University of Trieste

School of Doctorate in Environmental and Industrial Fluid Mechanics
XXIII Cycle

A SEMI-IMPLICIT, SEMI-LAGRANGIAN p-ADAPTIVE DISCONTINUOUS GALERKIN METHOD FOR THE ROTATING SHALLOW WATER EQUATIONS: ANALYSIS AND NUMERICAL EXPERIMENTS

by

Giovanni Tumolo

April 2011

SUPERVISOR

Dott. F. Giorgi

ASSISTANT SUPERVISOR

Dott. L. Bonaventura

Contents

Introduction

The thesis has been focused on the design and analysis of a novel semi-implicit and semi-Lagrangian Discontinuous Galerkin method for the rotating shallow water equations (SISLDG in the following), as a first step in the context of a more ambitious project to develop a new generation, non-hydrostatic, DG based, dynamical core for regional atmospheric modelling. More specifically, the resulting method should improve the numerical discretizations presently employed in RegCM (see Giorgi (1990)).

The shallow water equations actually contain all of the horizontal operators required in a three-dimensional atmospheric model and thus usually represent a necessary first test for new numerical schemes. The techniques proposed here are non-standard in the framework of Discontinuous Galerkin Methods (DG) methods for time dependent problems. Indeed, DG based dynamical cores are very appealing for their accuracy and flexibility, but a critical issue in their application to the numerical solution of CFD problems at low Froude/Mach numbers is represented by the stability limitation on the maximum allowable time step that can be used in practical computations. For example, for Runge-Kutta-DG schemes, stability is proved in Cockburn and Shu (1989) provided that the following CFL condition holds

$$|c|\frac{\Delta t}{h} < \frac{1}{2k+1},$$

where k is the polynomial degree and c the celerity of the fastest propagating waves. Therefore, in order to avoid that the choice of the maximum permissible time step is governed by considerations of stability rather than accuracy, especially having in mind high order approximations, it was decided not to follow the standard way of applying DG to time dependent problems as proposed e.g. in several papers by Cockburn and Shu. Instead, an approach was considered that has already been quite successfully exploited in finite differences (for example in Robert (1982), or Casulli (1990)) and finite elements (for example in Staniforth and Temperton (1986), or Miglio *et al.* (1999), or Le Roux *et al.* (1999), or Giraldo (2005)) frameworks, but, up to now, not fully explored within the DG context. The chosen and novel approach consists in coupling the spatial DG discretization with a combination of Semi-Implicit (SI) and Semi-Lagrangian (SL) time discretizations. This approach is justified by the encouraging results found by other authors in attempts at coupling DG with either SI (for example Restelli and Giraldo (2009)) or SL time discretizations (e.g. Restelli *et al.* (2006)).

The main original results of this thesis work can be summarized as follows:

- the effects of different element choices for the velocity-pressure pairs on the stability of the approximate solution have been investigated by numerical experiments, showing that mixed orders $Q_k - Q_{k-1}$ velocity-pressure pairs (structured meshes

of quadrilaterals are employed) work better then equal order ones, for which clear instabilities arise. Benefits on the stability from the use of mixed order velocity-pressure pair instead of $Q_k - Q_k$ for DG were proved for the Stokes problem (Toselli (2002), Schötzau *et al.* (2003)), but the fact that typical atmospheric flow regimes are characterized by small Froude/Mach numbers suggested the extension of the same strategy to SWE too. Moreover, this mixed order choice for the pressure-velocity pair can be regarded as the DG analogue of staggering in the finite difference framework (see e.g. Winninghoff (1968), Arakawa and Lamb (1977)).

- A simple p-adaptivity criterion has been employed, that allows to adjust dynamically the number of local degrees of freedom employed to the local structure of the solution. This goal has been achieved thanks to the flexibility of the DG spatial discretization and of the orthogonality property of the Legendre polynomial basis. As demonstrated by the one-dimensional and two-dimensional numerical experiments, p-adaptivity strategy employed is quite effective in reducing the computational cost, while being sufficiently simple and robust to be applied to complete climate and NWP models, where the physical parametrizations present in the source terms make it difficult to perform rigorous a posteriori error analysis.

- Thanks to the choice of 'stable' mixed order velocity-pressure pairs, after standard L^2 projection against test functions (chosen equal to the basis functions as in Direct Characteristic Galerkin scheme, see Morton *et al.* (1988)), and after integration by parts (where necessary), centred numerical fluxes were used to replace the (not-defined) traces of the solution at the inter-element boundaries, as in Bassi and Rebay (1997b). Moreover, the size of the final fully discrete problem was reduced by expressing the discrete velocity components in terms of the discrete free surface elevation from the momentum equations and then substituting the resulting expressions into the continuity equation, (as customary in SI methods, see e.g. Casulli and Greenspan (1984), Staniforth and Temperton (1986), Temperton and A.Staniforth (1987), Casulli and Cheng (1990), Casulli (1990), Casulli and Cattani (1994)), to obtain a single discrete Helmholtz equation in the free surface elevation unknown only, which takes the form of sparse (penta-diagonal in one-dimension, trideca-diagonal in two dimensions) block non symmetric linear system, which is solved via GMRES iteration.

- To fully exploit the power of semi-Lagrangian approach, the proposed SWE solver has been coupled with a SLDG passive tracers advection scheme in flux form (which is the extension of the scheme of Restelli *et al.* (2006)), whose properties in terms of conservation of constants (C-property, see e.g. Gross *et al.* (2002)) and compatibility with the continuity equation have been investigated . The p-adaptive treatment has been extended in independent way for each different passive tracer. As a result, the changes in the number of degrees of freedom are totally independent for each species, thus allowing to increase the accuracy for some specific variable without increasing the computational cost for other variables that do not need refinement.

- The proposed approach has been implemented in a modular FORTRAN95 code. The 1D implementation first developed has been used as template for the 2D implementation on Cartesian meshes.

The code has been used to perform a number of tests in order to analyse the stability and accuracy properties of the novel SISLDG method. Numerical results in the framework of one dimensional test cases prove that proposed the method captures accurately and effectively the main features of linear gravity and inertial gravity waves, as well as reproduces correct solutions in non-linear open channel flow tests and in all rarefaction Riemann problem. The effectiveness of the SISLDG method is also demonstrated by numerical results obtained at high Courant numbers and with automatic choice of the local approximation degree.

Numerical results in the framework of two-dimensional test cases show the effectiveness of the p-adaptivity strategy employed, as in the test of Smolarkiewicz, as well as the ability of well capture gravity waves also in two dimensions. Moreover the SLDG discretization for the advection has been already tested on a vector of an arbitrary number of tracers.

Chapter 1

The rotating shallow water equations

In this chapter the basics of the physical model employed in this thesis are summarized, namely the shallow water equations (SWE). SWE contain all the horizontal operators required in an atmospheric model and thus represent a good first test for newly proposed methods for atmospheric models.

1.1 Model background

For this model consider a sheet of fluid with constant and uniform density ρ (see Pedlosky (1987)). The height of the surface of the fluid above the reference level $z = 0$ is $\eta(x, y, t)$. We model the gravity body force as $\boldsymbol{g}\eta$ with atmosphere and ocean in mind: $\boldsymbol{g}$ is a vector directed perpendicular to the $z = 0$ surface, or $\boldsymbol{g}$ can be said to be anti-parallel to the vertical axis. The rotation axis of the fluid is the $z-$axis in this model. In this case the Coriolis parameter is $f = 2\Omega sin\varphi$, φ being the latitude, since $\boldsymbol{\Omega} = \Omega\boldsymbol{k}$. the rigid bottom is defined by the surface $z = b(x, y)$. Hence the total depth of the fluid is given by $h(x, y, t) = \eta(x, y, t) - b(x, y)$, see fig. 1.1. The velocity has components u, v, w in the x, y and z axis respectively. Though the pressure of the fluid surface can be arbitrarily imposed, for this model it will be assumed to be constant. Lastly the fluid is assumed *inviscid*, in other words, only the motions for which viscosity is not important are considered. In this model, because the depth of the fluid, $h(x, y, t) = \eta(x, y, t) - b(x, y)$, varies over time or space, let H be the average

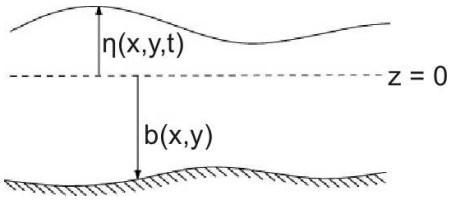


Figure 1.1: Notation for the shallow flow over a non-flat bottom.

depth of the fluid; H characterizes the vertical scale of the motion also. Let L be the characteristic horizontal scale for the motion. Then a fundamental condition which will characterize shallow water theory will be the thin sheet assumption

$$\delta := \frac{H}{L} \ll 1 \tag{1.1}$$

which is also called hydrostatic approximation with long wavelengths (see Gill (1987))
.

The shallow water model contains several of the important dynamical features of the atmosphere and ocean while being simple enough to be easily understood: therefore it represents usually the starting point for the development of new dynamical cores.

The major physical difference between this model and the reality is the absence of density stratification that is present in the real fluid such as earth's atmosphere or oceans. The hydrostatic approximation also allows ρ to vary with z, but we will consider ρ constant for this model. Recalling the equation of motion for rotating fluids (Haltiner and Williams (1980)), we have:

$$\frac{d\boldsymbol{u}}{dt} = -\frac{1}{\rho}\boldsymbol{\nabla} p - 2\boldsymbol{\Omega} \times \boldsymbol{u} + \boldsymbol{g} + \boldsymbol{F} \tag{1.2}$$

where $\frac{d}{dt}$ is the Lagrangian derivative:

$$\frac{d}{dt} = \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y} = \frac{\partial}{\partial t} + \boldsymbol{u}_H \cdot \boldsymbol{\nabla}_H,$$

$\boldsymbol{g}$ is the sum of gravitational and centrifugal forces per unit mass, while $\boldsymbol{F}$ is the force (per unit mass) due to friction. In the shallow water model we will be neglecting both $\boldsymbol{g}$ (except in the vertical direction) and $\boldsymbol{F}$ because under the assumption of shallow water theory they are much smaller in magnitude than the Coriolis forces. Thus in our assumptions, the momentum equation reduces to:

$$\frac{d\boldsymbol{u}}{dt} = -\frac{1}{\rho}\boldsymbol{\nabla} p - 2\boldsymbol{\Omega} \times \boldsymbol{u} - g\boldsymbol{k} \tag{1.3}$$

1.2 The rotating shallow water equations in primitive variables: advective form

There are several formulations in which the shallow water equations can be written: primitive variable formulation, vorticity divergence formulation or formulations using stream function and velocity potential (see, for example, Haltiner and Williams (1980)).

It is well known that straightforward finite-difference and finite-element discretizations of the shallow water equations, written in their primitive $(u - v)$ form, can lead to energy propagation in the wrong direction for the small scales (Schoenstadt (1980), Williams (1981), Coté *et al.* (1990)). This usually manifests itself as noise at the smallest scales and reduced accuracy. Two solutions to this problem have been proposed in the past literature. The first solution is to define the dependent variables on grids that are staggered with respect to one other (Arakawa (1966)), while the second is to use the governing equations in their differentiated (vorticity-divergence) form (Staniforth and Mitchell (1977), Williams and Zienkiewicz (1981)).

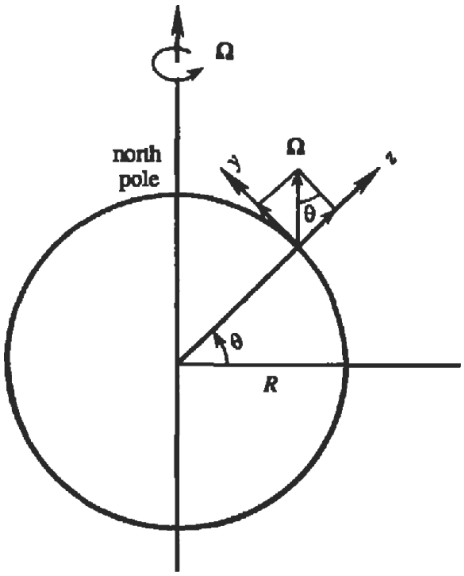


Figure 1.2: Local Cartesian coordinates. The x-axis is into the plane of the paper.

Since we are going to choose a semi-Lagrangian treatment of the advection together with, moreover, a suitable mixed order discontinuous finite element formulation, we will be able to use the primitive variable form of the shallow water governing equations, without have problems of noise at smallest scales (as showed in chap 6) . Therefore in the following the derivation of shallow water equations in primitive variable form is summarized.

There are different ways in which shallow water equations can be derived, for example via asymptotic expansions (Stoker (1957)) or via depth integration (Gill (1987)). Now we follow this second way.

We start by notice that the assumption of incompressibility and constant density decouples the dynamics from the thermodynamics (see e.g. Pedlosky (1987)); moreover the equation of mass conservation reduces to

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0.$$

Now before to pass to an estimate of the order of magnitude of various terms in the governing equations let us expand it in component form, hence let us fix a frame of reference .

Large-scale geophysical flow problems should be solved using spherical polar coordinates. If, however, the horizontal length scales are much smaller than the radius of the earth ($6371km$), then the curvature of the earth can be ignored, and the motion can be studied by adopting a *local* Cartesian system on a tangent plane (see fig. 1.2).

On this plane we take an xyz coordinate system, with x increasing eastward, y northward, and z upward. The corresponding velocity components are u (eastward), v (northward), and w (upward).

Then in this coordinate system the continuity equation takes the form:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{1.4}$$

Now the first two terms of this equation are of order $\frac{U}{L}$, where U can be considered the characteristic scale for the horizontal velocity. It the follows that the scale for the vertical velocity (W) is smaller than or equal to the order $\delta \cdot U$. This represents an upper bound for the vertical velocity, and it can be smaller than order $\delta \cdot U$ if there is cancellation between $\frac{\partial u}{\partial x}$ and $\frac{\partial v}{\partial y}$. In conclusion, since by the thin sheet assumption $\delta \ll 1$, the continuity equation states:

$$W \ll U \tag{1.5}$$

Regarding the momentum equation 1.3, we start by rewriting the Coriolis term. The earth rotates at a rate

$$\Omega = 2\pi rad/day = 0.73 \times 10^{-4} s^{-1}$$

around the polar axis, in an counterclockwise sense looking from above the north pole. From fig, 1.2, the components of angular velocity of the earth $\mathbf{\Omega}$ in the local Cartesian system are:

$$\begin{aligned}\Omega_x &= 0\\ \Omega_y &= \Omega cos\theta\\ \omega_z &= \Omega sin\theta\end{aligned}$$

then the Coriolis force (per unit mass) takes the form:

$$2\mathbf{\Omega} \times \boldsymbol{u} = 2\Omega \left[(wcos\theta - vsin\theta)\boldsymbol{i} + usin\theta\boldsymbol{j} - ucos\theta\boldsymbol{k}\right]$$

In the term multiplied by $\boldsymbol{i}$ we can use the condition $wcos\theta \ll vsin\theta$, that directly follows from (1.5). Then the three components of the Coriolis force (per unit mass) reduce to

$$\begin{aligned}(2\mathbf{\Omega} \times \boldsymbol{u})_x &= -(2\Omega sin\theta)v = -fv\\ (2\mathbf{\Omega} \times \boldsymbol{u})_y &= (2\Omega sin\theta)v = fu\\ (2\mathbf{\Omega} \times \boldsymbol{u})_z &= -(2\Omega cos\theta)u\end{aligned}$$

where we have defined the Coriolis parameter

$$f := 2\Omega sin\theta$$

The vertical component of the Coriolis force, namely $2\Omega cos\theta$, is generally negligible compared to the dominant terms in the vertical equation of motion, therefore the momentum equation (1.3), in component form, reduces to:

$$\begin{array}{ccccccccccccc}
\frac{\partial u}{\partial t} & + & u\frac{\partial u}{\partial x} & + & v\frac{\partial u}{\partial y} & + & w\frac{\partial u}{\partial z} & - & fv & = & -\frac{1}{\rho}\frac{\partial p}{\partial x} & & \\
\frac{U}{T} & & \frac{U^2}{L} & & \frac{U^2}{L} & & \frac{UW}{H} & & fU & & \frac{P}{\rho L} & & \\
\frac{\partial v}{\partial t} & + & u\frac{\partial v}{\partial x} & + & v\frac{\partial v}{\partial y} & + & w\frac{\partial v}{\partial z} & - & fv & = & -\frac{1}{\rho}\frac{\partial p}{\partial y} & & \\
\frac{U}{T} & & \frac{U^2}{L} & & \frac{U^2}{L} & & \frac{UW}{H} & & fU & & \frac{P}{\rho L} & & \\
\frac{\partial w}{\partial t} & + & u\frac{\partial w}{\partial x} & + & v\frac{\partial w}{\partial y} & + & w\frac{\partial w}{\partial z} & & & = & -\frac{1}{\rho}\frac{\partial p}{\partial z} & - & g \\
\frac{W}{T} & & \frac{UW}{L} & & \frac{UW}{L} & & \frac{W^2}{H} & & & & \frac{P}{\rho H} & & g
\end{array}$$

where each term has the order of magnitude written immediately below it in terms of the characteristic scales where T is the characteristic scale for time and P is the characteristic scale for the pressure field.

From this scale analysis and from the incompressibility condition (1.4), the last equation reduces to the hydrostatic approximation

$$\frac{\partial p}{\partial z} = -\rho g + O(\delta^2)$$

from which follows that the pressure p can be written as :

$$p(x, y, z, t) = -\rho g z + \pi(x, y, t)$$

or, in particular (see fig. 1.1)

$$p(x, y, z, t) = \rho g(-z + \eta(x, y, t)) + p_0,$$

(p_0 being the uniform ambient pressure on the free surface) then the horizontal pressure gradient is independent of z.

As a result, if the right hand side of the horizontal momentum equation is independent of z, then the left hand side also must be independent of z, therefore:

$$\frac{\partial u}{\partial z} = \frac{\partial v}{\partial z} = 0,$$

hence the horizontal momentum equations reduce to :

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -g\frac{\partial \eta}{\partial x} \tag{1.6}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -g\frac{\partial \eta}{\partial y} \tag{1.7}$$

Finally let us rewrite the continuity equation (1.4) as :

$$\frac{\partial w}{\partial z} = -\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)$$

then, integrating in dx from the bottom ($z = -b(x, y)$) up to the free surface ($z = \eta(x, y, t)$), we have

$$w(\eta(x, y, t)) = w(b(x, y)) - \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)(\eta - b)$$

. Finally the boundary conditions of no normal flow at the bottom and at the free surface require:

$$w(x, y, \eta(x, y, t)) = \frac{\partial \eta}{\partial t} + u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} \tag{1.8}$$

$$w(x, y, b(x, y)) = u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} \tag{1.9}$$

that inserted in the previous equation gives:

$$\frac{\partial \eta}{\partial t} + \frac{\partial (u(\eta - b))}{\partial x} + \frac{\partial (v(\eta - b))}{\partial y} = 0$$

In conclusion we derived the rotating shallow water equation system governing the unknowns η, u, v :

$$\begin{aligned} &\frac{\partial \eta}{\partial t} + \frac{\partial (uh)}{\partial x} + \frac{\partial (vh)}{\partial y} = 0 \\ &\frac{Du}{Dt} = -g\frac{\partial \eta}{\partial x} + fv \\ &\frac{Dv}{Dt} = -g\frac{\partial \eta}{\partial y} - fu \end{aligned} \tag{1.10}$$

where $\frac{D}{Dt}$ is the Lagrangian derivative

$$\frac{D}{Dt} := \frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y}$$

Chapter 2

The discontinuous Galerkin method

In this chapter we review the basic formulation of the Discontinuous Galerkin (DG) method for the solution of partial differential equations. As prototype problem the solution of the Laplace equation is considered. In sect. 2.1 a brief introduction to the key features of DG methods is outlined, then in sect 2.2 the discontinuous finite element spaces are defined, for which proper bases are illustrated in sects. 2.3, 2.4. Finally the DG technique is illustrated onto a test problem given by the Laplace equation in section 2.5

2.1 Introduction

Discontinuous Galerkin methods are finite element methods based on completely discontinuous finite element spaces, i.e. allowing for discontinuous fields at the element interfaces of the discretization. These methods combine different features commonly associated to finite element and to finite volume methods. As in classical finite element method, in fact, accuracy is obtained by means of high-order polynomial approximation within an element rather than by wide stencils as in the case of finite volume schemes. On the other hand DG methods rely on the introduction of suitable numerical fluxes (to approximate the traces of the functions on the inter-element interfaces of the discretization), like in finite volume schemes. In other words the order of discontinuous Galerkin methods, applied to problems with regular solutions, depends on the degree of the approximating polynomials only, which can easily be increased, dramatically simplifying the use of high order methods both on structured and unstructured meshes. Furthermore the stencil of most discontinuous Galerkin schemes is minimal, in the sense that each element communicates only with its direct neighbours, this compactness property of DG methods having clear advantages in parallelization, which does not require additional element layers at partition boundaries. In addition to this, the communication at element interfaces is identical for any order of the method which simplifies the use of methods of differing orders in adjacent elements. This flexibility property of DG methods allows for the variation of the order the numerical scheme over the computational domain, dramatically simplifying the implementation of p-adaptivity strategies.

We start the presentation of DG methods from the diffusion equation since its discretization represents the prototype for the SISLDG technique we will develop in chapter (5.3) for the SWE (4.1). Even if it may appear surprising that we took advantage from the discretization of an elliptic problem to study an hyperbolic one, this is not the case since, at discrete level, the two problems share many aspects, especially if we consider that, in order to reduce the computational effort, we have in mind to perform the substitution, again at discrete level, of velocity degree of freedom in terms of free surface elevation into the continuity equation, to end up with a discrete Helmholtz equation in the free surface elevation only.

2.2 Discontinuous finite element spaces

Equations (4.1) will be solved on a domain which is not an arbitrary open bounded connected subset of $\mathbb{R}^2$, but has the form $\Omega = (a,b)\times(c,d)$ with appropriate initial and boundary conditions. The domain Ω is partitioned in N non overlapping quadrilateral elements $K_I, I = 1,\dots,N$ whose width is denoted by $(\Delta x_I, \Delta y_I)$ and such that $\Omega = \bigcup_{I=1}^{N} K_I$. We denote the domain partition as $\mathcal{T}_h = \{K_I : I = 1,\dots,N\}$, where $h = max_I(diamK_I)$. The center of the generic element K_I is denoted by (x_I, y_I) while $(x_{I\pm1/2}, y_{I\pm1/2})$ denote its corners. The four edges of the element K_I are labelled as $e_{1,I}, e_{2,I}, e_{3,I}, e_{4,I}$ (see figure). It is immediate that each K_I is the image of the master element $K = [-1,1]\times[-1,1]$ via the affine local map $\boldsymbol{F}_I$, such that $x = F_{I,1}(\xi) = \xi\Delta x_I/2 + x_I, y = F_{I,2}(\eta) = \eta\Delta y_I/2 + y_I$. For a non-negative integer p, we denote by $\mathcal{Q}_p$ the set of all polynomials of degree less or equal to p in each coordinate on K. We will also define

$$\mathcal{Q}_p(K_I) = \left\{w : w = v\circ\boldsymbol{F}_I^{-1}, \quad v\in\mathcal{Q}_p\right\}$$

For each I, we denote by the non-negative integer p_I the *local* polynomial degree on K_I and by the non-negative integer s_I the *local* Sobolev index and we set $\boldsymbol{p} = \{p_I : I = 1,\dots,N\}$, $\boldsymbol{s} = \{s_I : I = 1,\dots,N\}$ and $\boldsymbol{F} = \{F_I : I = 1,\dots,N\}$, respectively. We then consider the finite element space:

$$V_h^{DG} \equiv S^{\boldsymbol{p}}(\Omega,\mathcal{T}_h,\boldsymbol{F}) = \left\{v\in L^2(\Omega) : v|_{K_I}\circ F_I^{-1}\in\mathcal{Q}_{p_I} \quad I = 1,\dots,N\right\}$$

Given the partition $\mathcal{T}_h$, we associate to it the broken Sobolev space of composite index $\boldsymbol{s}$, defined by:

$$H^{\boldsymbol{s}}(\Omega,\mathcal{T}_h) := \left\{v\in L^2(\Omega) : v|_{K_I}\circ F_I^{-1}\in H^{s_I}(K) \quad I = 1,\dots,N\right\},$$

equipped with the broken Sobolev norm

$$\|v\|_{\boldsymbol{s},\mathcal{I}_h} = \Big(\sum_{I=1}^{N}\|v\|^2_{H^{s_I}(K_I)}\Big)^{\frac{1}{2}}.$$

When $s_I = s$ for all $I = 1,\dots,N$, we shall write

$$H^s(\Omega,\mathcal{T}_h), \quad \|v\|_{s,\mathcal{T}_h}.$$

We will denote by $\mathcal{E}_h$ the set of all element boundary edges, which is naturally decomposed as $\mathcal{E}_h = \mathcal{E}_{h,int}\cap\mathcal{E}_b$, where $\mathcal{E}_{h,int}$ denotes the element boundary edges that belong to (the interior of) Ω and $\mathcal{E}_b = \partial(\Omega)$. Functions in $H^{\boldsymbol{s}}(\Omega,\mathcal{T}_h)$ are in general discontinuous across element boundaries. Thus, it is convenient to define jump and average

operators on each inter-element edge. For a given edge $e \in \mathcal{E}_{h,int}$ there exist two elements $K, K' \in \mathcal{T}_h$ such that $e = \partial K \cap \partial K'$, hence for each point $\boldsymbol{x} \in e$ and $v_h \in V_h^{DG}$, being $\boldsymbol{n}_{\partial K,e}$ the restriction of $\boldsymbol{n}_{\partial K}$ on the edge $e \subset \partial K$ we can thus define

$$\{\{v_h\}\}(\boldsymbol{x}) = \frac{1}{2}\Big(v_h|_K(\boldsymbol{x}) + v_h|_{K'}(\boldsymbol{x})\Big) \tag{2.1}$$

$$[[v_h]](\boldsymbol{x}) = v_h|_K(\boldsymbol{x})\boldsymbol{n}_{e,\partial K} + v_h|_{K'}(\boldsymbol{x})\boldsymbol{n}_{e,\partial K'}. \tag{2.2}$$

Notice that these definitions, following Cockburn and Shu (1991) and Arnold *et al.* (2002), in particular the definition of jumps, are symmetric with respect to the two elements K, K'sharing the boundary point at which the jump is defined .

While if $e \in \mathcal{E}_{h,b}$ then there exists a unique $K \in \mathcal{T}_h$ such that $e \in \partial K$ and hence we set

$$\{\{v_h\}\}(\boldsymbol{x}) = v_h|_K(\boldsymbol{x}) \tag{2.3}$$

$$[[v_h]](\boldsymbol{x}) = v_h|_K(\boldsymbol{x})\boldsymbol{n}_{e,\partial K}. \tag{2.4}$$

Finally we define mesh related (or broken) gradient and divergence operators. We define broken operators by restrictions to each element $\kappa \in \mathcal{T}_h$ as follows.

The broken gradient operator $\boldsymbol{\nabla}_h : H^1(\mathcal{T}_h) \to \left[L^2(\mathcal{T}_h)\right]^2$ is defined by:

$$(\boldsymbol{\nabla}_h v)|_\kappa := \nabla(v|_\kappa), \quad \kappa \in \mathcal{T}_h$$

for $v \in H^1(\mathcal{T}_h)$, where $(\nabla v)_i := \partial_{x_i} v, i = 1, 2$.

The broken divergence operator $\boldsymbol{\nabla}_h \cdot : \left[H^1(\mathcal{T}_h)\right]^2 \to L^2(\mathcal{T}_h)$ is defined by:

$$(\boldsymbol{\nabla}_h \cdot \tau)|_\kappa := \nabla \cdot (\tau|_\kappa) \quad \kappa \in \mathcal{T}_h$$

for $\tau \in \left[H^1(\mathcal{T}_h)\right]^2$, where $(\nabla \cdot \tau) := \sum_{1 \le i \le d} \partial_{x_i} \tau_i$

As usual in finite element methods, the choice of the finite-dimensional subspaces for the approximation of the velocity-pressure pair is a crucial issue. In our context, the role of the pressure will be played by the free surface elevation. In that sense, we follow results for incompressible flows (for example see Toselli (2002) and Schötzau *et al.* (2003) for the Stokes problem) where has been proven that $\mathcal{Q}_p - \mathcal{Q}_{p-1}$ pairs are not only *inf—sup* stable with respect to the mesh-size, but they are also uniformly stable with respect to the polynomial degree p. We will denote the bases of the local polynomial spaces as $\{\psi_{I,l}\}_{l=1}^{(p_I+1)^2}$. In principle, either Lagrangian or hierarchical bases could be employed. We will work mostly with hierarchical bases because they provide a natural environment for the implementation of a $p-$adaptation algorithm, see for example Zienkiewicz *et al.* (1983).

2.3 Hierarchical bases vs. Lagrangian bases in one dimension

In this section the concepts of *modal* and *nodal* polynomial expansions are introduced and the associated *hierarchical* and *Lagrangian* bases illustrated. Following Karniadalis and Sherwin (2005), to illustrate the difference between a *modal* and a *nodal* polynomial expansion we introduce three expansion sets denoted by $\Phi_k^A(x)$, $\Phi_k^B(x)$ and

$\Phi_k^C(x)$, $k = 0, \ldots, p$ in the region $\Omega_{ref} = [-1, 1]$. All of these sets are a basis for the set of polynomials up to order p on Ω_{ref} and are defined as:

$$\Phi_k^A(x) = x^k, \quad k = 0, \ldots, p$$
$$\Phi_k^B(x) = \frac{\prod_{i=0, i\neq k}^{p}(x - x_i)}{\prod_{i=0, i\neq k}^{p}(x_k - x_i)}, \quad k = 0, \ldots, p$$
$$\Phi_k^C(x) = L_k(x), \quad k = 0, \ldots, p$$

where $x_0, \ldots, x_p$ are $p+1$ fixed nodes on Ω_{ref} and $L_k(x)$ is the $k-$th Legendre polynomial, defined recursively by the three-term recurrence relation:

$$L_{k+1}(x) = \frac{2k+1}{k+1} x L_k(x) - \frac{k}{k+1} L_{k-1}(x), \quad k = 1, 2, \ldots$$
$$L_0(x) = 1, \quad L_1(x) = x. \tag{2.5}$$

The shape of these functions can be seen in fig 2.1-2.3.

The first basis simply increases the order of x in a monomial fashion and is called hierarchical since the basis of cardinality $p-1$ is contained within the basis of order p. There is a notion of hierarchy in the sense that higher-order bases are built from the lower order bases, i.e.

$$\{\Phi_k^A\}_{k=0}^{p-1} \subset \{\Phi_k^A\}_{k=0}^{p}$$

The second polynomial basis $\{\Phi_k^B\}_{k=0}^{p}$ is a Lagrange polynomial basis which is identified by $p+1$ nodal points $\{x_j\}_{j=0}^{p}$ which are chosen beforehand and could be, for example, equispaced in the interval Ω_{ref}. The Lagrange polynomials $\{\Phi_k^B\}_{k=0}^{p}$ build a non-hierarchical bases for $\mathcal{Q}_p(\Omega_{ref})$ in the sense that

$$\{\Phi_k^B\}_{k=0}^{p-1} \not\subset \{\Phi_k^B\}_{k=0}^{p}$$

because they are $p+1$ polynomials of order p, but instead they are called a *nodal* basis in the sense that the expansion coefficients of a given function $f \in \mathcal{Q}_p(\Omega_{ref})$ over the basis $\{\Phi_k^B\}_{k=0}^{p}$ can be interpreted in terms of the approximate values of f at the *nodes* $\{x_j\}_{j=0}^{p}$. This follows directly from the property (consequence of the definition of Φ_k^B) $\Phi_k^B(x_j) = \delta_{k,j}$ where $\delta_{k,j}$ represents the Kronecker delta:

$$f(x_j) = \sum_{k=0}^{p} \hat{f}_k \Phi_k^B(x_j) = \sum_{k=0}^{p} \hat{f}_k \delta_{kj} = \hat{f}_j$$

Finally the Legendre polynomial basis $\{\Phi_k^C\}_{k=0}^{p}$ is a hierarchical basis for $\mathcal{Q}_p(\Omega_{ref})$ that is also called *modal* since the expansion coefficients over such a basis have the physical interpretation of frequency components of the function to be represented over the ϕ_k^B. Moreover the Legendre polynomial basis $\{\Phi_k^C\}_{k=0}^{p}$, by definition, has another important property that is the orthogonality (in the L^2 inner product) :

$$(L_p, L_q)_{L^2} := \int_{-1}^{1} L_p(x) L_q(x) dx = \frac{2}{2p+1} \delta_{pq}.$$

Notice that this orthogonality property has important numerical implications for the Galerkin method since not only makes diagonal all the mass matrices but also because it gives in general improved conditioning of the discrete matrices (see Karniadalis and Sherwin (2005) or Zienkiewicz *et al.* (1983) for details).

Other examples of hierarchical bases are given by the wavelets basis, where the concept of hierarchy is naturally contained in the notion of Multi-Resolution-Analysis on which rely all definitions of wavelets.

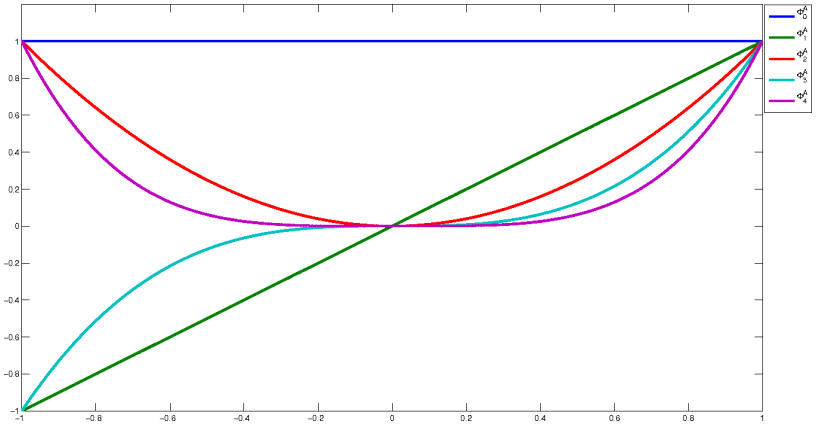


Figure 2.1: Monomial bases functions Φ_k^A for $k = 0, 1, 2, 3, 4$.

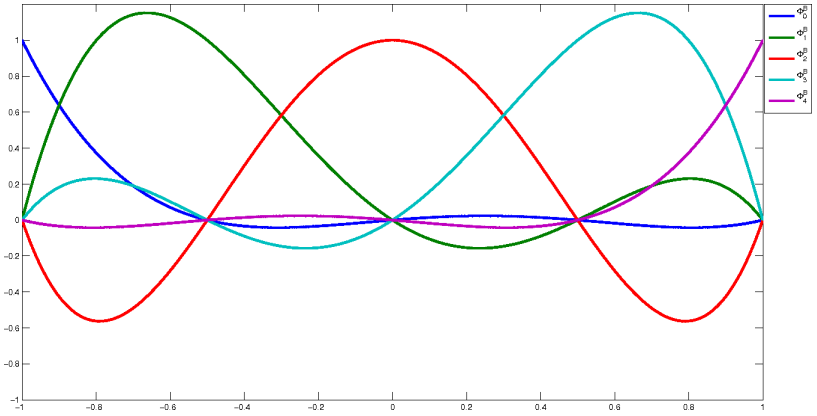


Figure 2.2: Lagrange polynomial bases functions Φ_k^B for $k = 0, 1, 2, 3, 4$.

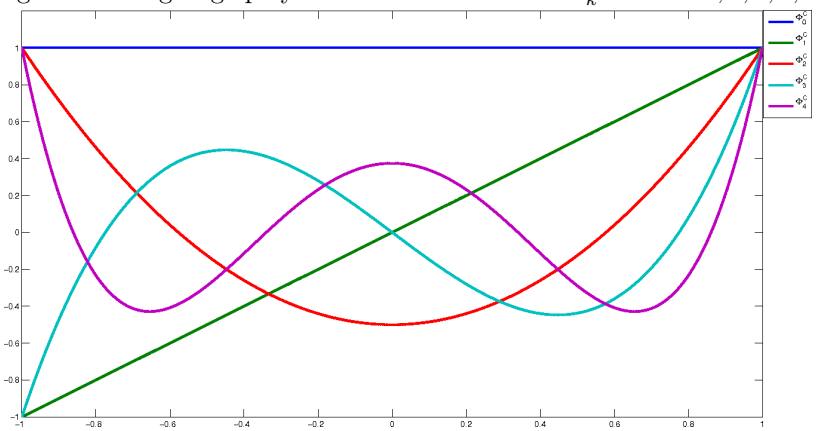


Figure 2.3: Legendre polynomial bases functions Φ_k^C for $k = 0, 1, 2, 3, 4$.

2.4 Tensor product bases

The extension to higher dimension within quadrilateral regions is relatively straightforward, if rather more involved than the one-dimensional case discussed in section 2.3. we start by defining the two-dimensional reference element (on which each element $K_I \in \mathcal{T}_h$ will be remapped):

$$\Omega_{ref} = \{-1 \leq \xi_1, \xi_2 \leq 1\}.$$

Since this reference element is trivially defined by a standard Cartesian coordinate system, the most natural and straightforward way to construct the two-dimensional basis is by taking a product of the one-dimensional basis, which can be thought of as one-dimensional tensors. Hence tensor basis in two dimensions we shall denote the polynomial bases $\{\varphi_{kl}(\xi_1, \xi_2)\}_{kl}$ (where ξ_1, ξ_2 are the standard Cartesian coordinates) that can be defined in terms of product of one-dimensional functions or tensor product, for example :

$$\varphi_{kl}(\xi_1, \xi_2) = \Phi_k(\xi_1)\Phi_l(\xi_2)$$

Notice that this type of extension may be applied equally well to either the modal- or nodal- type basis, indeed no distinction has been made between these two forms in this section.

2.5 DG methods for the diffusion equation

Following Arnold *et al.* (2002) we consider the elliptic model problem:

$$-\Delta u = f \quad \text{in} \quad \Omega, \qquad u = 0 \quad \text{on} \quad \partial\Omega.$$

Departing from Arnold *et al.* (2002) we consider in this section periodic boundary conditions only, in order to simplify the presentation. As a results if we call $\Gamma^{int} = \cup_{e \in \mathcal{E}_h^{int}} e$, we have that $\partial\Omega \subset \Gamma^{int}$. The problem is rewritten as first order system trough the introduction of the auxiliary variable $\boldsymbol{\sigma}$:

$$\boldsymbol{\sigma} = \boldsymbol{\nabla} u, \qquad -\boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = f, \qquad u = 0 \quad on \quad \partial\Omega.$$

Now assuming $u \in H^2(\mathcal{T}_h)$ and $\boldsymbol{\sigma} \in [H^1(\mathcal{T}_h)]^2$ we multiply first and second equation by test functions $\boldsymbol{\tau} \in [H^1(\mathcal{T}_h)]^2$ and $v \in H^1(\mathcal{T}_h)$, respectively, integrate over a element $\kappa \in \mathcal{T}_h$, integrate by parts and finally sum over all elements $\kappa \in \mathcal{T}_h$. Thus we get the system flux formulation:

$$\int_\Omega \boldsymbol{\sigma} \cdot \boldsymbol{\tau} d\boldsymbol{x} = -\int_\Omega u \boldsymbol{\nabla}_h \cdot \boldsymbol{\tau} d\boldsymbol{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \hat{u}\boldsymbol{\tau} \cdot \boldsymbol{n} ds \quad \forall \boldsymbol{\tau} \in \left[H^1(\mathcal{T}_h)\right]^2, \quad (2.6)$$

$$\int_\kappa \boldsymbol{\sigma} \cdot \boldsymbol{\nabla}_h v d\boldsymbol{x} = \int_\kappa f v d\boldsymbol{x} + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \hat{\boldsymbol{\sigma}} \cdot \boldsymbol{n} ds \quad \forall v \in H^1(\mathcal{T}_h). \quad (2.7)$$

where $\boldsymbol{n}$ is the unit outward normal vector to $\partial\kappa$. Notice that since u and $\boldsymbol{\sigma}$ may be discontinuous across inter-element faces $\partial\kappa \in \mathcal{E}_h^{int}$, they must be replaced by single valued numerical flux functions $\hat{u} : H^1(\mathcal{T}_h) \to T(\mathcal{T}_h)$ and $\hat{\boldsymbol{\sigma}} : H^2(\mathcal{T}_h) \times \left[H^1(\mathcal{T}_h)\right]^2 \to [T(\mathcal{T}_h)]^2$, where $T(\mathcal{T}_h) := \prod_{\kappa \in \mathcal{T}_h} L^2(\partial\kappa)$, which are approximations of u and $\boldsymbol{\sigma}$ on $\mathcal{E}_h^{int}$.

Depending on the particular choice of the numerical flux functions $\hat{u}$ and $\hat{\boldsymbol{\sigma}}$, several different DG discretizations can be derived, each with specific properties with respect to stability and accuracy.

Notice that equations (2.6)-(2.7) represent a first order system in u and σ with 3 as many unknowns as the original (scalar) problem in u. In order to reduce the problem in size, the auxiliary variable $\boldsymbol{\sigma}$ is usually eliminated to gain the so called primal formulation involving the primal variable u only (a second integration by parts in (2.6) is required):

find $u \in H^2(\mathcal{T}_h)$ such that

$$\hat{B}_h(u,v) = \int_\Omega f v d\boldsymbol{x} \quad \forall v \in H^2(\mathcal{T}_h),$$

where the bilinear form $\hat{B}_h(\cdot,\cdot) : H^2(\mathcal{T}_h) \times H^2(\mathcal{T}_h) \to \mathbb{R}$ is defined by

$$\hat{B}_h(u,v) := \int_\Omega \boldsymbol{\nabla}_h u \cdot \boldsymbol{\nabla}_h v d\boldsymbol{x} - \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} \hat{\boldsymbol{\sigma}} \cdot \boldsymbol{n} ds + \sum_{\kappa \in \mathcal{T}_h} \int_{\partial\kappa} (\hat{u} - u)\boldsymbol{n} \cdot \boldsymbol{\nabla}_h v ds. \quad (2.8)$$

Notice that the bilinear form has been denoted by $\hat{B}_h$ (and not by B_h) as it includes the (still unspecified) numerical fluxes $\hat{u}$ and $\hat{\boldsymbol{\sigma}}$. Furthermore, $\hat{B}_h$ includes - through the specification of $\hat{u}$ and $\hat{\boldsymbol{\sigma}}$ on the boundary $\partial\Omega$- all boundary data terms.

Finally note that $\hat{B}_h$ in (2.8) is an element-based bilinear form, i.e. it is given in terms of $\sum_\kappa \int_{\partial\kappa}$. This means that each interior face $e \in \mathcal{E}_h^{int}$ occurs twice in the sum over all elements κ, (once in $\int_{\partial\kappa}$ and once in $\int_{\partial\kappa'}$ for $\kappa' \neq \kappa$ end $e = \partial k \cap \partial k'$). Now we transfer the element the element-based bilinear form into a face-based bilinear form, i.e. we rewrite $\hat{B}_h$ in terms of $\int_{\Gamma^{int}}$, where each interior face occurs only once. After straightforward computations (see Hartmann (2008) for details) we get the following face-based primal form:

$$\begin{aligned} \hat{B}_h(u,v) &= \int_\Omega \boldsymbol{\nabla}_h u \cdot \boldsymbol{\nabla}_h v d\boldsymbol{x} + \int_{\Gamma^{int}} ([[\hat{u} - u]] \cdot \{\{\boldsymbol{\nabla}_h v\}\} - \{\{\hat{\boldsymbol{\sigma}}\}\} \cdot [[v]]) ds + \\ &+ \int_{\Gamma^{int}} (\{\{\hat{u} - u\}\}[[\boldsymbol{\nabla}_h v]] - [[\hat{\sigma}]]\{\{v\}\}) ds. \end{aligned} \quad (2.9)$$

Up to now the numerical fluxes $\hat{u}$ and $\hat{\boldsymbol{\sigma}}$ are still unspecified. Here some choices are written, defining some DG methods, viz. SIPG (Symmetric Interior Penalty Galerkin), NIPG (Non-symmetric Interior Penalty Galerkin) and BO(Baumann-Oden method):

$$\begin{aligned} \text{SIPG}: &\quad \hat{u}_h = \{\{u_h\}\}, &&\quad \hat{\boldsymbol{\sigma}} = \{\{\boldsymbol{\nabla}_h u_h\}\} - C_{IP}\frac{p^2}{h}[[u_h]] \\ \text{NIPG}: &\quad \hat{u}_h = \{\{u_h\}\} + \boldsymbol{n}_\kappa \cdot [[u_h]], &&\quad \hat{\boldsymbol{\sigma}} = \{\{\boldsymbol{\nabla}_h u_h\}\} - C_{IP}\frac{p^2}{h}[[u_h]] \\ \text{BO}: &\quad \hat{u}_h = \{\{u_h\}\} + \boldsymbol{n}_\kappa \cdot [[u_h]], &&\quad \hat{\boldsymbol{\sigma}} = \{\{\boldsymbol{\nabla}_h u_h\}\} \end{aligned}$$

Following again Arnold *et al.* (2002), the discretizations derived above are written in unified form as follows:

find $u_h \in V_h$ such that:

$$B_h(u_h, v_h) = F_h(v_h) \quad \forall v_h \in V_h,$$

where

$$\begin{aligned} B_h(u,v) &= \int_\Omega \boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v d\boldsymbol{x} + \\ &\int_{\Gamma^{int}} \theta[[u]] \cdot \{\{\boldsymbol{\nabla} v\}\} + (\delta[[u]] - \{\{\boldsymbol{\nabla} u\}\}) \cdot [[v]] ds. \end{aligned}$$

and the constants θ and δ are given by:

SIPG:	$\theta = -1,$	$\delta > 0$
NIPG:	$\theta = 1,$	$\delta > 0$
BO:	$\theta = 1,$	$\delta = 0$

In the following we investigate the experimental order of convergence in the L^2 norm of the SIPG ($\theta = -1$) and NIPG ($\theta = 1$) discretizations, where we choose the constant δ according the following (see Hartmann (2008)):

Theorem 1 *(A priori error estimate) Let $c \in H^{p+1}$ be the exact solution to Poisson's equation in Ω with Dirichlet homogeneous B.C. Furthermore let $c_h \in V_{h,p}^d$ be the solution to :*

$$B_h(c_h, v_h) = F_h(v_h) \quad \forall v_h \in V_{h,p}^d$$

where $B_h(c_h, v_h)$ is given as in (2.10)with $\theta = -1$ (SIPG) or $\theta = 1$(NIPG) and $\delta = C_{IP}\frac{p^2}{h}$, $C_{IP} > C_{IP}^0$ ($C_{IP}^0 = 0$ for NIPG, $C_{IP}^0 > 0$ for SIPG)

then

$$\|c - c_h\|_{L^2(\Omega)} \leq Ch^{p+1}|c|_{H^{p+1}(\Omega)} \quad \textit{for SIPG}$$

$$\|c - c_h\|_{L^2(\Omega)} \leq Ch^{p}|c|_{H^{p+1}(\Omega)} \quad \textit{for NIPG}$$

As model problem we considered the Poisson equation in one dimension with $\Omega = (a, b)$ and forcing function chosen so that the analytical solution is given by $u(x) = sin(2\pi\frac{x-a}{b-a})$. We imposed periodic boundary conditions. Figure 2.4-2.5 plots the L^2 norm of the error against the number of element used.

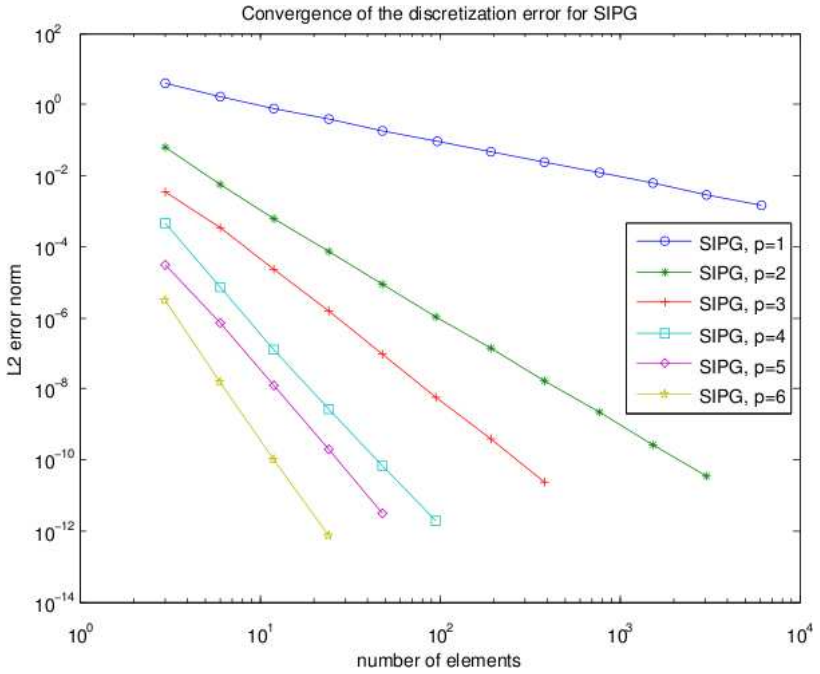


Figure 2.4: Order of convergence of SIPG in L^2 norm

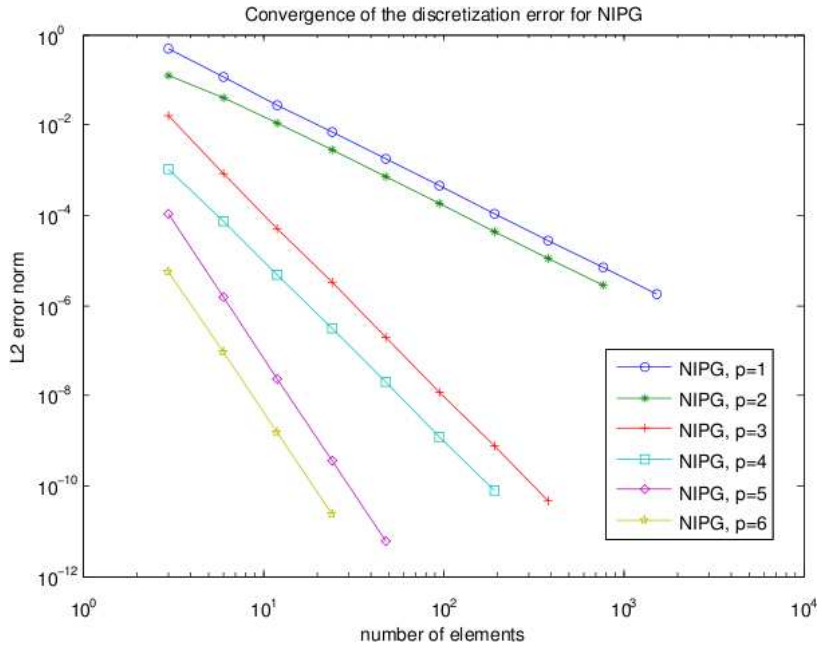


Figure 2.5: Order of convergence of NIPG in L^2 norm

Once we have computed the DG discretization of the diffusion operator, we used it also to integrate its time-dependent counterpart given by the heat equation: after DG space semidiscretization we get the following problem (matrix formulation):

$$M\frac{d\boldsymbol{c}}{dt}(t) + A\boldsymbol{c}(t) = \boldsymbol{f}(t)$$

where:

- M = mass matrix is diagonal
- A = stiffness matrix is block-tridiagonal
- $\boldsymbol{c}$ = vector of the modal coefficients of the approximate solution

A θ-method has implemented for the time integration:

$$M\frac{\boldsymbol{c}^{k+1} - \boldsymbol{c}^k}{\Delta t} + A\Big(\theta\boldsymbol{c}^{k+1} + (1-\theta)\boldsymbol{c}^k\Big) = \theta\boldsymbol{f}^{k+1} + (1-\theta)\boldsymbol{f}^k$$

where:

- $\theta = 0$ = Explicit Euler (EE)
- $\theta = 1$ = Implicit Euler (IE)
- $\theta = 0.5$ = Crank-Nicolson (CN)

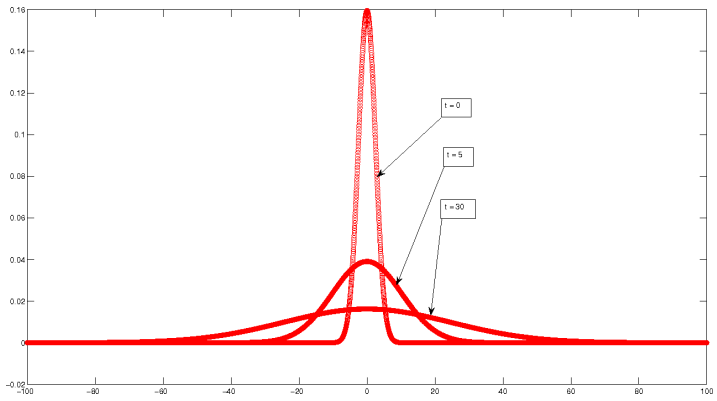


Figure 2.6: Computed solution of the heat equation in one dimension (SIPG discretization in space) at times $t = 0$, $t = 5s$, $t = 30s$.

Taking advantage from this well structured sparsity pattern of the DG space semidiscretization, two Krylov-family linear solver has been implemented where only the nonzero blocks of the stiffness matrix are stored:

- PCG for implicit SIPG time-step
- PmGMRES for implicit NIPG time-step

Results of the implicit time integration of homogeneous heat equation (SIPG) with Gaussian initial profile are collected in table 2.1 (where the exact solution has been computed analytically via convolution with the heat kernel on $\mathbb{R}$) for two different time stepping, i.e. Implicit Euler (IE), and Crank-Nicolson (CN), while a plot of the computed solution at successive times is given in figure 2.6.

Δt	$\|error(t=1)\|_{L^2} for IE$	$\|error(t=1)\|_{L^2} for CN$
4e-2	2.333e-4	9.047e-7
2e-2	1.173e-4	2.147e-7
1e-2	5.882e-5	4.780e-8
5e-3	2.945e-5	9.672e-9

Table 2.1: L^2 norm of the final error for IE and CN time integration coupled with high order SIPG space discretization with data: $a = -100, b = 100, n_{elem} = 400, t_0 = 0, T_{tot} = 1.0, poldegree = 8$

Chapter 3

The semi-Lagrangian time discretization technique

Since in this work the semi-Lagrangian technique is used to treat advective terms of the dynamical equations in meteorology, the advection equation only will be considered in this chapter. In particular in section 3.2 and 3.3 the two key steps of any semi-Lagrangian scheme are addressed, i.e. respectively, the foot of the trajectories computation and the advected quantities interpolation at such a points . Then two different forms of the advection equation are integrated in the context of a DG space discretization i.e. the quasilinear form (sect. 3.4), which will be used in our SISLDG method to discretize the advective terms in SWE, and the flux form, that has been used to design a conservative tracers advection scheme, which has been coupled with the SISLDG SWE solver.

3.1 Introduction

The numerical modelling of the advection is an ubiquitous issue in atmospheric problems, for a review see e.g. Rood (1987). The fundamental property of the advection equation that makes it more difficult to model than parabolic or elliptic equations is that this equation permits the formation and maintenance of fronts (shocks). The fact that numerical mechanisms that cause shock fronts are similar to the mechanisms that cause numerical instability makes it difficult to write absolutely stable advection schemes for long-time integrations.

Discretization schemes based on semi-Lagrangian treatment of advection have excited considerable interest in the past three decades for the efficient integration of weather forecast models, since they offer the promise of allowing larger time steps (with no loss of accuracy) than Eulerian-based advection schemes (whose time-step length is overly limited by considerations of stability); for a review see Staniforth and Coté (1991).

The key idea behind semi-Lagrangian time-integration is to combine the regular resolution property of Eulerian schemes and the enhanced stability property of Lagrangian ones, by following (backward in time) different sets of particles at each time step: the

sets of particles arriving exactly at the points of a regular Cartesian mesh at the end of the time step are tracked backward in time (see fig 3.1).

Therefore, independently of the type of spatial discretization one has in mind, any semi-Lagrangian method reduces the approximation of the advection equation to performing the two following key steps:

1. at a given time level, compute for each mesh point $\boldsymbol{x}$ (that will be a grid node in finite difference or a quadrature node or a element edge in Galerkin type methods) the foot $\boldsymbol{x}^*$ at time t^n of the trajectory of the particle arriving in $\boldsymbol{x}$ at time t^{n+1};

2. evaluate the advected variable at time level t^n at the point $\boldsymbol{x}^*$,that in general does not belong to the mesh, therefore interpolation is usually required at this stage.

In the following two sections, it will be discussed how the solution of each of these steps can be implemented so as to achieve the greatest accuracy and efficiency in the resulting numerical method.

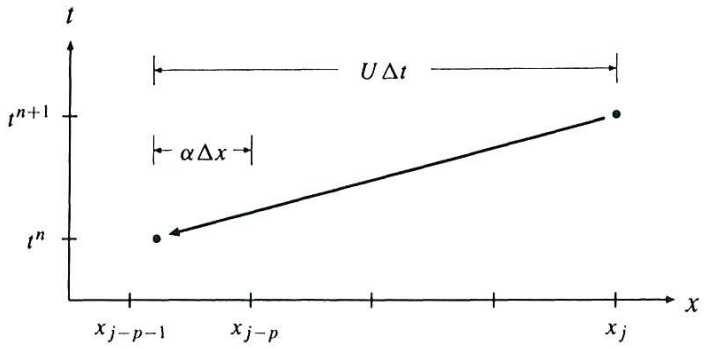


Figure 3.1: Schematic for two time level semi-Lagrangian advection.

Before to continue the discussion, let us remember that most of semi-Lagrangian schemes are based on discretizations over either two time levels (e.g. Bates and McDonald (1977), McDonald and Bates (1987)) or three time levels(e.g. Robert (1981), Robert (1982), Staniforth and Temperton (1986)). Two-time-level schemes have, in principle, several virtues: they are simpler to code, make fewer demands on computer memory (particularly important if we have in mind to combine them with a demanding space discretization like DG) and have no time computational modes. Furthermore they are potentially twice as efficient as three-time-level leapfrog based schemes, in fact these last schemes require time-steps half the size of two-time level ones in order to achieve the same level of time truncation error. Other advantages of two-time-level schemes over three-time level ones are showed in Temperton and A.Staniforth (1987).Therefore in the following we will concentrate on two-time-level semi-Lagrangian schemes.

3.2 Trajectory approximation techniques

Given a point $\boldsymbol{x} \in \Omega$, the problem of finding the point $\boldsymbol{x}^*$ such that a fluid particle placed in $\boldsymbol{x}^*$ at time t^n will be transported by the advective velocity field $\boldsymbol{u} : \Omega \times [0, T) \to \mathbb{R}^2$ to arrive exactly at point $\boldsymbol{x}$ at time t^{n+1}, is equivalent to find the solution $\boldsymbol{x}^* = \boldsymbol{X}(t^n; t^{n+1}, \boldsymbol{x})$ of the following backward Cauchy problem for a vector ODE:

$$\begin{cases} \frac{d}{dt}\boldsymbol{X}(t;t^{n+1},\boldsymbol{x}) = \boldsymbol{u}\Big(\boldsymbol{X}(t;t^{n+1},\boldsymbol{x}),t\Big) \\ \boldsymbol{X}(t^{n+1};t^{n+1},\boldsymbol{x}) = \boldsymbol{x} \end{cases} . \tag{3.1}$$

It is important to notice that the solution of (3.1) requires, in general, knowledge of the velocity field $\boldsymbol{u}$ at points which do not belong to the space-time computational mesh. For application to realistic models, this implies that some interpolation or in some cases, as far as the time dependence is concerned, some extrapolations are necessary.

Traditionally, three main strategies have been proposed and implemented for the approximation of the problem (3.1) of the particle trajectories computation:

- sub-stepping with explicit ODE solvers;
- fixed point iteration approach;
- Taylor expansion of the parametric representation for the trajectory.

The sub-stepping with explicit ODE solvers is perhaps the most natural way to tackle the problem, being (3.1) a system of ordinary differential equations. This trajectory method has been quite popular in the finite element and coastal modelling community, see e.g. Pironneau (1982), Casulli (1990), Quarteroni and Valli (1994), Miglio *et al.* (1999), although applications to atmospheric models have are also reported in the literature, see e.g. Bonaventura (2000), Giraldo (2000). Since in general trajectories are not straight, a key issue for this family of methods is the choice of the time sub-step $\Delta\tau = \frac{\Delta t}{N}$ (where Δt is the time we have to go back along the trajectory); furthermore, whenever the Courant number is larger than one, in order to guarantee that the approximated trajectories do not cross each other, a shorter time sub step has to be used in general for their approximation, (see e.g. for details Rosatti *et al.* (2005)) .

The simplest strategy one can imagine is to use a backward Euler scheme where, since (3.1) is non-linear, we linearise it in time by using the velocity at time t^n to compute $\boldsymbol{X}(t^n;t^{n+1})$; then (3.1) is discretized backward as:

$$\boldsymbol{x}^{(k-1)} = \boldsymbol{x}^{(k)} - \Delta\tau\boldsymbol{u}(\boldsymbol{x}^{(k)},t^n) \qquad k = N, N-1,\ldots,1. \tag{3.2}$$

An other possible option to go back along trajectories is given by more accurate Runge-Kutta schemes. If we set $s^k := t^{n+1} - k\Delta\tau$, for example, for second order accuracy one has to compute for $k = N, N-1,\ldots,1$

$$\begin{aligned} \boldsymbol{x}^{(k-\frac{1}{2})} &= \boldsymbol{x}^{(k)} - \Delta\tau\boldsymbol{u}(\boldsymbol{x}^{(k)},s^k) \\ \boldsymbol{x}^{(k-1)} &= \boldsymbol{x}^{(k)} - \frac{\Delta\tau}{2}\left[\boldsymbol{u}(\boldsymbol{x}^{(k-\frac{1}{2})},s^k-\Delta\tau) + \boldsymbol{u}(\boldsymbol{x}^{(k)},s^k)\right] \end{aligned} \tag{3.3}$$

or, for third order accuracy

$$\begin{aligned} \boldsymbol{x}^{(k-\frac{1}{3})} &= \boldsymbol{x}^{(k)} - \frac{\Delta\tau}{3}\boldsymbol{u}(\boldsymbol{x}^{(k)},s^k) \\ \boldsymbol{x}^{(k-\frac{2}{3})} &= \boldsymbol{x}^{(k)} - \frac{2\Delta\tau}{3}\boldsymbol{u}\left(\boldsymbol{x}^{(k-\frac{1}{3})},s^k-\frac{\Delta\tau}{3}\right) \\ \boldsymbol{x}^{(k-1)} &= \boldsymbol{x}^{(k)} - \frac{\Delta\tau}{4}\left[3\boldsymbol{u}(\boldsymbol{x}^{(k-\frac{2}{3})},s^k-\frac{2\Delta\tau}{3}) + \boldsymbol{u}(\boldsymbol{x}^{(k)},s^k)\right] \end{aligned} \tag{3.4}$$

Notice that in all the methods (3.2)-(3.4) we start from $\boldsymbol{x}^{(N)} = \boldsymbol{x}$, while the desired approximate solution $\boldsymbol{x}^* = \boldsymbol{X}(t^n;t^{n+1},\boldsymbol{x})$ of (3.1) is given by $\boldsymbol{x}^{(0)}$.

In practical applications of the semi-Lagrangian technique, the advective velocity field is also the output of a numerical method (in our case of the SISLDG solver for SWE), and the flow field is assumed to be sufficiently smooth. In this case, velocities at intermediate time levels, such as $\boldsymbol{u}(\boldsymbol{y}, s^k)$, can be replaced by consistent extrapolations from the values at previous time-steps without degrading the formal accuracy of the method. Taking $s \in [t^n, t^{n+1}]$, the appropriate extrapolation formula for the second order scheme is given by

$$\boldsymbol{u}(\boldsymbol{y}, s) = \left(1 + \frac{s - t^n}{\Delta t}\right) \boldsymbol{u}(\boldsymbol{y}, t^n) - \left(\frac{s - t^n}{\Delta t}\right) \boldsymbol{u}(\boldsymbol{y}, t^{n-1})$$

and for the third order scheme by

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{y}, s) \quad = \quad & \left[1 + \frac{3}{2}\left(\frac{s - t^n}{\Delta t}\right) + \frac{1}{2}\left(\frac{s - t^n}{\Delta t}\right)^2\right] \boldsymbol{u}(\boldsymbol{y}, t^n) + \\ & - \left[2\left(\frac{s - t^n}{\Delta t}\right) + \left(\frac{s - t^n}{\Delta t}\right)^2\right] \boldsymbol{u}(\boldsymbol{y}, t^{n-1}) + \\ & + \left[\frac{1}{2}\left(\frac{s - t^n}{\Delta t}\right) + \frac{1}{2}\left(\frac{s - t^n}{\Delta t}\right)^2\right] \boldsymbol{u}(\boldsymbol{y}, t^{n-2}). \end{aligned}$$

Notice that in many atmospheric applications the advective velocity field can be also taken constant in time over the interval $[t^n, t^{n+1}]$ and equal to its extrapolated value at time $t^n + \frac{\Delta t}{2}$, without degrading the effective accuracy of the method.

Furthermore, for each time sub-step k an interpolation must be performed in order to compute the velocity at $\boldsymbol{x}^{(k)}$ and at the other intermediate points. In this sense bilinear interpolation is usually recognised to be sufficiently accurate (see e.g. Staniforth and Coté (1991)) and piecewise linear interpolation has also been used with good results in Casulli (1990) and Casulli and Cheng (1992). Therefore, each of the N sub-steps can be performed at a low computational cost.

Regarding the second family of methods for the trajectories computation, the fixed point iteration technique was introduced in its best known form in Robert (1981), although an iterative technique had also been used in Mathur (1970). It is quite widely (if not almost exclusively) the technique of choice in atmospheric modelling. Let us define $\boldsymbol{\alpha}_m := \boldsymbol{x}_m - \boldsymbol{X}(t^n; t^{n+1}, \boldsymbol{x}_m)$ the displacement travelled by the fluid particles during Δt when following the approximated space-time trajectory AC'.

Then determine the foot $\boldsymbol{X}(t^n; t^{n+1}, \boldsymbol{x}_m)$ is equivalent to determine $\boldsymbol{\alpha}_m$. To determine $\boldsymbol{\alpha}_m$ note that $\boldsymbol{u}$ evaluated at point B of fig. 3.2 is just the inverse of the slope of the straight line $A'C$, and this gives the following $O(\Delta t^2)$ approximation Robert (1981):

$$\boldsymbol{\alpha}_m = \Delta t \boldsymbol{u}(\boldsymbol{x}_m - \boldsymbol{\alpha}_m, t^n)$$

that can be iteratively solved with a fixed point method, for example with Picard iteration:

$$\boldsymbol{\alpha}_m^{(k+1)} = \Delta t \boldsymbol{u}(\boldsymbol{x}_m - \boldsymbol{\alpha}_m^{(k)}, t^n)$$

Note that the time at which the velocity field has been frozen for the purposes of the trajectory computation is t^n only in a three time level scheme, while in the context of a two time level scheme (we have in mind), one should choose to take the velocity frozen at $t^{n+\frac{1}{2}}$, which leads to the need for a time extrapolation in case the velocity fields

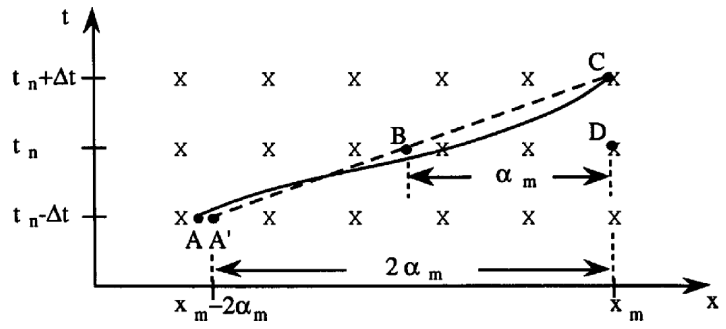


Figure 3.2: Schematic for the three time level advection. actual (solid) curve and approximated (dashed line) trajectories that arrive at mesh point $\boldsymbol{x}_m$ at time $t^n + \Delta t$. Here α_m is the distance the particle is displaced in x in time Δt

are only known up to time t^n. The advantage of the iterative technique is that it is unconditionally stable and that the iterations can be proven to converge under conditions that are relatively mild for atmospheric flows, see e.g.Pudykiewicz and Staniforth (1984), Smolarkiewicz and Pudykiewicz (1992).

Finally, the Taylor expansion technique was introduced instead by McGregor (1993) for Cartesian grids and extended in Giraldo (1999) to triangular finite element grids. It is based on the Taylor expansion in time of $\boldsymbol{X}(t^n; t^{n+1}, \boldsymbol{x})$ and on an approximate reconstruction of the time derivatives involved. The advantage over the previous techniques is that no interpolations or extrapolations are required. On the other hand, the approximated time derivative require extra storage and the derivation of the approximation is entirely heuristic and not generally valid for arbitrary flows. Another possible approach is that proposed by Purser and Leslie in Purser and Leslie (1994b), which employs forward in time trajectories.

3.3 Interpolation techniques

The other key ingredient in the implementation of a semi-Lagrangian method is the interpolation procedure employed in each time-step to reconstruct the solution values at the departure point of the streamline. Most of the work on evaluation of interpolation algorithms has been done for Cartesian grids, due to the context in which the methods were developed. Linear interpolation produces quite diffusive solutions, as it will be shown in section 6. In the case of one dimensional flow with Courant numbers smaller than one, it is easy to see that in fact the upwind method is recovered. Quadratic interpolation was used in Bates and McDonald (1977). Cubic Lagrange interpolation was used already in Sawyer (1963) and cubic spline was first applied in Purnell (1975). Cubic Lagrange interpolation was also used in the fundamental work of Robert Robert (1981) Robert (1982) and analysed in Pudykiewicz and Staniforth (1984). This is the interpolation technique most widely used in atmospheric applications. In general, it was shown by McCalpin in McCalpin (1988) that different properties are obtained depending on whether odd or even order polynomials are used in the interpolation. The so called cascade interpolation method was also proposed by R.J. Purser in Purser and Leslie (1994a) and an improved cascade interpolation technique is presented in Nair *et al.* (1999). However, successful attempts to achieve higher accuracy by means of kriging

or radial basis function interpolators have been reported in Le Roux *et al.* (1997), Le Roux *et al.* (1998), Behrens and Iske (2002). On unstructured meshes typical of finite element models, the finite element interpolators are generally used.

Moreover in the context of our numerical experiments on structured Cartesian meshes with the proposed SISLDG scheme we also used the the finite element high order interpolator, even if is well known that the maximum principle for the linear interpolation case cannot be proven for higher order interpolators. This is similar to what happens for many other numerical schemes for the advection equation, see e.g. LeVeque (1996), Quarteroni and Valli (1994). The issue was considered in Williamson and Rasch (1989), where various monotonic interpolation techniques based on Hermite interpolation were compared. A general prescription to overcome the intrinsic lack of monotonicity of all higher order interpolation and to monotonize a semi-Lagrangian method was given Bonaventura (2000), along with a proof of the monotonicity of the resulting scheme. This approach is also the most widely followed in operational implementations. In all the tests performed in chap 6, smooth solutions were considered (for which undershoots and overshoots are minimized) and no monotonization procedure was employed.

3.4 Advection equation in advective form

Let us consider the advection equation in quasilinear form:

$$\frac{\partial c}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} c = 0 \tag{3.5}$$

We take into account the fact that $\frac{\partial c}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} c$ can be written as $\frac{Dc}{Dt}$, the total derivative of c in the direction of the flow $\boldsymbol{u}$. Then equation (3.5) reduces to:

$$\frac{Dc}{Dt} = 0 \tag{3.6}$$

which states that c is just transported without change along the fluid particle trajectories. Therefore the solution of (3.7) is

$$c(\boldsymbol{x}, t^{n+1}) = c(\boldsymbol{x}^*, t^n) \tag{3.7}$$

where $\boldsymbol{x}^* = \boldsymbol{X}(t^n; t^{n+1}, \boldsymbol{x})$ is the solution of (3.1).

We now discretize in space by taking the L^2 projection against the test functions on a given element K_I . In particular, following Hasbani *et al.* (1982) or the Direct Characteristic Galerkin method proposed in Morton *et al.* (1988), we use as test functions the basis functions $\{\varphi_{I,l}(\boldsymbol{x})\}_{l=1}^{(p+1)^2}$:

$$\int_{K_I} c(\boldsymbol{x}, t^{n+1}) \varphi_{I,k}(\boldsymbol{x})\, d\boldsymbol{x} = \int_{K_I} c(\boldsymbol{x}^*, t^n) \varphi_{I,k}(\boldsymbol{x})\, d\boldsymbol{x} \quad k = 1, (p+1)^2. \tag{3.8}$$

Results of this algorithm are showed in fig 3.3 and 3.4 for the linear advection equation at a Courant number equal to 3.75. The same strategy will be adopted in section 5.3 for the discretization of the non-linear advective terms in the SWE. Finally, using the the evolution operator $E(t^n, \Delta t)$ defined in (5.11), and extensively used in section 5.3, eq(3.9) can be rewritten as follows:

$$\int_{K_I} \varphi_{I,k}(\boldsymbol{x}) c(\boldsymbol{x}, t^{n+1})\, d\boldsymbol{x} = \int_{K_I} \varphi_{I,k}(\boldsymbol{x}) \left[E(t^n, \Delta t) c^n\right](\boldsymbol{x})\, d\boldsymbol{x} \quad k = 1, (p+1)^2. \tag{3.9}$$

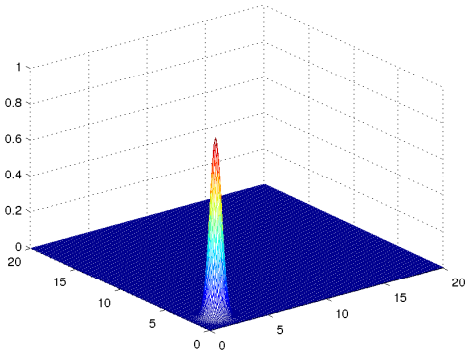


Figure 3.3: Gaussian profile solution of $\frac{\partial c}{\partial t} + \boldsymbol{U} \cdot \nabla c = 0$ with Periodic B.C., $Courant = 3.75$ after $T = 4s$, advected by a wind field $U = 3m/s$, $V = 3m/s$.

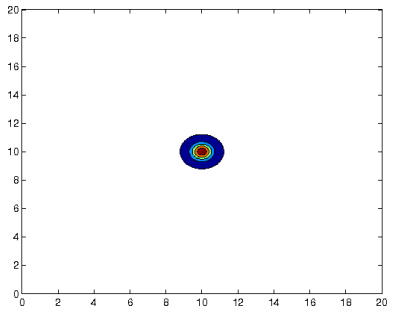


Figure 3.4: Gaussian initial profile, $p_x = 2$, $p_y = 2$.

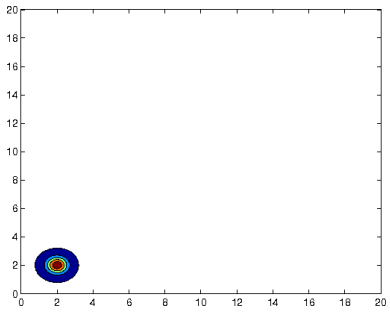


Figure 3.5: Gaussian profile after $T = 4s$, advected by a wind field $U = 3m/s$, $V = 3m/s$.

Other tests for the advection in two dimensions are showed in chapter 6, where results about solid body rotation and deformational flow of Smolarkiewicz are presented in the context of the p-adaptivity strategy presented in section 5.4.

Finally, to make the method robust also in the limit $\Delta t \to 0$ Gauss-Legendre-Lobatto (GLL) quadrature nodes could be used instead of Gauss-Legendre ones.

3.5 Advection equation in flux form

It has also been argued that by using the flux form it is easier to avoid the numerical instabilities of the type reported by Phillips Phillips (1959).

Moreover the flux form of the advection equation guaranties the conservation property and therefore has been chosen to model the passive tracers advection to be coupled with our proposed SISLDG solver for SWE equations. In the following the SLDG algorithm first proposed in Restelli *et al.* (2006) will be presented and commented.

Let us start from the (in general non-linear) scalar advection , written in flux form.

Find $u(\boldsymbol{x},t)$ s.t. :

$$\frac{\partial c}{\partial t}+\mathbf{u}(c)\cdot\nabla c=\frac{\partial c}{\partial t}+\nabla\cdot\mathbf{F}(c)=0\quad\text{in}\quad\mathbb{R}^n\supset\Omega\times(0,T)\tag{3.10}$$

with suitable initial and boundary conditions. Here $\mathbf{F}:\mathbb{R}\to\mathbb{R}^n$ is an assigned smooth flux vector function and $\mathbf{u}=\mathbf{F}'$ the corresponding characteristic velocity vector. Notice that consider the flux form of the advection equation is essential in order to obtain a discrete conservation property when the advection field is non-solenoidal. The weak formulation is obtained initially along the usual lines of DG methods.

Find $c(\cdot,t)\in H^1(\mathcal{T}_h)\quad\forall t\in(0,T)$ s.t.:

$$\frac{d}{dt}\int_\kappa vcd\boldsymbol{x}=-\int_\kappa v\nabla\cdot\mathbf{F}(c)d\boldsymbol{x}\quad\forall v\in H^1(\mathcal{T}_h)$$

and integrating by parts

$$\frac{d}{dt}\int_\kappa vcd\boldsymbol{x}=\int_\kappa\nabla v\cdot\mathbf{F}(c)d\boldsymbol{x}-\int_{\partial\kappa}v\boxed{\mathbf{n}\cdot\mathbf{F}(c)}d\mathbf{y}$$

Notice that the advective boundary term $\mathbf{n}\cdot\mathbf{F}(c)$ highlighted in the box does not have yet a precise meaning, because $c\in\mathcal{T}_h$ is a discontinuous function (in the sense of the traces) across inter-element boundaries. In order to resolve this ambiguity the fluxes $\hat{c}=\{\{c\}\}$ are introduced (see Bassi and Rebay (1997a)) to replace c on each edge $e\subset\partial\kappa$. The weak formulation of the original conservation law thus reads: Find $c\in H^1(\mathcal{T}_h)$ s.t.

$$\frac{d}{dt}\int_\kappa vcd\boldsymbol{x}=\int_\kappa\nabla v\cdot\mathbf{F}(c)d\boldsymbol{x}-\sum_{e\in\partial\kappa}\int_e v\mathbf{n}\cdot\mathbf{F}(\hat{c})d\mathbf{y}\quad\forall v\in H^1(\mathcal{T}_h)$$

At this point Restelli *et al.* (2006) departs from the usual DG technique and follows the interpretation of characteristics-Galerkin methods (for evolutionary hyperbolic problems) as generalized Godunov methods as proposed and analysed in Morton (1998). With this approach in mind let us integrate the equation in time between t^n and $t^{n+1}:=t^n+\Delta t$ (as usual in the application of a finite volume scheme to a conservation law) to give:

$$\begin{aligned}\int_\kappa v(\boldsymbol{x})c(\boldsymbol{x},t^{n+1})d\boldsymbol{x}=&\int_\kappa v(\boldsymbol{x})c(\boldsymbol{x},t^n)d\boldsymbol{x}\\&+\int_\kappa\nabla v(\boldsymbol{x})\cdot\int_{t^n}^{t^{n+1}}\mathbf{F}(\boxed{c(\boldsymbol{x},t)})dtd\boldsymbol{x}+\\&-\sum_{e\in\partial\kappa}\int_e v(\mathbf{y})\mathbf{n}(\mathbf{y})\cdot\int_{t^n}^{t^{n+1}}\mathbf{F}(\boxed{\hat{c}(\mathbf{y},t)})dtd\mathbf{y}\quad\forall v\in H^1(\mathcal{T}_h)\end{aligned}$$

The solution c of (3.10) is constant along characteristics $\frac{d\boldsymbol{x}}{dt}=\mathbf{u}(c)$, which are therefore straight (this being true only for the scalar case, but migrating to systems of conservation laws, like shallow water equations, we have to face with curvilinear trajectories). Therefore we are going to use this information to evaluate (at least in an approximated way) the unknown quantity $c(\cdot,t),\quad t>t^n$ highlighted in the box at right and side. This is the way in which Restelli *et al.* (2006) combines the characteristics method with

the Galerkin approach, that is the fact that $\frac{\partial c}{\partial t} + \mathbf{u}(c) \cdot \nabla c$ can be written as $\frac{dc}{dt}$, total derivative in the direction of the flow, is exploited not directly in writing the equation in advective form, but indirectly in the evaluation of the unknown function in the flux terms arising when the equations are written in conservation formulation. It is clear now that this approach rely an how well we are able to compute the trajectories (see section 3.2). For the moment let $E(t_0, \Delta t)$ denote the evolution operator defined in (5.11). In terms of $E(t_0, \Delta t)$, the weak formulation reads:

$$\begin{aligned}\int_\kappa v(\boldsymbol{x})c(\boldsymbol{x}, t^{n+1})d\boldsymbol{x} = &\int_\kappa v(\boldsymbol{x})c(\boldsymbol{x}, t^n)d\boldsymbol{x} \\ &+ \int_\kappa \nabla v(\boldsymbol{x}) \cdot \int_0^{\Delta t} \mathbf{F}\Big(\left[E(t^n, \tau)c^n\right](\boldsymbol{x})\Big)d\tau d\boldsymbol{x} + \\ &- \sum_{e\in\partial\kappa} \int_e v(s)\mathbf{n}(s) \cdot \int_0^{\Delta t} \mathbf{F}\Big(\left[E(t^n, \tau)\hat{c}^n\right](s)\Big)d\tau ds \quad \forall v \in H^1(\mathcal{T}_h)\end{aligned}$$

Given the previous weak formulation, the SLDG numerical method for equation (3.10) reads:

find $c_h \in V_h^{DG}$ such that:

$$\begin{aligned}\int_\kappa v_h(\boldsymbol{x})c_h(\boldsymbol{x}, t^{n+1})d\boldsymbol{x} = &\int_\kappa v_h(\boldsymbol{x})c_h(\boldsymbol{x}, t^n)d\boldsymbol{x} \\ &+ \int_\kappa \nabla v_h(\boldsymbol{x}) \cdot \int_0^{\Delta t} \mathbf{F}\Big(\left[E(t^n, \tau)c_h^n\right](\boldsymbol{x})\Big)d\tau d\boldsymbol{x} + \\ &- \sum_{e\in\partial\kappa} \int_e v_h(s)\mathbf{n}(s) \cdot \int_0^{\Delta t} \mathbf{F}\Big(\left[E(t^n, \tau)\hat{c_h}^n\right](s)\Big)d\tau ds \quad \forall v_h \in V_h^{DG}\end{aligned}$$

Now once the exact evolution operator $E(t^n, s)$ is replaced by its discrete counterpart $E^d(t^n, s)$ defined in section 5.2 in terms of the two steps of the semi-Lagrangian technique, the unknown variable c_h is expanded over a fixed orthogonal basis for $\mathcal{Q}_k(\kappa)$,(see section 2.4 for the notation), and as arbitrary test function v_h a generic basis function is taken, then the fully discrete problem is obtained.

In the following some numerical results of the implementation of this method in one dimension with high order polynomials are reported, that show how this method imposes no CFL restrictions on Δt, without simultaneously penalizing the accuracy of the method, making therefore it the ideal candidate to be coupled with the novel SISLDG solver for SWE.

3.5.1 Linear advection

In particular the conservative SLDG method has been tested in one dimension considering equation (3.10) on the interval $[-40, 40]$ of the real line and taking as initial datum a $\mathcal{C}^1$, compactly supported, bell shaped function with a maximum value of 0.4. The time step was taken to be $\Delta t = 1/10$ and the spatial resolution was $\Delta x = 1/10$. The advection velocity was increased from a value $1/2$ up to value 8 with the Courant number going from $1/2$ up to 8. The results obtained with the SLDG method using polynomial degree equal to 8 are shown in figure 3.6 (black line is used for the initial datum, red line for the profile at the final time $T = 1s$). The corresponding error in L^2 norm are collected in table 3.1. It should be remarked that in this linear test the exact value for

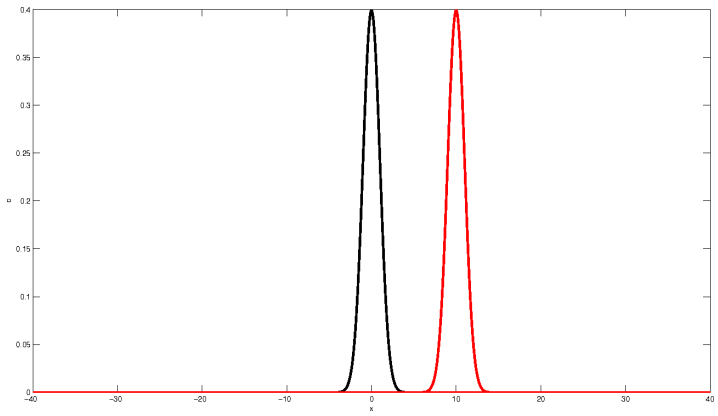


Figure 3.6: SLDG solution of LAE at Courant number = 10.

u	$Cu = u\frac{\Delta t}{h}$	$\|error(t=1)\|_{L^2}$
0.5	0.5	6.36e-14
1	1	6.36e-14
2	2	3.67e-13
4	4	7.34e-13
8	8	1.46e-13

Table 3.1: L^2 error norm for SLDG linear advection with polynomial degree = 8, $\Delta t = 0.1, \Delta x = 0.1$ and different advection velocity values.

the trajectory departure point was used, since all basic integration schemes would have given the exact solution results in this simple case of advective velocity constant in time and space. This is not the case of the Burgers equation.

3.5.2 Non-linear advection

The inviscid Burgers equation $c_t + (c^2/2)_x = 0$ is a special case of (3.10) with $F(c) = c^2/2$. The initial condition for this problem is set to $c_0(x) = sin(\frac{\pi}{2}x)$, over a periodic domain [0,2]. The domain is partitioned into 80 elements, and the SLDG scheme 3.11 employing cubic polynomials and 4 Gauss-Legendre quadrature points is used for the simulation. The time step used was $\Delta t = 0.06$ corresponding to a Courant number $C = 7.2$. In figure 3.7 the numerical solution is shown for this problem at $t = 0.3s$ just before the formation of the shock. Remember (see for example LeVeque (1992)) that for this problem the time of formation of the shock can be computed as $T_{shock} = -\frac{1}{minc_0'} \simeq 0.318$. The SLDG solution is marked in red where ball points, one value per each element, represent the average on each element (i.e. the first modal coefficient), while again black line is used for the initial datum. As time evolves the shock becomes severe and will pollute the numerical solution, but SLDG method is not devised to solve shock capturing problems. This test was only reported to show the good stability properties of SLDG method even in the non-linear case.

Finally let us notice that similar results were obtained even using the SLDG in advective form (not displayed here).

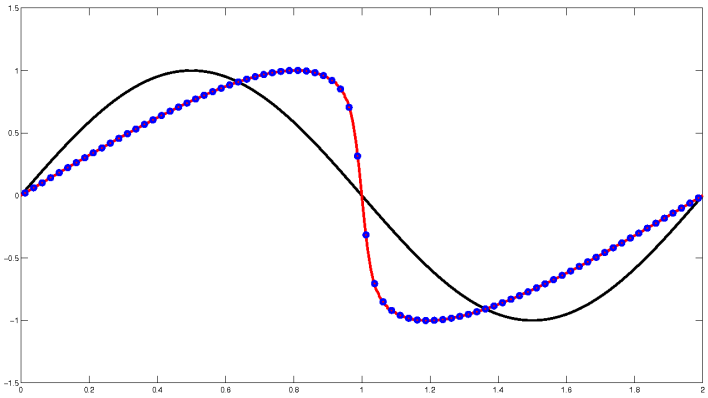


Figure 3.7: SLDG solution of the Burgers equation with Courant number $C = 7.2$. Black line represents the initial sinusoidal profile, while red line is the solution computed at $t = 0.3s$ i.e. near the shock formation time.

Chapter 4

The semi-implicit time discretization technique

In this chapter a presentation of the semi-implicit technique based on an characteristic analysis of the governing equations is summarized. In this context a pure Eulerian point of view is adopted, but the same analysis will hold also when, in next chapters, the semi-implicit technique will be coupled with a semi-Lagrangian treatment of the advection .

4.1 Introduction

A fully implicit discretization of the governing equations would lead to a method which is, of course, unconditionally stable; this procedure, however, involves the simultaneous solution of a large number of coupled non-linear equations. For efficiency, then, only some terms of the governing equations are discretized implicitly. Specifically, since in subsonic flows the time scale of the speed of sound is faster than the time scale of the flow, the semi-implicit method uses an implicit discretization only for those terms which are related to the speed of sound. As a result unconditional stability is reached with respect to the CFL condition for the speed of sound, which is the most restrictive condition for fully explicit methods in subsonic flow calculations. However, how to identify terms responsible for fast propagating gravity waves ? To answer this question in this chapter we closely follow the approach of Casulli Casulli and Greenspan (1984), Casulli and Cheng (1990) which starts from a characteristic analysis of the governing equations to derive the semi-implicit time discretization of shallow water equations we will use in the definition of our proposed SISLDG scheme.

4.2 Characteristic analysis of the rotating shallow water equations

Equations 4.1 form a quasilinear hyperbolic system of partial differential equations in three independent variables. Closely following Casulli and Cheng (1990), in order to determine the particular semi-implicit discretization whose stability is independent on the celerity, we first analyze the characteristic cone of the governing equations. To this purpose let us rewrite eqs. (4.1) in the equivalent form

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + g\frac{\partial \eta}{\partial x} + v\frac{\partial u}{\partial y} = fv \tag{4.1}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + g\frac{\partial \eta}{\partial y} = -fu \tag{4.2}$$

$$\frac{\partial \eta}{\partial t} + h\frac{\partial u}{\partial x} + u\frac{\partial \eta}{\partial x} + h\frac{\partial v}{\partial y} + v\frac{\partial \eta}{\partial y} = u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} \tag{4.3}$$

or, in matrix notation,

$$\frac{\partial \boldsymbol{w}}{\partial t} + A(\boldsymbol{w})\frac{\partial \boldsymbol{w}}{\partial x} + B(\boldsymbol{w})\frac{\partial \boldsymbol{w}}{\partial y} = D(\boldsymbol{w}) \tag{4.4}$$

where $\boldsymbol{w} := (u, v, \eta)^T$, and

$$A = \begin{bmatrix} u & 0 & g \\ 0 & u & 0 \\ h & 0 & u \end{bmatrix}, \quad B = \begin{bmatrix} v & 0 & 0 \\ 0 & v & g \\ 0 & h & v \end{bmatrix}, \quad \boldsymbol{w} := \begin{pmatrix} fu \\ -fv \\ u\frac{\partial b}{\partial x} + v\frac{\partial b}{\partial y} \end{pmatrix}.$$

If I denotes the identity matrix, then the characteristic equation of system (4.4) is given by

$$\det(qI + rA + sB) = 0,$$

that is

$$(q + ru + sv)\left[(q + ru + sv)^2 - gh(r^2 + s^2)\right] \tag{4.5}$$

The triples (q, r, s) satisfying eq. (4.5) are then directions normal to the characteristic cone at its vertex. Equation (4.5) decomposes into the two equations

$$q + ru + sv = 0 \tag{4.6}$$

and

$$(q + ru + sv)^2 - gh(r^2 + s^2) = 0 \tag{4.7}$$

Hence, as shown in fig. 4.1, the local characteristic cone with vertex in (x_0, y_0, t_0) parallel to the vector $(1, u, v)^T$ and the cone whose equation is:

$$[(x - x_0) - u(t - t_0)]^2 + [(y - y_0) - v(t - t_0)]^2 - gh(t - t_0)^2 = 0, \tag{4.8}$$

in fact, on the cone surface, the gradient of the left-hand side of eq. (4.8) satisfies eq (4.7)

Now notice that, whereas the first part of the characteristic cone, defined by eq. (4.6), depends only on the fluid velocity u, v, the second part of the characteristic cone, defined by eq. (4.7), depends also upon the celerity $\sqrt{gh}$. Note also that the term gh

in the characteristic equation (4.5) arises from the off-diagonal terms g and h in the matrices A and B. These are the coefficient of $\partial\eta/\partial x$ in eq. (4.2), the coefficient of $\partial\eta/\partial y$ in eq. (4.3), and the coefficient of $\partial u/\partial x$ and $\partial v/\partial y$ in eq. (4.3). As a result these terms in the shallow water equations must be discretized implicitly for a numerical method to be unconditionally stable with respect to the celerity.

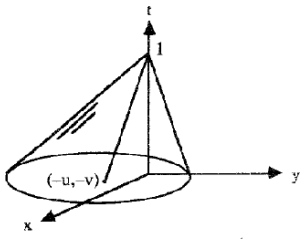


Figure 4.1: Characteristic cone through (0,0,1).

4.3 Semi-implicit time discretization

Based on the characteristic analysis of the governing equations reported in previous section we write a time semi-discretization of (4.1) in which the gradient of the free surface elevation in the momentum equations and the velocity divergence in the continuity equation will be discretized implicitly:

$$
\begin{aligned}
\frac{\eta^{n+1}-\eta^n}{\Delta t} &= -\theta\boldsymbol{\nabla}\cdot\left(\boldsymbol{u}^{n+1}h^n\right)-(1-\theta)\boldsymbol{\nabla}\cdot\left(\boldsymbol{u}^n h^n\right)\\
\frac{u^{n+1}-u^n}{\Delta t}+(\boldsymbol{u}^n\cdot\boldsymbol{\nabla})u^n &= -\theta g\left[\frac{\partial\eta}{\partial x}\right]^{n+1}+\theta f v^{n+1}-(1-\theta)g\left[\frac{\partial\eta}{\partial x}\right]^{n}+(1-\theta)fv^n\\
\frac{v^{n+1}-v^n}{\Delta t}+(\boldsymbol{u}^n\cdot\boldsymbol{\nabla})v^n &= -\theta g\left[\frac{\partial\eta}{\partial y}\right]^{n+1}-\theta f u^{n+1}-(1-\theta)g\left[\frac{\partial\eta}{\partial y}\right]^{n}-(1-\theta)fu^n
\end{aligned}
$$

where $\theta\in[0,1]$ is a fixed implicitness parameter.

Notice that in this context the *pure* semi-implicit technique has been considered, therefore an *Eulerian* point of view was adopted in all this chapter. As a result *no* evolution operators are involved in this pure semi-implicit (Eulerian) context. On the other hand, in the next chapter this semi-implicit time discretization strategy will be combined with a semi-Lagrangian treatment of the nonlinear advective terms to discretize in time the equations (4.1): in those context, instead, the introduction of an evolution operator associated to the semi-Lagrangian technique will play a crucial role .

Finally notice that when the semi-implicit scheme is combined with a two-time-level semi-Lagrangian treatment of the advective terms, it is customary to discretize semi implicitly also the Coriolis terms, which from a pure characteristic analysis of the equation, could be treated explicitly, indeed an explicit treatment of Coriolis terms in the context of a two time-level scheme could be computationally unstable as noticed in Temperton and A.Staniforth (1987).

Chapter 5

The p-SISLDG method for the SWE

In this chapter the semi-Lagrangian (chap. 3) and the semi-implicit (chap. 4) techniques are combined within a Discontinuous Galerkin formulation (chap. 2) to devise a novel discretization scheme for the approximation of the rotating shallow water equations, namely the Semi-Implicit Semi-Lagrangian Discontinuous Galerkin method (SISLDG). After an abstract reformulation of the SISL technique through the introduction of a proper evolution operator $E(t^n, \Delta t)$ in section 5.2, the SISLDG method is derived in section 5.3, then, to increase its efficiency, a simple $p-$adaptivity criterion is illustrated in section 5.4 .

5.1 Introduction

To achieve the goal of making the maximum permissible time step governed by considerations of stability rather than accuracy in the integration of NWP models, it is essential to associate a semi-Lagrangian treatment of advection with a sufficiently stable treatment of terms responsible for the propagation of gravitational oscillations. The powerful of this idea has been demonstrated by Roberts in his seminal papers (Robert (1981), Robert (1982)) where he combined a semi-Lagrangian (SL) treatment of advection with a semi-implicit (SI) treatment of gravity oscillations terms, in the context of three-time-level shallow-water finite-difference model in Cartesian geometry. From then on the semi-implicit semi-Lagrangian technique has been quite successfully exploited both in finite differences (for example Robert (1982), or Casulli (1990)) and finite elements (for example Staniforth and Temperton (1986), or Miglio *et al.* (1999),or Le Roux *et al.* (1999), or Giraldo (2005)) frameworks, but, up to now, not fully explored within the DG context. In this chapter an attempt has been made to fill this gap, encouraged by results found by other authors in attempts at coupling DG with either SI (for example Restelli and Giraldo (2009)) or SL time discretizations Restelli *et al.* (2006).

5.2 An abstract formulation of the semi-implicit semi-Lagrangian technique

In this section we do not have the ambition to introduce a thorough treatment from the continuum mechanics point of view (for whose foundations see e.g. Gurtin (1981) or Marsden and Hughes (1983)), but we only want to fix a notation to define the evolution operator E which will be massively used in the definition of the SISLDG method presented in the next section.

Let us consider the following (possibly nonlinear) first order system of PDE's

$$\frac{\partial \boldsymbol{q}}{\partial t} + (\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{q} = \boldsymbol{F}(\boldsymbol{q}) \tag{5.1}$$

where $\boldsymbol{q} = \boldsymbol{q}(\boldsymbol{x}, t)$ is the vector of the N unknown functions, $\mathbf{F} : \mathbb{R}^N \to \mathbb{R}^N$ is a given function, assumed to be sufficiently smooth, and $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$ is the velocity field.

Notice that in (5.1) the independent variables are the current position $\boldsymbol{x}$ and the time t: this formulation is called Eulerian description in the fluid mechanics context or spatial description in the continuum mechanics framework.

In order to derive the semi-Lagrangian semi-implicit time discretization of eq. (5.1), the most natural way to proceed is to make a change of variables and to reformulate our system using now a referential position $\boldsymbol{X}$ at a fixed reference time (identifying a fluid particle) and the time t as independent variables: this choice takes the name of Lagrangian description in fluid mechanics and the name of referential description in continuum mechanics.

The link between the two equivalent formulations is given by the function:

$$\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t) \tag{5.2}$$

whose value in $(\boldsymbol{X}, t)$ is the position at time t of the fluid particle [1] $\boldsymbol{X}$, (see fig. 5.1),

or by its inverse

$$\boldsymbol{X} = \boldsymbol{X}(\boldsymbol{x}, t) \tag{5.3}$$

whose value in $(\boldsymbol{x}, t)$ identifies the particle $\boldsymbol{X}$ which is passing in $\boldsymbol{x}$ at time t .

Now using the two different descriptions, Eulerian or Lagrangian, the functional dependence of the unknowns vector on the independent variables will be different and therefore denoted with $\boldsymbol{q}$ in the Eulerian description and $\boldsymbol{Q}$ in the Lagrangian one:

$$\boldsymbol{q}(\boldsymbol{x}(\boldsymbol{X}, t), t) = \boldsymbol{Q}(\boldsymbol{X}, t) \tag{5.4}$$

where (5.2) has been used.

If we now derive eq. (5.4) with respect to t, then, applying the chain rule, we get:

$$\frac{\partial \boldsymbol{q}}{\partial t} + \left[\frac{\partial}{\partial t}\boldsymbol{x}(\boldsymbol{X}, t)\right] \cdot \boldsymbol{\nabla}\boldsymbol{q} = \frac{\partial \boldsymbol{Q}}{\partial t}$$

but since $\boldsymbol{x}(\boldsymbol{X}, t)$ has been defined as the the position at time t of the fluid particle $\boldsymbol{X}$, then it follows:

$$\frac{\partial}{\partial t}\boldsymbol{x}(\boldsymbol{X}, t) = \boldsymbol{u}(\boldsymbol{x}(\boldsymbol{X}, t), t) \tag{5.5}$$

[1] With the expression fluid particle $\boldsymbol{X}$ we mean the fluid particle occupying the position $\boldsymbol{X}$ at the fixed reference time.

In other words we came to the well known expression of the Lagrangian time derivative in terms of Eulerian derivatives:

$$\frac{\partial \boldsymbol{q}}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{q} = \frac{\partial \boldsymbol{Q}}{\partial t} \tag{5.6}$$

Finally the original equation (5.1) can be re-casted in Lagrangian framework as :

$$\frac{\partial \boldsymbol{Q}}{\partial t} = \boldsymbol{F}(\boldsymbol{Q}) \tag{5.7}$$

This equation now represents the starting point for the semi-Lagrangian semi-implicit discretization. Let us integrate it in time between t^n and t^{n+1}:

$$\int_{t^n}^{t^{n+1}} \frac{\partial \boldsymbol{Q}}{\partial t} dt = \int_{t^n}^{t^{n+1}} \boldsymbol{F}(\boldsymbol{Q}) dt$$

Now the left hand side can be computed exactly via the fundamental theorem of calculus, while the right hand side can be approximated using the trapezoidal rule:

$$\boldsymbol{Q}(\boldsymbol{X}, t^{n+1}) - \boldsymbol{Q}(\boldsymbol{X}, t^n) = \frac{\Delta t}{2} \left[\boldsymbol{F}(\boldsymbol{Q}(\boldsymbol{X}, t^{n+1}) + \boldsymbol{F}(\boldsymbol{Q}(\boldsymbol{X}, t^n) \right] \tag{5.8}$$

Notice that the integration in time and its numerical approximation through the trapezoidal rule (at right hand side) have been both carried out in terms of $\boldsymbol{Q}$ keeping the fluid particle $\boldsymbol{X}$ fixed: this means that we have integrated in time *along the trajectories*.

Notice that, up to now, the reference configuration was just fixed in an arbitrary way. At this point the semi-Lagrangian technique consists in choosing the reference configuration in a clever way, i.e. in taking t^{n+1} as reference time. As a result we have

$$\boldsymbol{x}(\boldsymbol{X}, t^{n+1}) = \boldsymbol{X} \tag{5.9}$$

Now if eq. (5.9) together with eq. (5.4) is used, then eq. (5.8) becomes:

$$\begin{aligned} \boldsymbol{q}(\boldsymbol{X}, t^{n+1}) \quad = \quad & \boldsymbol{q}(\boldsymbol{x}(\boldsymbol{X}, t^n), t^n) + \\ & \frac{\Delta t}{2} \boldsymbol{F}\left(\boldsymbol{q}(\boldsymbol{X}, t^{n+1})\right) + \\ & \frac{\Delta t}{2} \boldsymbol{F}\left(\boldsymbol{q}(\boldsymbol{x}(\boldsymbol{X}, t^n), t^n)\right) \end{aligned} \tag{5.10}$$

At this point following evolution operator is introduced:

$$\left[E(t^n, \Delta t)h\right](\cdot) = h(\boldsymbol{x}(\cdot, t^n)) \tag{5.11}$$

where $h : \mathbb{R}^d \supset \Omega \to \mathbb{R}$ is a real valued function of the space only. From an operational point of view the action of the operator $E(t^n, \Delta t)$ onto the function h is a function of the space only, whose value at a point $\boldsymbol{X}$ can be defined in two steps:

- first the point $\boldsymbol{x}(\boldsymbol{X}, t^n)$ i.e. the foot at time level t^n of the trajectory arriving in $\boldsymbol{X}$ at time t^{n+1} has to be calculated. To this end, considering that:

$$\boldsymbol{x}(\boldsymbol{X}, t^{n+1}) = \boldsymbol{x}(\boldsymbol{X}, t^n) + \int_{t^n}^{t^{n+1}} \frac{\partial}{\partial t} \boldsymbol{x}(\boldsymbol{X}, t) dt$$

and using eq. (5.9) together with eq. (5.5), we have:

$$\boldsymbol{x}(\boldsymbol{X}, t^n) = \boldsymbol{X} - \int_{t^n}^{t^{n+1}} \boldsymbol{u}\left(\boldsymbol{x}(\boldsymbol{X}, t), t\right) dt.$$

Moreover, since in general the advective velocity field $\boldsymbol{u}$ in eq. (5.1) can be a function of the unknown $\boldsymbol{q}$ too, (as in the case of the nonlinear advective term of the momentum equations), the semi-Lagrangian approach consists in freezing the advective velocity field at the old time t^n, (that amounts into linearization in time of the possibly nonlinear advective term at left hand side of eq. (5.1)), therefore previous equation in a semi-Lagrangian context is replaced by:

$$\boldsymbol{x}(\boldsymbol{X}, t^n) = \boldsymbol{X} - \int_{t^n}^{t^{n+1}} \boldsymbol{u}\left(\boldsymbol{x}(\boldsymbol{X}, t), t^n\right) dt.$$

- then the function $\boldsymbol{q}^n(\cdot) := \boldsymbol{q}(\cdot, t^n)$ has to be evaluated at the foot computed in the previous step, i.e. :

$$\left[E(t^n, \Delta t)\boldsymbol{q}^n\right](\boldsymbol{X}) = \boldsymbol{q}^n\left(\boldsymbol{X} - \int_{t^n}^{t^{n+1}} \boldsymbol{u}\left(\boldsymbol{x}(\boldsymbol{X}, t), t^n\right) dt\right)$$

Let us notice that since in the first step of the definition of the semi-Lagrangian evolution operator the advective velocity field has been frozen at time t^n, then it follows that a different semi-Lagrangian evolution operator will be associated to each different time step, according to the different underlying advective field. This dependence from t^n has been underlined also in the notation $E(t^n, \Delta t)$.

Then, in terms of the evolution operator E, eq. (5.10) takes the form:

$$\begin{aligned} \boldsymbol{q}(\boldsymbol{X}, t^{n+1}) \;=\; & \left[E(t^n, \Delta t)\boldsymbol{q}^n\right](\boldsymbol{X}) + \\ & + \frac{\Delta t}{2}\boldsymbol{F}\left(\boldsymbol{q}(\boldsymbol{X}, t^{n+1})\right) + \\ & + \frac{\Delta t}{2}\boldsymbol{F}\left(\left[E(t^n, \Delta t)\boldsymbol{q}^n\right](\boldsymbol{X})\right) \end{aligned} \tag{5.12}$$

where it is understood that $\boldsymbol{q}^n(\cdot) := \boldsymbol{q}(\cdot, t^n)$.

At this point, if the previous two steps defining the 'exact' semi-Lagrangian evolution operator $E(t^n, \Delta t)$ are replaced by their discrete counterparts, then the 'discrete' semi-Lagrangian evolution operator $E^d(t^n, \Delta t)$ is defined in the sense that its action on a function h of the space only is a function of the space only, whose value at a point $\boldsymbol{X}$ can be defined in two steps:

- first the point $\boldsymbol{x}(\boldsymbol{X}, t^n)$ i.e. the foot at time level t^n of the trajectory arriving in $\boldsymbol{X}$ at time t^{n+1} is *approximated*, e.g. using one of the techniques presented in section 3.2 ;

- then the function $\boldsymbol{q}(\cdot, t^n)$ has to be *interpolated* at the foot approximated in previous step, e.g. by using one of the techniques presented in section 3.3.

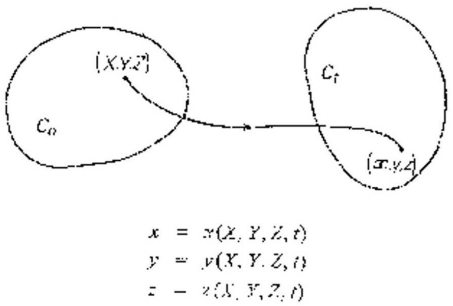


Figure 5.1: $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t)$ is the position at time t of the particle identified by its reference position $\boldsymbol{X}$.

5.3 The semi-implicit, semi-Lagrangian Discontinuous Galerkin discretization

The problem of a semi-implicit, semi-Lagrangian Discontinuous Galerkin discretization of (4.1) is now addressed.

Following the approach of Casulli and Greenspan (1984),Casulli and Cheng (1990), Casulli (1990),Casulli and Cattani (1994), a semi-implicit, semi-Lagrangian (more precisely, Eulerian-Lagrangian) discretization of (4.1) can be written as

$$\frac{\eta^{n+1} - \eta^n}{\Delta t} = -\theta \nabla \cdot \left(\boldsymbol{u}^{n+1} h^n \right) - (1-\theta) \nabla \cdot \left(\boldsymbol{u}^n h^n \right) \tag{5.13}$$

$$\frac{u^{n+1} - E(t^n, \Delta t)u}{\Delta t} = -\theta g \left[\frac{\partial \eta}{\partial x} \right]^{n+1} + \theta f v^{n+1} + \tag{5.14}$$
$$+(1-\theta) g E(t^n, \Delta t) \left[\frac{\partial \eta}{\partial x} \right]^{n} + (1-\theta) f E(t^n, \Delta t) v^n$$

$$\frac{v^{n+1} - E(t^n, \Delta t)v}{\Delta t} = -\theta g \left[\frac{\partial \eta}{\partial y} \right]^{n+1} - \theta f u^{n+1} + \tag{5.15}$$
$$+(1-\theta) g E(t^n, \Delta t) \left[\frac{\partial \eta}{\partial y} \right]^{n} - (1-\theta) f E(t^n, \Delta t) u^n$$

where E is the evolution operator defined in (5.11), $\theta \in [0, 1]$ is a fixed implicitness parameter (see chapter 4), and finally the notation $f^n(\boldsymbol{x}) := f(\boldsymbol{x}, t^n)$ has been used.

Notice that the continuity equation is considered in its original (Eulerian) flux form Casulli (1990) in order to endow the proposed scheme with the mass conservation property, while the momentum equations are written in advective form in order to replace the nonlinear advection operator with the Lagrangian derivative, this making sense since the momentum is not a conserved variable and since we are not interested in describe shocks.

Regarding the semi-implicit discretization, note that the determination of this specific form (5.13)-(5.16) follows directly from the characteristic analysis of the SWE

equations, see e.g. Casulli and Greenspan (1984) for details (also reported in section 4.2) ; in particular notice that after this characteristic analysis of SWE one find that in the divergence term of the continuity equation only the velocity derivatives need to be discretized implicitly, while the total water depth $h := \eta - b$ can be taken explicitly, resulting in this way to a linearization in time of the divergence term. Finally the Coriolis terms are treated implicitly, since an explicit treatment of these terms in the context of a two time-level scheme could be computationally unstable as noticed in Temperton and A.Staniforth (1987).

We now discretize in space by taking the L^2 projection against the test functions on a given element K_I . In particular, following Hasbani *et al.* (1982) or the Direct Characteristic Galerkin method proposed in Morton *et al.* (1988), we use as test functions the basis functions $\{\varphi_{I,l}(\boldsymbol{x})\}_{l=1}^{(p_I^\eta+1)^2}$ and $\{\psi_{I,m}(\boldsymbol{x})\}_{m=1}^{(p_I^u+1)^2}$.

Notice that in the DG formulation of the SWE system, we are led to consider the gradient of the free surface elevation η, which is a piece-wise polynomial function with no global continuity constraints. To this end, we follow the standard local discontinuous Galerkin method and define the discrete gradient operator $\boldsymbol{\mathcal{D}}_h\eta$ as follows. Given $\eta \in \mathcal{Q}_{\boldsymbol{p}_\eta}$, its discrete gradient $\boldsymbol{\mathcal{D}}_h\eta$ is the unique element of $(\mathcal{Q}_{\boldsymbol{p}_\eta})^2$ such that

$$\int_{K_I} \boldsymbol{\mathcal{D}}_h\eta \cdot \boldsymbol{w}\, d\boldsymbol{x} = -\int_{K_I} \eta \boldsymbol{\nabla}\cdot \boldsymbol{w}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\boldsymbol{w}\cdot\boldsymbol{n}\, ds \quad \forall \boldsymbol{w} \in (Q_{\boldsymbol{p}_\eta})^2.$$

Notice that, by integration by parts, we also have

$$\int_{K_I} \boldsymbol{\mathcal{D}}_h\eta \cdot \boldsymbol{w}\, d\boldsymbol{x} = \int_{K_I} \boldsymbol{\nabla}\eta \cdot \boldsymbol{w}\, d\boldsymbol{x} + \int_{\partial K_I} (\hat{\eta}-\eta)\,\boldsymbol{w}\cdot\boldsymbol{n}\, ds \quad \forall \boldsymbol{w} \in (Q_{\boldsymbol{p}_\eta})^2.$$

Hence, as observed also in Bassi and Rebay (1997a), it is possible to recast the discrete gradient operator as the sum of two contributions, of which the first one takes into account the element-wise gradient of η while the second one takes into account its jumps (across the element interfaces).

In the same way the discrete partial derivatives of η are defined:

$$\begin{aligned}\int_{K_I} \mathcal{D}_h^x\eta\,\psi_{I,m}\, d\boldsymbol{x} &= -\int_{K_I} \eta\frac{\partial\psi_{I,m}}{\partial x}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\psi_{I,m}\boldsymbol{i}\cdot\boldsymbol{n}\, ds \qquad (5.16)\\ &= \int_{K_I} \frac{\partial\eta}{\partial x}\psi_{I,m}\, d\boldsymbol{x} + \int_{\partial K_I} (\hat{\eta}-\eta)\psi_{I,m}\boldsymbol{i}\cdot\boldsymbol{n}\, ds, \quad m = 1,\dots (p_I^\eta+1)^2.\end{aligned}$$

and

$$\begin{aligned}\int_{K_I} \mathcal{D}_h^y\eta\,\psi_{I,m}\, d\boldsymbol{y} &= -\int_{K_I} \eta\frac{\partial\psi_{I,m}}{\partial y}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\psi_{I,m}\boldsymbol{j}\cdot\boldsymbol{n}\, ds \qquad (5.17)\\ &= \int_{K_I} \frac{\partial\eta}{\partial y}\psi_{I,m}\, d\boldsymbol{x} + \int_{\partial K_I} (\hat{\eta}-\eta)\psi_{I,m}\boldsymbol{j}\cdot\boldsymbol{n}\, ds, \quad m = 1,\dots (p_I^\eta+1)^2.\end{aligned}$$

where $\boldsymbol{i},\boldsymbol{j}$ are the unit vectors along the coordinate axes, while $\boldsymbol{n}$ is the unit normal vector outward from ∂K_I.

Finally notice that in our approach, the coefficients of these discrete partial derivatives of η over the fixed basis are computed trough the solution of (5.16) and (5.17), which is very cheap, since if orthogonal hierarchical bases $\{\psi_{I,m}\}_m$ are used, only diagonal *local* mass matrix are to be inverted, that is trivial.

Thus, we obtain for each element $K_I, I = 1, \ldots, N$:

$$\begin{aligned}
\int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} &= \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^{n}(\boldsymbol{x})d\boldsymbol{x} + \\
&\quad -\theta\Delta t \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\nabla\cdot\Big(\boldsymbol{u}^{n+1}(\boldsymbol{x})h^n(\boldsymbol{x})\Big)d\boldsymbol{x} + \\
&\quad -(1-\theta)\Delta t \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\nabla\cdot\Big(\boldsymbol{u}^{n}(\boldsymbol{x})h^n(\boldsymbol{x})\Big)d\boldsymbol{x}
\end{aligned}$$

$$\begin{aligned}
\int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x} &= -\theta\Delta t g \int_{K_I} \psi_{I,m}(\boldsymbol{x})\left[\mathcal{D}_h^x\eta\right]^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&\quad +\theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&\quad + \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&\quad -g(1-\theta)\Delta t \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^x\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&\quad +(1-\theta)\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x}
\end{aligned}$$

$$\begin{aligned}
\int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x} &= -\theta\Delta t g \int_{K_I} \psi_{I,m}(\boldsymbol{x})\left[\mathcal{D}_h^y\eta\right]^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&\quad -\theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&\quad + \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&\quad -(1-\theta)\Delta t g \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^y\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&\quad -(1-\theta)\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x}
\end{aligned}$$

where an approximation has been made consisting into take the Coriolis parameter constant $f = f_I$ within a given element K_I, as suggested by the high resolution of the mesh we have in mind.

Then after integration by parts in the continuity equation, and after use of definitions (5.16) and (5.17) in the momentum equations we have:

$$
\begin{aligned}
&\int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&-\theta\Delta t \int_{K_I} \nabla\varphi_{I,l}(\boldsymbol{x}) \cdot \boldsymbol{u}^{n+1}(\boldsymbol{x})h^n(\boldsymbol{x})d\boldsymbol{x} + \\
&+\theta\Delta t \int_{\partial K_I} \varphi_{I,l}(s)\hat{\boldsymbol{u}}^{n+1}(s) \cdot \boldsymbol{n}(s)\hat{h}^n(s)ds = \\
&= \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^n(\boldsymbol{x})d\boldsymbol{x} \\
&+(1-\theta)\Delta t \int_{K_I} \nabla\varphi_{I,l}(\boldsymbol{x}) \cdot \boldsymbol{u}^n(\boldsymbol{x})h^n(\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta t \int_{\partial K_I} \varphi_{I,l}(s)\hat{\boldsymbol{u}}^n(s) \cdot \boldsymbol{n}(s)\hat{h}^n(s)ds,
\end{aligned} \tag{5.18}
$$

$$
\begin{aligned}
&\int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&-\theta\Delta tg \int_{K_I} \frac{\partial\psi_{I,m}}{\partial x}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&+\theta\Delta tg \int_{\partial K_I} \psi_{I,m}(s)\hat{\eta}^{n+1}(s)\boldsymbol{i} \cdot \boldsymbol{n}(s)ds + \\
&-\theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x} = \\
&= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta tg \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^x\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&+(1-\theta)\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x},
\end{aligned} \tag{5.19}
$$

$$
\begin{aligned}
&\int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&-\theta\Delta tg \int_{K_I} \frac{\partial\psi_{I,m}}{\partial y}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} + \\
&+\theta\Delta tg \int_{\partial K_I} \psi_{I,m}(s)\hat{\eta}^{n+1}(s)\boldsymbol{j} \cdot \boldsymbol{n}(s)ds + \\
&+\theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x} = \\
&= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta tg \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^y\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x},
\end{aligned} \tag{5.20}
$$

where $\boldsymbol{i}, \boldsymbol{j}$ are the unit vectors along the coordinate axes, while $\boldsymbol{n}$ is the unit normal vector outward from ∂K_I.

Notice that elements of V^{DG} are not defined on the elements of $\mathcal{E}_{h,int}$: therefore terms with hat in the equations are not well defined and must be replaced by appropriate numerical fluxes that depend on *both* interface states and which introduce a coupling between the unknowns of neighboring elements which would be otherwise completely missing. In fact the hallmark of the DG method is that the solution is allowed to be discontinuous over the elemental boundaries and that the elements are coupled by numerical fluxes only, as in the finite volume method.

In this context we will use centered fluxes as in Bassi and Rebay (1997a), i.e. for each dependent variable $d \in V_h^{DG}$ we define:

$$\hat{d}\big|_e = \{\{d\}\}, \quad \forall e \in \mathcal{E}_{h,int}, \tag{5.21}$$

(see def (2.2)).

Expanding the dependent variables in terms of the basis functions one has

$$\eta^n(\boldsymbol{x})\big|_{K_I} = \sum_{r=1}^{(p_I^\eta+1)^2} \varphi_{I,r}(\boldsymbol{x})\eta_{I,r}^n, \qquad u^n(\boldsymbol{x})\big|_{K_I} = \sum_{r=1}^{(p_I^u+1)^2} \psi_{I,r}(\boldsymbol{x})u_{I,r}^n. \tag{5.22}$$

A central issue in finite element formulations for fluid problems is the choice of the approximation space for the velocity and pressure variables (in the context of SWE the role of the pressure is played by the free surface elevation). In fact an inconsistent choice of the two approximation spaces may result in a solution which is polluted by spurious modes (see for example Walters (1983) or Walters and Carey (1983) for the SWE or, for general reference, Quarteroni and Valli (1994)). We have not tried to address this issue from a theoretical point of view. Instead the effects of different element choices for the velocity-pressure pairs on the stability of the approximate solution have been investigated by numerical experiments, showing that mixed orders $Q_k - Q_{k-1}$ velocity-pressure pairs (structured meshes of quadrilaterals are employed) work better then equal order ones, for which clear instabilities arise (see section 6.1). Therefore in this work we will assume:

$$\boxed{p_I^u = p_I^\eta + 1, \quad I = 1, \ldots, N} \tag{5.23}$$

This choice is not surprising, a similar arrangement having been successfully used, investigated and analyzed in the context of continuous Galerkin discretization of SWE in Williams and Zienkiewicz (1981) in the case of low polynomial degrees. Up to now benefits on the stability from the use of mixed order velocity-pressure pair instead of $Q_k - Q_k$ for DG discretizations were proved for the Stokes problem (Toselli (2002), Schötzau *et al.* (2003)) only, but the fact that typical atmospheric flow regimes are characterized by small Froude/Mach numbers suggested the extension of the same strategy to SWE too. In fact, as well underlined in Le Roux *et al.* (1998), in the absence of much in the way of theoretical results for compressible flows, the general conclusion that can be drawn from the finite-element literature is that one should employ a finite-element pair that is suitable for the incompressible case, that is, that satisfies the LBB condition. After all, it seems reasonable to ask that a good element pair for compressible flow should also perform acceptably well for the special case of incompressible flow. Finally notice that this mixed order choice for the pressure-velocity pair can be regarded as the analog for hierarchical basis oriented schemes of staggering in the nodal basis oriented finite elements or in finite difference framework Winninghoff (1968), Arakawa and Lamb (1977) .

In fact the phenomenon of the presence of spurious solutions due to small-scale artifacts introduced by the spatial discretization scheme, can occur in both finite-element and finite-difference formulations Arakawa (1966).

A second key point to be discussed at this stage of the description of SISLDG however concerns the efficiency of the proposed method. Using a constant high polynomial degree of the basis functions in the whole computational domain can lead to an unreasonably high CPU effort. Hence to increase computational efficiency of the SISLDG method the choice has been done (as evident from (5.22)) to make the polynomial degrees p_I^η and p_I^u *local* i.e. dependent on the element K_I, in order to dynamically adjust the number of local degrees of freedom employed to the local structure of the solution . So at this point the question is how to choose the p_I^η and p_I^u *automatically* for each element K_I ? This issue is addressed in section 5.4 where a simple but effective (see 6.1) p-adaptivity criterion is employed, which massively takes advantage from the flexibility of DG discretization and from the orthogonality property of the hierarchical basis used.

Substituting expressions (5.22), into the numerical fluxes definitions (5.21) and then also into equations (5.18)-(5.20), one obtains the full space and time discretization of system (4.1), that can be conveniently presented employing a vector notation.

We denote by

$$\boldsymbol{\eta}_I^n = (\eta_{I,j}^n)_{j=1}^{(p_I^\eta+1)^2}, \quad \boldsymbol{u}_I^n = (u_{I,j}^n)_{j=1}^{(p_I^u+1)^2}, \quad \boldsymbol{v}_I^n = (v_{I,j}^n)_{j=1}^{(p_I^u+1)^2}$$

the vectors collecting all the discrete degrees of freedom associated to element K_I.

Furthermore, we introduce the mass matrices of η, u, v, denoted by M_I^η, M_I^u, M_I^v respectively. These matrices are diagonal if orthogonal (hierarchical) basis functions are used and their entries are given by

$$(M_I^\eta)_{l,r} = \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\varphi_{I,r}(\boldsymbol{x})d\boldsymbol{x}, \quad (M_I^u)_{l,r} = (M_I^v)_{l,r} = \int_{K_I} \psi_{I,l}(\boldsymbol{x})\psi_{I,r}(\boldsymbol{x})d\boldsymbol{x}, \quad (5.24)$$

were obviously one has $l, r = 1, \ldots, (p_I^\eta + 1)^2$ for η and $l, r = 1, \ldots, (p_I^u + 1)^2$ for u, v.

If we consider the partition $\partial K_I = e_{S,I} \cup e_{E,I} \cup e_{N,I} \cup e_{W,I}, \quad I = 1, \ldots, N$ according to the convention of figure 5.2, we also set (for the continuity equation) :

$$(L_I^{\eta u})_{l,r} = -\frac{1}{2}\int_{e_{W,I}} \varphi_{I,l}\hat{h}^n\psi_{I_W,r}dy, \quad (L_I^{\eta v})_{l,r} = -\frac{1}{2}\int_{e_{S,I}} \varphi_{I,l}\hat{h}^n\psi_{I_S,r}dx, \quad (5.25)$$

$$(D_I^{\eta u})_{l,r} = \int_{K_I} \frac{\partial\varphi_{I,l}}{\partial x}h^n\psi_{I,r}d\boldsymbol{x} - \frac{1}{2}\int_{e_{W,I}} \varphi_{I,l}\hat{h}^n\psi_{I,r}dy + \frac{1}{2}\int_{e_{E,I}} \varphi_{I,l}\hat{h}^n\psi_{I,r}dy,$$

$$(D_I^{\eta v})_{l,r} = \int_{K_I} \frac{\partial\varphi_{I,l}}{\partial y}h^n\psi_{I,r}d\boldsymbol{x} - \frac{1}{2}\int_{e_{S,I}} \varphi_{I,l}\hat{h}^n\psi_{I,r}dx + \frac{1}{2}\int_{e_{N,I}} \varphi_{I,l}\hat{h}^n\psi_{I,r}dx,$$

$$(U_I^{\eta u})_{l,r} = \frac{1}{2}\int_{e_{E,I}} \varphi_{I,l}\hat{h}^n\psi_{I_E,r}dy, \quad (U_I^{\eta v})_{l,r} = \frac{1}{2}\int_{e_{N,I}} \varphi_{I,l}\hat{h}^n\psi_{I_N,r}dx,$$

and (for the momentum equation along x):

$$(L_I^{u\eta})_{l,r} = -\frac{1}{2}\int_{e_{W,I}} \psi_{I,l}\varphi_{I_W,r}dy, \quad (U_I^{u\eta})_{l,r} = \frac{1}{2}\int_{e_{E,I}} \psi_{I,l}\varphi_{I_E,r}dy, \quad (5.26)$$

$$(D_I^{u\eta})_{l,r} = \int_{K_I} \frac{\partial\psi_{I,l}}{\partial y}\varphi_{I,r}d\boldsymbol{x} - \frac{1}{2}\int_{e_{S,I}} \psi_{I,l}\varphi_{I,r}dx + \frac{1}{2}\int_{e_{N,I}} \psi_{I,l}\varphi_{I,r}dx,$$

$$
\begin{aligned}
(\mathcal{U}_I^n)_l &= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta t g\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^x\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&+(1-\theta)\Delta t f_I\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x},
\end{aligned}
\tag{5.27}
$$

and (for the momentum equation along y):

$$
\begin{aligned}
(L_I^{v\eta})_{l,r} &= -\frac{1}{2}\int_{e_{S,I}} \psi_{I,l}\varphi_{I_S,r}dx, \quad (U_I^{v\eta})_{l,r} = \frac{1}{2}\int_{e_{N,I}} \psi_{I,l}\varphi_{I_N,r}dx, \\
(D_I^{u\eta})_{l,r} &= \int_{K_I} \frac{\partial\psi_{I,l}}{\partial x}\varphi_{I,r}d\boldsymbol{x} - \frac{1}{2}\int_{e_{W,I}} \psi_{I,l}\varphi_{I,r}dy + \frac{1}{2}\int_{e_{E,I}} \psi_{I,l}\varphi_{I,r}dy,
\end{aligned}
\tag{5.28}
$$

$$
\begin{aligned}
(\mathcal{V}_I^n)_l &= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta t g\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^y\eta\right]^n\big](\boldsymbol{x})d\boldsymbol{x} + \\
&-(1-\theta)\Delta t f_I\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x}.
\end{aligned}
\tag{5.29}
$$

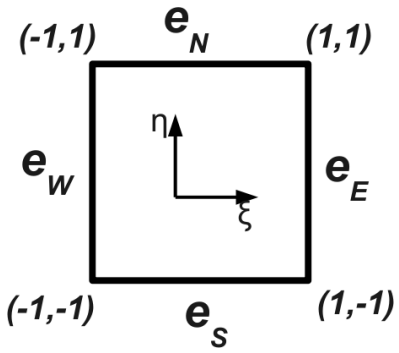


Figure 5.2: The geometry of the mapping on the reference element $[-1,1]^2$ and the associated metric information.

Notice that all the matrices whose entries are defined in (5.25), (5.26) and (5.28) are in general rectangular, of dimensions $(p_I^\eta+1)^2\times(p_J^u+1)^2$, where $J\in\{I,I_S,I_E,I_N,I_W\}$ for matrices defined in (5.25), while of dimension $(p_I^u+1)^2\times(p_J^\eta+1)^2$, where $J\in\{I,I_S,I_E,I_N,I_W\}$ for the matrices defined in (5.26) and (5.28).

Furthermore, all the matrices whose entries are defined in (5.26) and (5.28) are time-independent and therefore they can be computed once for all at the beginning of the computation. This allows a saving of computational effort and hence gives a motivation of the matrix approach here introduced.

After the definitions (5.25)-(5.29), the equations (5.18)-(5.20) defining the SISLDG method for the system (4.1) read in vector form:

$$
\begin{aligned}
&M_I^{\eta}\boldsymbol{\eta}_I^{n+1} + \\
&-\theta\Delta t\big(L_I^{\eta u}\boldsymbol{u}_{I_W}^{n+1} + D_I^{\eta u}\boldsymbol{u}_I^{n+1} + U_I^{\eta u}\boldsymbol{u}_{I_E}^{n+1} + \\
&\qquad +L_I^{\eta v}\boldsymbol{v}_{I_S}^{n+1} + D_I^{\eta v}\boldsymbol{v}_I^{n+1} + U_I^{\eta v}\boldsymbol{v}_{I_N}^{n+1}\big) = \\
&= M_I^{\eta}\boldsymbol{\eta}_I^{n} + \\
&-(1-\theta)\Delta t\big(L_I^{\eta u}\boldsymbol{u}_{I_W}^{n} + D_I^{\eta u}\boldsymbol{u}_I^{n} + U_I^{\eta u}\boldsymbol{u}_{I_E}^{n} + \\
&\qquad +L_I^{\eta v}\boldsymbol{v}_{I_S}^{n} + D_I^{\eta v}\boldsymbol{v}_I^{n} + U_I^{\eta v}\boldsymbol{v}_{I_N}^{n}\big),
\end{aligned}
\tag{5.30}
$$

$$
M_I^{u}\boldsymbol{u}^{n+1} - \theta\Delta t g\left(L_I^{u\eta}\boldsymbol{\eta}_{I_W}^{n+1} + D_I^{u\eta}\boldsymbol{\eta}_I^{n+1} + U_I^{u\eta}\boldsymbol{\eta}_{I_E}^{n+1}\right) - \theta\Delta f_I M_I^{v}\boldsymbol{v}_I^{n+1} = \boldsymbol{\mathcal{U}}_I^{n}, \tag{5.31}
$$

$$
M_I^{v}\boldsymbol{v}^{n+1} - \theta\Delta t g\left(L_I^{v\eta}\boldsymbol{\eta}_{I_S}^{n+1} + D_I^{v\eta}\boldsymbol{\eta}_I^{n+1} + U_I^{v\eta}\boldsymbol{\eta}_{I_N}^{n+1}\right) + \theta\Delta f_I M_I^{u}\boldsymbol{u}_I^{n+1} = \boldsymbol{\mathcal{V}}_I^{n}. \tag{5.32}
$$

As customary in semi-implicit methods, see e.g. Staniforth and Temperton (1986), Temperton and A.Staniforth (1987),Casulli and Cheng (1990), Casulli (1990),Casulli and Cattani (1994), and in the pressure methods Casulli and Greenspan (1984), $\boldsymbol{u}^{n+1}$ and $\boldsymbol{v}^{n+1}$ are expressed in terms of $\boldsymbol{\eta}^{n+1}$ and the resulting expressions substituted into the continuity equation (5.30), in order to obtain a discrete (vector) Helmholtz equation in the $\boldsymbol{\eta}^{n+1}$ unknown only. Notice that this substitution is performed at the *discrete* level following closer Casulli and Greenspan (1984),Casulli and Cheng (1990), Casulli (1990),Casulli and Cattani (1994), than Staniforth and Temperton (1986), Temperton and A.Staniforth (1987) where it is performed at continuous level. If the elements surrounding K_I are labeled according to figure 5.3, this equation can be rewritten as follows:

$$
\begin{aligned}
& & K_I^{WW}\boldsymbol{\eta}_{I_{WW}}^{n+1} & + & & \\
& +K_I^{SW}\boldsymbol{\eta}_{I_{SW}}^{n+1} & +K_I^{W}\boldsymbol{\eta}_{I_W}^{n+1} + & K_I^{NW}\boldsymbol{\eta}_{I_{NW}}^{n+1} & + \\
+K_I^{SS}\eta_{I_{SS}}^{n+1} + & K_I^{S}\boldsymbol{\eta}_{I_S}^{n+1} & +K_I\boldsymbol{\eta}_I^{n+1} + & K_I^{N}\boldsymbol{\eta}_{I_N}^{n+1} & + K^{NN}\boldsymbol{\eta}_{I_N N}^{n+1} + \\
& +K_I^{SE}\boldsymbol{\eta}_{I_{SE}}^{n+1} & +K_I^{E}\boldsymbol{\eta}_{I_E}^{n+1} + & K_I^{NE}\boldsymbol{\eta}_{I_{NE}}^{n+1} & + \\
& + & K_I^{EE}\boldsymbol{\eta}_{I_{EE}}^{n+1} & = & \\
& = & \boldsymbol{\mathcal{N}}_I^{n} & , &
\end{aligned}
\tag{5.33}
$$

where, if

$$
c_I = \theta\Delta t f_I, \quad c_{1,I} = \frac{-\left(\theta\Delta t\right)^2 g}{1+(\theta\Delta t f_I)^2}, \quad c_{2,I} = c_I c_{1,I},
$$

$$
c_{3,I} = \frac{1}{1+(\theta\Delta t f_I)^2}, \quad c_{4,I} = \frac{-\left(\theta\Delta t\right) g}{1+(\theta\Delta t f_I)^2}, \quad c_{5,I} = c_I c_{4,I},
$$

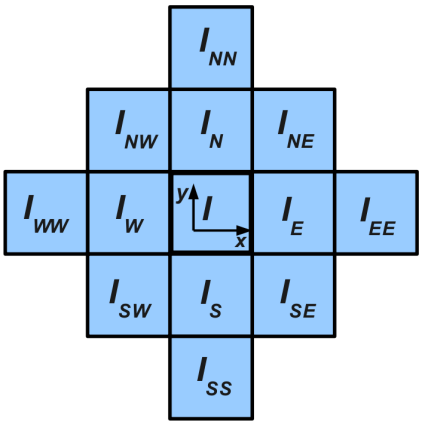


Figure 5.3: The computational stencil for the semi-implicit step of SISLDG and the names of the elements surrounding the elements K_I.

we have set (from 'west' to 'east' and from 'south' to 'north'):

$$
\begin{aligned}
K_I^{WW} &= c_{1,I} L_I^{\eta u} (M_{I_W}^u)^{-1} L_{I_W}^{u\eta}, \\
K_I^{SW} &= c_{2,I} L_I^{\eta u} (M_{I_W}^u)^{-1} L_{I_W}^{v\eta} - c_{2,I_S} L_I^{\eta v} (M_{I_S}^u)^{-1} L_{I_S}^{u\eta}, \\
K_I^{W} &= c_{1,I} \left[L_I^{\eta u} (M_{I_W}^u)^{-1} \left(D_{I_W}^{u\eta} + c_I D_{I_W}^{v\eta} \right) + \left(D_I^{\eta u} - c_I D_I^{\eta v} \right) (M_I^u)^{-1} L_I^{u\eta} \right], \\
K_I^{NW} &= c_{2,I} L_I^{\eta u} (M_{I_W}^u)^{-1} U_{I_W}^{v\eta} - c_{2,I_N} U_I^{\eta v} (M_{I_N}^u)^{-1} L_{I_N}^{u\eta}, \\
K_I^{SS} &= c_{1,I_S} L_I^{\eta v} (M_{I_S}^v)^{-1} L_{I_S}^{v\eta}, \\
K_I^{S} &= c_{1,I} \left(c_I D_I^{\eta u} + D_I^{\eta v} \right) (M_I^u)^{-1} L_I^{v\eta} + c_{1,I_S} L_I^{\eta v} (M_{I_S}^u)^{-1} (-c_{I_S} D_{I_S}^{u\eta} + D_{I_S}^{v\eta}), \\
K_I &= M_I^\eta + \\
&\quad + c_{1,I_s} L_I^{\eta v} (M_{I_S}^v)^{-1} U_{I_S}^{v\eta} + \\
&\quad + c_{1,I} [L_I^{\eta u} (M_{I_W}^u)^{-1} U_{I_W}^{u\eta} + \\
&\qquad + (D_I^{\eta u} - c_I D_I^{\eta v}) (M_I^u)^{-1} D_I^{u\eta} + \\
&\qquad + (c_I D_I^{\eta u} + D_I^{\eta v}) (M_I^u)^{-1} D_I^{v\eta} + \\
&\qquad + U_I^{\eta u} (M_{I_E}^u)^{-1} L_{I_E}^{u\eta}] + \\
&\quad + c_{1,I_N} U_I^{\eta v} (M_{I_N}^v)^{-1} L_{I_N}^{v\eta}, \\
K_I^{N} &= c_{1,I} (c_I D_I^{\eta u} + D_I^{\eta v}) (M_I^u)^{-1} U_I^{v\eta} + c_{1,I_N} U_I^{\eta v} (M_{I_N}^u)^{-1} (-c_{I_N} D_{I_N}^{u\eta} + D_{I_N}^{v\eta}), \\
K_I^{NN} &= c_{1,I_N} U_I^{\eta v} (M_{I_N}^v)^{-1} U_{I_N}^{v\eta}, \\
K_I^{SE} &= c_{2,I} U_I^{\eta u} (M_{I_E}^u)^{-1} L_{I_E}^{v\eta} - c_{2,I_S} L_I^{\eta v} (M_{I_S}^u)^{-1} U_{I_S}^{u\eta}, \\
K_I^{E} &= c_{1,I} \left[(D_I^{\eta u} - c_I D_I^{\eta v}) (M_I^u)^{-1} U_I^{u\eta} + U_I^{\eta u} (M_{I_E}^u)^{-1} \left(D_{I_E}^{u\eta} + c_{I_E} D_{I_E}^{v\eta} \right) \right], \\
K_I^{NE} &= c_{2,I} U_I^{\eta u} (M_{I_E}^u)^{-1} U_{I_E}^{v\eta} - c_{2,I_N} U_I^{\eta v} (M_{I_N}^u)^{-1} U_{I_N}^{u\eta}, \\
K_I^{EE} &= c_{1,I} U_I^{\eta u} (M_{I_E}^u)^{-1} U_{I_E}^{u\eta},
\end{aligned}
$$

and finally, as regards the right hand side:

$$
\begin{aligned}
\boldsymbol{\mathcal{N}}_I \;=\;& \boldsymbol{\eta}_I^n + \\
& -\Delta t L_I^{\eta u}\left[\theta c_{3,I_W}(M^u_{I_W})^{-1}\left(\boldsymbol{\mathcal{U}}^n_{I_W} + c_{I_W}\boldsymbol{\mathcal{V}}^n_{I_W}\right) + (1-\theta)\boldsymbol{u}^n_{I_W}\right] + \\
& -\Delta t D_I^{\eta u}\left[\theta c_{3,I}(M^u_I)^{-1}\left(\boldsymbol{\mathcal{U}}^n_I + c_I\boldsymbol{\mathcal{V}}^n_I\right) + \quad (1-\theta)\boldsymbol{u}^n_I\right] \quad + \\
& -\Delta t U_I^{\eta u}\left[\theta c_{3,I_E}(M^u_{I_E})^{-1}\left(\boldsymbol{\mathcal{U}}^n_{I_E} + c_{I_E}\boldsymbol{\mathcal{V}}^n_{I_E}\right) + \quad (1-\theta)\boldsymbol{u}^n_{I_E}\right] + \\
& -\Delta t L_I^{\eta v}\left[\theta c_{3,I_S}(M^u_{I_S})^{-1}\left(-c_{I_S}\boldsymbol{\mathcal{U}}^n_{I_S} + \boldsymbol{\mathcal{V}}^n_{I_S}\right) + \quad (1-\theta)\boldsymbol{v}^n_{I_S}\right] + \\
& -\Delta t D_I^{\eta v}\left[\theta c_{3,I}(M^u_I)^{-1}\left(-c_I\boldsymbol{\mathcal{U}}^n_I + \boldsymbol{\mathcal{V}}^n_I\right) + \quad (1-\theta)\boldsymbol{v}^n_I\right] + \\
& -\Delta t U_I^{\eta v}\left[\theta c_{3,I_N}(M^u_{I_N})^{-1}\left(-c_{I_N}\boldsymbol{\mathcal{U}}^n_{I_N} + \boldsymbol{\mathcal{V}}^n_{I_N}\right) + (1-\theta)\boldsymbol{v}^n_{I_N}\right] \quad .
\end{aligned}
$$

The non-symmetric linear system (5.33) is solved using the GMRES method (Saad and Schultz (1986)). Once $\boldsymbol{\eta}^{n+1}$ is known, $\boldsymbol{u}^{n+1}$ and $\boldsymbol{v}^{n+1}$ can be recovered by substituting $\boldsymbol{\eta}^{n+1}$ back into equations (5.31) and (5.32) respectively.

As final remark, notice that our SISLDG method does *not* rely on operator splitting.

5.3.1 Implementation details

All the integrals appearing in the elemental equations are evaluated by means of Gaussian numerical quadrature formulae with a number of quadrature nodes consistent with the local polynomial degree used. In particular notice that integrals containing functions which are the image through the evolution operator E defined in (5.11) (i.e. functions evaluated at the foot of the trajectories arriving in the quadrature nodes) cannot be computed exactly (see e.g. Morton *et al.* (1988), Priestley (1994)), except the special case of linear advection equation, since such functions are not polynomials, therefore a careful approximation of these integrals is needed.

Secondly, the computation of terms $(\mathcal{U}^n_I)_l$ and $(\mathcal{V}^n_I)_l$ defined in eqs.(5.27) and (5.29) contains the evaluation at quadrature nodes $\boldsymbol{x_q}$ of each element K_I of the action of the evolution operator $E(t^n, \Delta t)$ on $u, v, \mathcal{D}^x_h, \mathcal{D}^y_h$, but notice that the first step of the evaluation of $\left[E(t^n,\Delta t)u\right](\boldsymbol{x_q})$, $\left[E(t^n,\Delta t)v\right](\boldsymbol{x_q})$, $\left[E(t^n,\Delta t)\left[\mathcal{D}^x_h\eta\right]\right](\boldsymbol{x_q})$, $\left[E(t^n,\Delta t)\left[\mathcal{D}^y_h\eta\right]\right](\boldsymbol{x_q})$ i.e. the computation of the foot of the trajectory arriving at $(\boldsymbol{x_q})$ at t^{n+1} is exactly the *same* for $u, v, \mathcal{D}^x_h, \mathcal{D}^y_h$, and therefore can be carried only once for each $\boldsymbol{x_q}$. Moreover also in the interpolation step saving of computational effort can be obtained if *hierarchical* bases are used since in that case, the basis functions to be evaluated at the foot of the trajectory are the *same* for $u, v, \mathcal{D}^x_h, \mathcal{D}^y_h$.

Finally remember that the presence of matrices $(M^u_I)^{-1}$ in previous formulae does not constitute a computational problem since, taking advantage from the flexibility of the discontinuous Galerkin spatial semi-discretization, *orthogonal* bases can be employed in a simple way in order to make local mass matrices *diagonal* and therefore invertible without cost. Moreover if *orthonormal* bases are used, then mass matrices reduce to the identity, so they completely disappear from previous formulae. The use of normalized bases (w.r.t. L^2 norm) gives benefits also from the point of view of the conditioning of previous matrices.

All these remarks show how feasibility of the proposed method can be reached through a proper implementation, but the most powerful way to improve the efficiency of the proposed SISLDG method is represented by the introduction of an adaptivity strategy, one simple example of which is presented in next section.

5.4 A simple p-adaptivity criterion

The numerical method described in section 5.3 can be run in principle (and has been run effectively for several numerical tests, as shown in section 6.1) taking a constant value for the degree of the polynomials defining the local basis on each element. However, our aim is to exploit the great flexibility of the DG spatial discretization by providing our method with an automatic criterion to adapt the local number of degrees of freedom to the nature of the numerical solution. This approach to adaptivity is not only quite convenient from the computational viewpoint, but has in our opinion some specific advantages for realistic environmental applications. Indeed, in NWP, climate or ocean models, $h-$adaptivity approaches (that is, local mesh coarsening or refinement in which the size of some elements changes in time) can be cumbersome in practice, since a great amount of information that is necessary to carry out realistic simulations, such as orography/bathymetry profiles, data on land use and soil type, land-sea masks and very many others, is (as a rule, painfully) reconstructed on the computational mesh and has to be re-interpolated each time the mesh is changed. Furthermore, many physical parametrizations are highly sensitive to the mesh size. Although devising better parametrizations that require less mesh-dependent tuning is an important research goal, it is quite obvious that more conventional parametrizations will still be in use for quite some time. As a consequence, it would be useful to be able to improve the accuracy locally by adding supplementary degrees of freedom where necessary, without having to change the underlying computational mesh. This is exactly what $p-$adaptivity does. Furthermore, if simulations with a large number of reacting species are envisaged, as increasingly common in environmental applications, $h-$adaptivity approaches may lead to mesh refinement for all species due to the necessity of greater accuracy for just a few of them. In $p-$adaptive approaches, on the other hand, the increase in the number of degrees of freedom is totally independent for each species, thus allowing for increasing the accuracy for some specific variable without increasing the computational cost for other variables that do not need refinement.

Various approaches for $p-$adaptivity have been proposed in the literature, see e.g. Flaherty and Moore (1995), Remacle *et al.* (2003), Houston and Süli (2005), Eskilsson (2010). The technique we employ is extremely simple and relies on the use of orthogonal hierarchical tensor-product basis functions.

Consider the local (to the element K_I) representation of some dependent variable $\alpha(\boldsymbol{x})$:

$$\alpha(\boldsymbol{x})\big|_{K_I} = \sum_{j=1}^{(p_I^\alpha+1)^2} a_{I,j}\psi_{I,j}(\boldsymbol{x}) = \sum_{k=1}^{p_I^\alpha+1}\sum_{l=1}^{p_I^\alpha+1} \alpha_{I,k,l}\psi_{I_x,k}(x)\psi_{I_y,l}(y). \tag{5.34}$$

where (remember the construction of tensor product basis in chapter 2) $I = (I_x, I_y)$ is a suitable multi-index relabelling the two-dimensional elements in terms of the one-dimensional ones and $j = (k, l)$ is a suitable multi-index relabelling the two-dimensional degrees of freedom in terms of the one-dimensional ones, such that $a_{I,j} = \alpha_{I,k,l}$ and $\psi_{I,j}(\boldsymbol{x}) = \psi_{I_x,k}(x)\psi_{I_y,l}(y)$.

Notice that p_I^α will have different values for each different dependent variable α. Due to the finite element spaces chosen for the SISLDG discretization, one will have $p_I^u = p_I^\eta + 1$, while the local degree of the polynomials used for the tracers can in principle be chosen independently for each different tracer.

If a normalized *orthogonal hierarchical* basis is employed, due to the orthonormality

of the chosen basis it is well known (Parseval identity) that

$$\mathcal{E}^{tot} = \|P\alpha\|^2 = \sum_{j=1}^{(p_I^\alpha+1)^2} a_{I,j}^2 = \sum_{k,l=1}^{p_I^\alpha+1} \alpha_{I,k,l}^2 \tag{5.35}$$

where P is the L^2 projector onto the local polynomial subspace.

Now combining (5.35) with the *hierarchical* property of the basis, for any integer $r = 1, \dots, p^\alpha + 1$ we can define the "*energy*" contained in the $r-$th modal components of $\alpha\big|_{K_I}$ for a given element $K_I \in \mathcal{T}_h$ as:

$$\mathcal{E}^r := \sum_{max(k,l)=r} \alpha_{I,k,l}^2 \tag{5.36}$$

In other words $\mathcal{E}^{r-1}$ will contain the 'energy' associated to the region highlighted in figure 5.4 in the plane $p_x - p_y$.

Therefore, for any integer $r = 1, \dots, p_I^\alpha + 1$, the quantity

$$w_r = \sqrt{\frac{\mathcal{E}^r}{\mathcal{E}^{tot}}}$$

will measure the relative weight of the $r-$th modal components of α with respect to the best approximation available for the L^2 norm of α. Assuming that α denotes a generic model variable at the beginning of the computation of a generic time step, the proposed adaptation criterion simply consists in the following algorithm:

Given an error tolerance $\epsilon_I > 0$ for all $I = 1, \dots, N$, :

1) compute w_{p_I}

2.1) if $w_{p_I} \geq \epsilon_I$, then

2.1.1) set $p_I(\alpha) := p_I(\alpha) + 1$

2.1.2) set $\alpha_{I,p_I} = 0$, exit the loop and go the next element

2.2) if instead $w_{p_I} < \epsilon_I$, then

2.2.1) compute w_{p_I-1}

2.2.2) if $w_{p_I-1} \geq \epsilon_I$, exit the loop and go the next element

2.2.3) else if $w_{p_I-1} < \epsilon_I$, set $p_I(\alpha) := p_I(\alpha) - 1$ and go back to 2.2.1.

Once the values of p_I have been updated, in order to avoid too abrupt variation of the local polynomial degree and to ensure that the values employed do not go beyond a maximum and minimum allowed values p_{max}, p_{min}, respectively, for each element we set

$$p_I := \min\Big\{ \max\big\{p_I, \frac{p_{I_S}}{2}, \frac{p_{I_E}}{2}, \frac{p_{I_N}}{2}, \frac{p_{I_W}}{2}, p_{min}\big\}, p_{max}\Big\}.$$

At the beginning of the simulation, all the variables are initialized with the maximum possible number of local degrees of freedom. The adaptation algorithm then runs preliminary to any computation in each new time step, including the first.

As a result, from the first time step only the degrees of freedom up to the local maximum value are updated. In our simulations, the tolerance has been kept fixed for all elements and variables, but the method could be refined by choosing different values for particular purposes. Although one dimensional tests are definitely not sufficient to assess the efficiency improvement coming from the proposed approach, preliminary testing reported in section 6.1 seems to show that the presented $p-$adaptivity criterion effectively does work.

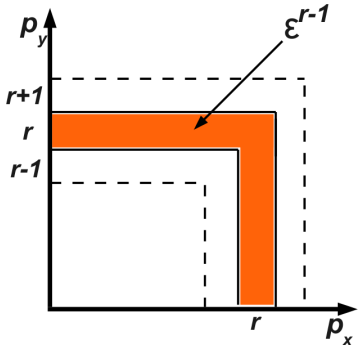


Figure 5.4: The region on the $p_x - p_y$ plane of the degree of freedom which contributes to $\mathcal{E}^{r-1}$.

Chapter 6

Numerical validation

In this chapter a numerical validation of the novel SISLDG discretization of the rotating shallow water equations is presented with two main purposes: asses the stability and accuracy properties of the proposed numerical scheme and analyse the impact of the p-adaptivity criterion on the computed solution. Stability is studied considering the size of the maximum allowable Courant number.

Introduction

The numerical method introduced in section 5.3 has been implemented and tested on a number of classical benchmarks. Due to the nature of semi-Lagrangian method, strong unstationary shocks cannot be approximated properly. Indeed, it is well known that an accurate computation of the shock wave speed can only be achieved if the numerical method is based on the conservative form of the momentum equation, while the non conservative formulation is used in the present approach. Therefore, it is to be expected that discontinuous solutions such as those arising in the 'dam break' Riemann problem will be reproduced with incorrect propagation speed, which indeed happens also with our method. On the other hand, as highlighted in the introduction, the present kind of numerical technique is aimed at achieving high computational efficiency for low Mach/Froude number problems, as those encountered in many typical environmental applications. Moreover, since we have focused on low Mach/Froude number flows, where the typical velocity is much smaller than that of the fastest propagating waves (this being also the motivation for the use of semi-implicit methods), in the following we will distinguish between the maximum Courant number based on the velocity $C_{vel} := \frac{u\Delta t}{\Delta x/p}$ and the the maximum Courant number based on the celerity $C_{cel} := \frac{(u+\sqrt{gh})\Delta t}{\Delta x/p}$. In all the test cases with analytic solution, the relative errors in the L^1, L^2 and L^∞ at time T are defined, for a generic variable ζ, as

$$E_1^\zeta = \frac{\|\zeta(\cdot,T) - \zeta_{ex}(\cdot,T)\|_{L^1}}{\|\zeta_{ex}(\cdot,T)\|_{L^1}}, \quad E_2^\zeta = \frac{\|\zeta(\cdot,T) - \zeta_{ex}(\cdot,T)\|_{L^2}}{\|\zeta_{ex}(\cdot,T)\|_{L^2}}, \quad E_\infty^\zeta = \frac{\|\zeta(\cdot,T) - \zeta_{ex}(\cdot,T)\|_{L^\infty}}{\|\zeta_{ex}(\cdot,T)\|_{L^\infty}}$$

where ζ_{ex} denotes the analytic solution and the integral norms are computed by appropriate numerical quadrature rules.

6.1 One dimensional tests

The performance of the proposed numerical method for the integration of (4.1) is first assessed in the one-dimensional case, both linearised and fully non-linear. While such analysis does not completely cover the general problem, it gives necessary conditions for stability, and anyway an important measure of performance (as underlined in Purnell (1975)). A more general discussion is given in S.A.Orszag (1971).

6.1.1 Accuracy assessment on non-linear solutions

A first test has been carried out in order to check that the spatial discretization is indeed implemented correctly. To do this on a non trivial benchmark, we considered an all rarefaction wave solution of the Riemann problem for equations (4.1) with $f = 0$ and $h(x,0) = h_0$, $u(x,0) = \begin{cases} u_l, & \text{if } x < 0 \\ \\ u_r, & \text{if } x \geq 0 \end{cases}$ whose exact solution is given by:

$$\begin{pmatrix} h(x,t) \\ u(x,t) \end{pmatrix} = \begin{cases} \begin{pmatrix} h_0 \\ u_l \end{pmatrix}, & \text{if } x \leq (u_l - \sqrt{gh_0})t \\ \begin{pmatrix} \frac{1}{9g}(u_l + 2\sqrt{gh_0} - x/t)^2 \\ \frac{1}{3}(u_l + 2\sqrt{gh_0} + 2x/t) \end{pmatrix}, & \text{if } (u_l - \sqrt{gh_0})t \leq x \leq (\frac{1}{4}u_l + \frac{3}{4}u_r - \sqrt{gh_0})t \\ \begin{pmatrix} \frac{1}{16g}(u_l - u_r + 4\sqrt{gh_0})^2 \\ \frac{u_l + u_r}{2} \end{pmatrix}, & \text{if } (\frac{1}{4}u_l + \frac{3}{4}u_r - \sqrt{gh_0})t \leq x \leq (\frac{3}{4}u_l + \frac{1}{4}u_r + \sqrt{gh_0})t \\ \begin{pmatrix} \frac{1}{9g}(-u_r + 2\sqrt{gh_0} + x/t)^2 \\ \frac{1}{3}(u_r - 2\sqrt{gh_0} + 2x/t) \end{pmatrix}, & \text{if } (\frac{3}{4}u_l + \frac{1}{4}u_r + \sqrt{gh_0})t \leq x \leq (u_r + \sqrt{gh_0})t \\ \begin{pmatrix} h_0 \\ u_r \end{pmatrix}, & \text{if } x \geq (u_r + \sqrt{gh_0})t \end{cases}$$

In all the computed solutions, $\theta = 0.6$. A plot of the solution is displayed in figure 6.1.

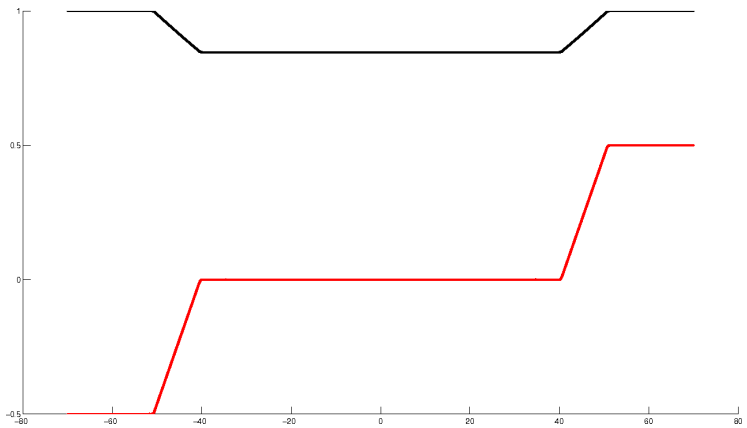


Figure 6.1: Solution of all rarefaction wave Riemann problem for at $T = 16.0$ s.

Firstly, a convergence test for the spatial discretization was performed, assuming a small and fixed value Δt, taking $p_\eta = 1$ and increasing the number of elements. Results are reported in table 6.1.

Another convergence test was performed at fixed small Courant number C_{cel}. Results are reported in table 6.2.

N_{el}	C_{vel}	C_{cel}	E_2^η	E_1^{eta}	E_∞^η
6	6.43e-3	4.68e-2	0.203	0.15	0.37
13	1.26e-2	9.14e-2	7.56e-2	5.78e-2	0.11
25	2.32e-2	0.17	2.90e-2	2.12e-2	7.15e-2
50	4.56e-2	0.33	1.26e-2	7.14e-3	4.18e-2
100	9.02e-2	0.65	4.21e-3	1.98e-3	2.07e-2

Table 6.1: Relative errors on the free surface elevation in L^2, L^1, L^∞ norms, $p_\eta = 1$ and increasing number of elements at fixed Δt.

N_{el}	Δt	N_{tstep}	E_2^η	E_1^η	E_∞^η
6	0.125	8	0.203	0.15	0.37
13	0.0625	16	7.56e-2	5.77e-2	0.11
25	0.0312	32	2.88e-2	2.11e-2	7.29e-2
50	0.0156	64	1.27e-2	7.40e-3	4.02e-2
100	0.0078	128	4.47e-3	2.38e-3	2.09e-2

Table 6.2: Relative errors on the free surface elevation in L^2, L^1, L^∞ norms, $p_\eta = 1$ and increasing number of elements at fixed small C_{cel}.

Finally, the same test was run keeping mesh size and time step constant, while increasing the degree of the polynomial spaces, see tab. 6.3.

p_η	p_u	E_2^η	E_1^η	E_∞^η
0	1	3.89e-2	1.90e-2	1.50e-1
1	2	4.27e-3	1.55e-3	2.48e-2
2	3	3.18e-4	1.36e-4	2.39e-3

Table 6.3: Relative errors on the free surface elevation in L^2, L^1, L^∞ norms of when the the polynomial degree is increased and Δx, Δt are kept constant.

6.1.2 Gravity wave propagation and geostrophic adjustment

For environmental applications, correct reproduction of gravity wave solutions of (4.1) is essential, both in the rotating and non rotating case.

Gravity waves propagation with no rotation

The one dimensional linear and non-linear shallow water equations (4.1) have been solved using the scheme presented in section 5.3. In order to simulate propagation of gravity waves on geophysical scales, we considered first the non rotating case $f = 0$ on a domain with length $L = 1.8 \times 10^4$ km and a reference height $H = 2$ km. The solutions computed by discretizing the linear and non-linear equations are shown in figures 6.2 and 6.3a, respectively, as computed using $\Delta x = 180$ km, $\theta = 0.54$ and a value of the Courant number $C \ll 1.$. The computed solutions are in good agreement with the theory and the deformation of the initial profile due to the non-linear effects is clearly displayed in figure 6.3a. Moreover figure 6.3a should be compared with 6.3b where, for equal order pressure-velocity pair, clear instabilities arise.

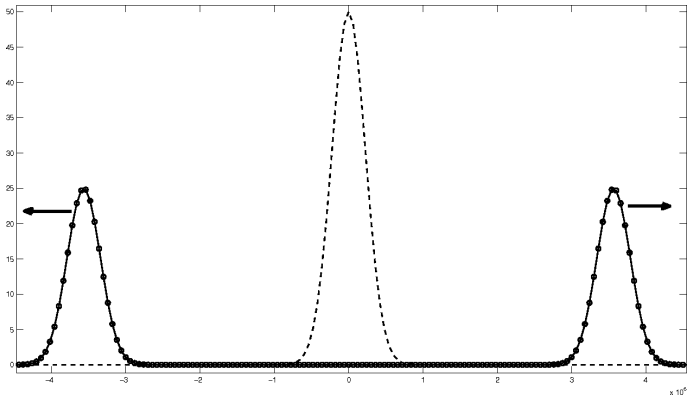


Figure 6.2: Numerical solution for the free surface elevation field for the linear one-dimensional SWE. Dashed curve: initial condition. Solid curve: solution at $T = 10$ h. Circles denote nodal values.

For a more quantitative assessment of the spatial accuracy of the method, the errors with respect to the analytic solution available in the linear case are reported in table 6.4. Again, a small and fixed Courant number is chosen, in order to let the spatial discretization error be the dominant error component, while employing different polynomial orders for the local representation of the solution.

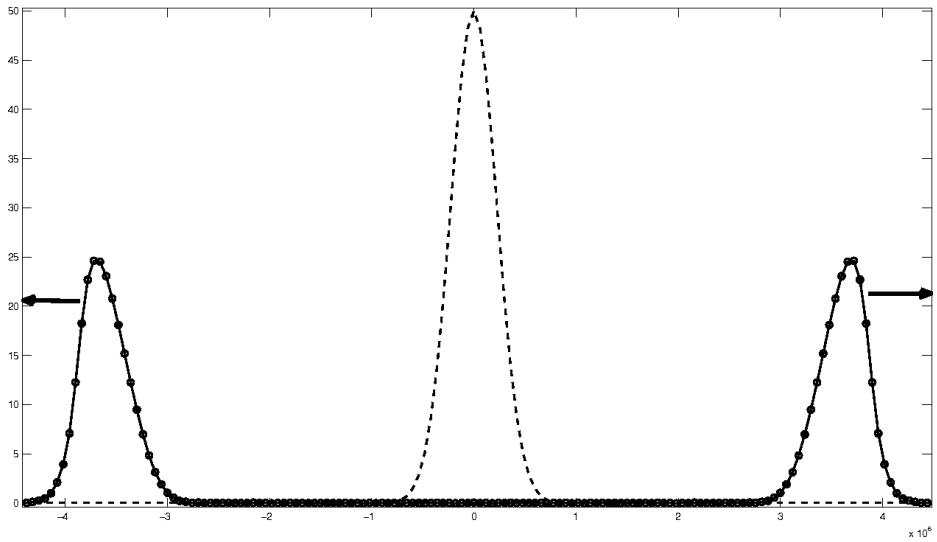
p_η	p_u	E^η_∞	E^η_2	E^η_1
1	2	0.44	0.48	0.67
2	3	1.62×10^{-2}	2.21×10^{-2}	2.12×10^{-2}
3	4	5.05×10^{-4}	7.34×10^{-4}	6.56×10^{-4}
4	5	3.07×10^{-5}	4.70×10^{-5}	4.31×10^{-5}

Table 6.4: Relative errors in L^∞, L^2, L^1 norm for linearised non-rotating SWE at fixed Courant number $C \ll 1$, $\theta = 0.501$.

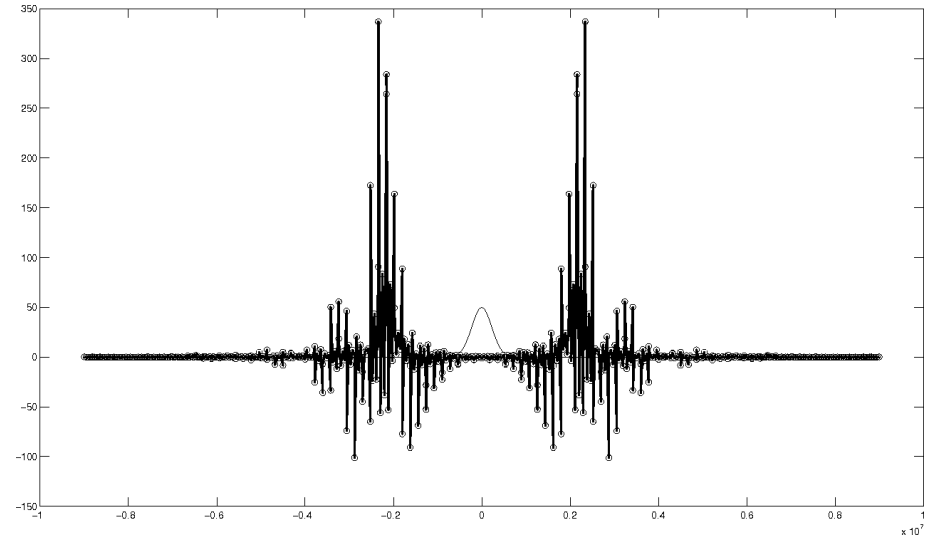
On the other hand, in order to assess the accuracy and robustness of the time discretization, the same test was run with reference height $H = 1$ km, $\Delta x = 18$ km and $p_\eta = 3, p_u = 4$ using different values for the time step Δt. The results are summarized in table 6.5. A more comprehensive set of results in the same test case is also shown in table 6.6.

Δt	C_{cel}	E^η_2	E^η_1	E^η_∞
500	4.47	8.40×10^{-2}	9.37×10^{-2}	8.79×10^{-2}
250	2.23	2.17×10^{-2}	2.40×10^{-2}	2.19×10^{-2}
125	1.14	5.51×10^{-3}	6.07×10^{-3}	5.79×10^{-3}
62.5	0.57	1.42×10^{-3}	1.55×10^{-3}	1.56×10^{-3}
31.25	0.28	3.90×10^{-4}	4.23×10^{-4}	4.49×10^{-4}

Table 6.5: Relative errors in L^∞, L^2, L^1 norm for linearised non-rotating SWE at fixed spatial resolution when the time step Δt is decreased.



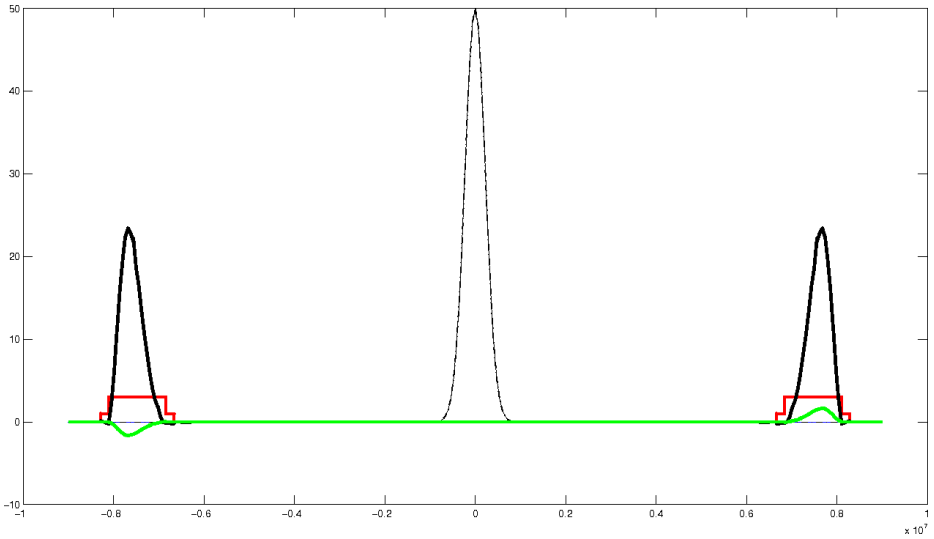
(a) Solid curve: $Q_4 - Q_5$ solution at $T = 10$ hours. The dashed curve indicates the initial condition.



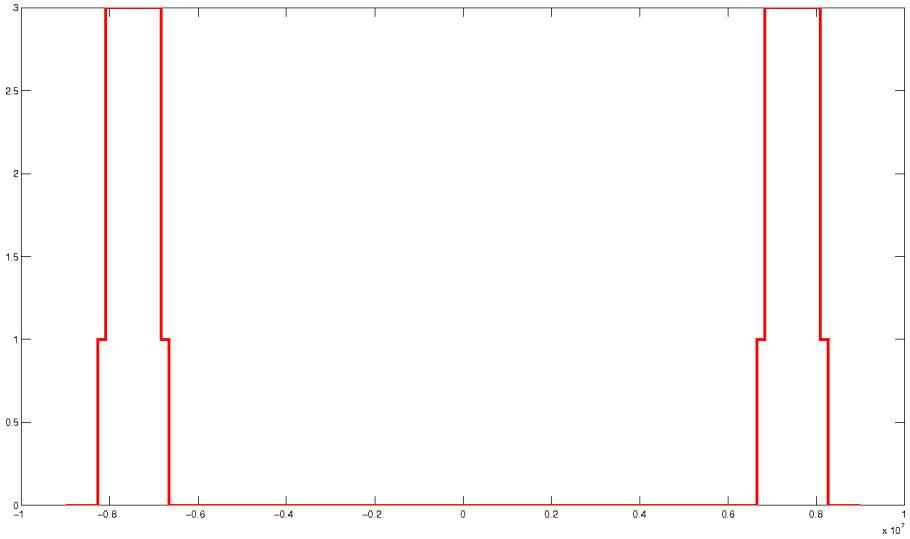
(b) $Q_4 - Q_5$ solution at $T \ll 10$ hours.

Figure 6.3: Numerical solution of non-linear one-dimensional SWE ($f = 0$) for the free surface elevation field.

The effectiveness of the proposed p-adaptivity criterion has also been analysed in the context of gravity wave propagation. To this end the same test as described above was run up to $T = 15$ h with automatic choice of the local polynomial degree up to a maximum value of $p_\eta = 3$. A cut-off tolerance of $\epsilon = 2 \times 10^{-3}$ was employed in the adaptivity criterion. The computed solution is shown in figure 6.4a, while the local polynomial degrees p_η are shown in figure 6.4b, displaying clearly how the adaptivity criterion is able to track adequately the two travelling waves.



(a) free surface elevation (black), u-velocity (green), local p_η (red).



(b) local polynomial degree p_η for the free surface elevation.

Figure 6.4: p-adaptive gravity wave solution for the nonlinear SWE: Cut-off tolerance $\epsilon = 2 \times 10^{-3}$, $H = 2km$, $T = 15hours$.

Geostrophic equilibrium

In the rotating case $f \neq 0$, geostrophic equilibrium is one of the exact solutions of the nonlinear shallow water equations. In order to assess the effectiveness of the proposed spatial discretization in representing accurately this very important component of the solution on geophysical scales, an initial datum in geostrophic equilibrium was chosen on the same spatial domain considered in section 6.1.2 and departures from geostrophic equilibrium and stationarity were measured by the error norms

$p\|_\eta$	p_u	C_{vel}	C_{cel}	E^η_∞	E^η_2	E^η_1	E^u_∞	E^u_2	E^u_1
$\Delta t = 60.0$ s									
0	1	1.17e-3	0.47	6.18e-2	4.03e-2	4.26e-2	8.68e-2	4.90e-2	4.82e-2
2	3	3.50e-3	1.40	3.90e-3	3.63e-3	3.99e-3	3.93e-3	3.63e-3	3.99e-3
6	7	8.17e-3	3.28	3.90e-3	3.63e-3	3.99e-3	3.90e-3	3.63e-3	3.99e-3
$\Delta t = 600.0$ s									
0	1	1.0e-2	4.68	0.29	0.26	0.32	0.31	0.26	0.32
2	3	3.23e-2	14.05	0.29	0.29	0.36	0.29	0.29	0.36
6	7	7.55e-2	32.8	0.29	0.29	0.36	0.29	0.29	0.36
$\Delta t = 3600.0$ s									
0	1	3.38e-2	28.0	0.71	1.07	2.38	0.71	1.07	2.38
2	3	0.1	84.0	0.71	1.07	2.38	0.71	1.07	2.38
6	7	0.24	196	0.71	1.07	2.38	0.71	1.07	2.38
$\Delta t = 6.0$ s									
0	1	1.17e-4	4.70e-2	6.59e-2	4.36e-2	4.64e-2	9.10e-2	5.19e-2	5.14e-2
2	3	3.5e-4	0.14	1.0e-4	8.09e-5	8.86e-5	1.23e-4	8.19e-5	8.95e-5
6	7	8.17e-4	0.33	9.47e-5	8.08e-5	8.85e-5	9.47e-5	8.08e-5	8.85e-5
$\Delta t = 1.0$ s									
0	1	1.95e-5	7.80e-3	6.59e-2	4.36e-2	4.64e-2	9.10e-2	5.19e-2	5.14e-2
2	3	5.84e-5	2.34e-2	2.09e-5	1.28e-5	1.42e-5	4.23e-5	1.82e-5	1.96e-5
6	7	1.36e-4	5.47e-2	1.40e-5	1.21e-5	1.35e-5	1.40e-5	1.21e-5	1.35e-5

Table 6.6: Relative errors in L^∞, L^2, L^1 norm for linearised non rotating SWE.

$$E_1^{geo} := \frac{1}{\beta - \alpha} \int_\alpha^\beta \left| g \frac{\partial \eta}{\partial x} - fv \right| dx \tag{6.1}$$

$$E_2^{std} = \frac{\left\| \eta^{n+1} - \eta^n \right\|_{L^2}}{\left\| \eta^{n+1} \right\|_{L^2}}.$$

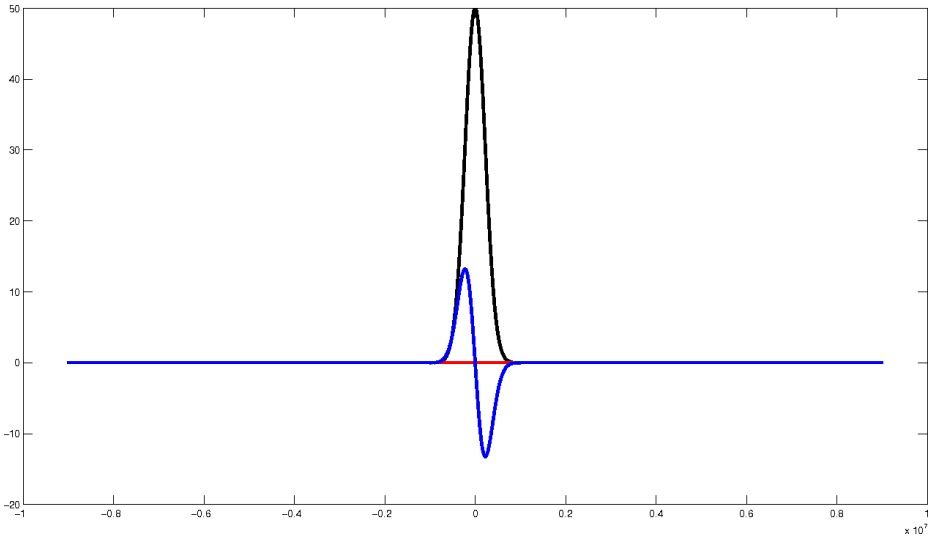
The values of these indicators at approximately $T = 150$ h are displayed in table 6.7, as computed for a simulation with $H = 1$ km, $\Delta x = 36$ km, $\Delta t = 500$ s, $\theta = 0.54$. Fixed polynomial degrees were considered in this case.

p_η	p_u	C_{cel}	$\frac{\sqrt{gH}/f}{\Delta x/p}$	E_1^{geo}	E_2^{std}
2	3	4.2	82.5	4.48×10^{-7}	1.56×10^{-10}
4	5	7.0	137	3.26×10^{-10}	5.32×10^{-14}
6	7	9.9	192	1.42×10^{-13}	1.53×10^{-17}

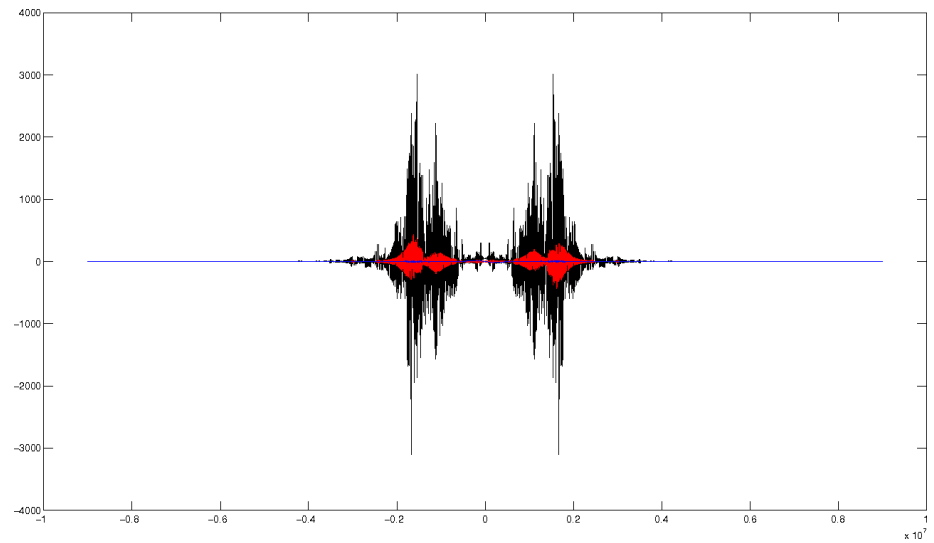
Table 6.7: Departures from geostrophic equilibrium and stationarity in solutions of the nonlinear SWE.

Here, the values α, β in definition 6.1 were assumed to coincide with the boundaries of the 1D domain. A plot of the solution in a similar test is also shown in figure 6.5a, clearly displaying the absence of any spurious oscillation.

It is interesting to observe that, if equal polynomial orders are chosen, the discretization is instead unable to maintain geostrophic equilibrium. A plot of the solution



(a) Numerical solution of the non-linear SWE initialized with a profile in geostrophic equilibrium, computed with $Q_3 - Q_4$ at $C_{cel} = 11.3$ ($\Delta x = 18km$, $\Delta t = 500s$). The solid curves show the solutions (black for the free surface elevation η, red for u, blue for v) at $T = 2.5 \times 10^5 s$.



(b) Numerical solution of the non-linear SWE with rotation ($f = 10^{-4}$)computed with $Q_2 - Q_2$ elements. solution at $T = 2500$s. The solid curves show the solutions (black for the free surface elevation η, red for u) at $T = 1$ rmhs.

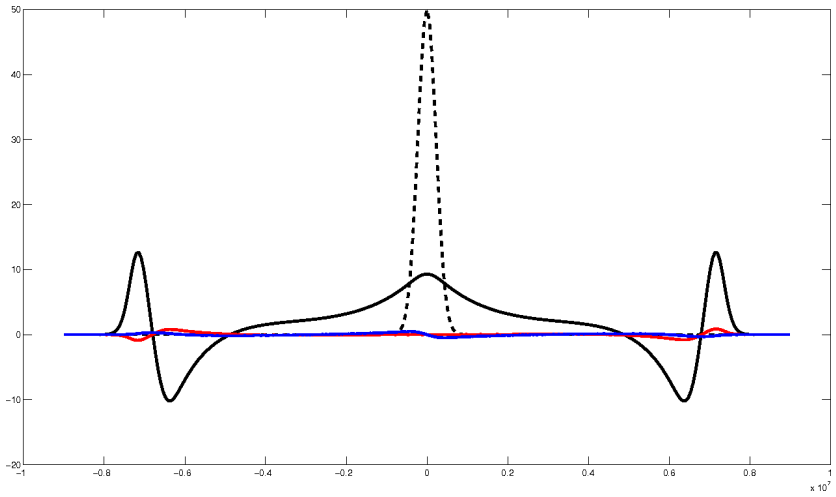
Figure 6.5

computed in the same test at approximately $T = 1$ rmh and taking $p^\eta = p^u = 2$ is shown in figure 6.5b. This justifies the choice of different polynomial orders for the approximation of the free surface and velocity variables, respectively, that was motivated in section 5.3 based on theoretical results available in the stationary, incompressible case.

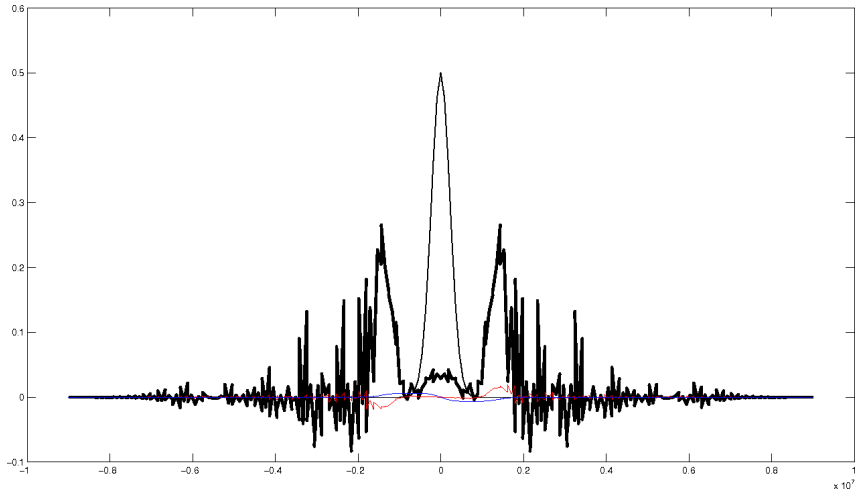
Geostrophic adjustment

If the non rotating case is considered with unbalanced initial data, the solutions will present both gravity wave components, representing the geostrophic adjustment process, and a geostrophically balanced component. The numerical solution computed at $T = 10$ h with $\theta = 0.51$, $H = 2$ km, $\Delta x = 180$ km, $p_\eta = 4$, $p_u = 5$ is shown in figure 6.6a.

Also in this case, when equal order finite element spaces $Q_k - Q_k$ are employed, the numerical solution is entirely disrupted by spurious oscillations, as apparent in figure 6.6b for the case $p_\eta = 2$, $p_u = 2$.



(a) Numerical solution for the free surface elevation field for the nonlinear rotating one-dimensional SWE. Dashed curve: initial condition. Solid curves (black for η, red for u, blue for v): solution at $T = 10$ h.



(b) Numerical solution of the nonlinear one-dimensional SWE with rotation computed with $p_\eta = 2$, $p_u = 2$. Black line: free surface elevation, red line: u component of velocity at $T = 2 \times 10^2$ s.

Figure 6.6: p-adaptive gravity wave solution for the nonlinear SWE: Cut-off tolerance $\epsilon = 2 \times 10^{-3}$, $H = 2km$, $T = 15hours$.

Also in this case, departures from geostrophic equilibrium have been computed for the geostrophic component, choosing α and β in 6.1 in order to exclude the two travelling gravity waves. Results are summarized in table 6.8 for different values of the Rossby deformation radius.

p_η	p_u	C_{cel}	$\frac{\sqrt{gH}/f}{\Delta x/p}$	E_1^{geo}
0	1	0.8	7.78	4.5×10^{-5}
2	3	2.4	23.3	2.6×10^{-6}
6	7	5.5	54.0	2.5×10^{-6}

Table 6.8: Departures from geostrophic equilibrium in the solution of nonlinear SWE at $T = 5 \times 10^4$ s. Solution computed for $\Delta t = 100$ s, $\|\eta_0\|_{L^\infty} = 50$.

While in the previous test the Rossby deformation was always well resolved on the computational mesh considered, it is also of interest to check how the geostrophic adjustment process is reproduced when the Rossby deformation radius is poorly resolved. Encouraging results in this case are collected in table 6.9.

p_η	p_u	C_{cel}	$\frac{\sqrt{gH}/f}{\Delta x/p}$	E_1^{geo}
0	1	0.01	0.87	2.15×10^{-6}
2	3	0.02	2.6	6.22×10^{-7}
6	7	0.06	6	5.22×10^{-7}

Table 6.9: Departures from geostrophic equilibrium in the solution of nonlinear SWE at $T = 5 \times 10^5$ s. Solution computed for $H = 100$ m, $\Delta t = 100$ s, $\|\eta_0\|_{L^\infty} = 0.5$.

6.1.3 Free surface open channel flows with non constant bathymetry

In order to check the performance of the nonlinear model in a case with non constant bathymetry where an analytic solution is available, some of the river hydraulics benchmarks considered in Rosatti *et al.* (2011) and Vazquez-Cendon (1999) have been considered. In particular, inviscid open channel flow was considered in a 25 m long channel with a parabolic obstacle placed in the middle. $N_{el} = 100$ elements of degrees $p_\eta = 3$, $p_u = 4$ were employed in all tests and the solutions were computed in general at $T = 800$ s.

Firstly, the canonical still water test case has been successfully performed over very long integration times. As discussed in Rosatti *et al.* (2011), methods employing the non conservative formulation of the momentum equation satisfy automatically the so called C-property and are naturally well balanced, so that no problems arise due to the variable bathymetry in the still water case.

A steady state test in sub-critical regime was then considered, for which the reference depth us $H = 2.0$ m,, the (constant) exact value of discharge is $Q_{ex} = h_{ex}u_{ex} = 4.42$ m^3/s and the obstacle maximum height is $h_d = 2.0$m. The relative errors for the computed discharge are for this case $E_Q^\infty = 8.07 \times 10^{-4}$, $E_Q^2 = 4.68 \times 10^{-5}$, respectively, and a plot of the free surface elevation is shown in figure 6.7.

Finally, a steady state test in transcritical regime was considered, for which the reference depth is $H = 0.66$ m,, the (constant) exact value of discharge is $Q_{ex} = h_{ex}u_{ex} = 1.53$ m^3/s and the obstacle maximum height is $h_d = 2.0$m. The relative errors

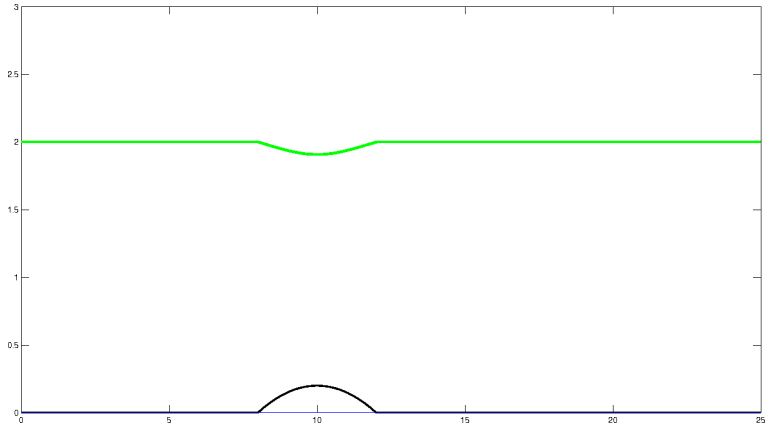


Figure 6.7: Steady flow over a parabolic bump, free surface elevation in sub-critical case, $C_{vel} = 2.09$, $C_{cel} = 5.67$.

for the computed discharge are for this case $E_Q^\infty = 1.559 \times 10^{-3}$, $E_Q^2 = 5.68 \times 10^{-5}$, respectively, and a plot of the free surface elevation is shown in figure 6.8.

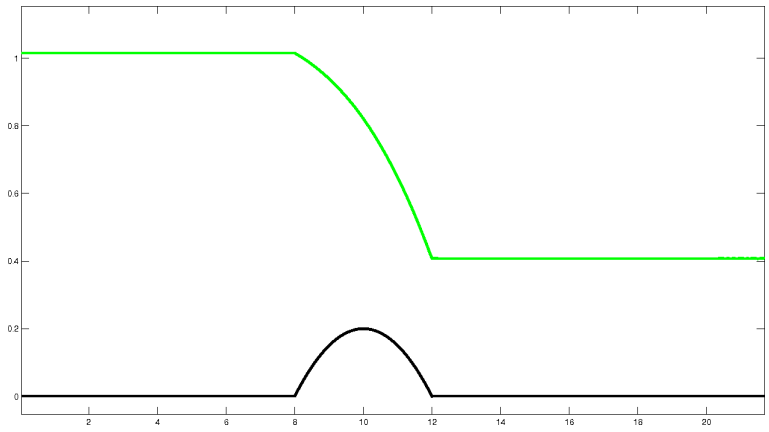


Figure 6.8: Steady flow over a parabolic bump, free surface elevation in transcritical case, $C_{vel} = 0.62$, $C_{cel} = 1.13$.

6.1.4 Coupling with tracers advection

To fully exploit the power of semi-Lagrangian approach, the proposed SISLDG solver for SWE has been coupled with a SLDG passive tracers advection scheme in flux form (see section 3.5), which is the extension of the scheme of Restelli *et al.* (2006), whose properties in terms of conservation of constants (C-property, see e.g. Gross *et al.* (2002)) and compatibility with the continuity equation have been investigated .

As well underlined in Gross *et al.* (2002), a minimal monotonicity requirement, which is, however, desirable of any numerical scheme for a tracers advection equation , is that an initially uniform scalar field remains uniform in the absence of sources and sinks.

This condition has been called the constancy condition in Leonard *et al.* (1996), where it is also shown that instability can arise in methods in which it is violated.

It has to be noticed that in general this property does not always follow for conservative discretizations of a passive tracers advection equation coupled to free-surface flows. In fact, this property is granted if the conservative scheme employed for the tracers advection is consistent with the discretization of the continuity equation. The concept of consistency with continuity has been discussed by various authors (see for example LeVeque (1996) or Lin and Rood (1996)), especially in connection with the links of conservative schemes to their respective advective versions. In Lin and Rood (1996) a discretization of the advection equation is defined consistent with continuity if, given a spatially uniform scalar field as an initial datum, and a general flow field, the discretized scalar advection equation reduces to the discretized continuity equation.

The purpose of this section is to verify the consistency of the semi-implicit discretization of the continuity equation proposed in section 5.3 with the SLDG discretization approach for the scalar transport equation presented in section 3.5.

First the tracer profile was initialized with a constant and check was made that the profile remained constant even after long time. To take into account a situation where the nonlinear terms play an important role, the steady state flow of section 6.1.3 was considered: for such a flow , going on with the simulation even after reaching the steady state, we found that the error in L^∞ norm on the (constant) tracer concentration remains of the order of the machine precision, $\|c - c_{ex}\|_\infty = 4.6 \times 10^{-15}$, see fig. 6.9 .

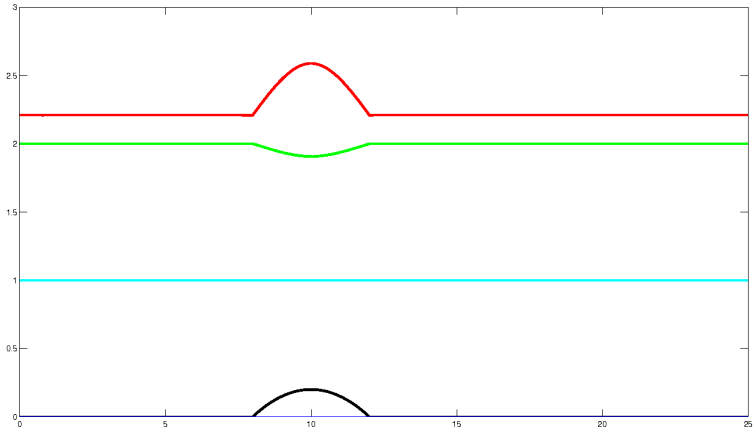


Figure 6.9: Tracer advection by the steady flow over a parabolic bump considered in section 6.1.3 (bathymetry in black, free surface in green, velocity in red), $C_{vel} = 0.45$.. Starting at $t = 0$ with a concentration $c = 1$, the tracer concentration c (in blue) is shown after $\Delta t = 5 \times 10^4 s$: at this time $\|c - c_{ex}\|_\infty = 4.6 \times 10^{-15}$.

Moreover the effect of the passage of a gravity wave on a localized concentration of tracer was considered: in figure 6.10 the initial datum is shown for the elevation and the tracer concentration, while in figure 6.11, 6.12 the solution after 12 hours is shown in linearised and nonlinear case respectively, for $C_{cel} = 1.19$. It can be recognized how the tracer profile remains unaffected by the passage of the gravity wave.

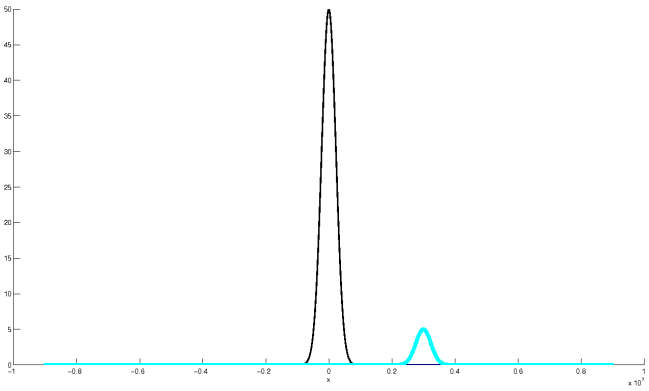


Figure 6.10: Initial datum for the free surface elevation (black) and the tracer concentration (blue).

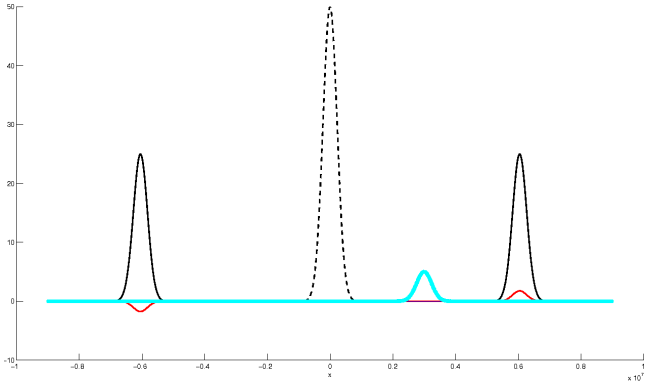


Figure 6.11: Tracer concentration (blue) at $t = 12h$, when the gravity wave has been passed. Linearised SWE, $C_{cel} = 1.19$.

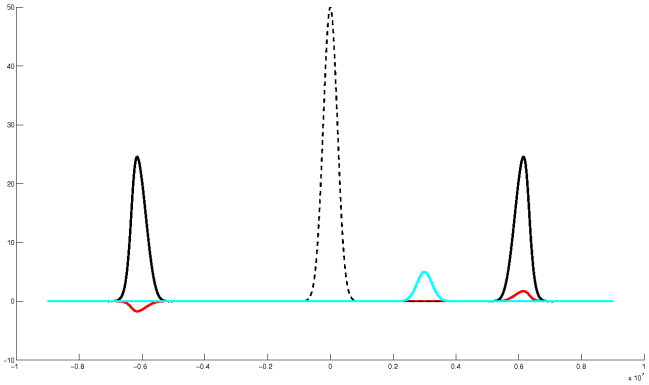


Figure 6.12: Tracer concentration (blue) at $t = 12h$, when the gravity wave has been passed. Nonlinear SWE, $C_{cel} = 1.19$.

6.2 Two dimensional tests

6.2.1 p-adaptivity on the advective part: solid body rotation

As a first test of the two-dimensional implementation, we consider two-dimensional passive tracer advection, in order to achieve a first assessment of the effectiveness of the multidimensional p-adaptivity strategy.

First the most basic test case was considered, namely solid body rotation. For this test, a stationary velocity field is considered, representing a rotating flow with frequency $\omega = 2\pi/300\ \mathrm{s}^{-1}$ around the center of the square domain $\Omega = [0, 20] \times [0, 20]$. The initial datum is a continuous and compactly supported function with the shape of a cosine hill centred in $(7.5, 10)$. Results of the non conservative, $p-$adaptive SL advection after 4 revolutions at maximum $C = 0.5$ are shown in figure 6.13, while in figure 6.14 the values of the local polynomial degrees actually used are displayed, showing how the proposed adaptation algorithm is able to follow correctly the main solution features.

6.2.2 p-adaptivity on the advective part: deformational flow test

Finally the well known test proposed in Smolarkiewicz (1982) has been considered. The number of elements is $N_{el} = 2500$ and the maximum polynomial degree used is fixed to four. As pointed out in Staniforth *et al.* (1986), where an analytic solution for this test has been described, there are two flow regimes for which different evaluation criteria are appropriate. On a time scale of the order of the characteristic period of the flow, accurate numerical methods are assumed to reproduce the analytic solution correctly. On the other hand, on a much longer time scale, it can only be expected that only the average behaviour of the analytic solution is recovered. The computed solution at time $T = 30$ s, corresponding to 3/4 of the characteristic period of the flow, is displayed in figure 6.15. This result compares well with the plots of the analytic solution presented in Staniforth *et al.* (1986). In figure 6.16, on the other hand, the values of the local polynomial degrees actually used are displayed, showing again that the proposed adaptation algorithm is able to follow correctly the main solution features.

On the longer time scale, the results of the advective form of SLDG scheme are similar to large scale average of the analytic solution, as displayed in figures

6.17 and 6.18. Notice that all these computations on the Smolarkiewicz test case have been carried out using a mesh of 2500, a time step corresponding to a Courant number $C \approx 4$, while the tolerance for the adaptivity criterion has been chosen $\epsilon = 10^{-2}$.

6.2.3 Gravity waves propagation

After in previous section the correctness of the semi-Lagrangian part of the code in two dimensions was checked through different advection tests, now the the correctness of the semi-implicit part in two dimensions is investigated first considering the gravity waves propagation (exactly as first done in the one-dimensional case) .

In order to simulate propagation of gravity waves on geophysical scales, nonlinear shallow water equations (4.1) have been solved using the scheme presented in section 5.3 first in the non rotating case $f = 0$, on a rectangular domain $[-L, L]^2$ with length $L = 7000$km and a reference height $H = 2$ km. The solution after 18750s computed by discretizing the nonlinear equations with an initial datum given by zero velocity and a Gaussian perturbation of the free surface placed in the center of the domain is shown in

figure 6.19 and 6.20, where polynomial degrees $p_\eta = 4, p_u = 5$, 2500 elements, a value of the Courant number $C \approx 1$, and $\theta = 0.52$ were used.

The computed solutions are in good agreement with the theory as the propagation velocity of the gravity wave given by $\sqrt{gH}$ is well reproduced.

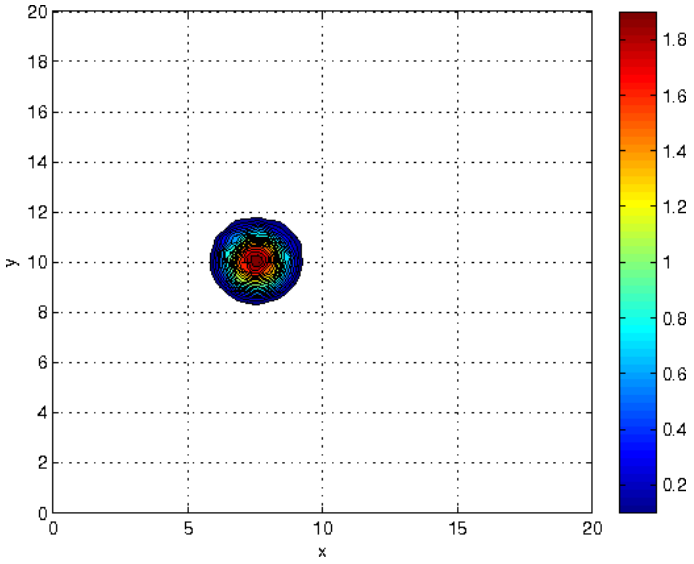


Figure 6.13: Solid body rotation: contour lines of solution after 4 revolutions at $C = 0.5$.

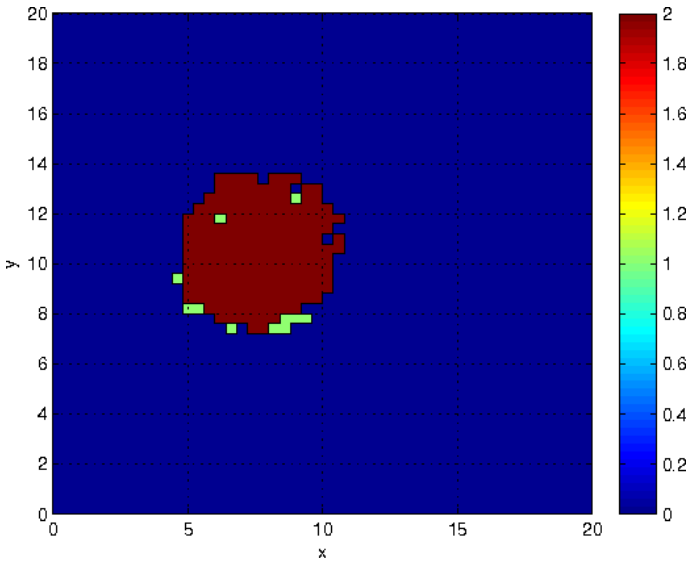


Figure 6.14: Distribution of the local polynomial degrees for the solid body rotation test case.

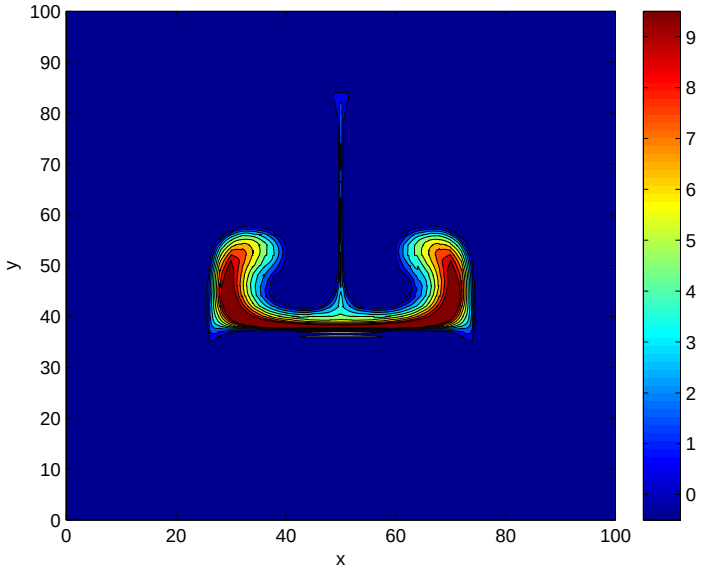


Figure 6.15: Contour lines of the computed solution for the Smolarkiewicz test at $T = 30s$. The maximum Courant is $C \approx 4$, while the tolerance for the adaptivity criterion has been chosen $\epsilon = 10^{-2}$.

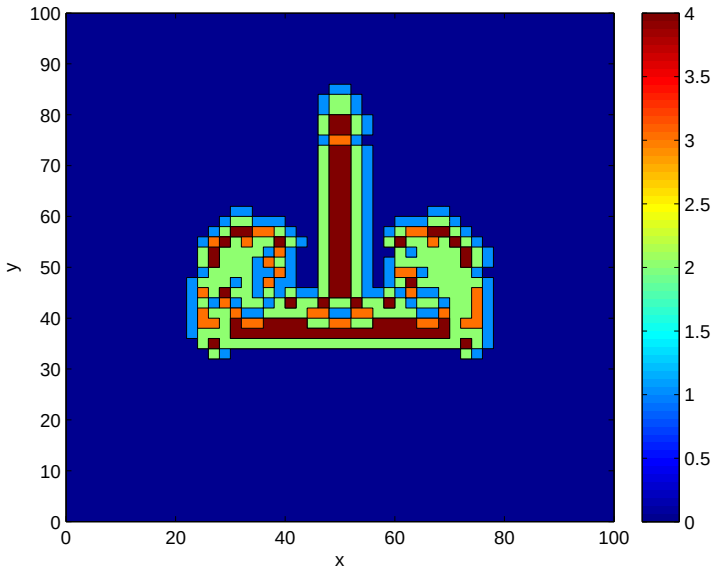


Figure 6.16: Contour lines of the automatically chosen polynomial degree for the computed solution of the Smolarkiewicz test at $T = 30s$. The maximum Courant is $C \approx 4$, while the tolerance for the adaptivity criterion has been chosen $\epsilon = 10^{-2}$.

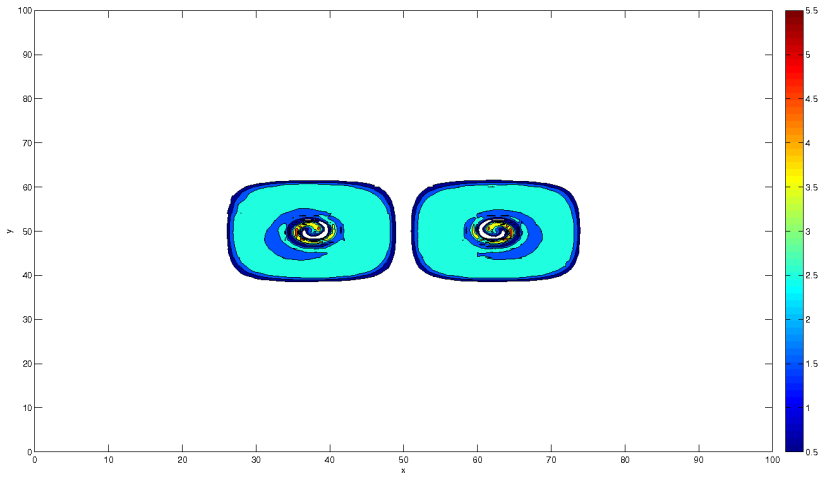


Figure 6.17: Contour lines of the computed solution for the Smolarkiewicz test at $T = 900s$. The maximum Courant is $C \approx 4$, while the tolerance for the adaptivity criterion has been chosen $\epsilon = 10^{-2}$.

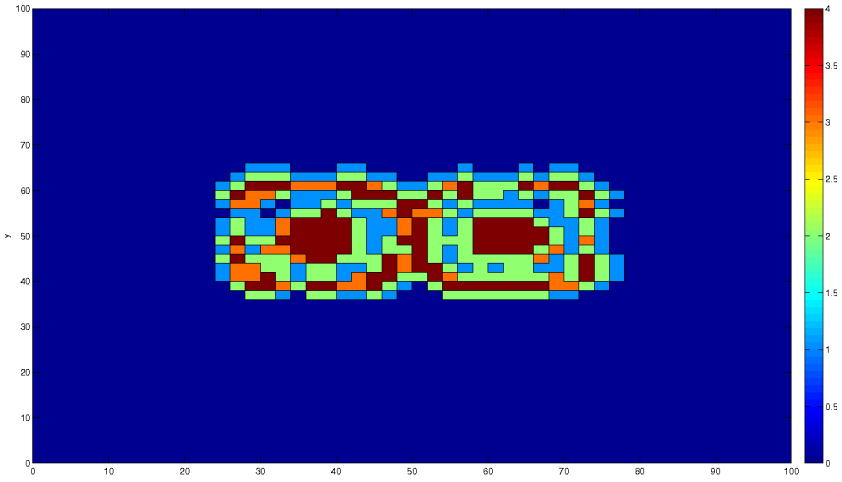


Figure 6.18: Contour lines of the automatically chosen polynomial degree for the computed solution of the Smolarkiewicz test at $T = 900s$. The maximum Courant is $C \approx 4$, while the tolerance for the adaptivity criterion has been chosen $\epsilon = 10^{-2}$.

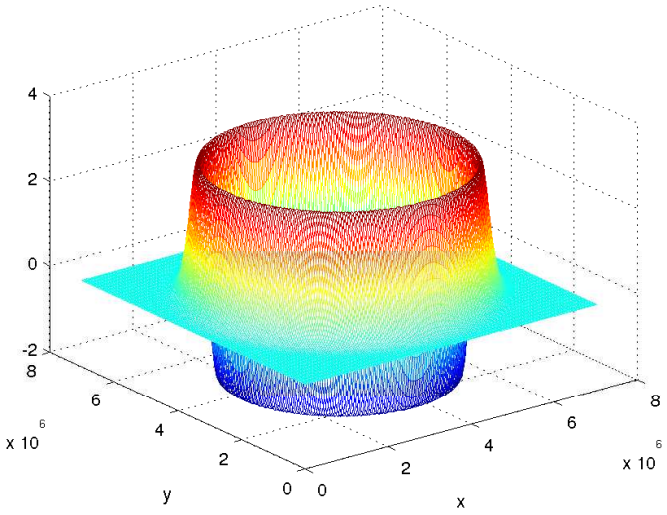


Figure 6.19: Solution of the nonlinear shallow water equations representing the propagation of a gravity wave after 18750s sec. The maximum Courant is $C \approx 1$.

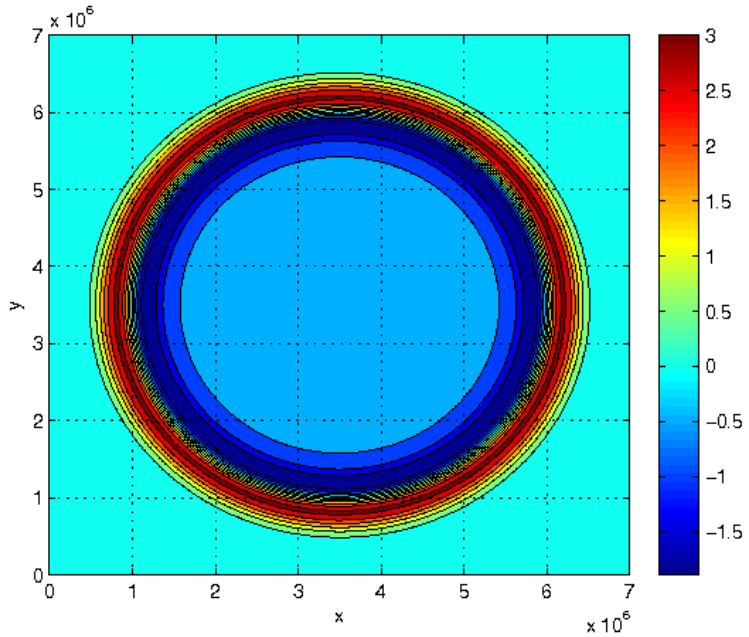


Figure 6.20: Contour lines of the solution of the nonlinear shallow water equations representing the propagation of a gravity wave after 18750s sec. The maximum Courant is $C \approx 1$.

Chapter 7

Conclusions and future perspectives

The main purpose of the present thesis has been the design and analysis of a novel semi-implicit and semi-Lagrangian Discontinuous Galerkin method for the rotating shallow water equations (SISLDG), as a first step in the context of a more ambitious project to develop a new generation, non-hydrostatic, DG based, dynamical core for regional atmospheric modelling.

The main original results of this thesis work can be summarized as follows:

- the effects of different element choices for the velocity-pressure pairs on the stability of the approximate solution have been investigated by numerical experiments, showing that mixed orders $Q_k - Q_{k-1}$ velocity-pressure pairs (structured meshes of quadrilaterals are employed) work better then equal order ones, for which clear instabilities arise. Benefits on the stability from the use of mixed order velocity-pressure pair instead of $Q_k - Q_k$ for DG were proved for the Stokes problem (Toselli (2002), Schötzau *et al.* (2003)), but the fact that typical atmospheric flow regimes are characterized by small Froude/Mach numbers suggested the extension of the same strategy to SWE too. Moreover, this mixed order choice for the pressure-velocity pair can be regarded as the DG analogue of staggering in the finite difference framework (see e.g. Winninghoff (1968), Arakawa and Lamb (1977)).

- A simple p-adaptivity criterion has been employed, that allows to adjust dynamically the number of local degrees of freedom employed to the local structure of the solution. This goal has been achieved thanks to the flexibility of the DG spatial discretization and of the orthogonality property of the Legendre polynomial basis. As demonstrated by the one-dimensional and two-dimensional numerical experiments, p-adaptivity strategy employed is quite effective in reducing the computational cost, while being sufficiently simple and robust to be applied to complete climate and NWP models, where the physical parametrizations present in the source terms make it difficult to perform rigorous a posteriori error analysis.

- Thanks to the choice of 'stable' mixed order velocity-pressure pairs, after standard L^2 projection against test functions (chosen equal to the basis functions as in Direct Characteristic Galerkin scheme, see Morton *et al.* (1988)), and after

integration by parts (where necessary), centred numerical fluxes were used to replace the (not-defined) traces of the solution at the inter-element boundaries, as in Bassi and Rebay (1997b). Moreover, the size of the final fully discrete problem was reduced by expressing the discrete velocity components in terms of the discrete free surface elevation from the momentum equations and then substituting the resulting expressions into the continuity equation, (as customary in SI methods, see e.g. Casulli and Greenspan (1984), Staniforth and Temperton (1986), Temperton and A.Staniforth (1987), Casulli and Cheng (1990), Casulli (1990), Casulli and Cattani (1994)), to obtain a single discrete Helmholtz equation in the free surface elevation unknown only, which takes the form of sparse (penta-diagonal in one-dimension, trideca-diagonal in two dimensions) block non symmetric linear system, which is solved via GMRES iteration.

- To fully exploit the power of semi-Lagrangian approach, the proposed SWE solver has been coupled with a SLDG passive tracers advection scheme in flux form (which is the extension of the scheme of Restelli *et al.* (2006)), whose properties in terms of conservation of constants (C-property, see e.g. Gross *et al.* (2002)) and compatibility with the continuity equation have been investigated . The p-adaptive treatment has been extended in independent way for each different passive tracer. As a result, the changes in the number of degrees of freedom are totally independent for each species, thus allowing to increase the accuracy for some specific variable without increasing the computational cost for other variables that do not need refinement.

- The proposed approach has been implemented in a modular FORTRAN95 code. The 1D implementation first developed has been used as template for the 2D implementation on Cartesian meshes.

  The code has been used to perform a number of tests in order to analyse the stability and accuracy properties of the novel SISLDG method. Numerical results in the framework of one dimensional test cases prove that proposed the method captures accurately and effectively the main features of linear gravity and inertial gravity waves, as well as reproduces correct solutions in nonlinear open channel flow tests and in all rarefaction Riemann problem. The effectiveness of the SISLDG method is also demonstrated by numerical results obtained at high Courant numbers and with automatic choice of the local approximation degree.

  Numerical results in the framework of two-dimensional test cases show the effectiveness of the p-adaptivity strategy employed, as in the test of Smolarkiewicz, as well as the ability of well capture gravity waves also in two dimensions. Moreover the SLDG discretization for the advection has been already tested on a vector of an arbitrary number of tracers.

Future extensions of the numerical methodologies proposed in this thesis may include the development of a dispersion and stability analysis of the proposed method, the design of optimal solvers for the semi-implicit step (with adequate preconditioning), the comparison with other time integration techniques (e.g. exponential integrators).

Moreover, future applications of this work and its extensions will include switching to lat-lon coordinates (through the introduction of suitable metric terms) in order to perform numerical tests on the sphere and finally, (last but not least) the use of the same numerical technique (SISLDG) to develop a non-hydrostatic model for regional atmospheric simulations: more specifically, the resulting method should be used to

improve the numerical discretizations presently employed in RegCM, see Giorgi (1990)
.

Acknowledgements

First of all I would like to thank my advisor Dott. Filippo Giorgi for his strong support, his great humanity and his patience. Thanks for the opportunity of working in this wonderful field of environmental modelling and for the trust and encouragement he provided everyday in the last three years, especially in difficult moments. I am sincerely grateful to Dott. Luca Bonaventura, my co-advisor, for his help and support, for his competence and for introducing me to the fascinating world of numerical methods applied to the atmosphere. I am also greatly indebted with Dott. Marco Restelli, for his valuable advices and for his trust, especially in the last period of my work. Special thanks to Prof. Vincenzo Armenio, for his trust, his enthusiasm and for sharing with me his passion for Fluid Dynamics. A huge thank you goes to my parents, Daniela and Rino. I owe everything to them and I thank them immensely for their love, their patience and their support, even moral, especially during the difficult moments: without them nothing would have been possible in my growth. A special person I would like to thank is my girlfriend Alice, for her sensitivity, and for giving me the ability of looking at the world from a different perspective. I am very grateful to Prof. Renato De Zan, for his invaluable guide during these years. I am also indebted with Proff. Piermilo Vanzella, Raul Serapioni, Pierpaolo Omari, Daniele Del Santo, Alfredo Bellen, Igor Moret, Enzo Tonti, Luigi Quartapelle, Fausto Saleri and Alessandro Veneziani for having communicated to me the joy and the passion in my work and for their precious lessons.

I would like to thank my friend Francesco Dorigatti, since he makes me feel what does it mean to have a brother. I also wish to thank Gabriele Rotondi and Vincenzo Pecunia my other two friends-like-brothers. In general, I wish to thank all the friends and people who helped me during these years, in particular Mauro Montalbano, Giuseppe Aramini, Filippo Bressan, Valentina Stocca, Andrea Petronio,Rondrotiana Barimalala, Eugenio Gallini, Paolo Dassi and Gabriele Rucchin. Finally I would like to thank all the friends of the Earth System Physics section at ICTP, Gulilat Tefera Diro, Mouhamadou Bamba Sylla, Laura Mariotti, Laura Feudale, Fred Kucharski, Erika Coppola, Adrian Tompkins, Carola Bertolini, Graziano Giuliani, Riccardo Farneti, Anna Pirani, Fabien Solmon, Pandora Malchose Pieri, Susanne Henningsen, Karim Aoudia, Nellie Elguindi, Mariangela Guidarelli, Ian Hamling, Sanai Li, Rachel Lowe. Thanks for turning any working day into a special day, for the smiles and for making me feel as a member of a family, thanks to everybody!

Bibliography

Arakawa A. 1966. Computational Design for Long-Term Numerical Integration of the Equations of Fluid Motion: Two-Dimensional Incompressible Flow. Part i l. *Journal of Computational Physics* **1**: 119–143.

Arakawa A, Lamb V. 1977. Computational Design of the Basic Dynamical Processes of the UCLA General Circulation Model. *Methods in Computational Physics* **17**: 174–265.

Arnold D, Brezzi F, Cockburn B, Marini L. 2002. Unified analysis of Discontinuous Galerkin methods for elliptic problems. *SIAM Journal of Numerical Analysis* **39**: 1749–1779.

Bassi F, Rebay S. 1997a. A high-order accurate discontinuous finite element method for numerical solution of the compressible Navier-Stokes equations. *Journal of Computational Physics* **131**: 267–279.

Bassi F, Rebay S. 1997b. High-order discontinuous finite element solution of the 2d Euler equations. *Journal of Computational Physics* **131**: 267–279.

Bates J, McDonald R. 1977. Multiply-Upstream, Semi-Lagrangian Advective Schemes: Analysis and Application to a Multi-Level Primitive Equation Model. *Methods in Computational Physics* **17**: 174–265.

Behrens J, Iske A. 2002. Grid-free adaptive semi-Lagrangian advection using radial basis functions. *Computational and Applied Mathematics* **43**: 319–327.

Bonaventura L. 2000. A semi-implicit, semi-Lagrangian scheme using the height coordinate for a nonhydrostatic and fully elastic model of atmospheric flows. *Journal of Computational Physics* **158**: 186–213.

Casulli V. 1990. Semi-implicit finite difference methods for the two dimensional shallow water equations. *Journal of Computational Physics* **86**: 56–74.

Casulli V, Cattani E. 1994. Stability, accuracy and efficiency of a semi-implicit method for three-dimensional shallow water flow. *Comp. Math. Appl.* **27**: 99–112.

Casulli V, Cheng R. 1990. Stability analysis of eulerian-lagrangian methods for the one-dimensional shallow-water equations. *Applied Mathematical Modelling* **14**: 122–131.

Casulli V, Cheng R. 1992. Semi-Implicit Finite Difference Methods for Three-Dimensional Shallow Water Flow. *International Journal of Numerical Methods in Fluids* **15**: 629–648.

Casulli V, Greenspan D. 1984. Pressure method for the numerical solution of transient, compressible fluid flow. *International Journal of Numerical Methods in Fluids* **4**: 1001–1012.

Cockburn B, Shu C. 1989. TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws ii: General Framework. *Mathematics of Computation* **52 (186)**: 411–435.

Cockburn B, Shu C. 1991. The Runge-Kutta local projection P1 Discontinuous Galerkin method for scalar conservation laws. *Mathematical Modelling and Numerical Analysis* **25**: 337–361.

Coté J, Gravel S, Staniforth A. 1990. Improving Variable-Resolution Finite-Element Semi-Lagrangian Integration Schemes by Pseudostaggering. *Monthly Weather Review* **118**: 2718–2731.

Eskilsson C. 2010. An hp-adaptive discontinuous galerkin method for shallow water flows. *International Journal of Numerical Methods in Fluids* .

Flaherty J, Moore PK. 1995. Integrated space-time adaptive hp-refinement methods for parabolic systems. *Applied Numerical Mathematics* **16**: 317–341.

Gill A. 1987. *Atmospheric-Ocean Dynamics.* Academic Press.

Giorgi F. 1990. Simulation of Regional Climate Using a Limited Area Model Nested in a General Circulation Model. *Journal of Climate* **3**(9): 941–964.

Giraldo F. 1999. Trajectory computations for spherical geodesic grids in cartesian space. *Monthly Weather Review* **127**: 1651–1662.

Giraldo F. 2000. Lagrange-Galerkin methods on spherical geodesic grids: The shallow water equations. *Journal of Computational Physics* **160**: 336–368.

Giraldo F. 2005. Semi-implicit time-integrators for a scalable spectral element atmospheric model. *Quarterly Journal of the Royal Meteorological Society* **131**: 2431–2454.

Gross E, Bonaventura L, Rosatti G. 2002. Consistency with continuity in conservative advection schemes for free-surface models. *International Journal of Numerical Methods in Fluids* **38**: 307–327.

Gurtin M. 1981. *An Introduction to Continuum Mechanics.* Academic Press.

Haltiner G, Williams R. 1980. *Numerical Prediction and Dynamic Meteorology.* Wiley.

Hartmann R. 2008. Numerical analysis of higher order discontinuous Galerkin finite element methods. In: *CFD - ADIGMA course on very high order discretization methods, Oct. 13-17, 2008, Von Karman Institute for Fluid Dynamics, Rhode Saint Genèse, Belgium.* VKI LS 2008-08.

Hasbani Y, ELivne, MBercovier. 1982. Finite elements and characteristics applied to advection-diffusion equations. *Computers and Fluids* **11**: 71–83.

Houston P, Süli B. 2005. A note on the design of hp-adaptive finite element methods for elliptic partial differential equations. *Computed methods in applied mechanics and engineering* **194**: 229–243.

Karniadalis G, Sherwin S. 2005. . Oxford University Press.

Le Roux D, Lin C, AStaniforth. 1999. A Semi-implicit Semi-Lagrangian FiniteElement Shallow-Water ocean model. *Monthly Weather Review* **128**: 1384–1400.

Le Roux D, Lin C, Staniforth A. 1997. An accurate interpolating scheme for semi-Lagrangian advection on an unstructured grid for ocean modelling. *Tellus A* **49A**: 119–138.

Le Roux D, Staniforth A, Lin C. 1998. Finite elements for shallow water equation ocean models. *Monthly Weather Review* **126**: 1931–1951.

Leonard B, Lock A, MacVean M. 1996. Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Monthly Weather Review* **124**: 2588–2606.

LeVeque R. 1992. *Numerical Methods for Conservation Laws.* Birkhäuser.

LeVeque R. 1996. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal of Scientific Computing* **33**(2): 627–665.

Lin S, Rood RB. 1996. Multidimensional flux-form semi-Lagrangian transport schemes. *Monthly Weather Review* **124**: 2046–2070.

Marsden J, Hughes T. 1983. *Mathematical Foundations of Elasticity.* Prentice-Hall.

Mathur M. 1970. A note on an improved quasi-Lagrangian advective scheme for primitive equations. *Monthly Weather Review* **98**: 214–219.

McCalpin J. 1988. A quantitative analysis of the dissipation inherent in semi-Lagrangian advection. *Monthly Weather Review* **116**: 2330–2336.

McDonald A, Bates J. 1987. Improving the estimate of the departure point in a two time level semi-Lagrangian and semi-implicit model. *Monthly Weather Review* **115**: 737–739.

McGregor J. 1993. Economical determination of departure points for semi-Lagrangian models. *Monthly Weather Review* **121**: 221–330.

Miglio E, Quarteroni A, Saleri F. 1999. Finite element approximation of quasi-3d shallow water equations. *Comp. Methods in Applied Mechanics and Engineering* **174**: 355–369.

Morton KW. 1998. On the analysis of finite volume methods for evolutionary problems. *SIAM Journal of Numerical Analysis* **35**: 2195–2222.

Morton KW, Priestley A, Süli E. 1988. Stability of the Lagrange-Galerkin scheme with inexact integration. *RAIRO Modellisation Matemathique et Analyse Numerique* **22**: 625–653.

Nair R, Coté J, Staniforth A. 1999. Monotonic cascade interpolation for semi-lagrangian advection. *Quarterly Journal of the Royal Meteorological Society* **125**: 197–212.

Pedlosky J. 1987. *Geophysical fluid dynamics.* Springer Verlag.

Phillips N. 1959. *An example of nonlinear computational instability, in the Atmosphere and Sea in Motion*. Rockfeller Institute Press, New York, pp. 501–504.

Pironneau O. 1982. On the transport-diffusion algorithm and its applications to the Navier-Stokes equations. *Numerische Mathematik* **38**: 309–332.

Priestley A. 1994. Exact Projections and the Lagrange-Galerkin Method: A Realistic Alternative to Quadrature. *Journal of Computational Physics* **112**: 316–333.

Pudykiewicz J, Staniforth A. 1984. Some Properties and Comparative Performance of the Semi-Lagrangian Method of Robert in the Solution of the Advection-Diffusion Equation. *Atmosphere-Ocean* **22**: 283–304.

Purnell D. 1975. Solution of the Advective Equation by Upstream Interpolation with a Cubic Spline. *Monthly Weather Review* **104**: 42–47.

Purser R, Leslie LM. 1994a. An efficient interpolation procedure for high order three dimensional semi-Lagrangian models. *Monthly Weather Review* **119**: 2492–2498.

Purser R, Leslie LM. 1994b. An efficient semi-Lagrangian scheme using third order semi-implicit time integration and forward trajectories. *Monthly Weather Review* **122**: 745–756.

Quarteroni A, Valli A. 1994. *Numerical approximation of partial differential equations*. Springer Verlag.

Remacle J, Flaherty J, Shephard MS. 2003. An adaptive discontinuous galerkin technique with an orthogonal basis applied to compressible flow problems. *SIAM Review* **45**(1): 53–72.

Restelli M, Bonaventura L, Sacco R. 2006. A semi-Lagrangian Discontinuous Galerkin method for scalar advection by incompressible flows. *Journal of Computational Physics* **216**: 195–215.

Restelli M, Giraldo F. 2009. A conservative Discontinuous Galerkin semi-implicit formulation for the Navier-Stokes equations in nonhydrostatic mesoscale modeling. *SIAM Journal of Scientific Computing* **31**: 2231–2257.

Robert A. 1981. A stable numerical integration scheme for the primitive meteorological equations. *Atmosphere-Ocean* **19**: 35–46.

Robert A. 1982. A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations. *Journal of the Meteorological Society of Japan* **60**: 319–325.

Rood R. 1987. Numerical Advection Alogorithms and Their Role in Atmospheric Transport and Chemistry Models. *Reviews of Geophysics* **25**: 71–100.

Rosatti G, Bonaventura L, Cesari D. 2005. Semi-implicit, semi-Lagrangian environmental modelling on cartesian grids with cut cells. *Journal of Computational Physics* **204**: 353–377.

Rosatti G, Bonaventura L, Deponti A, Garegnani G. 2011. An accurate and efficient semi-implicit method for section-averaged free-surface flow modelling. *International Journal of Numerical Methods in Fluids* **65**: 448–473.

Saad Y, Schultz M. 1986. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* **7**: 856–869.

SAOrszag. 1971. Numerical simulation of incompressible flows within simple boundaries: accuracy. *Journal of Fluid Mechanics* **49**: 75–112.

Sawyer J. 1963. A semi-Lagrangian method of solving the vorticity advection equation. *Tellus* **15**: 336–342.

Schoenstadt A. 1980. A transfer function analysis of numerical schemes used to simulate geostrophic adjustment. *Monthly Weather Review* **108**: 1248–1259.

Schötzau D, Schwab C, Toselli A. 2003. Stabilized DGFEM for incompressible flows. *Mathematical Models and Methods in Applied Sciences* **13**: 1413—-1436.

Smolarkiewicz P. 1982. The multi–dimensional Crowley advection scheme. *Monthly Weather Review* **110**: 1968–1983.

Smolarkiewicz P, Pudykiewicz J. 1992. A class of semi-Lagrangian approximations for fluids. *Journal of the Atmospheric Sciences* **49**: 2082–2096.

Staniforth A, Coté J. 1991. Semi-lagrangian integration schemes for atmospheric models-a review. *Monthly Weather Review* **119**: 2206–2223.

Staniforth A, Côté J, Pudykiewicz J. 1986. Comments on "Smolarkiewicz's Deformational Flow". *Monthly Weather Review* **115**: 894–900.

Staniforth A, Mitchell H. 1977. A semi-implicit finite-element barotropic model. *Monthly Weather Review* **105**: 154–169.

Staniforth A, Temperton C. 1986. Semi-Implicit Semi-Lagrangian Integration Schemes for a Barotropic Finite-Element Regional Model. *Monthly Weather Review* **114**: 2078–2090.

Stoker J. 1957. *Water waves: The mathematical theory with applications*. Interscience.

Temperton C, AStaniforth. 1987. An efficient two-time-level semi-Lagrangian semi-implicit integration scheme. *Quarterly Journal of the Royal Meteorological Society* **113**: 1025–1039.

Toselli A. 2002. H-p Discontinuous Galerkin approximations for the Stokes problem. *Mathematical Models and Methods in Applied Sciences* **12**: 1565—-1597.

Vazquez-Cendon M. 1999. Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal of Computational Physics* **148**: 497–526.

Walters R. 1983. Numerically induced oscillations in finite-element approximations to the shallow-water equations. *International Journal of Numerical Methods in Fluids* **3**: 591–604.

Walters R, Carey G. 1983. Analysis of spurious oscillation modes for the shallow-water and Navier-Stokes equations. *Computers and Fluids* **11**: 51–68.

Williams R. 1981. On the formulation of finite-element prediction models. *Monthly Weather Review* **109**: 463–466.

Williams R, Zienkiewicz O. 1981. Improved finite-elements forms for the shallow-water wave equations. *International Journal of Numerical Methods in Fluids* **1**: 81–97.

Williamson D, Rasch P. 1989. Two dimensional semi-Lagrangian transport with shape-preserving interpolation. *Monthly Weather Review* **117**: 102–129.

Winninghoff F. 1968. On the adjustment toward a geostrophic balance in a simple primitive equation model with application to the problem on initialization and objective analysis. PhD thesis, University of California, Los Angeles.

Zienkiewicz O, JPGago, Kelly D. 1983. The hierarchical concept in finite element analysis. *Computers and Structures* **16**: 53–65.