with a contour-advective semi-Lagrangian method, while we considered with our approach the icosahedral mesh of 1280 curved elements with a fourth order mapping and solution. This corresponds to 19200 degrees of freedom, i.e. about 3.5 less degrees of freedom than the reference method. A good agreement between the two approaches is once again observed.

Gravity waves on irregular manifolds

We finally validate the robustness and the generality of the method by observing the propagation of gravity waves on the surface of very irregular manifolds. The non-linear shallow water equations are solved without any source term with the same approximate solver and flux function definitions than for the computations on the sphere.
The first computation considers the mesh of the cube depicted in Figure (5.30) with fourth order shape functions to represent the solution. The initial solution is an arbitrary gaussian elevation on the top and bottom of the cube, with a zero velocity. We observe in Figure (5.31) that the gravity waves propagate without any perturbation, even in the presence of large discontinuities of the normals directions on the sharp edges of the cube. Indeed, we considered a conservation of the velocities amplitudes through edges in the approximate Riemann solver.

The second computation deals with the gravity waves propagation on the mesh of a frog depicted in the right part of Figure (5.30). We observe that the frog is quite well represented with a coarse second order curved mesh. The initial gaussian elevation propagates and becomes highly distorted due to the irregularity of the manifold, while the fourth order solution remains perfectly smooth. This test case also illustrates the robustness and high level of generality of the method, the more complicated part of the simulation consisting in the creation of high order meshes while no analytical expression of the manifold is required for computing PDE's.

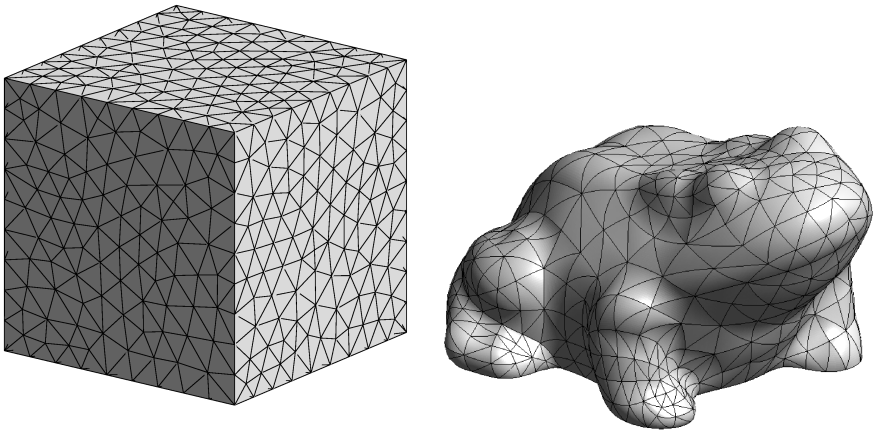


Figure 5.30: High-order meshes of irregular curved manifolds

5.3 Conclusions

One of the more crucial issues for performing oceanic computations with high-order DG methods is the treatment of the geometry. As high-order spectral elements, the use of high-order polynomials with the DG method tends to generate spurious oscillations in irregular topography. Moreover, reaching the optimal convergence rates requires the interpolation order to be consistent with the representation of the geometry by means of, for instance, isoparametric computations.

In the first section of the chapter, the DG method has been applied to solve tidal flows around the Rattray island. It has been shown that it is possible to accurately compute oceanic flows in complex irregular domains with the high-order DG method, and therefore to benefit from the high-order convergence properties. This is a difficult task since it involves complex irregular bathymetry and geometry. Indeed, the island boundaries have to be represented accurately and the bathymetry has to be treated carefully to avoid oscillations. Moreover, this computation is complicated to achieve because of the quasi-steady nature of the flow. The oscillations have the opportunity to grow and propagate to the entire solution only if the flow is sufficiently steady.
Realistic parameters and data such as the tidal forcing, the bathymetry

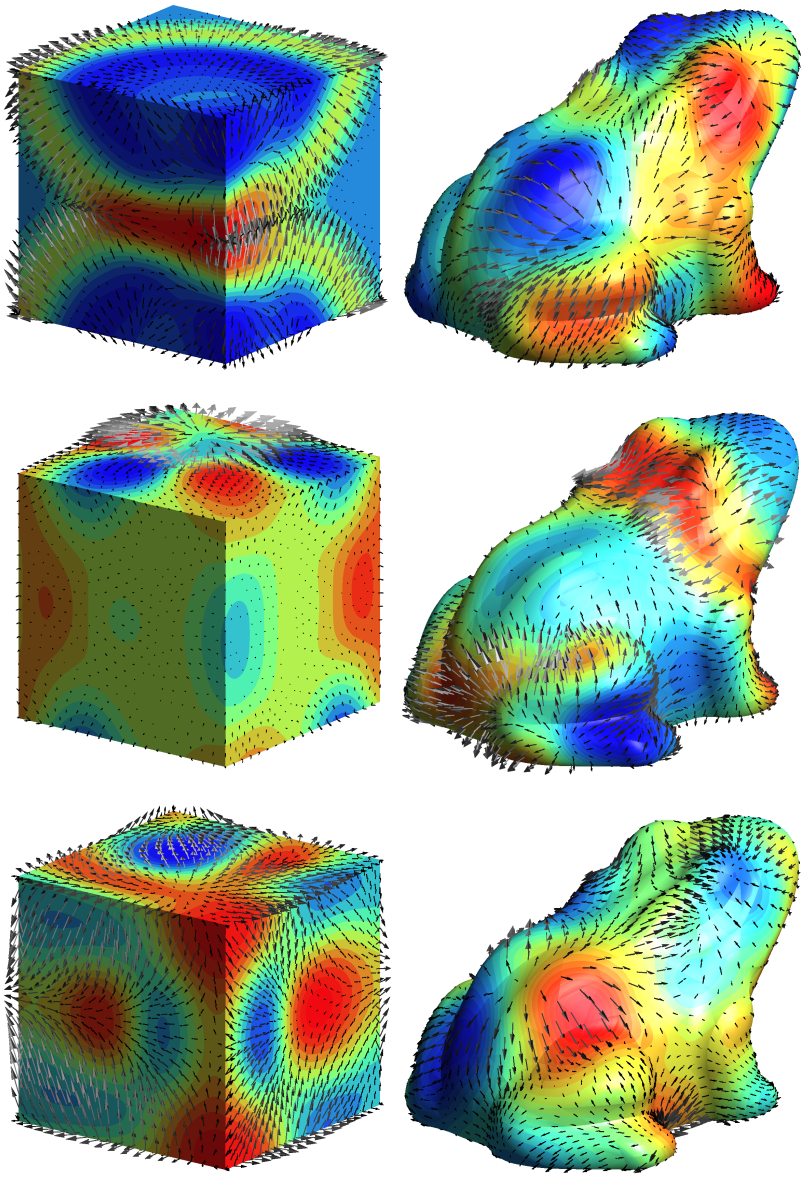


Figure 5.31: Propagation of gravity waves on the surface of a cube (left) and of a high-order frog mesh (right), illustrating the robustness and generality of the method.

and the complex island shape were used. We have observed the importance of the boundary numerical representation in the context of ocean processes. In the framework of an ocean model with thousands of coasts and islands, we cannot afford the same boundary discretization than for cylinder or airfoils studies. With very complex bluff bodies and very coarse meshes, some usually perfectly working techniques as the smoothing of the normals may reveal inefficient to avoid oscillations and spurious eddies, while a second order mapping still leads to very smooth solutions. Moreover, the continuous reconstruction of the normals on very coarse grids does not represent accurately the islands shapes as the high order curved meshes.
The Rattray test case highlights the fact that some non physical eddies may have the right shapes and amplitudes for wrong reasons when using piecewise linear representations of the boundaries and high order shape functions. Moreover, such a poor representation may lead to an unstable scheme for some specific data as a strong varying shallow bathymetry. The DG method with suitable boundary discretization provides an accurate model with a very low amount of diffusion so that no spurious eddy is created. A variable diffusivity may then be calibrated to take into account the subgrid scale processes. The tidal flow around the Rattray island with the real configuration has then been compared to in situ measurements. The DG results are in very good agreement with the velocity data, considering the lack of accuracy on the tide direction, the island shape or the bathymetry.
This work finally enlightened the fact that the generation of curved meshes for high order polynomial mappings is a critical issue in this framework, since the distortion of the elements has to be kept to a minimum to avoid large quadrature errors leading to spurious waves. Major improvements to the method could eventually be achieved by improving the mapping for high order elements and by implementing p-adaptivity to reduce the computational costs for multi-scale computations in large domains.

If high-order DG computations of oceanic flows have been successfully accomplished, the full potential of the DG method is clearly unleashed when considering smoother domains with less geometrical constraints, as in atmospheric computations. In the second section was proposed an alternative method for solving vectorial conservation laws

on general curved manifolds, combining the advantages of both classical approaches in the literature. The alternative approach based on a local tangent basis allows one to keep simple expressions of the fluxes, minimizes the number of unknowns and is general and robust for any manifold. Although the expression of the vectorial fields in a local tangent basis may seem complicated, since the tangent computation also requires a metric tensor information, the computation becomes very efficient, simple and general when considering finite elements, and especially the discontinuous Galerkin method. Indeed, the tangent basis can be computed locally, based on the high-order elements mappings, without continuity requirement of the basis since using a discontinuous approach. These discontinuities have just to be taken into account into the Riemann solver ensuring the information transfer between elements. The approach has been validated with the shallow water equation on the rotating sphere, using some classical Williamson test cases and the Perturbed Rossby-Haurwitz benchmark of Smith and Dritschel. The expected convergence rate of the discretization error has been reached for several polynomial orders, using a sufficiently accurate mapping to keep the geometrical errors lower than the discretization ones. The results on more complicated benchmarks have been compared to accurate spectral solutions, revealing a good agreement. The alternative approach has eventually been tested on very irregular high-order manifolds, presenting very distorted shapes and/or large normals discontinuities. Those benchmarks enlighten the robustness and generality of the method, since the only modification between those computations is the manifolds' meshes.

Therefore, the discontinuous Galerkin method with the local tangent basis approach and the use of coarse curved meshes exhibits a great potential for computing oceanic or especially atmospheric processes on the Earth's surface. Indeed, it is possible to benefit from the high potential of high-order polynomials using a very low number of elements.

Chapter 6

Modal analysis of spectral properties on unstructured meshes

Contents

6.1 Introduction 172

6.2 Modal analysis method on unstructured grids 174

6.2.1 Basics of dispersion error analysis 174

6.2.2 General method on unstructured grids 176

6.2.3 One-dimensional numerical dispersion relation and grid cutoff 179

6.3 DG Spectral convergence for dispersive waves 181

6.3.1 f-plane approximation on structured grids . . 183

6.3.2 Structured and unstructured meshes with variable Coriolis parameter 187

6.4 Comparison with the mixed $P_1^{NC} - P_1$ approach 196

6.4.1 Discrete formulation using the mixed $P_1^{NC} - P_1$ formulation 196

6.4.2 Dispersion errors for the $P_1^{NC} - P_1$ pair . . . 199

6.4.3 Influence of the subgrid scale viscosity on the $P_1^{NC} - P_1$ pair 207

6.5 Conclusions 208

This chapter is devoted to the analysis of the spectral properties of numerical methods, and in particular in the context of ocean modeling on unstructured grids.
A new general method is proposed to analyze the dissipation and dispersion properties of any numerical method on unstructured grids. This analysis is performed on the two-dimensional dispersive shallow water waves using the DG method. A comparison is then achieved with the mixed $P_1^{NC} - P_1$ formulation, as the analysis of the impact of unstructured grids on the convergence properties.

6.1 Introduction

On the one hand, ocean processes exhibit a wide range of time and space scales. Most types of motion are unsteady and some of them lead occasionally to quasi-discontinuities. Complex bathymetry and topology have to be considered since they generate small scale features containing a significant part of the ocean energy. Those constraints justify the shift from traditional structured grids models to unstructured meshes using finite elements or finite volumes [e.g. Hanert et al., 2004, Pietrzak et al., 2005].
On the other hand, the shallow water equations exhibit three kinds of waves, namely the Poincaré, Kelvin and Rossby waves. Those waves play a crucial role in ocean dynamics. Indeed, the Rossby waves are known to intensify the western boundary currents and are responsible for the transfer of huge amounts of heat and energy while the Kelvin waves carry large amounts of energy and play a crucial role in mixing and dissipation processes [Cushman-Roisin, 1994, Pedlosky, 1979, Majda, 2003]. Minor errors in those waves dispersion relations may lead to large lack of accuracy in climate and weather predictions. It is thus mandatory for an ocean model to propagate those waves in an accurate way.

We thus wish to study the accuracy of numerical methods for propagating those waves, especially on unstructured meshes in this context of a new ocean model. It turns out that only structured grids have been considered so far in the literature to study the spectral proper-

ties of numerical methods for ocean modeling. Indeed, many studies have been carried out since three decades to investigate the dispersion properties of many numerical schemes. In particular, dispersion relations on finite differences schemes have been studied for gravity waves [Arakawa and Lamb, 1977, Batteen and Han, 1981, Randall, 1994, Beckers and Deleersnijder, 1993] and for the Rossby waves [Dukowicz, 1995, Wajsowicz, 1986]. The most common technique often used with finite differences on regular grids consists in introducing Fourier modes in the discrete representation of the spatial operators. After some algebraic manipulation, one thus obtain the analytical expression of a numerical dispersion relation corresponding to this particular scheme, grid, and Fourier mode. More recently, dispersion for finite elements techniques were investigated [Atkinson et al., 2004, Walters, 1983]. In particular, Le Roux et. al investigated many finite element pairs for solving the two-dimensional shallow water equations on structured grids [Le Roux et al., 2007, Le Roux and Pouliot, 2008]. It has been shown that pairs as the $P_2 - P_2$, MINI, $P_1^{NC} - P_1$ or $P_1 iso P_2 - P_1$ present no spurious elevation modes [Le Roux, 2005, Le Roux et al., 2007, 2005]. Some have been disregarded because of poor dispersion error properties compared to the $P_1^{NC} - P_1$ pair [Le Roux, 2005, Hanert et al., 2004, 2005], which seems to be the best compromise. This pair is now used since a few years to represent shallow water oceanic flows [e.g. Lambrechts et al., 2008b, Hanert et al., 2004, 2005, 2007, White and Deleersnijder, 2007]. However, the authors considered this analysis as *one step of the selection process* since other important factors as variable element sizes and unstructured grids should be considered in further investigations.

Furthermore, the DG method does not present any elevation modes as well and is believed to be a good candidate for ocean modeling. One of the advantages of the DG method is its superconvergence properties and its ability to propagate waves without excessive dissipation or dispersion. Hu and Atkins studied the dispersion properties of the DG method applied to a one-dimensional scalar advection equation [Hu and Atkins, 2002]. By using a computational approach it has been shown that the DG method of polynomial order p exhibits for this simple problem a superconvergence of order $2p + 3$ for the dispersion errors, and of order $2p + 2$ for the dissipation ones. Those superconvergence rates, proven by Ainsworth in [Ainsworth, 2004], exceed the orders of accuracy

obtained for the traditional continuous finite element methods [Thompson and Pinsky, 1994].

Therefore, we wish to develop a new general approach, independent of both the numerical method and the grid, in order to

- analyze the convergence properties of the DG method for the two-dimensional dispersive shallow water waves on unstructured meshes,
- compare the convergence properties of several numerical methods, especially the DG method and the mixed $P_1^{NC} - P_1$ approach, in a relatively fair way,
- analyze the impact of unstructured grids on the spectral properties of those methods.

6.2 Modal analysis method on unstructured grids

This section first contains a brief reminder of the basics of dispersion errors analysis on one-dimensional structured grids. In a second step, the general two-dimensional method on unstructured grids is developed.

6.2.1 Basics of dispersion error analysis

Let us first consider the linear advection equation of the field $u(x,t)$ in a one-dimensional domain Ω:

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0 \ , \quad x \in \Omega \ , \tag{6.1}$$

with c a constant advection velocity.
We then write the unknown field as the product of a time-dependent function $T(t)$ and a space-dependent function $X(x)$:

$$u(x,t) = T(t)X(x) \ .$$

The advection equation then reads:

$$\frac{dT}{dt}X = -cT\frac{dX}{dx} \ . \tag{6.2}$$

Following a classical Fourier analysis, the one-dimensional solution $X(x)$ can be decomposed into a sum of sines and cosines function $X_k(x)$,

$$X_k(x) = e^{ikx} \ ,$$

where k is the wave number and i denotes the imaginary number. Using this decomposition into the advection equation and dividing by X_k yields:

$$\frac{dT}{dt} = -cikT \ . \tag{6.3}$$

This clearly corresponds to an eigenvalue problem, whose eigenvalues

$$\lambda_k = -ick \ \in Im \tag{6.4}$$

are pure imaginary values. The time solution thus reads:

$$T(t) = T_0 e^{-i(ck)t}$$

providing the exact dispersion relation of the frequency ω depending on the wave number k:

$$\omega = kc \ .$$

Notice that when considering diffusion equations, the second derivative of X will provide $i^2 = -1$ instead of i, leading to pure real values of λ.

Considering now a given discretization, we clearly observe the three following error sources: the discretization of $X(x)$ will be responsible for interpolation errors, $\frac{dT}{dt}$ will yield the time scheme errors while discretizing the spatial operator $\frac{dX}{dx}$ will introduce dispersion and dissipation errors. Indeed, let us discretize the spatial operator using for instance a simple centered finite differences scheme:

$$X(x_i) = e^{ikx_i} \ ,$$

which yields the discrete space operator:

$$\frac{dX(x_i)}{dx} = \frac{e^{ikx_{i+1}} - e^{ikx_{i-1}}}{2h} = \frac{X(x_i)}{h} i \sin(kh)$$

with h the element size. The eigenvalue problem then reads:

$$\frac{dT}{dt} = -i\frac{c}{h} \sin(kh) T \ . \tag{6.5}$$

A dispersion error has thus been introduced, especially for high-frequencies compared to the element size, i.e. large values of kh, and the numerical dispersion relation reads:

$$\omega = \frac{c}{h} \sin(kh) \tag{6.6}$$

instead of ck. It can be shown easily that using upwinded finite differences with the same method yields eigenvalues with non-zero imaginary and real parts, which corresponds thus to the introduction of numerical diffusion. Finally, using downwinded differences changes the sign of the real part of the eigenvalues, which corresponds to negative diffusion and leads to an unstable scheme.

This kind of Fourier approach presents several major drawbacks. Indeed, it becomes rapidly complicated when considering two-dimensional high-order methods, finite elements and especially unstructured meshes and complex geometries. Moreover, we considered the decomposition of the spatial dependence as a sum of sines and cosines functions. Although exact for a simple linear advection equation, the modes of the solution may be quite different from sine functions for more complicated processes as for instance dispersive waves in a rotating framework. A more general method is thus required.

6.2.2 General method on unstructured grids

Following the classical approach, we discretize the equations in the space domain only. We first consider the semi-discrete formulation of the linearized shallow water equations:

$$\frac{\partial \boldsymbol{U}^h}{\partial t} = \boldsymbol{L}\boldsymbol{U}^h \tag{6.7}$$

where $\boldsymbol{L} = \boldsymbol{M}^{-1}\boldsymbol{A}$ is the square matrix of the discrete space operators, $\boldsymbol{M}$ is the mass matrix, $\boldsymbol{A}$ is the matrix corresponding to the convective terms and $\boldsymbol{U}^h$ is the discretization of the vector of unknowns, using for instance a finite element method.

Let us assume the discrete shallow water fields $\boldsymbol{U}^h$ to be the real part of:

$$\boldsymbol{U}^h(x,y,t) = \boldsymbol{X}^h(x,y)\exp(i\omega t) \ . \tag{6.8}$$

Incorporating (6.8) into (6.7) leads to the following eigenvalue problem:

$$[\boldsymbol{L} - \lambda \boldsymbol{I}]\boldsymbol{X}^h = \boldsymbol{0} \ , \tag{6.9}$$

which can be seen as a generalization of the finite differences eigenvalue problem (6.5) for a system of discrete equations, except that no assumption has been required for the shapes of the modes in the space domain.

For each eigenvalue λ_j ($j = 1 \dots \mathcal{N}_d$), the corresponding discrete frequency ω_j and the associated numerical dissipation μ_j are found as the imaginary and real parts of the eigenvalue respectively:

$$\lambda_j = \mu_j + i\omega_j \quad .$$

For spatial operators computed with a Riemann-like solver, introducing some dissipation in the numerical scheme, the eigenvalue spectrum exhibits a real part different from zero as depicted in Figure (6.4). On the other hand, numerical schemes without upwind or Riemann solver present pure imaginary eigenvalues as depicted in Figure (6.18). The dissipation error of mode j is then given by the absolute value of μ_j.

We still need to compute the associated wave numbers in the x and y directions to derive a numerical dispersion relation. This information is contained in the eigenvectors. To each eigenvector is associated an elevation mode $\eta_j(x,y)$ and a velocity mode $(u_j(x,y), v_j(x,y))$. We consider for instance the elevation mode and perform a 2D Fourier Transform to obtain the dominating wave number $\boldsymbol{k}_j = (k_{x,j}, k_{y,j})$ corresponding to this mode. This Fourier analysis is only applied to the resolved modes of the discrete operator. For discontinuous finite elements as the DG method or non-conformal P_1^{NC} elements, this distinction between resolved and unresolved numerical modes is done by means of the *a posteriori* error estimation procedure based on the jumps, as described in Chapter (4). Typically, a mode is considered to be resolved if the relative spatial error is below a threshold, e.g. 5%. The Fourier spectrum of the elevation mode η_j is given by:

$$A_{pq} = \int_0^L \int_0^L \eta_j(x,y) \exp\left(2i\pi(\frac{px}{L} + \frac{qy}{L})\right) dxdy \ .$$

The L^2 norm of the field can then be obtained by taking advantage of the Parseval's theorem:

$$\| \eta_j \|^2 = \int_0^L \int_0^L \eta_j^2(x,y)dxdy = L^2 \sum_p \sum_q |A_{pq}|^2 \ .$$

This analysis gives us, as result, a single numerical wave number vector $\boldsymbol{k}_j$. Because of the periodicity of the domain, only even wavenumbers $k_x = 2p\pi/L$ have to be considered and we can write $k_{x,j} = 2\bar{p}\pi/L$, $k_{y,j} = \bar{q}\pi/L$ where $\bar{p}$ and $\bar{q}$ are the values corresponding to the dominant wave number. The spatial error of this mode, denoted by e_j, can then be identified as the remainder in the Fourier spectrum, i.e. the L^2 norm of the difference between the exact mode and the numerical mode:

$$e_j^2 = L^2 \sum_{p \neq \bar{p}} \sum_{q \neq \bar{q}} |A_{pq}|^2 \ .$$

We thus obtained a discrete dispersion relation $\omega_j(\boldsymbol{k_j})$. A discrete dispersion analysis is then performed by comparing this relation to the reference or exact dispersion relations $\omega(\boldsymbol{k_j})$ for the Kelvin waves (2.34) and for the Poincaré and Rossby waves following the Sturm-Liouville approach (2.43). The absolute dispersion error of mode j is given by:

$$\kappa_j = |\omega_j - \omega(\boldsymbol{k_j})| \quad .$$

It must be noticed that the Fourier transform only provides integer wave numbers. For the shallow water equations, those Fourier wave numbers are exact in the x direction, i.e. the longitudinal dimension, only if periodic boundary conditions are used, ensuring a sine dependence in x. But there is no guarantee to obtain such integer wave numbers in the y direction since the Coriolis parameter varies with the latitude and generates dispersive waves. With a non zero Coriolis force and dispersive waves, the y component of the wave number $\boldsymbol{k}$ has to be replaced by the reference exact wave number. The reference dispersion relations $\omega(\boldsymbol{k_j})$ for the dispersive shallow water waves have no analytical expression. They are computed numerically by solving the one-dimensional Sturm-Liouville relationship (2.42).

6.2.3 One-dimensional numerical dispersion relation and grid cutoff

Lets us compare the high-order DG method to classical finite differences using the one-dimensional advection equation (6.1) with an advection velocity $c = 1\ ms^{-1}$. It has been shown that the numerical dispersion relation using finite differences corresponds to the sine function (6.6), instead of the linear relation $\omega = ck$. This analytical analysis becomes quite complicated with the high-order DG method, even for 1D equations on structured grids. We thus apply the alternative numerical approach and analyze the discrete spatial operators matrix.

We consider 2th, 4th and 6th order finite differences on a 200 elements grid, and the DG method using $p = 1, 3$ and 7 with 100, 50 and 25 elements respectively. We also consider the advection equation in a periodic domain. Therefore, all computations present an even number of degrees of freedom of 200. We observe in Figure (6.1) the first nine resolved modes of the discrete operators corresponding to sine functions. Only even wave numbers are observed, ranging in $[0, 16]$. This restriction is clearly due to the periodicity of the domain.

The modal analysis provides the numerical dispersion and dissipation relations depicted in Figure (6.2). The relations are plotted in a non-dimensional version, using the non-dimensional frequency $\omega h_{\text{ref}}/\pi$ and wave number kh_{ref}. For each computation, the value h_{ref} is chosen accordingly to the smallest wavelength that can be represented on the grid. On a same given grid, the value h_{ref} is therefore eight times smaller for DG $p7$ than for finite differences. The theoretical sine function is obtained as expected for the 2th order finite differences. We observe that the DG method using a polynomial order $p = 1$ still matches approximately the analytical straight line for a non-dimensional wave number of 0.5 with an error of approximately 10%, while the finite differences FD2 already presents a 10% error for $k/200 = 0.25$. Increasing the polynomial order reduces the error on resolved modes, i.e. with a small wave number lower than 0.5, while it increases the dissipation on the unresolved wavenumbers. The DG method using the simple $p1$ shape functions presents better dispersion properties than high-order finite differences for the low wavenumbers, and introduces dissipation for high wavenum-

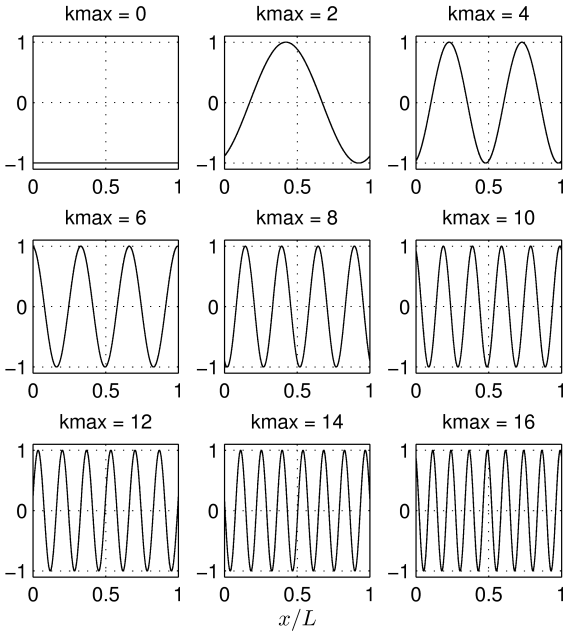


Figure 6.1: First resolved modes obtained as eigenvectors of the discrete operators. Periodicity restrict the solution to even wave numbers.

bers, while these unresolved modes are conserved in the system using finite differences.

The following analysis on 2D unstructured grids considers this zone of well resolved modes, and more precisely the convergence rates of the dispersion errors in this region.

6.3 Spectral convergence of the DG method for dispersive waves on 2D unstructured meshes

We need to build the global matrix $\boldsymbol{L}$ of the discrete space operators, and to compare the results to the reference dispersion relations. Therefore, we restrict ourselves to the shallow water equations in their linearized form (2.24), (2.25) and (2.26) on the semi-periodic domain used in Section (2.3) and depicted in Figure (2.3).

For all computations, the geometrical and physical parameters are $L = 10^6\ m$, $g = 10\ ms^{-2}$, $H_0 = 10^3\ m$, $\beta = 3\ 10^{-10}\ m^{-1}s^{-1}$ and $f_0 = 3\ 10^{-4}\ s^{-1}$. The shallow water equations are solved in a classical non dimensional version [Cushman-Roisin, 1994] in order to obtain a better numerical accuracy. This set of parameters presents an increased Coriolis parameter with an increased β-effect to visually differentiate the shape of Rossby modes in the y direction from sine functions. Indeed, a zero Coriolis parameter leads to gravity waves composed of sine and cosine modes, while a larger Coriolis parameter increases the dispersive nature of the Rossby and Poincaré waves. The only purpose of this arbitrary set of parameters is to visually identify this dispersive behavior and the difference with sine functions. For those values, the typical non dimensional numbers are given by:

$$Ro = \frac{\sqrt{gH_0}}{Lf_0} = \frac{1}{3}, \qquad \frac{\beta L}{f_0} = 1\ ,$$

with Ro denoting the Rossby number.

We apply here the modal analysis to the 2D linear shallow water equations with both constant and variable Coriolis parameter, using the DG method and an exact Riemann solver for the flux computation. The

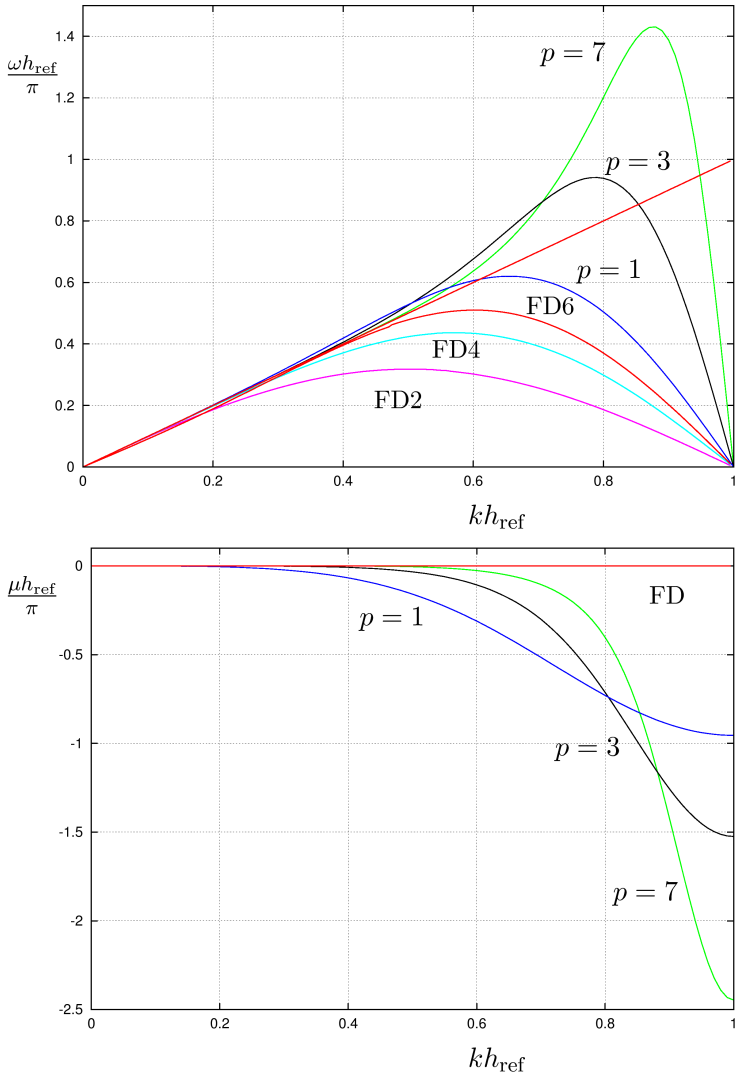


Figure 6.2: Numerical dispersion relation (top) and dissipation (bottom) obtained for the finite differences of 2th, 4th and 6th orders and for the high-order DG method. Using higher polynomial orders enlarges the range of resolved modes and increases the dissipation on the unresolved ones. The straight red line represents the analytical dispersion relation $\omega = ck$.

resulting eigenvalue spectrum depicted in Figure (6.4) presents thus non-zero real values. This numerical diffusion essentially corresponds to the non-resolved modes, i.e. the high-wavenumber modes. The convergence rates are analyzed for several polynomial orders for the dissipation and dispersion errors, and compared between the structured and unstructured meshes depicted in Figure (6.3).

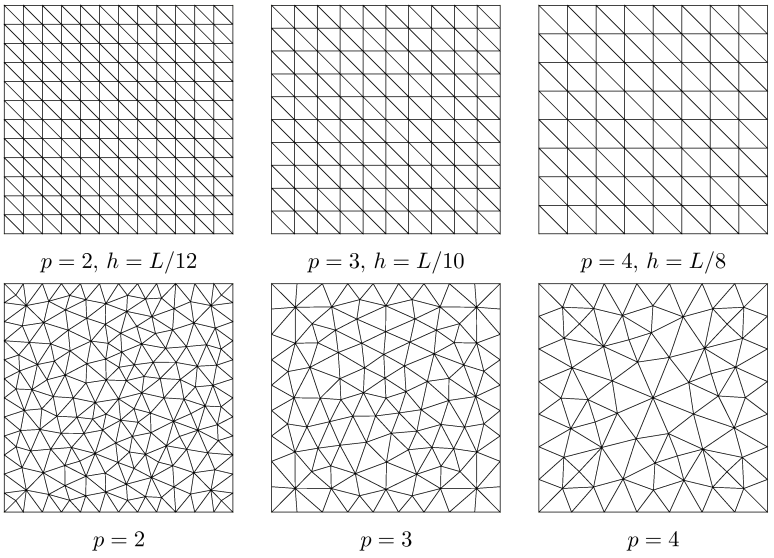


Figure 6.3: Definition of the structured and unstructured meshes used for computing discrete dispersion relations with the DG method. The size fields of the unstructured grids have been computed to reach approximately the same number of degrees of freedom than for the structured grids.

6.3.1 f-plane approximation on structured grids

Let us consider in a first time a constant Coriolis parameter in order to compare the analytical and numerical dispersion relations and to observe steady geostrophic modes. We use the structured meshes depicted in Figure (6.3). The mesh of 128 elements using for instance fourth order polynomial function, leads to a total number of degrees of freedom of $128 \times 3 \times 15 = 5760$. With the help of MATLAB, the computation of

the whole spectrum of $\boldsymbol{L}$, involving 5760 eigenvalues and eigenfunctions, requires approximately 30 minutes on a desktop computer and about 500 MB of RAM. In order to observe the convergent behavior, the same computation is also performed with polynomial order $p = 3$ and $p = 2$.

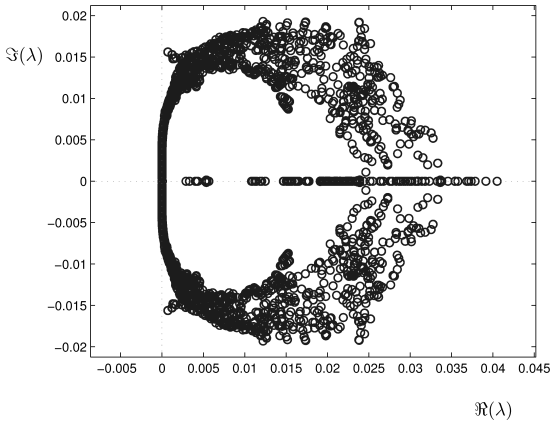


Figure 6.4: Spectrum of eigenvalues λ $[s^{-1}]$ of the DG discretization $\boldsymbol{L}$ of the spatial operators. The wave frequency ω is the imaginary part and is used to compute the dispersion error while the real part is the dissipation error of the numerical scheme.

The reference dispersion curves and the numerical dispersion relations are presented in Figure (6.5). Only the positive frequencies are considered. In Figure (6.6) are represented some Poincaré (upper row) and Kelvin (lower row) modes for a constant Coriolis coefficient. The color levels represent the elevation η while super-imposed vectors represent the velocities. We obtain as expected Poincaré modes composed of sines and cosines functions, while the Kelvin modes are a combination of sines and cosines in the x-direction and exhibit the expected exponential decay of equation (2.35) in the y-direction. Finally, geostrophic modes are depicted in Figure (6.7). They correspond to the null eigenvalues of the discrete operator. Therefore, any combination of two modes in geostrophic balance is in geostrophic balance and correspond to a null

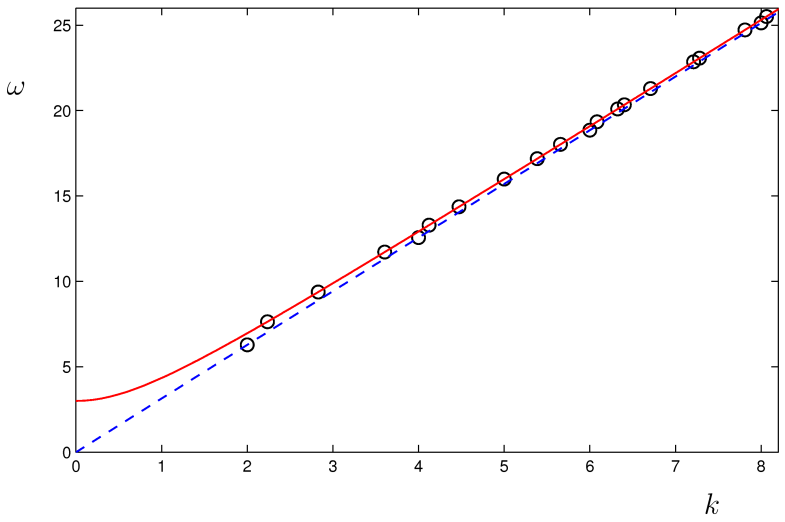


Figure 6.5: Poincaré and Kelvin numerical and analytical dispersion relations for a constant Coriolis factor, with the polynomial order $p = 4$ and the parameters $L = 10^6\ m$, $g = 10\ ms^{-2}$, $H_0 = 10^3\ m$ and $f_0 = 3\ 10^{-4}\ s^{-1}$. Analytical Poincaré and Kelvin relations correspond to the continuous and dashed line respectively while the dots are the numerical DG results.

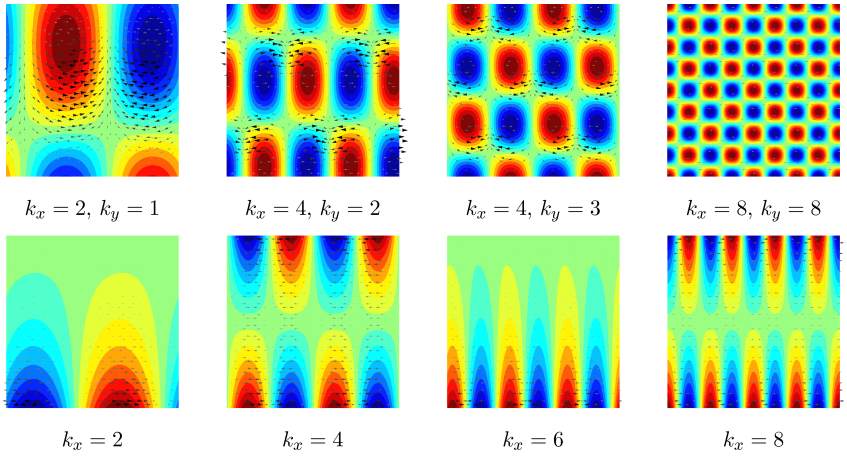


Figure 6.6: Some Poincare (upper row) and Kelvin (lower row) modes for a constant Coriolis coefficient and a Rossby radius of 3 10^{-6}.

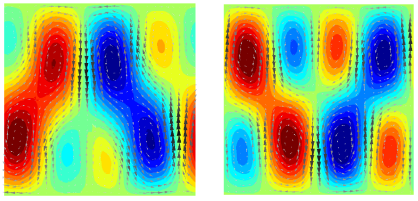


Figure 6.7: Modes in geostrophic balance.

eigenvalue of the operator. Those modes are not separated by the numerical process so that their Fourier spectrum may contain a variety of normal modes. The velocity vectors are aligned with the lines of iso-elevation and the divergence of the velocity is zero, which is clearly not the case for Poincaré waves.
Dispersion errors for the Poincaré and Kelvin waves are shown in the left part of Figure (6.8) while we see dissipation errors on the right part. Hu and Atkins showed that one of the DG method's properties is the superconvergent behavior of the dispersion and dissipation errors [Hu and Atkins, 2002]. In particular, it was demonstrated [Ainsworth, 2004] for the 1D transport equation that the DG method superconverges at a rate of $2p+3$ for every polynomial order p, if a Riemann solver is used for the flux computation. This $2p+3$ rate is reached for centered schemes only for even polynomial orders, while a $2p+1$ rate is obtained for odd orders. Same kind of results were demonstrated for the dissipation introduced by a Riemann solver, which superconverges at a rate of $2p+2$.

A least square fit of the dispersion error curves in the left part of Figure (6.8) shows that the error is converging at rate $\mathcal{O}(kh)^7$, $\mathcal{O}(kh)^9$ and $\mathcal{O}(kh)^{11}$ for $p = 2, 3$ and 4 respectively, i.e. a super convergence of order $2p+3$. Using the same fit, a super convergence of order $2p+2$ is reached in the right part of Figure (6.8) for the dissipation errors. We observe in these figures that the use of upwind fluxes introduces some numerical dissipation, but the resulting error is very low for resolved modes and superconverges to zero.

The theoretical rates of convergence for DG with a Riemann solver and a 1D transport equation are thus clearly observed as well for the 2D shallow water equations with a constant Coriolis coefficient f.

6.3.2 Structured and unstructured meshes with variable Coriolis parameter

Let us now perform the same computation with a variable Coriolis parameter, i.e. with the β-plane approximation and $\beta = 3\ 10^{-10}\ m^{-1}s^{-1}$, on both structured and unstructured meshes. The analytical dispersion relation for the non-dispersive Kelvin waves (2.34) is still valid, but the

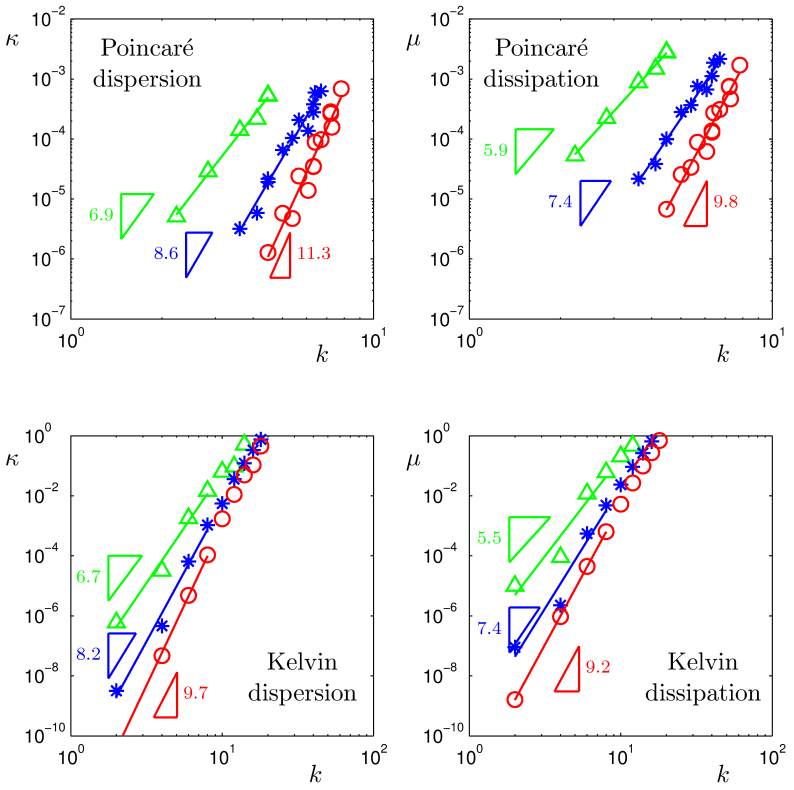


Figure 6.8: Convergence of the dispersion κ and dissipation μ errors for the Poincaré (top row) and Kelvin (bottom row) waves with a constant Coriolis coefficient on structured grids. The dots represent the numerical results while the lines are their $\mathcal{L}^2$ approximation to obtain the convergence rate. The second, third and fourth order elements correspond to the green, blue and red lines respectively. Both the dispersion and dissipation errors exhibit the superconvergence of order $2p+3$ and $2p+2$ respectively.

j	1	2	3	4	5	6
$k_{y,j}$	3.1331	6.3411	9.4651	12.5966	15.7321	18.8696
$j\pi/L$	3.1416	6.2832	9.4248	12.5664	15.7080	18.8496

Table 6.1: Wavenumbers $k_{y,j}$ of the Sturm-Liouville problem (2.42) compared to their approximation $j\pi/L$.

Sturm-Liouville approach has to be used in order to obtain a reference dispersion relation for the Rossby and Poincaré waves. In Table (6.1) are given some of the first exact eigenvalues $k_{y,j}$ computed with the Sturm-Liouville approach. These are compared with the approximated wavenumbers $j\pi/L$. The frequencies ω are then obtained by using the relation (2.43).

The dispersion relations for Rossby and Poincaré are shown in Figure (6.9). Only the positive part of the frequencies is considered, even if the two Poincaré waves lose their exact symmetry because of the β-effect. We see on the close up view the lack of accuracy of the approximated analytical expression with those parameters. We may also point out that the single dispersion curve of the constant Coriolis case has been replaced by a set of dispersion curves, one for every k_y, since the wavenumbers are not the same in the x and y directions. This means that modes with very similar wavenumbers k may exhibit different frequencies.

With those parameters, we clearly observe in Figure (6.10) that the resulting Poincaré and Rossby modes are affected by the variability of the Coriolis parameter, while the Kelvin modes remain unchanged. As expected, the sines and cosines dependence of those modes in the y-direction with a constant Coriolis coefficient has been replaced, leading to a general y-dependence $Y(y)$ depending on the value of β. Note that both Rossby and Poincaré modes exhibit this same y-dependence for a same wave number k_y. We observe in Figure (6.11) the functions $Y(y)$ for $\beta = 3\ 10^{-10}\ m^{-1}s^{-1}$, computed with the one dimensional Sturm-Liouville problem (2.42) in continuous lines while the dots represents the DG modal analysis.

Figure (6.12) presents the Poincaré Kelvin and Rossby dispersion and dissipation errors. The same theoretical convergence rates as for the $\beta = 0$ case are obtained for the Poincaré and Kelvin waves, for

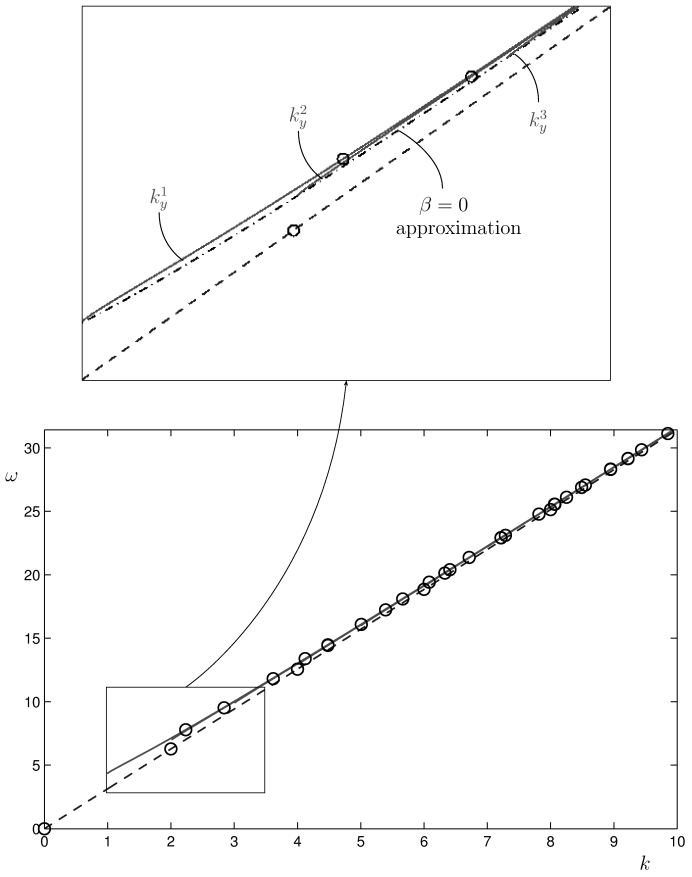


Figure 6.9: Reference and numerical dispersion relations. The continuous lines are the exact Poincaré dispersion relations for different values of k_y, the dashed line is the analytical Kelvin dispersion relation and the dots are the DG results. We see on the close up view the different dispersion curves for the three first wavenumbers k_y^n, $n = 1, 2, 3$, and the lack of accuracy of the constant Coriolis parameter approximation with those parameters, compared to the Sturm-Liouville approach.

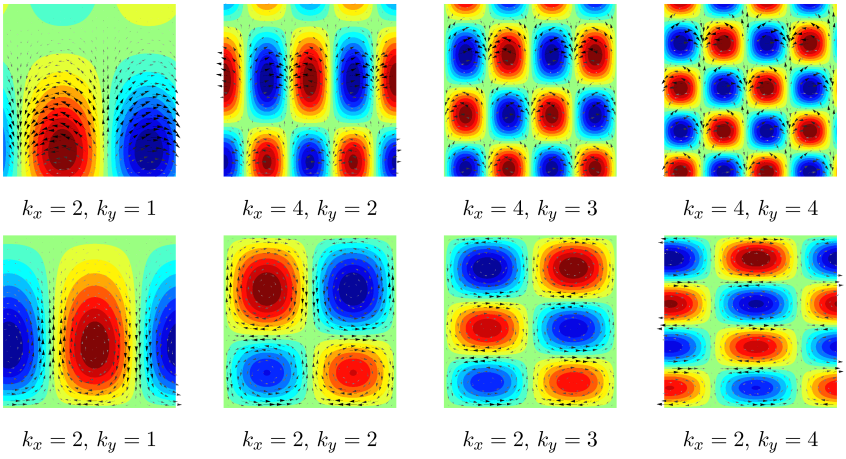


Figure 6.10: Shape of some Poincaré (upper row) and Rossby (lower row) modes for a variable Coriolis coefficient with $\beta = 3\ 10^{-10}\ m^{-1}s^{-1}$ and $f_0 = 3\ 10^{-4}\ s^{-1}$. Note that the shape of the Poincaré modes in the y-directions is now exactly the same than the shape of the Rossby modes.

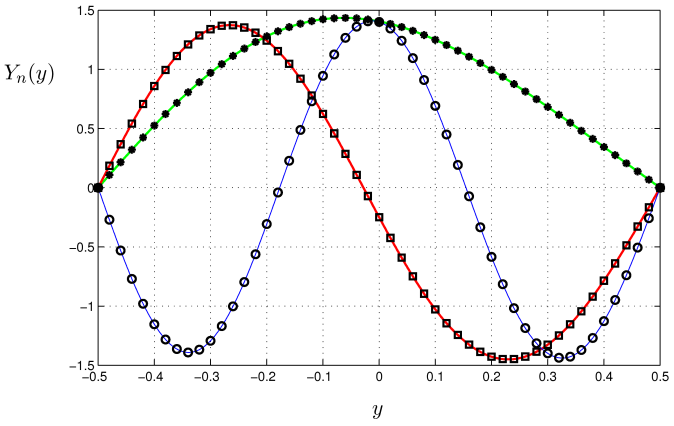


Figure 6.11: Shape of the first three eigenmodes in the y direction $Y(y)$ computed with the one dimensional Sturm-Liouville problem (2.42) (continuous lines) and with the DG modal analysis (dots) for $\beta = 3\ 10^{-10}\ m^{-1}s^{-1}$.

both dispersion and dissipation errors. The reference numerical dispersion relation from the Sturm-Liouville approach was computed with a 6000 points 1D continuous FEM. This solution is accurate enough, compared to the DG dispersion relation, until an absolute error of about 10^{-7}. Below this threshold, the numerical accuracy of the reference solution becomes insufficient as for the dispersion rate computation on the Rossby waves with $p = 4$. For the other Rossby rates, the dispersion and dissipation errors do not seem to fit as well the theoretical predictions, an order 2 seems to be missing. This result can be explained as follows: in the analytical developments [Ainsworth, 2004], Ainsworth has considered a 1D transport equation with constant coefficients, leading to non dispersive waves. The first theoretical convergence rates obtained in this study were $2p + 2$ and $2p + 1$ for the dispersion and dissipation respectively, in terms of relative errors. Those rates are then extended to $2p + 3$ and $2p + 2$ for the absolute errors. This extension to absolute values is not valid for the Rossby waves, since those waves experience a strong dispersive behavior. The correspondence between relative and absolute error is then no more relevant. We observe in Figure (6.14) that the theoretical convergence rates are reached in terms of relative dispersion and dissipation errors, even for the dispersive Rossby waves. Notice that the relative dissipation error was computed by dividing the absolute error by the norm of the corresponding eigenvalue, since the analytical dissipation in the system is zero. The theoretical convergence rate is then reached for the three shallow water waves, provided that the relative error is considered instead of the absolute one in the case of strongly dispersive waves.

Finally, the same modal analysis provides in Figure (6.13) the Poincaré, Kelvin and Rossby dispersion and dissipation errors for unstructured meshes. In those convergence plots, unstructured meshes do not affect the DG method superconvergence rates obtained on structured grids. This general grid-independent modal analysis is thus a promising tool to compare different numerical schemes in terms of dispersion and dissipation.

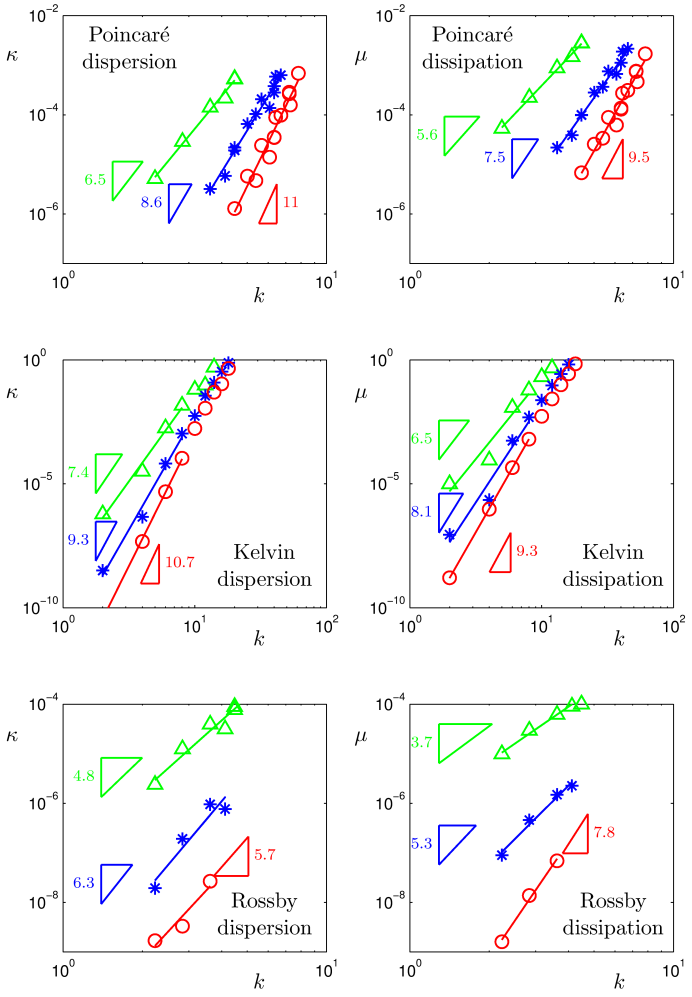


Figure 6.12: Convergence of the absolute dispersion κ and dissipation μ errors with a variable Coriolis coefficient on structured grids for $p = 2, 3$ and 4 (green, blue and red lines respectively). Superconvergence rates of order $2p+3$ and $2p+2$ are reached for the Poincaré and Kelvin waves while the dispersive Rossby waves require relative errors.

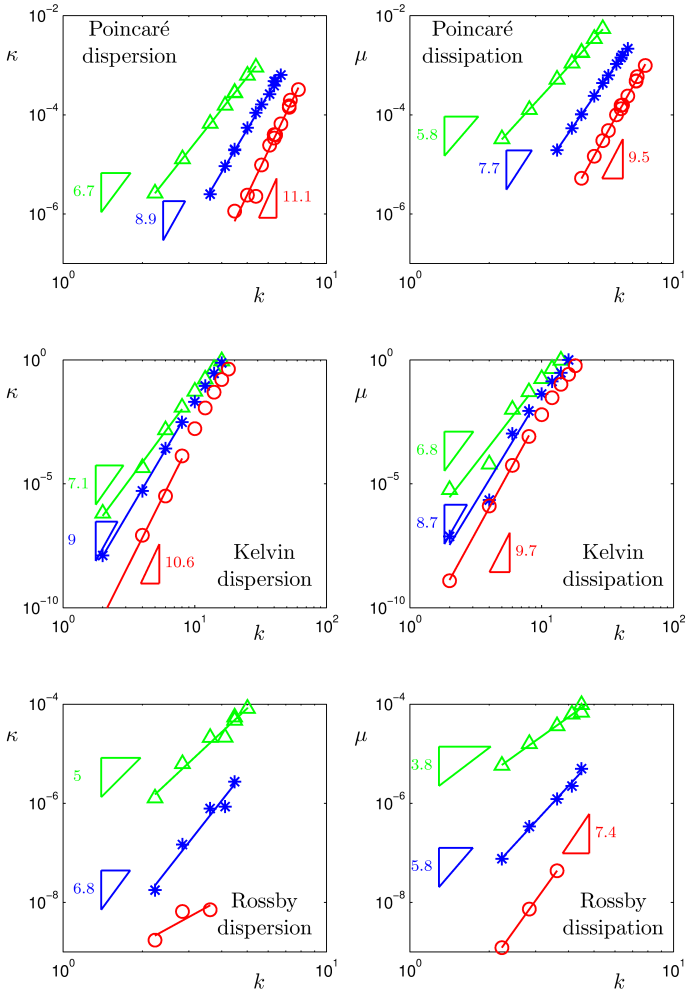


Figure 6.13: Convergence of the absolute dispersion κ and dissipation μ errors with a variable Coriolis coefficient on unstructured meshes for $p = 2, 3$ and 4 (green, blue and red lines respectively). Superconvergence rates of order $2p+3$ and $2p+2$ are reached for the Poincaré and Kelvin waves while the dispersive Rossby waves require relative errors.

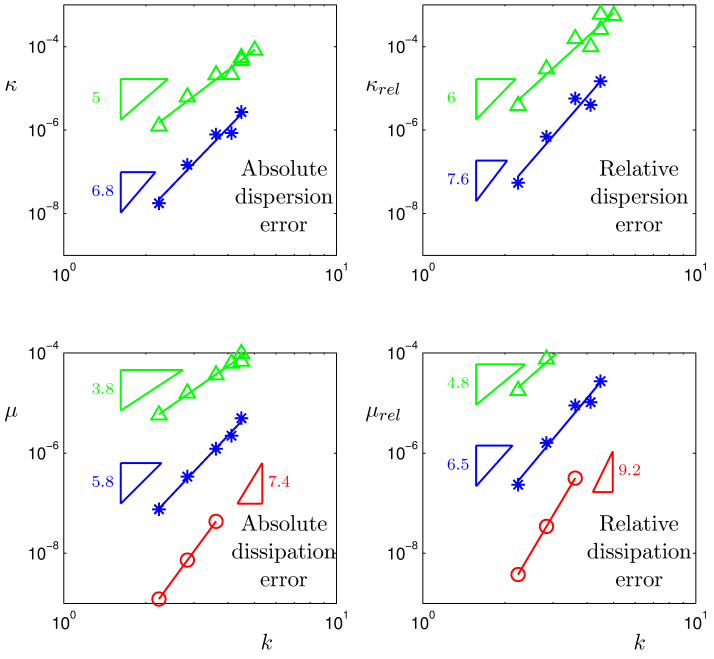


Figure 6.14: Comparison between convergence rates for the absolute (left) and the relative (right) dispersion and dissipation errors for the Rossby waves on unstructured grids. The theoretical rates of convergence of $2p+2$ and $2p+1$ for the relative dispersion and dissipation errors respectively are reached for the Rossby wave. The rates of $2p+3$ and $2p+2$ for the absolute errors are not reached since the correspondence between absolute and relative errors are based on the assumption of a non dispersive wave and the Rossby waves precisely experience a strong dispersive behavior.

6.4 Comparison of spectral convergence with the mixed $P_1^{NC} - P_1$ approach

Dispersion properties of many finite elements techniques have been investigated lately [Atkinson et al., 2004, Walters, 1983]. In particular, Le Roux et. al investigated many finite element pairs for solving the two-dimensional shallow water equations on structured grids [Le Roux et al., 2007, Le Roux and Pouliot, 2008]. It has been shown that pairs as the $P_2 - P_2$, MINI, $P_1^{NC} - P_1$ or $P_1 iso P_2 - P_1$ present no spurious elevation modes [Le Roux, 2005, Le Roux et al., 2007, 2005]. Some have been disregarded because of poor dispersion error properties compared to the $P_1^{NC} - P_1$ pair [Le Roux, 2005, Hanert et al., 2004, 2005], which seems to be the best compromise. This pair is now used since a few years to represent shallow water oceanic flows [e.g. Lambrechts et al., 2008b, Hanert et al., 2004, 2005, 2007, White and Deleersnijder, 2007]. However, the authors considered these analyses as *one step of the selection process* since other important factors as variable element sizes and unstructured grids should be considered in further investigations.

The new modal analysis developed here provides a way to investigate the dispersion properties of those elements on unstructured grids. In particular, we investigate here the use of the $P_1^{NC} - P_1$ pair and compare the results with the classical standard $P_1 - P_1$ pair. We also analyze the impact of unstructured meshes on the convergence properties.

6.4.1 Discrete formulation using the mixed $P_1^{NC} - P_1$ formulation

Let us first discretize the linear shallow water equations for the mixed formulation with the $P_1^{NC} - P_1$ pair. A weak formulation may be obtained by multiplying the momentum and continuity equations by test functions $\widehat{\boldsymbol{v}} = (\widehat{u}, \widehat{v})$ and $\widehat{\eta}$ respectively, by integrating on the domain Ω and taking advantage of the divergence theorem for the continuity equation. More informations on the standard Galerkin procedure may be found in [Ciarlet, 1978, Hughes et al., 2000]. A possible weak formulation is:

Find $\boldsymbol{u} = (u, v)$ and η such that:

$$\begin{aligned}
\int_\Omega \left(\frac{\partial u}{\partial t} + g\frac{\partial \eta}{\partial x} - fv \right) \widehat{u}\, d\Omega &= 0, \\
\int_\Omega \left(\frac{\partial v}{\partial t} + g\frac{\partial \eta}{\partial y} + fu \right) \widehat{v}\, d\Omega &= 0, \\
\int_\Omega \frac{\partial \eta}{\partial t}\widehat{\eta}\, d\Omega - \int_\Omega H_0 \left(u\frac{\partial \widehat{\eta}}{\partial x} + v\frac{\partial \widehat{\eta}}{\partial y} \right) d\Omega + \int_{\partial\Omega} H_0 u_n \widehat{\eta}\, d\Gamma &= 0,
\end{aligned} \tag{6.10}$$

where u_n denotes the normal velocity across the edge $\partial\Omega$. Setting this integration on boundaries to zero allows to weakly impose the impermeability conditions on the solid walls [e.g. Hanert et al., 2006]. Moreover, such a weak formulation allows us to consider non-continuous velocity fields.

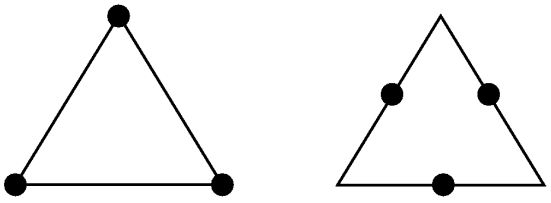


Figure 6.15: P_1 linear conforming element (left) representing the water elevation and the P_1^{NC} linear non-conforming element (right) representing the velocity field.

We consider the use of the $P_1^{NC} - P_1$ pair depicted in Figure (6.15). The pressure, i.e. the water elevation, is represented as a continuous piecewise linear field and the velocity is a discontinuous piecewise linear field except at the mid-edges. The $P_1^{NC} - P_1$ element pair has been used for the first time to build a noise-free finite element two-layer shallow water model [Hua and Thomasset, 1984]. This pair became popular because it clearly exhibits the better numerical properties in terms of dispersion error and computational cost among the set of element pairs which do not suffer from elevation modes (i.e. the $P_2 - P_2$, MINI, $P_1^{NC} - P_1$ or $P_1 iso P_2 - P_1$ pairs) [Le Roux et al., 2007, Le Roux and Pouliot, 2008]. Indeed, the orthogonality of the non-conforming shape functions permits to reduce the computational costs. The discontinuities in the

non-conforming elements represent a compromise between discontinuous and continuous fields and make the $P_1^{NC} - P_1$ a better suited element than continuous ones to represent advection dominated processes. Notice that some element pairs do not suffer from spurious elevation modes only in some particular cases, as the classical $P_1 - P_1$ with the use of weakly imposed no normal flow boundary conditions on smooth domains [Hanert et al., 2006].

We then proceed to the spatial discretization of the unknown fields. The computational domain Ω is divided into a set of N non overlapping elements Ω_e with a total of N_e edges and N_v vertices, leading to $N_d = N_v + 2N_e$ degrees of freedom. We discretize the elevation and velocity fields by assuming a linear combination of nodal values and linear shape functions on each element:

$$u = \sum_{n=1}^{3} U_n \psi_n, \quad v = \sum_{n=1}^{3} V_n \psi_n, \quad \eta = \sum_{n=1}^{3} H_n \phi_n,$$

with (U_n, V_n) and H_n the nodal values and ψ_n and ϕ_n the shape functions for the velocity and elevation fields respectively.

Introducing this discretization into the weak formulation (6.10) with the weak imposition of the no normal flow boundary conditions , we obtain the following finite element system:

$$\boldsymbol{M}\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}\boldsymbol{U} = \boldsymbol{0} \tag{6.11}$$

where $\boldsymbol{U}$ is the vector of nodal values

$$\boldsymbol{U} = [U_j \; V_j \; H_l]^T,$$

$\boldsymbol{M}$ is the global $N_d \times N_d$ mass matrix and $\boldsymbol{A}$ contains the gradient, divergence and Coriolis discrete operators:

$$\boldsymbol{A} = \left[\begin{array}{c|c|c} 0 & -\int_\Omega f\psi_i\psi_j \; d\Omega & \int_\Omega g\psi_i \frac{\partial \phi_l}{\partial x} \; d\Omega \\ \hline \int_\Omega f\psi_i\psi_j \; d\Omega & 0 & \int_\Omega g\psi_i \frac{\partial \phi_l}{\partial y} \; d\Omega \\ \hline -\int_\Omega H_0 \frac{\partial \phi_k}{\partial x}\psi_j \; d\Omega & -\int_\Omega H_0 \frac{\partial \phi_k}{\partial y}\psi_j \; d\Omega & 0 \end{array}\right] \tag{6.12}$$

with the subscripts $i, j = 1 \ldots N_e$ and $k, l = 1 \ldots N_v$.

The specific pattern of this matrix $\boldsymbol{A}$ allows the appearance of both velocity and elevation modes. Steady spurious modes are solutions of

the system which are not physically relevant. They do not propagate in time, but are trapped into the mesh and are thus solution of the steady state formulation:

$$\boldsymbol{AU} = \mathbf{0} \ . \tag{6.13}$$

Notice that specific spurious modes may also appear in some time-dependent problems, but those will not be considered in this paper. An analysis of such modes may be found in [Platzman, 1981].

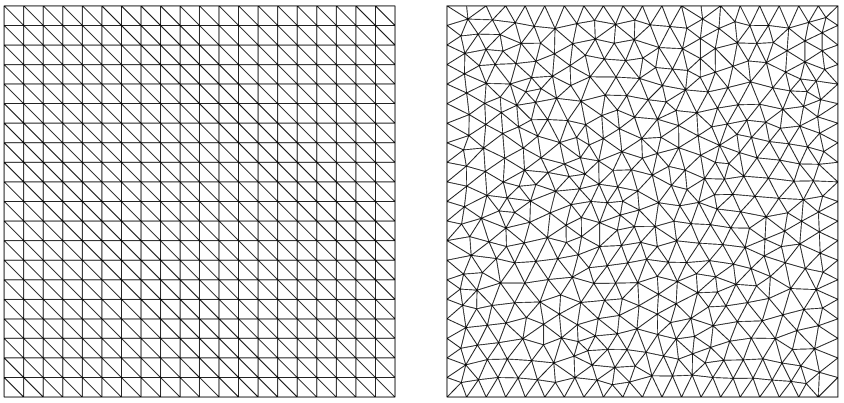
To each eigenvector of the matrix $\boldsymbol{A}$ corresponds an elevation mode and a velocity mode.
A spurious pressure (or elevation) mode can be defined as a mode corresponding to a stationary eigenmode of the discrete system which has zero velocity components and a non constant pressure belonging to the kernel of the discrete gradient operator. A spurious velocity mode can be defined as a mode corresponding to a zero elevation with the velocity components belonging to the intersection of the kernel of the divergence operator and the kernel of the Coriolis operators. Such spurious modes have been found for the C grid and the $RT_0 - P_0$, $RT_0 - P_1$, $BDM_1 - P_0$ and $BDM_1 - P_1$ finite element schemes [Rostand and Le Roux, 2007]. Notice that RT_0 denotes here the velocity discretization used in the Raviart-Thomas finite element approximation, while P_0 or P_1 denotes the elevation discretization.

6.4.2 Dispersion errors for the $P_1^{NC} - P_1$ pair

The following computations are performed with the same equations, same domain and same set of parameters than for the DG computation. The structured and unstructured meshes used in our computations are depicted in Figure (6.16) and Figure (6.17) for the $P_1^{NC} - P_1$ and the $P_1 - P_1$ formulations respectively.

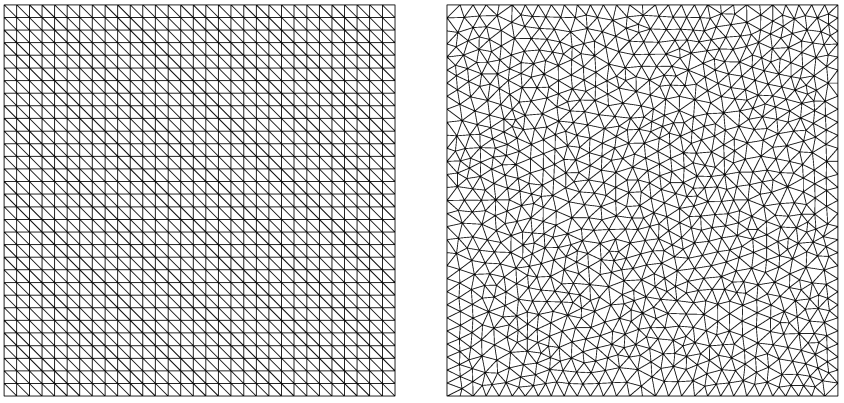
Dispersion errors for the $P_1^{NC} - P_1$ pair

The computation of the eigenvalue problem yields the set of eigenvalues depicted in Figure (6.18). As expected with the linear shallow water equations and the $P_1^{NC} - P_1$ pair, the scheme does not introduce any numerical dissipation, leading to fully imaginary eigenvalues. The



2921 degrees of freedom 3152 degrees of freedom

Figure 6.16: Structured and unstructured meshes used for the computations on the semi-periodic domain with 800 and 866 elements respectively with the $P_1^{NC} - P_1$ finite element pair.



2700 degrees of freedom 2703 degrees of freedom

Figure 6.17: Structured and unstructured meshes used for the computations on the semi-periodic domain with 1922 and 1920 elements respectively with the $P_1 - P_1$ finite element pair.

reference dispersion curves and the numerical dispersion relations are presented in Figure (6.19) where we only considered the positive frequencies.

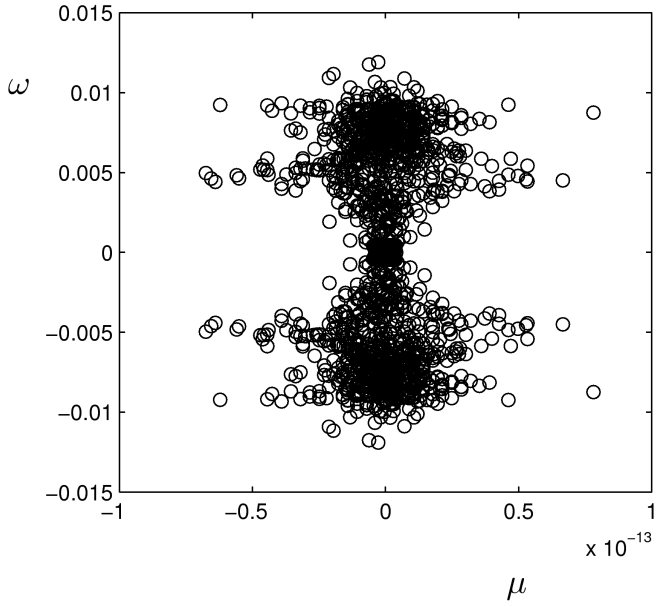


Figure 6.18: Spectrum of eigenvalues λ corresponding to the $P_1^{NC} - P_1$ pair. The scheme does not introduce any dissipation to filter the unresolved modes, leading to fully imaginary eigenvalues with the dissipation $\mu = Im(\lambda) \approx 10^{-14}$ and the frequency $\omega = Re(\lambda)$.

Relative dispersion errors for the Poincaré, Rossby and Kelvin waves are shown in Figure (6.20) for the $P_1^{NC} - P_1$ pair (left column) and the $P_1 - P_1$ pair (right column). The convergence rates are computed as the slope of the continuous lines, i.e. slopes of the linear approximations of the first resolved modes in the sense of the $\mathcal{L}_2$ norm. Since the error is proportional to the product of the element size h and the wave number k, $\mathcal{O}(kh)$, the convergence study is performed on only one grid,

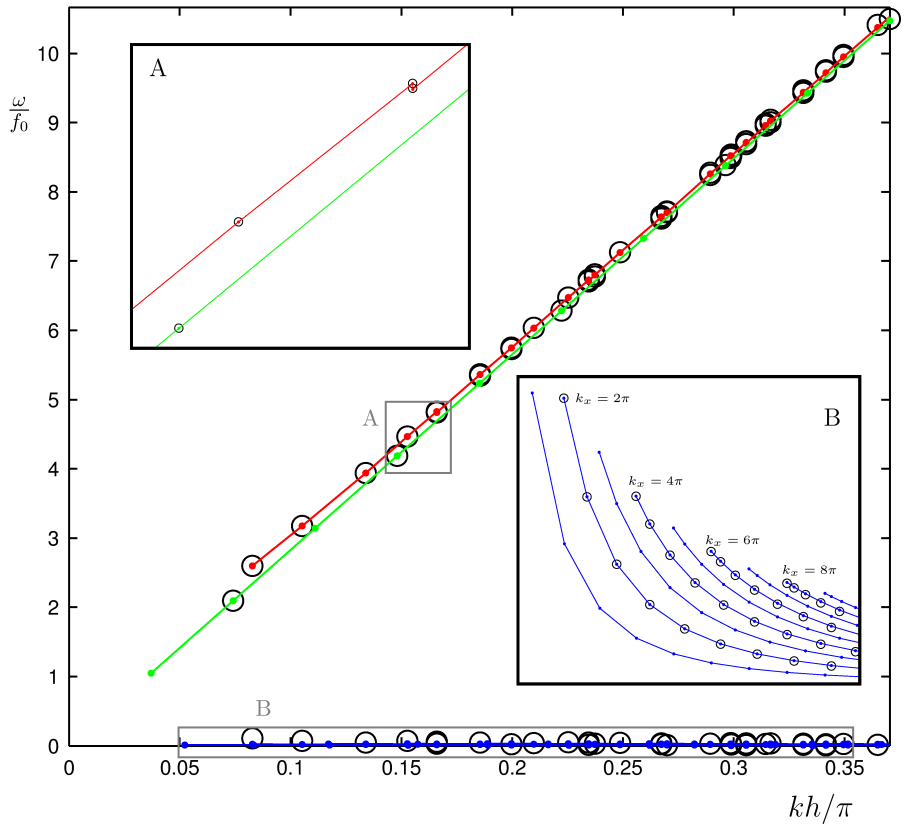


Figure 6.19: Discrete dispersion relation of the Poincaré (red), Kelvin (green) and Rossby (blue) waves. The black circles correspond to the frequencies from the 2D finite element modal analysis while the dots from continuous lines are the reference frequencies from the 1D Sturm-Liouville approach. Poincaré and Kelvin modes are shown on the close up view A: the reference solution is made of different curves, so that two different couples (k_x, k_y) with two different frequencies may correspond to a same wavenumber. Rossby dispersion relations for different wavenumbers k_x are depicted on close up view B, where only even k_x wavenumbers are obtained because of periodic boundary conditions.

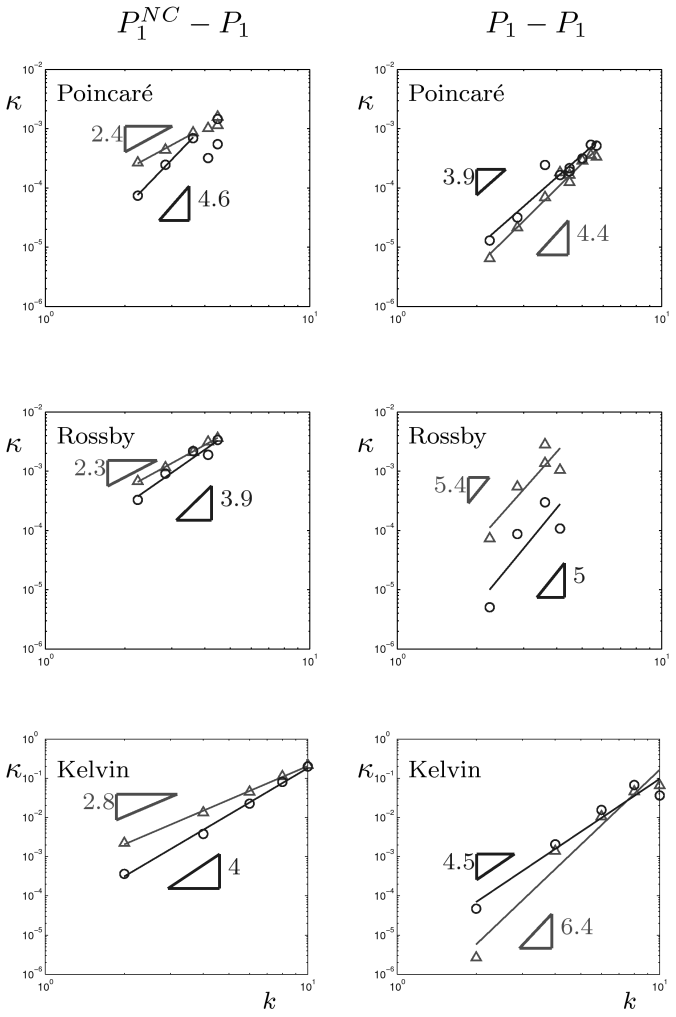


Figure 6.20: Convergence of the relative dispersion error for the first resolved modes corresponding to the dispersive Poincaré (upper row), Rossby and Kelvin(lower row) waves on structured (circles) and unstructured (triangles) meshes with the $P_1^{NC} - P_1$ pair (left) and the $P_1 - P_1$ pair (right). The use of unstructured meshes strongly decreases the convergence rates for the $P_1^{NC} - P_1$ pair while the rates remain unchanged for the $P_1 - P_1$ pair.

by considering the error variation with the wave number. Indeed, one way of computing the convergence of the dispersion errors is to consider several computations on different mesh with different element sizes h, and to compare the dispersion errors for one given mode. For sake of simplicity and in order to reduce the computational costs, we choose the second way which is to consider only one mesh and to compare the different resolved modes, which means considering a variation of k instead of a variation of h. Of course, this second approach is totally valid and equivalent to the first one and the convergence will be observed only if we consider the resolved modes far from the grid cutoff and if the shapes of the modes under study are close to sine functions, which is the case in this study, considering the physical parameters, the domain shape and boundary conditions used.

It has been shown that the Discontinuous Galerkin method exhibits superconvergence properties for the dissipation and dispersion errors [e.g. Hu and Atkins, 2002]. For polynomial shape functions of order p, the DG method provides a convergence rates of $2p+2$ for the relative dispersion errors and $2p+1$ for the relative dissipation errors if an even order is used. Same rates are reached for odd orders only if a Riemann solver is used in the flux computation instead of a center scheme. Those results have also been demonstrated [e.g. Ainsworth, 2004] for a 1D scalar transport equation. The same rates were obtained for the DG method with the same modal analysis for both structured and unstructured grids, for the Poincaré, Kelvin and Rossby waves [e.g. Bernard et al., 2008].
A fair comparison between a high order discontinuous method and other finite element pairs as the non-conforming $P_1^{NC} - P_1$ pair is not a simple task since the results depend in a large part on the problem under study, and in particular on the geometry and topology of the problem. A first comparison based on the dispersion errors criterion is provided by the modal analysis exposed in this paper, since both formulations are compared with the same method and the same number of degrees of freedom.

We observe in Figure (6.20) the difference between the $P_1^{NC} - P_1$ pair on structured grids (blue lines) and the same pair on unstructured grids (red lines). A first polynomial order DG method provides a convergence rate of 4 with a Riemann solver and of 2 with a centered scheme for the relative dispersion error. We observe here that the $P_1^{NC} - P_1$ pair

exhibits a convergence rate as high as the P_1 DG method for structured grids, but the use of unstructured grids considerably reduces the rate, reaching about 2.5, which is similar to the rate of convergence of a finite volume method.

Comparison with the $P_1 - P_1$ pair

The same computation is performed on both kinds of grids with the $P_1 - P_1$ pair. We consider here a transient computation and the no normal flow boundary conditions are weakly imposed to avoid any spurious elevation mode with this pair. Moreover, solving the linear shallow water equations with this pair does not require the use of SUPG formulation to stabilize the scheme. The $P_1 - P_1$ pair is thus well suited for this particular simple computation, even though its is usually not recommended for solving mixed formulations.
Under those assumptions, we observe in the right column of Figure (6.20) the relative dispersion errors. The $P_1 - P_1$ pair exhibits the same rate of convergence of 4. For this modal analysis, such an element pair appears to be a better discrete space, as the formulation (6.9) is no more a constrained problem and the issue of spurious modes is no more relevant. For pure wave propagation, the use of the same discretization then appears to be a better choice in term of accuracy. It is still not quite clear why the $P_1^{NC} - P_1$ accuracy is strongly affected and why the convergence rate of the $P_1 - P_1$ pair remains unchanged when considering unstructured grids.

Influence of mesh perturbation with the $P_1^{NC} - P_1$ pair

Given the strong degradation of the convergence rate of the $P_1^{NC} - P_1$ pair on unstructured grids, one might be interested in computing the convergence rates on slightly distorted structured grids. We consider the perturbed meshes depicted in Figure (6.21) and apply the same analysis. The convergences are depicted in Figure (6.22) for both structured-distorted meshes.
We observe a fast degradation of the convergence rate when perturbing the structured nature of the mesh: the first slightly distorted mesh leads

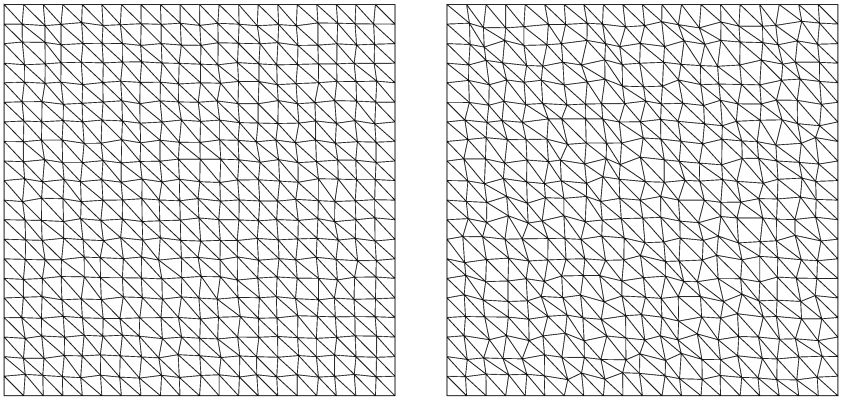


Figure 6.21: Structured-distorted meshes used for the computations on the semi-periodic domain with 2921 degrees of freedom and 800 elements with the $P_1^{NC} - P_1$ finite element pair. The perturbation on the nodes coordinates is a random function presenting a maximum value of 10% of the element size h on the less distorted mesh (left) and of 20% on the more distorted mesh (right).

to a convergence rate of about 3.5 instead of 4 on the original structured mesh. The second larger perturbation leads to a rate of about 2.7, which is almost as low as the convergence rate on fully unstructured meshes.

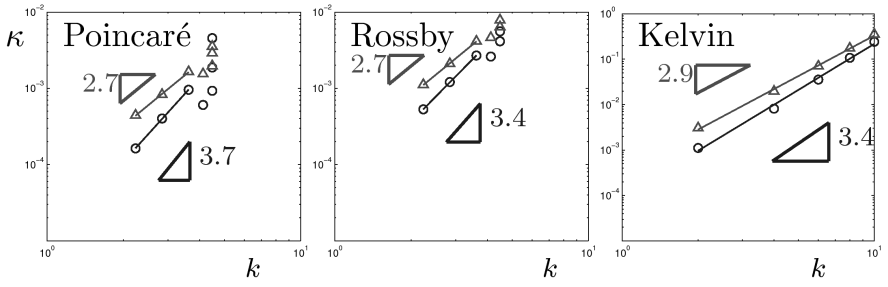


Figure 6.22: Convergence of the relative dispersion error for the first resolved modes corresponding to the Poincaré (left), Rossby and Kelvin (right) waves on the less distorted structured mesh (blue circles) and the more distorted structured mesh (red triangles) with the $P_1^{NC} - P_1$ pair. The first slight perturbation of the mesh leads to a convergence rate of about 3.5 instead of 4, while the second perturbation leads to a rate equivalent to the rate on fully unstructured meshes.

6.4.3 Influence of the subgrid scale viscosity on the $P_1^{NC} - P_1$ pair

Finally, we observe that the higher order of convergence can be recovered on unstructured grids with the $P_1^{NC} - P_1$ pair by introducing a sufficient amount of subgrid scale viscosity scaling as $\mathcal{O}(h^2)$. The convergence for the Poincaré, Kelvin and Rossby waves are depicted in Figure (6.23). For a small viscosity of $\nu = 0.02h^2$ (red lines), the convergence rates are not much improved compared to the inviscid computation, while the use of a larger viscosity $\nu = 0.2h^2$ (green lines) leads to recovering approximately the convergence of 4.

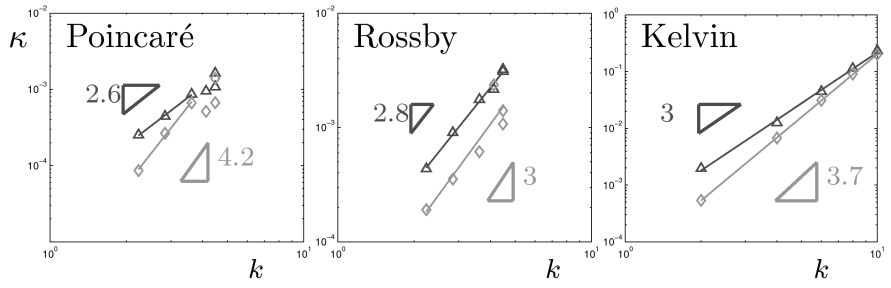


Figure 6.23: Convergence of the relative dispersion error with sudgrid scale viscosities of $\nu = 0.02h^2$ (red triangles) and $\nu = 0.2h^2$ (green diamonds) for the first resolved modes corresponding to the Poincaré, Rossby and Kelvin waves on unstructured meshes with the $P_1^{NC} - P_1$ pair. The convergence rate on unstructured meshes is highly improved with the use of a sufficiently large subgrid scale viscosity.

6.5 Conclusions

A first aspect of this study is not directly related to ocean modeling, but more generally to the analysis of numerical properties of numerical schemes. The comparison between numerical methods on unstructured grids is a quite difficult task and the conclusions may eventually lean towards one method or another, depending on the problem under study. A new method has been proposed here, providing a way to fairly compare the dispersion and dissipation errors introduced by different numerical schemes on unstructured grids, and bringing thus complementary informations for ranking different methods on the basis of dissipation and dispersion errors.
Starting from an implementation of the two-dimensional linearized shallow water equations with the DG method, we developed a modal analysis of the scheme by computing the eigenvalues and eigenvectors of the discretization of the space operators. This analysis is therefore fully independent of the numerical scheme and of the mesh. Only periodic boundary conditions on the eastern and western boundaries were considered in order to simplify algebra and to reduce the computational costs while keeping the three kind of waves in the system. The extension of this test case to non periodic conditions is straightforward and

would not affect the efficiency of the method but the convergence study would then become prohibitive in terms of computational costs, since we would have to perform the convergences on several meshes instead of simply considering the wavenumbers.

The modal analysis has been applied in particular to the study of the DG method. It has been shown that the DG method still exhibits its superconvergence properties when applied to the two-dimensional dispersive shallow water equations. Convergence rates of $2p+2$ and $2p+3$ were observed on the dissipation and dispersion errors, for both Poincaré and Kelvin waves. The theoretical rates were also obtained for the very dispersive Rossby waves, provided that the rate computation is based on the relative errors. Note that in the framework of an ocean model which aims to capture small scale processes, say between 1 and 25 km, the very large Rossby wavelengths will always be resolved in a very accurate way. Moreover, the Rossby frequencies are significantly different from zero and have to be resolved only for the smaller wavenumbers. It has been finally shown that the convergence properties remain unchanged when using both structured and unstructured meshes.

In the context of ocean modeling, the method has been used to compare the DG method to the mixed $P_1^{NC} - P_1$ FE pair. Solving the mixed problem with the $P_1^{NC} - P_1$ pair avoids the issue of the spurious elevation modes generation. Moreover, the velocity discontinuities are quite useful for introducing upwinding schemes when computing non-linear advection terms. The $P_1^{NC} - P_1$ has been chosen as the best candidate for modeling ocean processes with mixed formulations because of its numerical properties observed on structured grids. However, this study has enlightened the fact that the $P_1^{NC} - P_1$ pair exhibits a *high sensitivity to the mesh structure and leads to a significative loss of convergence for the dispersion error on unstructured meshes* or even slightly distorted structured meshes. Indeed, we obtained different convergence properties on structured and unstructured grids. The $P_1^{NC} - P_1$ pair leads to a convergence rate of the relative dispersion errors of 4 on structured grids, which is equivalent to the superconvergence properties of the piecewise linear DG method with the use of a Riemann solver, i.e. $2p + 2$. This convergence of the $P_1^{NC} - P_1$ pair is *highly reduced on unstructured grids* and reach about 2.5 while the convergence rates of schemes as the DG

method or the $P_1 - P_1$ pair remain unchanged on both kinds of grids. The convergence rate may however be recovered on unstructured meshes by introducing a sufficient amount of viscosity scaling as $\nu \propto \mathcal{O}(h^2)$ where h is the element size. However, the dissipation errors generated to recover the dispersion convergence rate often exceed the dispersion errors itself. Although certainly unacceptable for solving the wave equation, this option is however valid for an ocean model if the diffusivity associated to the subgrid scale processes exceeds the viscosity needed to recover the dispersion convergence rate.

Although numerical scheme comparison is a complex matter and cannot be reduced to a single criterion as the convergence properties of dispersion errors, this lack of accuracy of the $P_1^{NC} - P_1$ formulation on unstructured meshes is a real issue, especially in the framework of a finite element ocean model based on the use of unstructured grids. The use of more accurate pairs or methods on unstructured meshes should then be investigated, as for instance the discontinuous Galerkin method which does not suffer from such convergence rate reduction. Because of its dissipation and dispersion properties, the high-order DG method reveals as very accurate to simulate wave propagation, and is therefore a good candidate for ocean applications.
This study eventually points out the fact that the conclusions from analysis of pairs based on structured grids may be irrelevant in terms of convergence errors for unstructured grids. The results obtained on structured meshes cannot always be directly extended to unstructured ones and such dispersion analysis should then be carried on on unstructured meshes.

Chapter 7

Conclusions and Perspectives

We have developed during this doctoral research a two-dimensional model on possibly curved manifolds to solve the shallow water equations using the high-order discontinuous Galerkin method, with the aim to investigate some key aspects of ocean modeling on unstructured meshes. This research work, realized in the framework of the SLIM project, is a contribution to the creation of second generation OGCM's based on unstructured meshes. Only a few research groups worldwide are currently studying such models, and the optimal choice in term of numerical methods is still an open question. The work undertook in this thesis consists in one step forward to the selection process of better suited numerical techniques.

We have investigated many aspects of the DG method applied to ocean modeling, and have shown the large potential of this approach, especially when using polynomial orders as high as possible. In particular, the DG method has revealed very appealing considering the following key aspects of ocean modeling.
Ocean dynamics is often dominated by **advection** processes. Continuous finite element methods were not designed for advection processes in the first place and usually require additional stabilization terms, introducing dissipation as SUPG methods. Mixed semi-continuous formulations as the $P_1^{NC} - P_1$ pair allows one to introduce dissipation by per-

forming upwind on the non-linear transport terms, while no additional dissipation is required to solve the linearized equations. The DG method has been shown to be particularly efficient for advection, since accurate advection schemes as Riemann solver can be easily introduced at the inter-element discontinuities, as in finite volume methods. Even at the low Froude numbers experienced in the ocean, such advection schemes are mandatory for a relevant solution. Indeed, Riemann solvers introduce small amount of dissipation on the non-physical high-frequency modes of the solution, while the relevant low frequency modes does not experience dissipation. The solution obtained is therefore more physically relevant since the amount of dissipation introduced is controlled by the underlying physics. The computational costs associated to Riemann-like solvers are not significant compared to the whole computation, especially at high polynomial orders. Notice finally that issues have been reported in the literature concerning oscillations due to Riemann-like solvers for the Euler equations at very low Mach number. Although the equations are very similar, this drawback has not been observed using the DG method at low Froude numbers in the ocean.

The high-order DG method has then been shown to be very efficient for computing **wave propagation on unstructured meshes**. Indeed, it presents interesting superconvergence properties of the **dispersion and dissipation** errors for the Kelvin and the dispersive Poincaré and Rossby waves, on both structured and unstructured grids. The method has been compared to the mixed formulation with the $P_1^{NC} - P_1$ pair. The latter reveals surprisingly much less efficient for propagating waves on unstructured grids, the convergence rate of dispersion error being highly reduced. Such reductions of the convergence properties on unstructured grids have also been observed for the discretization errors using this mixed formulation. The new method developed for this analysis is independent of the numerical scheme and of the mesh, and could be used for comparing with other elements or methods.

An **efficient adaptive strategy** has been developed using the DG method to solve the very transient and local processes as the mesoscale eddies, the polar fronts displacements or the western boundary currents. The approach considers a very simple error estimator based on the jumps contained in the fields, while only local operations are performed. The conservation of the projected fields is therefore locally ensured while the memory footprint required is drastically reduced com-

pared to other approaches. An automatic control of the new optimal mesh metric has been proposed with the aim either to minimize the total number of elements to reach a relative target error, or to minimize the error while keeping about constant the number of elements. We have shown on standard benchmarks the efficiency of the method, and in particular the coupling of adaptive mesh refinement with high-order polynomials. A one-dimensional benchmark has finally been studied to illustrate the simplicity and efficiency of $p-$adaptivity. The DG method is clearly very well suited for such strategies and the associated computational costs are not significant, while mixed finite element formulations may highly increase the level of complexity of such procedures.
The creation of an general circulation model requires to solve the conservation laws **on the sphere**. A new general approach has been proposed to solve vectorial partial differential equations on **curved manifolds**. The approach combines the advantages of both classical approaches described in the literature, i.e. uses a minimum number of unknowns, ensures the tangency constraint and does not require to rewrite the PDE's according to an analytical metric tensor. The method is therefore efficient, robust and general, and the DG method is particularly well suited to this approach. Moreover, the use of high-order polynomials allows one to accurately represent the manifold's geometry with a very few number of elements. The approach has been successfully applied to some standard atmospheric benchmarks and to more non-conventional irregular manifolds. Although efficient for ocean modeling, the approach using the DG method is clearly well suited to any processes on any curved manifold.
The use of **high-order polynomials** has finally been investigated in the context of realistic complex geometries. If efficient high-order methods as spectral elements are usually disregarded because of their inability to represent complex geometries and forcings, it has however been shown that it is possible to use the high-order DG method for computing such flows. A method has been proposed based on the use of curved elements, in order to represent accurately the geometry of coasts and islands, avoiding usual drawbacks as oscillations around bluff bodies. The technique has been applied to the computation of tidal flows around the Rattray island using a **very coarse curved meshes**, realistic forcings and irregular bathymetry. Good agreement has been observe with insitu measurements.

Therefore, the DG method has been shown as particularly efficient and well suited for ocean modeling. Indeed, the method fulfills in a large part the first requirements of ocean modeling. The comparison with other numerical techniques on unstructured grids leads ton consider the DG method as a good candidate for such models. Finally, some of the methods and tools proposed in this work are general and not directly related to ocean modeling or to the DG method. Those tools could thus be applied to other concerns, with other finite element techniques, or used to study a larger range of elements for mixed formulations.

Future developments

However, lots of work has still to be done for achieving a complete, robust and general ocean model based on the DG method.

If the efficiency of the mesh adaptation strategy has been clearly shown, many improvements are possible to apply the concept to efficient ocean modeling, especially for large scales of time where $h-$adaption with bathymetry variations could be an issue. These improvements consist in coupling in a smart way the use of $h-$ and $p-$ adaptivity and the treatment of complex geometries. Consider for instance a computation on a very smooth domain, with possible h/p adaptation. Clearly, the fastest and easiest way to increase the accuracy is to first increase the polynomial order, because of the polynomial superconvergence properties of the DG method. Now, considering irregular domains with coasts and islands, it has been shown that it is possible to perform high-order computations on coarse meshes using high-order curved elements. Therefore, the maximum elements sizes is clearly controlled by the desired resolution on the coastlines, while the accuracy of the scheme is first related to the polynomial order. An efficient strategy would be to initially adapt the element sizes and curvatures to represent the geometrical features as coasts or strong bathymetry gradients in a sufficiently accurate way, while using elements as large as possible in other regions. Therefore, efficiency would be ensured by $p-$adaptivity while the accurate high-order representation of the coasts would avoid issues related to high-order polynomials in complex geometries.

Considering more local applications on small time scales, i.e. where accurate conservation in time is less crucial (computations on the Great Bareer Reef, the Mururoa Atoll or the North Sea), a more intensive use of $h-$adaption could be considered if necessary. Besides the initial relative smoothing of the bathymetry according to the resolved scales and the intersection of both bathymetry and solution based metrics, an essential component would be the efficient adaptation of the coasts. The adaptation should take into account the possible curvature of the boundary elements, by adapting according to some high-order parametrized representation of the coastlines, and not only according to the element edges.

In this framework of curved elements and the representation of geometrical features, many improvement could also be brought to the generation of high-order curved meshes. Indeed, if the generation of classical piecewise linear meshes is now perfectly controlled and automated, curved meshes generation is still a challenging issue because of the increased number of degrees of freedom on the elements orders, the position of the additional nodal coordinates, etc. An efficient high-order ocean model based on the DG method clearly requires the efficient automated generation of curved meshes. The elements sizes and orders should be automatically adapted to the coastlines curvature, and the mesh should then be efficiently optimized by moving some nodal coordinates in order to minimize the mapping distortion of the elements. Indeed, highly distorted curved meshes will prevent from convergence and possibly generate oscillations in presence of strong bathymetry gradients, since the accuracy of the quadrature rules become insufficient.

Many other aspects still need to be developed. The model should be extended to three-dimensional domains. According to the previous studies of 3D ocean models, the extension of the DG model should not present particular issues since using the same representation of elevation and velocity fields. This extension should take into account a flexible representation of the bottom of the ocean, in order to analyze its impact on high-order polynomials. Indeed, one could imagine that the advantages and drawbacks of the usual representations of the bathymetry, e.g. $\sigma-$ and $z-$ coordinates, might be modified using high-order representations.
The efficiency of the DG method could also clearly be used for solving

tracers advections. However, considering ecological models with chemical and biological modelizations of several components, the use of high-order methods would probably raise many issues and requires further investigations. Indeed, such computations often imply complex piecewise modelizations of the reactions. High-order computations might then introduce Gibbs oscillations and lack of monotonicity not well suited for such parameterizations.
Finally, the continuous improvements in computers technology will clearly lead to finer meshes and to an inevitable extension of the range of resolved scales. Therefore, unstructured meshes coupled to adaptive strategies will probably raise many questions in term of modelization. If the mesh adaptation technology and the parametrization of subgrid scale processes become now sufficiently mature to be used in actual numerical codes, the next step will probably consist in adapting the equations to the element size. Indeed, some features as the hydrostatic assumption might become more and more questionable with finer meshes.

Bibliography

A. Adcroft, J.-M. Campin, C. Hill, and J. Marshall. Implementation of an atmosphere-ocean general circulation model on the expanded spherical cube. *Monthly Weather Review*, 134:2845–2863, 2004.

M. Ainsworth. Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods. *Journal of Computational Physics*, 198:106–130, 2004.

M. Ainsworth and J.T. Oden. A procedure for a posteriori error estimation for h-p finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 101:73–96, 1992.

M. Ainsworth and J.T. Oden. A posteriori error estimation in finite element analysis. *Computer Methods in Applied Mechanics and Engineering*, 142:1–88, 1997.

M. Ainsworth and J.T. Oden. *A posteriori error estimation in finite element analysis*. Pure and Applied Mathematics. Wiley-Interscience Publication, 2000.

V. Aizinger and C. Dawson. A discontinuous galerkin method for two-dimensional flow and transport in shallow water. *Advances in Water Resources*, 25:67–84, 2002.

V. Aizinger and C. Dawson. The local discontinuous galerkin method for three-dimensional shallow water flow. *Computer Methods in Applied Mechanics and Engineering*, 196:734–746, 2007.

A. Arakawa and V.R. Lamb. Computational design of the basic dynamical processes of the UCLA general circulation model. In J. Chang,

editor, *Methods in Computational Physics*, pages 173–265. Academic Press, 1977.

D.N. Arnold, F. Brezzi, B. Cockburn, and L.D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM journal on numerical analysis*, 39:1749–1779, 2002.

H. L. Atkins and C.-W. Shu. Quadrature-free implementation of discontinuous Galerkin methods for hypebolic equations. *AIAA Journal*, 36 (5):775–782, 1998.

J.H. Atkinson, J.J. Westerink, and R.A. Luettich. Two dimensional dispersion analyses of finite element approximations to the shallow water equations. *International Journal for Numerical Methods in Fluids*, 45: 715–749, 2004.

I. Babuska and M. Zlamal. Non conforming elements in the finite element method with penalty. *SIAM Journal on Numerical Analysis*, 10:863–875, 1973.

P.L. Baehmann, M.S. Shephard, and J.E. Flaherty. A posteriori error estimation for triangular and tetrahedral quadratic elements using interior residuals. *International Journal For Numerical Methods In Engineering*, 34:979–996, 1992.

F. Bassi and S. Rebay. A high-order accurate discontinuous finite element solution of the 2d Euler equations. *Journal of Computational Physics*, 138:251–285, 1997a.

F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible navier-stokes equations. *Journal of Computational Physics*, 131:267–279, 1997b.

M.L. Batteen and Y.-J. Han. On the computational noise of finite-difference schemes used in ocean models. *Tellus*, 33:387–396, 1981.

J.-M. Beckers and E. Deleersnijder. Stability of a fbtcs scheme applied to the propagation of shallow-water inertia-gravity waves on various space grids. *Journal of Computational Physics*, 108:95–104, 1993.

J. Behrens. Atmospheric and ocean modeling with an adaptive finite element solver for the shallow-water equations. *Applied Numerical Mathematics*, 26:217–226, 1998.

P.-E. Bernard, N. Chevaugeon, V. Legat, E. Deleersnijder, and J.-F. Remacle. High-order h-adaptative discontinuous Galerkin methods for ocean modelling. *Ocean Dynamics*, 57:109–121, 2007.

P.-E. Bernard, J.-F. Remacle, and V. Legat. Dispersion analysis of discontinuous galerkin schemes applied to Poincaré, Kelvin and Rossby waves. *Journal of Scientific Computing*, 34:26–47, 2008.

K.S. Bey and J.T. Oden. A runge-kutta discontinuous galerkin finite element method for high speed flows. In *Proceedings of AIAA 10th computational Fluid Dynamics Conference, Honolulu, Hawaii*, 1991.

R. Biswas and R.C. Strawn. New procedure for dynamic adaptation of three dimensional unstructured grids. *Applied Numerical Mathematics*, 13:437–452, 1994.

R Biswas, K.D. Devine, and J. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Applied Numerical Mathematics*, 14: 255–283, 1994.

R. Bleck. On the use of hybrid vertical coordinates in numerical weather prediction models. *Monthly Weather Review*, 106:1233–1244, 1978.

J.P. Boyd. High-order models for the non-linear shallow water waves equations on the equatorial beta-plane with application to the kelvin wave frontogenesis. *Dynamics of Atmosphere and Oceans*, 28:69–91, 1998.

K. Bryan. A numerical method for the study of the circulation of the world ocean. *Journal of Computational Physics*, 4:347–376, 1969.

K. Bryan and M.D. Cox. A numerical investigation of the oceanic general circulation. *Tellus*, 19:54–80, 1967.

V. Casulli and R.A. Walters. An unstructured grid, three-dimensional model based on the shallow water equations. *International Journal for Numerical Methods in Fluids*, 32:331–348, 2000.

V. Casulli and P. Zanolli. Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems. *Mathematical and Computer Modelling*, 32:331–348, 2000.

N. Chevaugeon, J.-F. Remacle, X. Gallez, P. Ploumans, and S. Caro. Efficient discontinuous galerkin methods for solving acoustic problems. In *11th AIAA/CEAS Aeroacoustics Conference*, 2005a.

N. Chevaugeon, J. Xin, P. Hu, X. Li, D. Cler, J. E. Flaherty, and M. S. Shephard. Discontinuous Galerkin methods applied to shock and blast problems. *Journal of Scientific Computing*, 22-23:227–243, 2005b.

N. Chevaugeon, K. Hillewaert, X. Gallez, P. Ploumhans, and J.-F. Remacle. Optimal numerical parametrization of discontinuous Galerkin method applied to wave propagation problems. *Journal of Computational Physics*, 223:188–207, 2007.

J.Y-K. Cho and L.M. Polvani. The emergence of jets and vortices in freely evolving, shallow-water turbulence on a sphere. *Physics of Fluids*, 8:1531–1552, 1996.

B.J. Choi, M. Iskandarani, J.C. Levin, and D.B. Haidvogel. A spectral finite volume method for the shallow water equations. *Monthly Weather Review*, page submitted, 2004.

P.G. Ciarlet. *The finite element method for elliptic problems.* Studies in Mathematics and its Applications. North-Holland/American, 1978.

P.G. Ciarlet. *Basic error estimates for elliptic problems.* Handbook of Numerical Analysis, Vol. II:Finite Element Methods (Part I). North-Holland, Amsterdam, 1991.

B. Cockburn and C.-H. Shu. Tvd Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws ii: General framework. *Mathematics of Computation*, 52(186):411–435, 1989.

B. Cockburn and C.-W. Shu. The Runge-Kutta discontinuous galerkin method for conservation laws v - multidimensional systems. *Journal of Computational Physics*, 141:191–224, 1998a.

B. Cockburn and C.W. Shu. The runge-kutta local projection p1-discontinuous galerkin method for scalar conservation laws. *Mathematical Modelling and Numerical Analysis*, 25:337–361, 1991.

B. Cockburn and C.W. Shu. The local discontinuous galerkin finite element method for convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35:2440–2463, 1998b.

B. Cockburn, G. E. Karniadakis, and C. W. Shu. *Discontinuous Galerkin Methods. Theory, computation and applications.* Lecture Notes in Computational Science and Engineering, 11. Springer-Verlag Berlin, 2000.

R. Comblen, S. Legrand, E. Deleersnijder, and V. Legat. A finite element method for solving the shallow water equations on the sphere. *Ocean Modelling*, (accepted for publication), 2008.

J. Côté. A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere. *Quarterly Journal of the Royal Meteorological Society*, 114:1347–1352, 1988.

T. Coupez. Generation de maillages et adaptation de maillages par optimisation locale. *Revue européenne des elements finis*, 9:403–423, 2000.

B. Cushman-Roisin. *Introduction to Geophysical Fluid Dynamics.* Prenctice Hall, 1994. ISBN 0-13-353301-8.

S. Danilov, G. Kivman, and J. Schröter. A finite-element ocean model: principles and evaluation. *Ocean Modelling*, 6:125–150, 2004.

A.M. Davies and J.E. Jones. On the numerical solution of the turbulence energy equations for wave and tidal flows. *International Journal of Numerical Mathods in Fluids*, 12:17–41, 1991.

C. Dawson and J. Proft. Coupled discontinuous and continuous galerkin finite element methods for the depth-integrated shallow water equations. *Computer Methods in Applied Mechanics and Engineering*, 193: 289–318, 2004.

H.L. deCougny, K.D. Devine, J.E. Flaherty, R.M. Loy, C. Ozturan, and M.S. Shephard. High-order accurate discontinuous finite element solution of the 2d euler equations. *Applied Numerical Mathematics*, 16: 157–182, 1994.

E. Deleersnijder. On the numerical treatment of a lateral boundary layer in a shallow sea model. *Journal of Marine Systems*, 8:107–117, 1996.

E. Deleersnijder, A. Norro, and E. Wolanski. A three-dimensional model of the water circulation around an island in shallow water. *Continental Shelf Research*, 12:891–906, 1992.

K.D. Devine, J.E. Flaherty, S.R. Wheat, and A.B. Maccabe. A massively parallel adaptive finite element method with dynamic load balancing. *Proceedings of Supercomputing - IEEE Computer Society Press*, SAND 93-0936C:2–11, 1993.

K.D. Devine, J.E. Flaherty, R.M. Loy, and S.R. Wheat. Parallel partitionning strategies for the adaptive solution of conservation laws. *Rensselaer Polytechnic Institute Report No.94-1*, 1994.

M. Dindar, M.S. Shephard, J.E. Flaherty, and K. Jansen. Adaptive cfd analysis for rotorcraft aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 189:1055–1076, 2000.

C. Dobrzynski. *Adaptation de Maillage anisotrope 3D et application à l'aéro-thermique des bâtiments*. PhD thesis, Université Pierre et Marie Curie - Paris VI, 2005.

J. Douglas and T. Dupont. *Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods*. Lecture Notes in Physics. Springer Verlag, Berlin, 1976.

J.K. Dukowicz. Mesh effects for rossby waves. *Journal of Computational Physics*, 119:188–194, 1995.

R.A. Falconer, E. Wolanski, and L. Mardapitta-Hadjipandeli. Modeling tidal circulation in an island's wake. *Journal of Waterway, Port, Coastal and Ocean Engineering*, 112,2, 1986.

G.J. Fix. Finite element models for ocean circulation problems. *SIAM J. Appl. Math.*, 3:371–387, 1975.

J.A. Font. Numerical hydrodynamics in general relativity. *Living Reviews in Relativity*, 2000.

R. Ford, C.C. Pain, M. Piggott, A. Goddard, C.R. de Oliveira, and A. Umpleby. A non-hydrostatic finite element model for three-dimensional stratified oceanic flows, Part II: Model validation. *Monthly Weather Review*, 132:2832–2844, 2004a.

R. Ford, C.C. Pain, M. Piggott, A. Goddard, C.R. de Oliveira, and A. Umpleby. A non-hydrostatic finite element model for three-dimensional stratified oceanic flows, Part I: Model formulation. *Monthly Weather Review*, 132:2816–2831, 2004b.

O.B. Fringer, M. Gerritsen, and R.L. Street. An unstructured-grid, finite-volume, nonhydrostatic, parallel coastal ocean simulator. *Ocean Modelling*, 14:139–278, 2006.

P. Gent and J.C. McWilliams. Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography*, 20:150–155, 1990.

P.-L. George. Tet meshing: construction, optimization and adaptation. In *Proceedings of 8th Int. Meshing Roundtable, South Lake Tao, CA, USA*, 1999.

C. Geuzaine and J.-F. Remacle. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, (submitted), 2008.

P.A. Gilman. Magnetohydrodynamics "shallow water" equations for the solar tachocline. *Astrophysical Journal*, 544:L79–L82, 2000.

F. X. Giraldo and T. Warburton. A nodal triangle-based spectral element method for the shallow water equations on the sphere. *Journal of Computational Physics*, 207:129–150, 2005.

F. X. Giraldo, J. S. Hesthaven, and T. Warburton. Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations. *Journal of Computational Physics*, 181:499–525, 2002.

F.X. Giraldo. Lagrange-Galerkin methods on spherical geodesic grids : The shallow water equations. *Journal of Computational Physics*, 160: 336–368, 2000.

F.X. Giraldo. High-order triangle-based discontinuous Galerkin methods for hyperbolic equations on a rotating sphere. *Journal of Computational Physics*, 214:447–465, 2006. doi: doi:10.1016/j.jcp.2005.09.029.

S.K. Godunov. Finite difference methods for the computation of discontinuous solutions of the equations of fluid dynamics. *Math. Sbornik*, 47:271–306, 1959.

O. Gourgue, E. Deleersnijder, and L. White. Toward a generic method for studying water renewal, with application to the epilimnion of lake tanganyika. *Estuarine, Coastal and Shelf Science*, 74:764–776, 2007.

S. M. Griffies, C. Böning, F.O. Bryan, E.P. Chassignet, R. Gerdes, H. Hasumi, A. Hirst, A.-M. Treguier, and D. Webb. Developments in ocean climate modeling. *Ocean Modelling*, 2:123–192, 2000.

D.A. Ham, J. Pietrzak, and G.S. Stelling. A scalable unstructured grid 3-dimensional finite volume model for the shallow water equations. *Ocean Modelling*, 10:153–169, 2005.

E. Hanert, V. Legat, and E. Deleersnijder. A comparison of three finite elements to solve the linear shallow water equations. *Ocean Modelling*, 5:17–35, 2002.

E. Hanert, D.Y. Le Roux, V. Legat, and E. Deleersnijder. Advection schemes for unstructured grid ocean modelling. *Ocean Modelling*, 7: 39–58, 2004.

E. Hanert, D.Y. Le Roux, V. Legat, and E. Deleersnijder. An efficient eulerian finite element method for the shallow water equations. *Ocean Modelling*, 10:115–136, 2005.

E. Hanert, E. Deleersnijder, and V. Legat. An adaptive finite element water column model using the Mellor-Yamada level 2.5 turbulence closure scheme. *Ocean Modelling*, 12:205–223, 2006.

E. Hanert, E. Deleersnijder, S. Blaise, and J.-F. Remacle. Capturing the bottom boundary layer in finite element ocean models. *Ocean Modelling*, 17:153–162, 2007.

T. Heinze and A. Hense. The shallow water equations on the sphere and their Lagrange-Galerkin-solution. *Meteorology and Atmospheric Physics*, 81:129–137, 2002.

K. Hillewaert, N. Chevaugeon, P. Geuzaine, and J.-F. Remacle. Hierarchic multigrid iteration strategy for the discontinuous Galerkin solution of the steady Euler equations. *International Journal for Numerical Methods in Fluids*, 51:1157–1176, 2006.

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

F. Hu and H. Atkins. Eigensolution analysis of the discontinuous Galerkin method with nonuniform grids, part i: one space dimension. *Journal of Computational Physics*, 182(2):516–545, 2002.

B.-L. Hua and F. Thomasset. A noise-free finite element scheme for the two-layer shallow water equations. *Tellus*, 36A:157–165, 1984.

T.J.R. Hughes, G. Engel, L. Mazzei, and M.G. Larson. The continuous galerkin method is locally conservative. *Journal of Computational Physics*, 163:467–488, 2000.

R. Iacono, M.V. Struglia, and C. Ronchi. Spontaneous form of equatorial jets in freely decaying shallow water turbulence. *Physics of Fluids*, 11: 1272–1274, 1999.

M. Iskandarani, D.B. Haidvogel, and J.B. Boyd. A staggered spectral element model with application to the oceanic shallow water equations. *International Journal for Numerical Methods in Fluids*, 20:393–414, 1995.

M. Iskandarani, D. B. Haidvogel, and J. C. Levin. A three-dimensional spectral element model for the solution of the hydrostatic primitive equations. *Journal of Computational Physics*, 186:397–425, 2003.

G.A. Jacobs, H.E. Hurlburt, J.C. Kindle, E.J. Metzger, J.L. Mitchell, W.J. Teague, and A.J. Wallcraft. Decade-scale trans-Pacific propagation and warming effects of an El Niño anomaly. *Nature*, 370:360–363, 1994.

G.E. Karniadakis and S.J. Sherwin. *Spectral/hp Element Methods for Computational Fluid Dynamics*. Oxford Scholarship Online Monographs, 2005.

L. Krivodonova and M. Berger. High-order accurate implementation of solid wall boundary conditions in curved geometries. *Journal of Computational Physics*, 211:492–512, 2006.

L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon, and J. E. Flaherty. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. *Applied Numerical Mathematics*, 48:323–338, 2004.

E.J. Kubatko, J.J. Westerink, and C. Dawson. *hp* discontinuous galerkin methods for advection dominated problems in shallow water flows. *Computational Methods in Applied Mechanics and Engineering*, 196: 437–451, 2006.

J. Lambrechts, R. Comblen, V. Legat, C. Geuzaine, and J.-F. Remacle. Multiscale mesh generation on the sphere. *Ocean Dynamics (in preparation)*, 2008a.

J. Lambrechts, E. Hanert, E. Deleersnijder, P.-E. Bernard, V. Legat, J.-F. Remacle, and E. Wolanski. A multiscale model of the whole great barrier reef hydrodynamics. *Estuarine, Coastal and Shelf Science (accepted for publication), doi:10.1016/j.ecss.2008.03.016*, 2008b.

D. Y. Le Roux, A. Sene, V. Rostand, and E. Hanert. On some spurious mode issues in shallow-water models using a linear algebra approach. *Ocean Modelling*, 10:83–94, 2005.

D.Y. Le Roux. A new triangular finite-element with optimum constraint ratio for compressible fluids. *SIAM Journal on Scientific Computing*, 23:66–80, 2001.

D.Y. Le Roux. Dispersion relation analysis of the $p^n c_1 - p_1$ finite-element pair in shallow-water models. *SIAM Journal on Scientific Computing*, 27:394–414, 2005.

D.Y. Le Roux and B. Pouliot. Analysis of numerically-induced oscillations in 2d finite-element shallow water models, part ii: Planetary waves. *SIAM Journal on Scientific Computing*, (in press), 2008.

D.Y. Le Roux, A. Staniforth, and C.A. Lin. Finite elements for shallow-water equation ocean model. *Monthly Weather Review*, 126:1931–1951, 1998.

D.Y. Le Roux, C.A. Lin, and A. Staniforth. A semi-implicit semi-lagrangian finite-element shallow-water ocean model. *Monthly Weather Review*, 128:1384–1401, 2000.

D.Y. Le Roux, V. Rostand, and B. Pouliot. Analysis of numerically-induced oscillations in 2d finite-element shallow water models, part i: Inertia-gravity waves. *SIAM Journal on Scientific Computing*, 29: 331–360, 2007.

S. Legrand, V. Legat, and E. Deleersnijder. Delaunay mesh generation for an unstructured-grid ocean circulation model. *Ocean Modelling*, 2: 17–28, 2000.

S. Legrand, E. Deleersnijder, E. Hanert, V. Legat, and E. Wolanski. High-resolution, unstructured meshes for hydrodynamic models of the great barrier reef, australia. *Estuarine, Coastal and Shelf Science*, 68: 36–46, 2006.

S. Legrand, E. Deleersnijder, E.J.M. Delhez, and V. Legat. Unstructured, anisotropic mesh generation for the northwestern european continental shelf, the continental slope and the neighbouring ocean. *Continental Shelf Research*, 27:1344–1356, 2007.

R. LeVeque. *Numerical methods for Conservation Laws*. Lectures in Mathematics. Birkhäuser-Verlag, 1992.

R.J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002.

X. Li. *Mesh Modification procedure for General 3-D Non-Manifold Domaines.* PhD thesis, Renselear Polytechnic Indtitute, 2003.

J.-L. Lions. Problèmes aux limites non homogènes à données irrégulières: Une méthode d'approximation. *Numerical Analysis of Partial Differential Equations*, pages 283–292, 1968.

R. Löhner. Three-dimensional fluid-structure interaction using a finite element solver and adaptive remeshing. *Computing Systems and Engineering*, 1:257–272, 1990.

R. Löhner and J.D. Baum. Adaptive h-refinement on 3d unstructured grids for transient problems. *International Journal For Numerical Methods In Fluids*, 14:1407–1419, 1992.

I. Lomtev and G.E. Karniadakis. A discontinuous galerkin method for the navier-stokes equations. *International Journal for Numerical Methods in Fluids*, 29:587–603, 1999.

I. Lomtev, C.B. Quillen, and G.E. Karniadakis. Spectral/hp methods for viscous compressible flows on unstructured 2d meshes. *Journal of Computational Physics*, 33:325–357, 1998.

M. S. Longuet-Higgins. Planetary waves on a rotating sphere, part ii. *Proceedings of the Royal Society of London*, 284:40–68, 1965.

H. Ma. A spectral element basin model for the shallow water equations. *Journal of Computational Physics*, 109:133–149, 1993.

A. Majda. *Introduction to PDE's and waves for the atmosphere and ocean.* American Mathematical Society, 2003.

E. Marchandise and J.-F. Remacle. A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows. *Journal of Computational Physics*, 219:780–800, 2006. doi: DOI 10.1016/j.jcp.2006.04.015.

E. Marchandise, N. Chevaugeon, and J.-F. Remacle. Spatial and spectral superconvergence of discontinuous galerkin method for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 215: 484–494, 2006a.

E. Marchandise, N Chevaugeon, and JF Remacle. Spatial and spectral superconvergence of discontinuous galerkin method for hyperbolic problems. *Journal of Computational and Applied Mathematics*, 2006b.

E Marchandise, JF Remacle, and N Chevaugeon. A quadrature-free discontinuous galerkin method for the level set equation. *Journal of Computational Physics*, 212:338–357, 2006c. doi: DOI 10.1016/j.jcp.2005.07.006.

E. Marchandise, P. Geuzaine, N. Chevaugeon, and JF Remacle. A stabilized finite element method using a discontinuous level set approach for the computation of bubble dynamics. *Journal of Computational Physics*, 225:949–974, 2007.

J.C. McWilliams and P.R. Gent. A coupled air and sea model for the tropical pacific. *Journal of the Atmospheric Sciences*, 35:962–989, 1978.

J.C. McWilliams and P.R. Gent. Intermediate models of planetary circulations in the atmosphere and ocean. *Journal of the Atmospheric Sciences*, 37:1657–1678, 1980.

G.L. Mellor and T. Yamada. Development of a turbulence closure model for geophysical fluids problems. *Review of Geophysics and Space Physics*, 20:851–875, 1982.

R.D. Nair, S.J. Thomas, and R.D. Loft. A discontinuous Galerkin global shallow water model. *Monthly Weather Review*, 133:876–888, 2005.

J. Naithani, E. Deleersnijder, and P.-D. Plisnier. Analysis of wind-induced thermocline oscillations of lake tanganyika. *Environmental Fluid Mechanics*, 3:23–39, 2003.

C. Ozturan, H.L. deCougny, M.S. Shephard, and J.E. Flaherty. Parallel adaptive mesh refinement and redistribution on distributed memory computers. *Computational Methods in Applied Mechanics and Engineering*, 119:123–137, 1994.

C.C. Pain, A.P. Humpleby, C.R.E. de Oliveira, and A.J.H. Goddard. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Computer Methods In Applied Mechanics and Engineering*, 190:3771–3796, 2001.

C.C. Pain, M.D. Piggott, A.J.H. Goddard, F. Fang, G.J. Gorman, D.P. Marshall, M.D. Eaton, P.W. Power, and C.R.E. de Oliveira. Three-dimensional unstructured mesh ocean modelling. *Ocean Modelling*, 10:5–33, 2005.

J. Pedlosky. *Geophysical Fluid Dynamics*. Springer-Verlag, Heidelberg, 1979.

J. Peraire, J. Peiro, and K. Morgan. Adaptive remashing for three-dimensional compressible flow computations. *Journal of Computational Physics*, 103:269–285, 1992.

J. Pietrzak, E. Deleersnijder, and J. Schroeter (Editors). The second international workshop on unstructured mesh numerical modelling of coastal, shelf and ocean flows (delft, the netherlands, september 23-25, 2003). *Ocean Modelling (special issue)*, 10:1–252, 2005.

J. Pietrzak, A. Socquet, D. Ham, W. Simons, C. Vigny, R.J. Labeur, E. Schrama, G. Stelling, and D. Vatvani. Defining the source region of the indian ocean tsunami from gps, altimeters, tide gauges and tsunami models. *Earth and Planetary Science Letters*, 261:49–64, 2007.

M.D. Piggott, G.J. Gorman, C.C. Pain, P.A. Allison, A.S. Candy, B.T. Martin, and M.R. Wells. A new computational framework for multi-scale ocean modelling based on adapting unstructured meshes. *International Journal for Numerical Methods in Fluids*, 56:1003–1015, 2008.

G.W. Platzman. Some response characteristics of finite-element tidal models. *Journal of Computational Physics*, 40:36–63, 1981.

J.A. Pons, J.A. Font, J.M. Ibanez, J.M. Marti, and J.A. Miralles. General relativistic hydrodynamics with special relativistic riemann solvers. *Astronomy and Astrophysics*, 339:638–642, 1998.

M. Rancic, R.J. Purser, and F. Mesinger. A global shallow-water model using an expanded spherical cuge: gnomonic versus conformal coordinates. *Quaternary Journal of Royal Meteorology Society*, 122:959–982, 1996.

D.A. Randall. Geostrophic adjustment and the finite-difference shallow-water equations. *Monthly Weather Review*, 122:1371–1377, 1994.

W.H. Reed and T.R. Hill. Triangular mesh method for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory Report LA-UR-73-479, 1973.

J.-F. Remacle, J. E. Flaherty, and M.S. Shephard. Parallel algorithm oriented mesh database. In *Tenth International Meshing Roundtable*, 2001.

J.-F. Remacle, X. Li, N. Chevaugeon, and M.S. Shephard. Transient mesh adaptation using conforming and non conforming mesh modifications. In *Proceedings of 11th Int. Meshing Roundtable, Ithaca, NY, USA*, 2002.

J.-F. Remacle, J. E. Flaherty, and M. S. Shephard. An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems. *SIAM review*, 45:53–72, 2003.

J.-F. Remacle, X. Li, M. S. Shephard, and J. E. Flaherty. Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods. *International Journal for Numerical Methods in Engineering*, 62(7):899–923, 2005.

J.-F. Remacle, S. Soares Frazão, X. Li, and M.S. Shephard. An adaptive discretization of shallow-water equations based on discontinuous-Galerkin methods. *International Journal for Numerical Methods in Fluids*, 52:903–923, 2006.

B. Riviere, M. F. Wheeler, and V. Girault. Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems i. *Computational Geosciences*, 3:337–360, 1999.

P. L. Roe. Approximate Riemann solvers, parameters vectors and difference schemes. *Journal of Computational Physics*, 135:250–258, 1981.

C. Ronchi, R. Iacono, and P. S. Paolucci. The "cubed sphere" : a new method for the solution of partial differential equations in spherical geometry. *Journal of Computational Physics*, 124:93–114, 1996.

J.A. Rossmanith. A wave propagation method for hyperbolic systems on the sphere. *Journal of Computational Physics*, 213:629–658, 2006.

J.A. Rossmanith, D.S. Bale, and R.J. LeVeque. A wave propagation algorithm for hyperbolic systems on curved manifolds. *Journal of Computational Physics*, 199:631–662, 2004.

V. Rostand and D.Y. Le Roux. Raviart-thomas and brezzi-douglas-marini finite element approximations of the shallow water equations. *International Journal for Numerical Methods in Fluids*, (published online), 2007. doi: 10.1002/fld.1668.

D. Schwanenberg and J. Kongeter. A discontinuous Galerkin method for the shallow water equations with source terms. *Computational Science and Engineering*, 11:419–424, 2000.

M. S. Shephard, J. E. Flaherty, K. E. Jansen, X. Li, X. Luo, N. Chevaugeon, J.-F. Remacle, M. W. Beall, and R. M. O'Bara. Adaptive mesh generation for curved domains. *Applied Numerical Mathematics*, 52: 251–271, 2005.

J. Smagorinsky. General circulation experiments with the primitive equations. *Monthly Weather Review*, 91:99–164, 1963.

R.K. Smith and D.G. Dritschel. Revisiting the Rossby-Haurwitz wave test case with contour advection. *Journal of Computational Physics*, 217:473–484, 2006.

S. Soares Frazão and Y. Zech. Undular bores and secondary waves - experiments and hybrid finite-volume modelling. *Journal of Hydraulic Research*, 40:33–34, 2002a.

S. Soares Frazão and Y. Zech. Dam-break in channels with 90-degree bend. *Journal of Hydraulic Engineering, American Society of Civil Engineers*, 128:956–968, 2002b.

A. St-Cyr, J.M. Dennis, S.J. Thomas, and H. Tufo. An adaptive non-conforming spectral element atmospheric model. *SIAM Journal of Scientific Computing (submitted)*, 2006.

H. Stommel. The westward intensification of wind-driven ocean currents. *Transactions - American Geophysical Union*, 29:202–206, 1948.

G. R. Stuhne and W. R. Peltier. New icosahedral grid-point discretizations of the shallow water equations on the sphere. *Journal of Computational Physics*, 148:23–58, 1999.

G. R. Stuhne and W. R. Peltier. A robust unstructured grid discretization for 3-dimensional hydrostatic flows in spherical geometry: A new numerical structure for ocean general circulation modeling. *Journal of Computational Physics*, 213:704–729, 2006.

P.N. Swarztrauber, D.L. Williamson, and J.B. Drake. The Cartesian method for solving partial differential equations in spherical geometry. *Dynamics of Atmospheres and Oceans*, 27:679–706, 1997.

M. Taylor, J. Tribbia, and M. Iskandarani. The spectral element method for the shallow water equations on the sphere. *Journal of Computational Physics*, 130:92–108, 1997.

Maik Thomas, Jürgen Sündermann, and Ernst Maier-Reimer. Consideration of ocean tides in an ogcm and impacts on subseasonal to decadal polar motion excitation. *Geophysical Research Letters*, 28:2457–2460, 2001.

L. Thompson and P. Pinsky. Complex wavenumber fourier analysis of the p-version finite element method. *Computational Mechanics*, 13: 255–275, 1994.

J. Thuburn and Y. Li. Numerical simulations of Rossby-Haurwitz waves. *Tellus*, 52A:181–189, 2000.

E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics, a Practical Introduction.* Springer, Berlin, 1997.

B. van Leer. Towards the ultimate conservation difference scheme, ii. *Journal of Computational Physics*, 14:361–376, 1974.

B. van Leer. Towards the ultimate conservation difference scheme, v. *Journal of Computational Physics*, 32:1–136, 1979.

C.B. Vreugdenhill. *Numerical methods for shallow water flow.* Water Science and Technologie Library. Kluwer Academic Publishers, 1994a.

C.B. Vreugdenhill. *Numerical Methods for Shallow-Water Flow*. Water Science and Technology Library. Kluwer Academic Publishers, 1994b.

R.C. Wajsowicz. Free planetary waves in finite-difference numerical models. *Journal of Physical Oceanography*, 16:773–789, 1986.

R.A. Walters. Numerically induced oscillations in finite element approximations to the shallow water equations. *International Journal for Numerical Methods in Fluids*, 3:591–604, 1983.

R.A. Walters. Design considerations for a finite element coastal ocean model. *Ocean Modelling*, 15:90–100, 2006.

Q. Wang, S. Danilov, and J. Schroter. Comparison of overflow simulations on different vertical grids using the finite element ocean circulation model. *Ocean Modelling*, 20:313–335, 2008.

T. Warburton and G. E. Karniadakis. A discontinuous galerkin method for the viscous mhd equations. *Journal of Computational Physics*, 152:608–641, 1999.

T.C. Warburton, I. Lomtev, R.M. Kirby, and G.E. Karniadakis. A discontinuous galerkin method for the navier-stokes equations on hybrid grids. Technical report, Center for Fluid Mechanics n.97-14, Division of Applied Mathematics, Brown University, 1997.

G. Wentzel. A generalization of quantum conditions for the purposes of wave mechanics. *Zeits. F. Phys.*, pages 38–518, 1926.

L. White and E. Deleersnijder. Diagnoses of vertical transport in a three-dimensional finite element model of the tidal circulation around an island. *Estuarine Coastal and Shelf Science*, 74:735–749, 2007.

L. White, V. Legat, E. Deleersnijder, and D.Y. Le Roux. A one-dimensional benchmark for the propagation of Poincaré waves. *Ocean Modelling*, 15:101–123, 2006.

L. White, V. Legat, and E. Deleersnijder. Tracer conservation for three-dimensional, finite-element, free-surface, ocean modeling on moving prismatic meshes. *Monthly Weather Review*, 136:420–442, 2008.

D.L. Williamson. Difference approximation for fluid flow on a sphere. In WMO-ICSU, editor, *Numerical Methods used in Atmospheric Models*, volume 17. Garp Publication, 1979.

D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, and P.N. Swarztrauber. A standard test set for numerical approximations to the shallow water equations in spherical geometry. *Journal of Computational Physics*, 102:211–224, 1992.

E. Wolanski. *Physical Oceanographic Processes of the Great Barrier Reef*. Boca Raton, FL: CRC Press, 1994.

E. Wolanski, J. Imberger, and M. L. Heron. Island wakes in shallow coastal waters. *Journal of Geophysical Research*, 89, Issue C14:10553–10569, 1984.

E. Wolanski, T. Asaeda, T. Tanaka, and E. Deleersnijder. Three-dimensional island wakes in the field, laboratory and numerical codes. *Continental Shelf Research*, 16:1437–1452, 1996.