MOX–Report No. 04/2012

A semi-implicit, semi-Lagrangian, p-adaptive Discontinuous Galerkin method for the shallow water equations

Tumolo, G.; Bonaventura, L.; Restelli, M.

MOX, Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox@mate.polimi.it http://mox.polimi.it

A semi-implicit, semi-Lagrangian, $p-$adaptive Discontinuous Galerkin method for the shallow water equations

Giovanni Tumolo[(1)]
Luca Bonaventura[(2)], Marco Restelli[(2)],

January 14, 2012

[(1)] Earth System Physics Section
The Abdus Salam International Center for Theoretical Physics
Strada Costiera 11, 33151 Trieste, Italy
gtumolo@ictp.it

[(2)] MOX – Modelling and Scientific Computing,
Dipartimento di Matematica "F. Brioschi", Politecnico di Milano
Via Bonardi 9, 20133 Milano, Italy
luca.bonaventura@polimi.it, marco.restelli@polimi.it

Keywords: Geophysical flows, Discontinuous Galerkin methods, semi-implicit discretizations, semi-Lagrangian discretizations, shallow water equations

AMS Subject Classification: 35L02, 65M60, 65M25, 76U05, 86A10

Abstract

A semi-implicit and semi-Lagrangian Discontinuous Galerkin method for the shallow water equations is proposed, for applications to geophysical scale flows. A non conservative formulation of the advection equation is employed, in order to achieve a more treatable form of the linear system to be solved at each time step. The method is equipped with a simple $p-$adaptivity criterion, that allows to adjust dynamically the number of local degrees of freedom employed to the local structure of the solution. Numerical results show that the method captures well the main features of gravity and inertial gravity waves, as well as reproducing correct solutions in nonlinear test cases with analytic solutions. The accuracy and effectiveness of the method are also demonstrated by numerical results obtained at high Courant numbers and with automatic choice of the local approximation degree.

1 Introduction

The Discontinuous Galerkin (DG) method, after proving itself a very valuable tool for applications to high Mach number aerodynamics (see e.g. [2], [3], [7], [8], [9]), has also become increasingly popular for applications to geophysical flows. Indeed, its appealing combination of high order accuracy, local mass conservation and ease of massively parallel implementation have turned it into one of the possible choices for next generation climate and NWP dynamical cores, as well as for other environmental fluid dynamics applications, see e.g. [13], [17], [26].

One of the main drawbacks of DG discretizations, however, is that, when coupled to standard explicit time discretizations, they imply rather severe stability restrictions. Denoting by C the Courant number and by p the order of the polynomial basis employed, the equivalent of the standard Courant-Friedrichs-Lewy stability condition turns out to be $C \leq 1/(2p+1)$, as proven e.g. in [7]. If high order spatial discretizations are envisaged, this is quite restrictive and implies a serious computational burden, that may lead to question the overall effectiveness of DG methods for real time, operational NWP models or for long range climate simulations.

Traditionally, remarkable efficiency gains have been achieved in NWP models by application of semi - implicit (SI) and semi - Lagrangian (SL) techniques. The combination of these two approaches, starting with the seminal work by André Robert [34], has produced a number of efficient SISL models for a number of environmental applications, see e.g. [4], [11], [12], [33], [39], [42], [43], [44].

In previous work [30], [31], [32], we have presented the first attempts at increasing the efficiency of DG methods by separate application of either SL or SI techniques. The aim of this paper is to present a complete semi - implicit and semi - Lagrangian Discontinuous Galerkin (SISLDG) discretization approach for low Mach number, compressible fluid dynamics problems. To further increase the efficiency of the proposed method, the flexibility of DG discretization with respect to the local number of degrees of freedom is fully exploited, by introducing a simple $p-$adaptivity strategy that allows to change dynamically the number of degrees of freedom used in each element, without degrading the overall accuracy of the method.

In section 2, the shallow water equations are introduced as a model problem. In section 3, we define the finite element spaces used then in section 4 to introduce the SISLDG discretization of the shallow water equations. In section 5 the algorithm for $p-$

adaptivity is presented. Numerical results are presented in section 6, while in section 7 we try to draw some conclusions and outline the path towards application of the concepts introduced here in the context of a complete multidimensional dynamical core.

2 The shallow water equations

We will consider as a model problem the two-dimensional shallow water equations with rotation in cartesian coordinates,

$$\begin{aligned} &\frac{\partial \eta}{\partial t} + \frac{\partial (uh)}{\partial x} + \frac{\partial (vh)}{\partial y} = 0 \\ &\frac{Du}{Dt} = -g\frac{\partial \eta}{\partial x} + fv \\ &\frac{Dv}{Dt} = -g\frac{\partial \eta}{\partial y} - fu. \end{aligned} \tag{1}$$

Here, $\frac{D}{Dt}$ denotes the Lagrangian derivative

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

η is the free surface elevation, g is the acceleration of gravity, f is the Coriolis parameter, $h = \eta - b$, where b is the bathymetry profile, u is the velocity in the coordinate direction x and v is the velocity in the coordinate direction y. These equations are a standard test bed for numerical methods to be applied to the full equations of motion of atmospheric or oceanic circulation models. It should be observed that the continuity equation is kept in conservation form, in order to obtain a mass conservative method, along the lines of the approach pursued e.g. in [4],[5],[12].

3 Discontinuous finite element spaces

The discretization approach proposed in this work is independent of the nature of the mesh and could also be implemented for fully unstructured and even non conforming meshes. For simplicity, however, in this paper we only consider implementations on structured, Cartesian meshes. Therefore, we assume that equations (1) are to be solved on a rectangular domain $\Omega = (a, b) \times (c, d)$ with appropriate initial and boundary conditions. The domain Ω is partitioned in N non overlapping quadrilateral elements $K_I, I = 1, \ldots, N$ 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 $\mathcal{T}_h = \{K_I : I = 1, \ldots, N\}$, where $h = \max_I(\mathrm{diam} K_I)$. 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 $e_{S,I}, e_{E,I}, e_{N,I}, e_{W,I}$ (see figure 1). It is immediate that each K_I is the image of the master element $\hat{K} = [-1, 1] \times [-1, 1]$ via the affine local map $\boldsymbol{F}_I$, such that $x = F_{I,1}(\xi_1) = \xi_1 \Delta x_I/2 + x_I, y = F_{I,2}(\xi_2) = \xi_2 \Delta y_I/2 + y_I$, where $(x, y) \in K_I$ and $(\xi_1, \xi_2) \in \hat{K}$. 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 $\hat{K}$. 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 we set

$$\boldsymbol{p} = \{p_I : I = 1, \ldots, N\}.$$

We then consider the finite element space:

$$V_{h,\boldsymbol{p}} = \left\{ v \in L^2(\Omega) : v|_{K_I} \in \mathcal{Q}_{p_I(K_I)} \quad I = 1, \ldots, N \right\}. \tag{2}$$

We will denote by $\mathcal{E}_h$ the set of all element boundary edges, which is naturally decomposed as $\mathcal{E}_h = \mathcal{E}_{h,int} \cup \mathcal{E}_b$, where $\mathcal{E}_{h,int}$ denotes the element boundary edges that belong to the interior of Ω and $\mathcal{E}_b$ denotes the element boundary edges that belong to $\partial\Omega$. Functions in $V_{h,\boldsymbol{p}}$ 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 $v_h \in V_{h,\boldsymbol{p}}$, being $\boldsymbol{n}_{\partial K,e}$ the restriction of $\boldsymbol{n}_{\partial K}$ on the edge $e \subset \partial K$, we can define

$$\{\{v_h\}\}(\boldsymbol{x}) = \frac{1}{2}\Big(v_h|_K(\boldsymbol{x}) + v_h|_{K'}(\boldsymbol{x}) \Big) \tag{3}$$

$$[[v_h]](\boldsymbol{x}) = v_h|_K(\boldsymbol{x}) \boldsymbol{n}_{\partial K,e} + v_h|_{K'}(\boldsymbol{x}) \boldsymbol{n}_{\partial K',e}. \tag{4}$$

Notice that these definitions, following [1] and [8], are symmetric with respect to the two elements K, K' sharing the boundary point at which the jump is defined. If instead $e \in \mathcal{E}_{h,b}$, then there exists a unique $K \in \mathcal{T}_h$ such that $e \in \partial K$ and hence we set

$$\{\{v_h\}\}(\boldsymbol{x}) = v_h|_K(\boldsymbol{x}) \tag{5}$$

$$[[v_h]](\boldsymbol{x}) = v_h|_K(\boldsymbol{x}) \boldsymbol{n}_{\partial K,e}. \tag{6}$$

We now consider the issue of defining a basis for $V_{h,\boldsymbol{p}}$. Thanks to the definition (2), this basis can be defined by first choosing a basis for $\mathcal{Q}_p$ which induces a set of local bases on the spaces $\mathcal{Q}_{p_I}(K_I)$. The basis of $V_{h,\boldsymbol{p}}$ is then obtained by collecting the extensions of the latter functions to zero outside of their corresponding element. 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 [50].

Hierarchical bases can be defined starting with the one dimensional case. More specifically, let $\hat{K}^{1D} = [-1, 1]$ and consider, for $\xi \in \hat{K}^{1D}$, the k-th Legendre polynomial, defined recursively by the three-term recurrence relation:

$$\begin{aligned} &L_{k+1}(\xi) = \frac{2k+1}{k+1}\xi L_k(\xi) - \frac{k}{k+1}L_{k-1}(\xi), \quad k = 1, 2, \dots \\ &L_0(\xi) = 1, \quad L_1(\xi) = \xi. \end{aligned} \tag{7}$$

The Legendre polynomials form an orthogonal basis for polynomials on $\hat{K}^{1D}$, since

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

Notice that this orthogonality property has important numerical implications for Galerkin type methods, since it not only implies that all the mass matrices are diagonal, but also gives in general improved conditioning of the resulting discretizations (see e.g. [20] or [50] for details). The extension to higher dimensions over quadrilateral regions is relatively straightforward, although rather more involved than the one-dimensional case. Turning now to the two-dimensional case, for $(\xi_1, \xi_2) \in \hat{K}$, a hierarchical basis of $\mathcal{Q}_p$ can be constructed by taking tensor products of the Legendre polynomials introduced for the one-dimensional case, so that our basis functions are of the form $L_k(\xi_1)L_l(\xi_2)$ for $0 \leq k, l \leq p$. On regular Cartesian meshes, the most natural and straightforward way to construct the two-dimensional basis is by taking products of the one-dimensional bases.

Finally, a central issue in finite element formulations for fluid problems is the choice of appropriate approximation spaces for the velocity and pressure variables. In the context of SWE, the role of the pressure is played by the free surface elevation. Indeed, an inconsistent choice of the two approximation spaces may result in a solution that is polluted by spurious modes (see for example [47] or [48] for the specific case of SWE or, as a general reference, [28]).

Although no attempt is made here to analyze this issue at a theoretical level, results available for incompressible flows (see for example [45] and [38] for the Stokes problem) have shown that $\mathcal{Q}_p - \mathcal{Q}_{p-1}$ pairs are not only *inf - sup* stable with respect to the mesh-size, but also uniformly stable with respect to the polynomial degree p. Since the applications of interest for the present discretization approach are essentially in the small Froude number regime, this element pair will be chosen for the results presented in this paper, see also similar recommendations in [21]. As a result, different polynomial degrees will be assumed in each element for the representation of velocity and pressure variables, specifically

$$p_I^u = p_I^\eta + 1, \quad I = 1, \dots, N \tag{8}$$

and the notation $\{\phi_{I,l}\}_{l=1}^{(p_I^\eta+1)^2}$, $\{\psi_{I,m}\}_{m=1}^{(p_I^u+1)^2}$ will be used to denote the corresponding basis functions of $\mathcal{Q}_{p_I^\eta}(K_I)$, $\mathcal{Q}_{p_I^u}(K_I)$, respectively. The corresponding global finite element spaces will be denoted by $V_{h,\boldsymbol{p}^\eta}$, $V_{h,\boldsymbol{p}^u}$, respectively. A similar arrangement has been successfully used, investigated and analyzed in the context of continuous Galerkin discretization of SWE in [49] in the case of low polynomial degrees. It should be remarked, however, that we have implemented the proposed model with the freedom to choose either (8) or $p_I^u = p_I^\eta$ and that the model appears to produce correct results also if equal degrees are employed for the velocity and free surface variables. As it will be seen in the numerical tests reported in section 6, the observed differences between the two versions of the model have been marginal in all the tests carried out so far.

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

The problem of a SISLDG discretization of (1) is now addressed. Following the approach of [4], [5], [6], [32], a semi - implicit, semi - Lagrangian discretization of (1) can be written as

$$\frac{\eta^{n+1}-\eta^n}{\Delta t} = -\theta\Big[\frac{\partial(u^{n+1}h^n)}{\partial x}+\frac{\partial(v^{n+1}h^n)}{\partial y}\Big] \quad (9)$$
$$-(1-\theta)\Big[\frac{\partial(u^nh^n)}{\partial x}+\frac{\partial(v^nh^n)}{\partial y}\Big]$$

$$\frac{u^{n+1}-E(t^n,\Delta t)u^n}{\Delta t} = -\theta g\frac{\partial\eta}{\partial x}^{n+1}+\theta f v^{n+1} \quad (10)$$
$$+(1-\theta)gE(t^n,\Delta t)\frac{\partial\eta}{\partial x}^{n}$$
$$+(1-\theta)E(t^n,\Delta t)(fv^n)$$

$$\frac{v^{n+1}-E(t^n,\Delta t)v^n}{\Delta t} = -\theta g\frac{\partial\eta}{\partial y}^{n+1}-\theta f u^{n+1} \quad (11)$$
$$+(1-\theta)gE(t^n,\Delta t)\frac{\partial\eta}{\partial y}^{n}$$
$$-(1-\theta)E(t^n,\Delta t)(fu^n).$$

Here, the notation $G^n(\boldsymbol{x}) = G(\boldsymbol{x},t^n)$ has been used throughout, $\theta\in[0,1]$ is the fixed implicitness parameter of a two time-level averaged semi-implicit discretization. E is a numerical evolution operator that approximates the exact evolution operator associated to the velocity field $\boldsymbol{u}^n=(u^n,v^n)$ (see e.g. [23], [25]). More precisely, if $\boldsymbol{X}(t,t^{n+1},\boldsymbol{x})$ denotes the solution of

$$\frac{d\boldsymbol{X}(t,t^{n+1},\boldsymbol{x})}{dt} = \boldsymbol{u}^n(\boldsymbol{X}(t,t^{n+1},\boldsymbol{x}) \quad (12)$$

with initial datum $\boldsymbol{X}(t^{n+1},t^{n+1},\boldsymbol{x})=\boldsymbol{x}$ at time $t=t^{n+1}$, $E(t^n,\Delta t)G$ denotes a numerical approximation of $G(\boldsymbol{X}(t^n,t^{n+1},\boldsymbol{x}))$. Different methods can be employed to approximate equation (12); in this paper, we have employed a simple Euler method with substepping, see e.g. [16], [35].

The continuity equation is considered in its Eulerian flux form, 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.

We now discretize in space by taking the L^2 projection against the test functions on a given element K_I . In particular, following [18] or the Direct Characteristic Galerkin method proposed in [24], 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 SISLDG formulation of the SWE system, we have to consider the numerical evolution operator $E(t^n, \Delta t)$ applied to the gradient of the free surface elevation η, which is a piecewise polynomial function with no global continuity constraints. To compute this function, we follow the standard local discontinuous Galerkin method and define first the discrete gradient operator $\boldsymbol{\mathcal{D}}_h\eta$ as follows. Given $\eta \in V_{h,\boldsymbol{p}^\eta}$, its discrete gradient $\boldsymbol{\mathcal{D}}_h\eta$ is the unique element of $[V_{h,\boldsymbol{p}^u}]^2$ such that

$$\int_{K_I} \boldsymbol{\mathcal{D}}_h\eta \cdot \boldsymbol{w}\, d\boldsymbol{x} = -\int_{K_I} \eta \boldsymbol{\nabla}\cdot \boldsymbol{w}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\boldsymbol{w}\cdot\boldsymbol{n}\, ds \quad \forall \boldsymbol{w} \in [V_{h,\boldsymbol{p}^\eta}]^2. \tag{13}$$

Notice that, by integration by parts, we also have

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

Hence, as observed also in [2], it is possible to recast the discrete gradient operator as the sum of two contributions, the first one taking into account the elementwise gradient of η, while the second one taking into account its jumps across the element interfaces. As a consequence, the discrete partial derivatives of η are defined as:

$$\begin{aligned}\int_{K_I} \mathcal{D}_h^x\eta\, \psi_{I,m}\, d\boldsymbol{x} &= -\int_{K_I} \eta\frac{\partial\psi_{I,m}}{\partial x}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\psi_{I,m}\boldsymbol{i}\cdot\boldsymbol{n}\, ds \quad (14)\\ &= \int_{K_I} \frac{\partial\eta}{\partial x}\psi_{I,m}\, d\boldsymbol{x} + \int_{\partial K_I} (\hat{\eta}-\eta)\psi_{I,m}\boldsymbol{i}\cdot\boldsymbol{n}\, ds,\end{aligned}$$

for $m = 1, \dots (p_I^u+1)^2$ and

$$\begin{aligned}\int_{K_I} \mathcal{D}_h^y\eta\, \psi_{I,m}\, d\boldsymbol{y} &= -\int_{K_I} \eta\frac{\partial\psi_{I,m}}{\partial y}\, d\boldsymbol{x} + \int_{\partial K_I} \hat{\eta}\psi_{I,m}\boldsymbol{j}\cdot\boldsymbol{n}\, ds \quad (15)\\ &= \int_{K_I} \frac{\partial\eta}{\partial y}\psi_{I,m}\, d\boldsymbol{x} + \int_{\partial K_I} (\hat{\eta}-\eta)\psi_{I,m}\boldsymbol{j}\cdot\boldsymbol{n}\, ds,\end{aligned}$$

for $m = 1, \dots (p_I^\eta+1)^2$, where $\boldsymbol{i}, \boldsymbol{j}$ are the unit vectors along the coordinate axes, while $\boldsymbol{n}$ is the unit normal vector pointing outward from ∂K_I and $\hat{\eta}$ is a numerical flux that will be defined in the following formula (19). The coefficients of the discrete partial derivatives with respect to the chosen local polynomial basis are computed via the solution of (14) and (15). If orthogonal hierarchical bases $\{\psi_{I,m}\}_m$ are used, this only requires the inversion of diagonal local mass matrices. Notice that $\boldsymbol{\mathcal{D}}_h\eta$ could in principle also be defined as an element of $[V_{h,\boldsymbol{p}^\eta}]^2$. Numerical experiments have shown that

this does not lead to significant changes in the results, so that we adopted definition (13) for convenience.

We then obtain for each element $K_I, I = 1, \ldots, N$:

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

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

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

Notice that the Coriolis parameter constant $f = f_I$ has been taken to be constant within a given element K_I in order to simplify the structure of the resulting implicit discretization. This is consistent with the target of applying the method on relatively high resolution meshes. After integration by parts in the continuity equation and use of definitions (14) and (15) in the momentum equations, we have:

$$\int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} - \theta\Delta t \int_{K_I} \nabla\varphi_{I,l}(\boldsymbol{x}) \cdot \boldsymbol{u}^{n+1}(\boldsymbol{x})h^n(\boldsymbol{x})d\boldsymbol{x}$$
$$+\theta\Delta t \int_{\partial K_I} \varphi_{I,l}(s)\hat{\boldsymbol{u}}^{n+1}(s) \cdot \boldsymbol{n}(s)\hat{h}^n(s)ds \tag{16}$$
$$= \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\eta^n(\boldsymbol{x})d\boldsymbol{x} + (1-\theta)\Delta t \int_{K_I} \nabla\varphi_{I,l}(\boldsymbol{x}) \cdot \boldsymbol{u}^n(\boldsymbol{x})h^n(\boldsymbol{x})d\boldsymbol{x}$$
$$-(1-\theta)\Delta t \int_{\partial K_I} \varphi_{I,l}(s)\hat{\boldsymbol{u}}^n(s) \cdot \boldsymbol{n}(s)\hat{h}^n(s)ds,$$

$$\int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x} - \theta\Delta tg \int_{K_I} \frac{\partial\psi_{I,m}}{\partial x}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} \tag{17}$$
$$+\theta\Delta tg \int_{\partial K_I} \psi_{I,m}(s)\hat{\eta}^{n+1}(s)\boldsymbol{i} \cdot \boldsymbol{n}(s)ds - \theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x}$$
$$= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x}$$
$$-(1-\theta)\Delta tg \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^x\eta^n\right]\big](\boldsymbol{x})d\boldsymbol{x} +$$
$$+(1-\theta)\Delta t \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)fv^n\big](\boldsymbol{x})d\boldsymbol{x},$$

$$\int_{K_I} \psi_{I,m}(\boldsymbol{x})v^{n+1}(\boldsymbol{x})d\boldsymbol{x} - \theta\Delta tg \int_{K_I} \frac{\partial\psi_{I,m}}{\partial y}(\boldsymbol{x})\eta^{n+1}(\boldsymbol{x})d\boldsymbol{x} \tag{18}$$
$$+\theta\Delta tg \int_{\partial K_I} \psi_{I,m}(s)\hat{\eta}^{n+1}(s)\boldsymbol{j} \cdot \boldsymbol{n}(s)ds + \theta\Delta t f_I \int_{K_I} \psi_{I,m}(\boldsymbol{x})u^{n+1}(\boldsymbol{x})d\boldsymbol{x}$$
$$= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x} +$$
$$-(1-\theta)\Delta tg \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^y\eta^n\right]\big](\boldsymbol{x})d\boldsymbol{x} +$$
$$-(1-\theta)\Delta t \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)fu^n\big](\boldsymbol{x})d\boldsymbol{x},$$

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 $V_{h,\boldsymbol{p}^\eta}$, $V_{h,\boldsymbol{p}^u}$ are not defined on the elements of $\mathcal{E}_{h,int}$. Therefore, the resulting boundary terms are indeed replaced by appropriate numerical fluxes $\hat{u}, \hat{v}$, $\hat{\eta}$ and $\hat{h}$ which are single valued functions of both interface states and thus introduce the necessary coupling between the unknowns of neighboring elements. In this work, we will use centered fluxes as in [2]. Therefore,

using the notation in (4), or each dependent variable $a \in V_{h,\boldsymbol{p}^\eta}$ or $a \in V_{h,\boldsymbol{p}^u}$ we define:

$$\hat{a}\big|_e = \{\{a\}\}, \quad \forall e \in \mathcal{E}_{h,int}. \tag{19}$$

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

$$\eta^n(\boldsymbol{x})\big|_{K_I} = \sum_{r=1}^{(p_I^\eta+1)^2} \varphi_{I,r}(\boldsymbol{x})\eta_{I,r}^n, \tag{20}$$

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

The issue of the appropriate and solution-dependent choice of the values p_I^η and p_I^u is addressed in section 5, where a simple but effective $p-$adaptivity criterion will be introduced.

Substituting now expressions (20) into the numerical fluxes definitions (19) and then also into equations (16)-(18), one obtains the full space and time discretization of system (1), that can be conveniently presented employing a vector notation. We denote by

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

the vectors collecting all the discrete degrees of freedom associated to element K_I. Furthermore, we introduce the local mass matrices associated to η, u, v, denoted by M_I^η, M_I^u, M_I^v respectively. If hierarchical basis functions are used, these matrices are diagonal and their entries are given by

$$(M_I^\eta)_{l,r} = \int_{K_I} \varphi_{I,l}(\boldsymbol{x})\varphi_{I,r}(\boldsymbol{x})d\boldsymbol{x}, \tag{21}$$

$$(M_I^u)_{m,r} = (M_I^v)_{m,r} = \int_{K_I} \psi_{I,m}(\boldsymbol{x})\psi_{I,r}(\boldsymbol{x})d\boldsymbol{x},$$

were obviously one has $l, r = 1, \ldots, (p_I^\eta + 1)^2$ for η and $m, r = 1, \ldots, (p_I^u + 1)^2$ 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 1, we also define a number of auxiliary matrices. Specifically, for the continuity equa-

tion we set:

$$(L_I^{\eta u})_{l,r} = -\frac{1}{2}\int_{e_{W,I}} \varphi_{I,l}\hat{h}^n\psi_{I_W,r}dy, \quad (L_I^{\eta v})_{l,r} = -\frac{1}{2}\int_{e_{S,I}} \varphi_{I,l}\hat{h}^n\psi_{I_S,r}dx, \tag{22}$$

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

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

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

for the momentum equation along x:

$$(L_I^{u\eta})_{m,r} = -\frac{1}{2}\int_{e_{W,I}} \psi_{I,m}\varphi_{I_W,r}dy, \quad (U_I^{u\eta})_{m,r} = \frac{1}{2}\int_{e_{E,I}} \psi_{I,m}\varphi_{I_E,r}dy, \tag{23}$$

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

$$\begin{aligned}(\mathcal{U}_I^n)_m &= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)u^n\big](\boldsymbol{x})d\boldsymbol{x} + \\ &-(1-\theta)\Delta t g\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^x\eta^n\right]\big](\boldsymbol{x})d\boldsymbol{x} + \\ &+(1-\theta)\Delta t\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)fv^n\big](\boldsymbol{x})d\boldsymbol{x},\end{aligned} \tag{24}$$

and for the momentum equation along y:

$$(L_I^{v\eta})_{m,r} = -\frac{1}{2}\int_{e_{S,I}} \psi_{I,m}\varphi_{I_S,r}dx, \quad (U_I^{v\eta})_{m,r} = \frac{1}{2}\int_{e_{N,I}} \psi_{I,m}\varphi_{I_N,r}dx, \tag{25}$$

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

$$\begin{aligned}(\mathcal{V}_I^n)_m &= \int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)v^n\big](\boldsymbol{x})d\boldsymbol{x} + \\ &-(1-\theta)\Delta t g\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)\left[\mathcal{D}_h^y\eta^n\right]\big](\boldsymbol{x})d\boldsymbol{x} + \\ &-(1-\theta)\Delta t\int_{K_I} \psi_{I,m}(\boldsymbol{x})\big[E(t^n,\Delta t)fu^n\big](\boldsymbol{x})d\boldsymbol{x}.\end{aligned} \tag{26}$$

Notice that all the local matrices defined in (22), (23) and (25) are in general rectangular, of dimensions $(p_I^\eta+1)^2\times(p_J^u+1)^2$, where

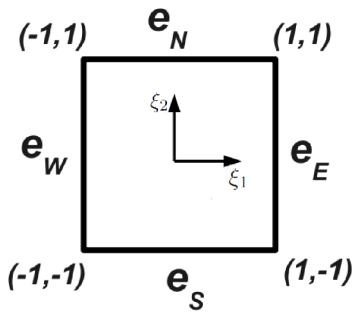


Figure 1: The geometry of the mapping on the reference element $\hat{K} = [-1,1]^2$ and the associated metric information.

$J \in \{I, I_S, I_E, I_N, I_W\}$ for matrices defined in (22), 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 (23) and (25). Furthermore, all the matrices defined in (23) and (25) are time-independent and can be computed *once for all* at the beginning of the computation.

After the definitions (22)-(26), the equations (16)-(18) defining the SISLDG method for the system (1) read in matrix form:

$$\begin{aligned} M_I^\eta \boldsymbol{\eta}_I^{n+1} \quad &- \quad \theta\Delta t\big(L_I^{\eta u}\boldsymbol{u}_{I_W}^{n+1} + D_I^{\eta u}\boldsymbol{u}_I^{n+1} + U_I^{\eta u}\boldsymbol{u}_{I_E}^{n+1} \\ &+ \quad L_I^{\eta v}\boldsymbol{v}_{I_S}^{n+1} + D_I^{\eta v}\boldsymbol{v}_I^{n+1} + U_I^{\eta v}\boldsymbol{v}_{I_N}^{n+1}\big) \\ &= M_I^\eta \boldsymbol{\eta}_I^n - (1-\theta)\Delta t\big(L_I^{\eta u}\boldsymbol{u}_{I_W}^n + D_I^{\eta u}\boldsymbol{u}_I^n + U_I^{\eta u}\boldsymbol{u}_{I_E}^n \\ &+ \quad L_I^{\eta v}\boldsymbol{v}_{I_S}^n + D_I^{\eta v}\boldsymbol{v}_I^n + U_I^{\eta v}\boldsymbol{v}_{I_N}^n\big), \end{aligned} \tag{27}$$

$$\begin{aligned} M_I^u \boldsymbol{u}^{n+1} \quad &- \quad \theta\Delta t g\left(L_I^{u\eta}\boldsymbol{\eta}_{I_W}^{n+1} + D_I^{u\eta}\boldsymbol{\eta}_I^{n+1} + U_I^{u\eta}\boldsymbol{\eta}_{I_E}^{n+1}\right) \\ &- \quad \theta\Delta f_I M_I^v \boldsymbol{v}_I^{n+1} = \boldsymbol{\mathcal{U}}_I^n, \end{aligned} \tag{28}$$

$$\begin{aligned} M_I^v \boldsymbol{v}^{n+1} \quad &- \quad \theta\Delta t g\left(L_I^{v\eta}\boldsymbol{\eta}_{I_S}^{n+1} + D_I^{v\eta}\boldsymbol{\eta}_I^{n+1} + U_I^{v\eta}\boldsymbol{\eta}_{I_N}^{n+1}\right) \\ &+ \quad \theta\Delta f_I M_I^u \boldsymbol{u}_I^{n+1} = \boldsymbol{\mathcal{V}}_I^n. \end{aligned} \tag{29}$$

As customary in semi-implicit methods, see e.g. [4], [5], [6], [32], [40], [43], $\boldsymbol{u}^{n+1}$ and $\boldsymbol{v}^{n+1}$ are expressed in terms of $\boldsymbol{\eta}^{n+1}$ and the resulting expressions substituted into the continuity equation (27), in order to obtain a discrete (vector) Helmholtz equation in the $\boldsymbol{\eta}^{n+1}$ unknown only. If the elements surrounding K_I are labeled as E,N,W,S in counterclockwise order (see figure 2), this equation can be rewritten as follows:

$$
\begin{aligned}
&K_I^{WW}\boldsymbol{\eta}_{I_{WW}}^{n+1} + K_I^{SW}\boldsymbol{\eta}_{I_{SW}}^{n+1} + K_I^{W}\boldsymbol{\eta}_{I_W}^{n+1} + K_I^{NW}\boldsymbol{\eta}_{I_{NW}}^{n+1} \\
&+K_I^{SS}\eta_{I_{SS}}^{n+1} + \quad K_I^{S}\boldsymbol{\eta}_{I_S}^{n+1} + K_I\boldsymbol{\eta}_I^{n+1} + K_I^{N}\boldsymbol{\eta}_{I_N}^{n+1} \quad + K^{NN}\boldsymbol{\eta}_{I_NN}^{n+1} + \\
&+K_I^{SE}\boldsymbol{\eta}_{I_{SE}}^{n+1} + K_I^{E}\boldsymbol{\eta}_{I_E}^{n+1} + K_I^{NE}\boldsymbol{\eta}_{I_{NE}}^{n+1} + K_I^{EE}\boldsymbol{\eta}_{I_{EE}}^{n+1} = \boldsymbol{\mathcal{N}}_I^n. \qquad (30)
\end{aligned}
$$

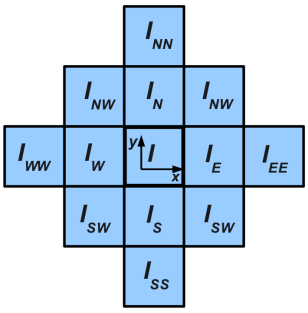


Figure 2: The computational stencil for the semi-implicit step of SISLDG and the names of the elements surrounding the elements K_I.

Here, we have set :

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

and defined the right hand side as:

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

while constants $c_{1,I}, \ldots, c_{5,I}$ are defined by

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

The non-symmetric linear system (30) can be solved using e.g. the GMRES method [37]. In the preliminary implementation available so far, only a simple diagonal preconditioning was employed. 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 (28) and (29), respectively.

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 being used. In particular, notice that integrals of terms in the image of the evolution operator E, i.e. functions evaluated at the departure points of the trajectories arriving at the quadrature nodes, cannot be computed exactly (see e.g. [24], [27]), since such functions are not polynomials. Therefore, a sufficiently accurate approximation of these integrals is needed, which may entail the need to employ numerical quadrature formulae with more nodes than the minimal requirement implied by the local polynomial degree.

5 A simple $p-$adaptivity criterion

The numerical method described in section 4 can be implemented 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 supplying the 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 for example orography/bathymetry profiles, data on land use and soil type, land-sea masks 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 parameterizations are highly sensitive to the mesh size. Although devising better parameterizations that require less mesh-dependent tuning is an important research goal, more conventional parameterizations will still be in use for quite some time. As a consequence, it would be useful 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 chemical 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 to increase 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. [14], [15], [19], [29]. 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 $a(\boldsymbol{x})$:

$$\begin{aligned} a(\boldsymbol{x})\big|_{K_I} &= \sum_{j=1}^{(p_I^\alpha+1)^2} a_{I,j}\psi_{I,j}(\boldsymbol{x}) \\ &= \sum_{k=1}^{p_I^a+1}\sum_{l=1}^{p_I^\alpha+1} a_{I,k,l}\psi_{I_x,k}(x)\psi_{I_y,l}(y), \end{aligned} \tag{31}$$

where $I = (I_x, I_y)$ 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 de-

grees of freedom in terms of the one-dimensional ones, so that we can write $a_{I,j} = a_{I,k,l}$ and $\psi_{I,j}(\boldsymbol{x}) = \psi_{I_x,k}(x)\psi_{I_y,l}(y)$.

Notice that p_I^a will have different values for each different dependent variable a. Due to the choice of finite element spaces 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 hierarchical basis is employed, by Parseval's identity one will have

$$\mathcal{E}^{tot} = \|Pa\|^2 = \sum_{j=1}^{(p_I^a+1)^2} a_{I,j}^2 = \sum_{k,l=1}^{p_I^a+1} a_{I,k,l}^2, \tag{32}$$

where P is the L^2 projector onto the local polynomial subspace. Combining (32) with the hierarchical property of the basis, for any integer $r = 0, \ldots, p^a$ we can define the quantity

$$\mathcal{E}^r = \sum_{\max(k,l)=r} a_{I,k,l}^2, \tag{33}$$

that can be interpreted as the energy associated to the $r-$th degree modal components of $a\big|_{K_I}$ for a given element $K_I \in \mathcal{T}_h$. The degrees of freedom associated with $\mathcal{E}^r$ are highlighted in figure 3 in the plane $p_x - p_y$, where p_x, p_y denote the polynomial degrees associated to the one dimensionals degrees of freedom in the x, y directions, respectively. Therefore, for any integer $r = 0, \ldots, p_I^a$, the quantity

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

will measure the relative weight of the $r-$th degree modal components of a with respect to the best approximation available for the L^2 norm of a. Assuming that a denotes a generic model variable at the beginning of the computation of a generic time step, the proposed adaptation criterion can be described as follows:

Given an error tolerance $\epsilon_I > 0$ for each element $I = 1, \ldots, N$, :

1) compute w_{p_I}

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

2.1.1) set $p_I(a) := p_I(a) + 1$

2.1.2) set $a_{I,p_I} = 0$, 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* $p_I(a) := p_I(a) - 1$ *and go back to 2.2.1.*

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 this adaptation criterion is purely heuristic, the numerical tests reported in section 6 show that it is effective in reducing the computational cost with no essential loss in the accuracy of the computed solutions. Indeed, in all the test a computational cost reduction of at least 40% was observed (actually, the reduction was much larger in some of the tests), while the relative error with respect to the solution computed employing a fixed number of degrees of freedom equal to the maximum allowed value was always consistent with the value chosen for the error tolerance ϵ.

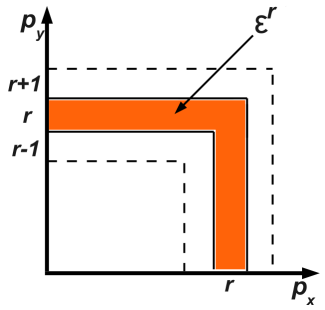


Figure 3: The region on the $p_x - p_y$ plane of the degree of freedom which contributes to $\mathcal{E}^r$.

6 Numerical experiments

The numerical method introduced in section 4 has been implemented and tested on a number of relevant test cases, in order to assess its accuracy and stability properties and to analyse the impact of the $p-$adaptivity strategy on its efficiency. Whenever a reference solution was available, the relative errors were computed in the L^1, L^2 and L^∞ norms at the final time T of the simulation as

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

where ζ_{ref} denotes the reference solution and the integral norms are computed by appropriate numerical quadrature rules. In all the tests reported below, we have used $p^u = p^\eta + 1$, but as remarked previously only marginal differences were observed if the same number of degrees of freedom was employed for both velocity and free surface.

For tests on a closed domain, mass conservation has been checked by computing at each time level t^n the following measure of deviation from the initial mass value:

$$E_{mass}^n = \frac{\left|\int_\Omega h^n d\mathbf{x} - \int_\Omega h^0 d\mathbf{x}\right|}{\int_\Omega h^0 d\mathbf{x}}.$$

Values just above the order of machine precision were obtained in all such tests.

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), a distinction has been made between the maximum Courant number based on the velocity the maximum Courant number based on the celerity, defined as

$$C_{vel} = \max \frac{|u|\Delta t}{\Delta x/p} \qquad C_{cel} = \max \frac{(|u \pm \sqrt{gh}|)\Delta t}{\Delta x/p}.$$

We have taken $p = p^u$ in all tests reported in this paper. In order to assess the efficiency gain given by the the adaptivity strategy, for each model variable and for each time level t^n, the computational effort reduction has been measured as:

$$\Delta_{dof}^n = \frac{\sum_{I=1}^N (p_I^n + 1)^2}{N(p_{max} + 1)^2},$$

where N is the total number of elements.

A first accuracy assessment of the proposed scheme has been carried out in the framework of two non trivial one-dimensional benchmarks, for which exact solutions are available. Two dimensional flows on geophysical scale have then been simulated.

6.1 Riemann problem with continuous solution

Due to the nature of the semi-Lagrangian method, strong non stationary 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. Therefore, we consider the Riemann problem for equations (1) with $f = 0$ and with initial values yielding an all rarefaction wave solution, specifically $h(x,0) = 2$ m, $u(x,0) = -0.5$ m/s, $x < 0$ and $u(x,0) = 0.5$ m/s, $x \geq 0$. An expression for the exact solution is derived e.g. in [22]. This one dimensional problem was solved using the two dimensional implementation of our method on the domain $[-L_x, L_x] \times [0, L_y]$ with $L_x = 70$ m and $L_y = 1$ m. The initial datum was taken to be equal to the analytical solution of the Riemann problem at time $t = 12$ s and the errors were computed at $t = 14$ s.

Two convergence tests for the spatial discretization were performed. Firstly, the local number of degrees of freedom and the Courant number were kept constant at $C_{cel} = 0.02, p^{\eta} = 1, p^{u} = 2$ and the number of elements N_x in the x direction was increased progressively, while the number of elements in the transversal direction was kept constant at $N_y = 5$ in all cases.

Results are reported in table 1 , showing that theoretical convergence rates are achieved at low spatial resolution, although the first order time discretization error dominates as the spatial resolution increases. The test was then repeated keeping mesh size and Courant number constant at $N_x = 8$ and $C_{cel} = 0.02$, respectively, while increasing the degree of the local polynomial spaces. Results are reported in table 2, showing that the method is convergent, although the achieved convergence rates are heavily influenced by the low order time discretization and by the low regularity of the solution.

N_x	E_2^η	E_1^η	E_∞^η
8	1.91e-2	9.55e-2	7.81e-2
16	5.00e-3	2.59e-2	2.20e-2
32	2.18e-3	9.16e-3	1.22e-2
64	1.12e-3	4.67e-3	6.29e-3
128	5.32e-4	1.99e-3	3.52e-3
256	3.93e-4	1.47e-3	2.49e-3

Table 1: Relative errors on the free surface elevation at fixed $C_{cel} = 0.02, \quad p^\eta = 1, p^u = 2$ and for increasing number of elements.

p^η	p^u	E_2^η	E_1^η	E_∞^η
0	1	3.62e-2	1.73e-1	9.04e-2
1	2	1.92e-2	9.51e-2	7.70e-2
2	3	8.97e-3	5.45e-2	2.57e-2
3	4	4.13e-3	2.58e-2	1.74e-2
4	5	3.21e-3	1.79e-2	1.43e-2

Table 2: Relative error norms on the free surface elevation at fixed $\Delta x = 2L_x/N_x$, $N_x = 8$, $C_{cel} = 0.02$ and for increasing polynomial degree.

6.2 River hydraulics benchmark

To check the performance of the method in a case with non constant bathymetry, we have considered the open channel flow problem with a parabolic obstacle placed in the middle described e.g. in [46] and [36]. An analytic steady state solution can be easily obtained from the Bernoulli equation. The spatial domain is given by the interval $[0, L] \times [0, 1]$, with $L = 25$ m, covered with 100 elements of equal size in the x direction, while again 5 elements were employed in the y direction. The profile of the parabolic bump is given by:

$$b(x) = \begin{cases} -H + 0.2 - 0.05(x - 10)^2, & \text{if } 8 < x < 10 \\ -H, & \text{otherwise} \end{cases}$$

The sub-critical steady state regime was considered, achieved by imposing as inflow boundary condition a discharge $Q_{in} = 4.42$ m^3/s and as outflow boundary condition a total fluid depth $h = 1$ m.

The relative errors on the free surface elevation and on the discharge have been computed at time $T = 300$ s, when the departure from steady state measured by

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

was seen to yield negligible values. The errors were computed for different time-step sizes that cover a wide range of Courant numbers, see table 3. It is apparent how the proposed scheme is able to reproduce correctly the steady state solution even for very large Courant numbers.

For this configuration, the canonical still water test case has also been successfully performed over very long integration times, assuming no flux boundary conditions and taking even very large time steps. As discussed in [36], 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.

Δt	C_{vel}	C_{cel}	E_2^{std}	E_∞^η	E_2^η	E_∞^Q	E_2^Q
0.00625	0.20	0.33	2.38e-14	2.61e-4	1.58e-4	1.91e-5	1.33e-6
0.0125	0.40	0.66	2.56e-13	3.44e-4	1.83e-4	6.36e-4	3.85e-5
0.025	0.80	1.32	3.13e-12	5.66e-4	1.31e-4	1.23e-3	7.41e-5
0.05	1.56	2.64	6.25e-14	1.28e-3	2.54e-4	5.15e-3	3.75e-4
0.1	3.10	5.31	5.79e-14	3.03e-3	5.07e-4	7.46e-4	5.85e-5
0.2	6.20	10.60	4.88e-14	6.14e-3	1.17e-3	2.89e-3	2.18e-4

Table 3: Relative errors at steady state on free surface elevation and discharge with different time step values.

Finally, we show that even transcritical regimes can be correctly reproduced by the proposed method. Considering for example a reference depth $H = 0.66$ m in the bathymetry profile, 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 (see e.g. [36]). The relative errors 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 4.

6.3 Propagation of pure gravity waves

As a first two dimensional test case, the propagation of pure gravity waves is considered, i.e. we assume that $f = 0$. For this case, a rectangular $[0, L] \times [0, L]$ domain with $L = 10^7$ m was considered, for which the initial mean water depth H was set to 10^3 m and no-flux boundary conditions were assumed. Initially, the fluid is at rest

and the free surface elevation is given by the following expression:

$$\eta_0(x,y) = 5\exp\left[-\frac{(x-L/2)^2+(y-L/2)^2}{2(L/20)^2}\right]. \qquad (34)$$

The computed solution is observed at $t = 36000$s, so that the distance covered by the wave crest in the radial direction can be estimated by the linear theory as $\Delta r = \sqrt{gH}(t_f - t_i) = 3.5656 \times 10^6$ m. The method was applied on a structured mesh composed by 2500 rectangles and elements of degree $p^\eta = 5$ were employed. The test is repeated at different Courant numbers. A snapshot of the solution at $t = 36000$ s is shown in figure 5, while relative errors on the celerity of the gravity wave, measured as

$$E_{cel} = \frac{\frac{(r^f_{peak} - r^i_{peak})}{(t^f - t^i)} - \sqrt{gH}}{\sqrt{gH}},$$

are summarized in table 4. It can be observed that the solution is reproduced correctly, up to the precision allowed by this rather rough indicator. At very large Courant numbers, as expected, the well known distortion of the wave pattern due to the semi-implicit discretization is observed.

The test is then repeated using the automatic adaptation criterion of the local polynomial degree described in section 5. The tolerance in the adaptation criterion was set to $\epsilon = 5 \times 10^{-3}$ for all elements. The same value was used also in all subsequent adaptive tests. The method appears to be effective in reducing the computational cost without affecting the accuracy of the solution, as it can be observed from the plots of the solution at time $t = 36000$ s shown in figure 6. More precisely, the L_2 relative error between the adaptive and nonadaptive solution at $t = 36000$ s has been measured, obtaining the value 3.3×10^{-3} for the free surface in the adaptive solution reported in figures 5 and 6. A plot of the error contours is also shown in figure 8. Indeed, in this case we observe values of E_{cel} entirely analogous to those of the non adaptive case for all the values of the Courant number that were considered. For this test of the adaptivity criterion, the time evolution of the computational cost reduction Δ^n_{dof} is reported in figure 7, showing that a saving of more than 50% was achieved. On the other hand, for the present implementation of the adaptation criterion an increase in the number of local degrees of freedom can be triggered by very small amplitude perturbations, as it can be seen comparing figures 6(a), 6(b).

C_{cel}	0.53	1.07	2.14	4.30	8.55
E_{cel}	6.90e-2	6.37e-2	4.90e-2	2.88e-2	2.91e-2

Table 4: Error on the gravity wave propagation speed at different Courant numbers.

6.4 Geostrophic adjustment and preservation of geostrophic equilibrium

In presence of the Coriolis force, we have tested the ability of the proposed method to reproduce the process of geostrophic adjustment and to maintain an initially balanced solution for a long time. In this case, the so called $f-$plane approximation is considered, assuming the constant value $f = 10^{-4}$ s^{-1} for the Coriolis parameter. Departure from geostrophic equilibrium at time t^n has been measured by the error indicator

$$E^n_{geo} = \frac{\int_\alpha^\beta \int_\gamma^\delta \left[\left| g\frac{\partial \eta}{\partial x} - fv + (\boldsymbol{u}\cdot\nabla)u \right| + \left| g\frac{\partial \eta}{\partial y} + fu + (\boldsymbol{u}\cdot\nabla)v \right| \right] \; dxdy}{(\beta-\alpha)(\delta-\gamma)}.$$

The average is computed either on the whole domain, in the test on preservation of an initially balanced datum, or on a smaller area containing the balanced part of the solution, for the geostrophic adjustment test case.

In the adjustment test case, we have considered again the same initial datum (34) with the same spatial domain and boundary conditions as in the non rotating pure gravity wave case. The test was then repeated using the automatic adaptation criterion of the local polynomial degree described in section 5, again with $\epsilon = 5 \times 10^{-3}$. Since the results were entirely analogous, we only show those obtained with the adaptive method.

More precisely, the distance between the adaptive and nonadaptive solution at $t = 36000$ s has been measured again by the relative L_2 error between the adaptive and nonadaptive case , obtaining a value $1.2e-3$ for the solution shown in figure 10.

In table 5, deviations from geostrophic equilibrium in the interior part of the domain are reported for different Courant numbers at $t = 36000$ s, while a snapshot of the solution at the same time is shown in figures 10. It can be observed that the solution is accurately reproduced and the geostrophic component of the flow was found to be insensitive to the value of the Courant number. The time evolution of the ageostrophy indicator is reported in figure 9. Furthermore, also in this case the adaptation criterion appears to be

effective in reducing the computational as it can be observed from the plots of the time evolution of the computational cost reduction Δ^n_{dof} reported in figure 11.

C_{cel}	0.53	1.07	2.14	4.30	8.55
E^{geo}	4.7e-7	4.68e-7	4.7e-7	4.75e-7	5.8e-7

Table 5: Geostrophic adjustment: final deviation from geostrophic equilibrium at different Courant numbers.

In the second test, a balanced initial datum was considered, in order to check the ability of the proposed scheme to maintain a geostrophic equilibrium solution. A balanced initial datum is computed assuming the profile (34) for the free suface. The initial velocity profiles were obtained by fixed point iteration applied to the equations of nonlinear geostrophic equilibrium, using the linear equilibrium solutions as an initial guess. Different spatial resolutions have been considered and results are summarized in table 6.

N_{elem}	E^{geo}	E^{η}_2	E^{η}_1	E^{η}_{∞}
25^2	6.80e-8	3.16e-7	8.85e-8	1.59e-5
50^2	2.16e-10	2.16e-9	1.04e-9	4.93e-8
100^2	9.95e-12	2.14e-10	6.81e-11	5.38e-9

Table 6: Geostrophic equilibrium preservation at fixed $p^{\eta} = 4$, $\Delta t = 100$s with increasing spatial resolution.

6.5 Stommel gyre

Finally, the numerical simulation of a Stommel gyre, see e.g. [41], was considered, in order to perform a test on longer time scales and to allow for non constant values of the Coriolis coefficient. The so called β-plane approximation was assumed, the Coriolis parameter is given by $f = f_0 + \beta y$, with $f_0 = 10^{-4}$ s^{-1} and $\beta = 2 \times 10^{-11}$ m^{-1}s^{-1}, corresponding to mid-latitudes in the northern hemisphere. For this test, a rectangular $[0, L] \times [0, L]$ domain with $L = 10^6$ m was considered, for which the initial mean water depth H was set to 10^3 m and no-flux boundary conditions were assumed. According to Stommel's model, the flow is forced by a wind stress and a linear dissipation term. For these terms, we employed the same form and coefficient values proposed in [10]. More specifically, terms $\boldsymbol{\tau}^s, \boldsymbol{\tau}^b$ are added to the right hand side of the momentum equation, defined

as:

$$\boldsymbol{\tau}^s = \frac{10^{-4}}{h} \sin\left(\pi \frac{(y - L/2)}{L}\right) \mathbf{e}_x, \quad \boldsymbol{\tau}^b = -\gamma \mathbf{u},$$

with $\gamma = 10^{-6}$ s^{-1}. The sinusoidal wind stress induces a a clockwise circulation, while the linear dissipation term balances the wind stress forcing.

An initial condition of rest with zero surface elevation is assumed and the flow is simulated for a total of three months, after which a steady state is approximately reached. The solution obtained after three months, with $p^{\eta,max} = 4$, $\theta = 0.65$ and $\epsilon = 5 \times 10^{-4}$ is reported in figure 13, for which $C_{cel} \approx 2$. Notice that, in this nonlinear Stommel problem, the magnitude of the advection terms along the western boundary is quite large, which then does not allow the wind stress and the rotation to turn the flow around the top-left corner of the basin. For this reason, the solution of the nonlinear problem shows an accumulation of fluid along the top-left corner of the basin. However, the extrema of the free surface height of the linear and nonlinear problem remain similar. The result obtained are in good qualitative and quantitative agreement with the reference solutions presented in [10]. The computational saving achieved by the adaptation algorithm is shown in figure 14 to be around 40% throughout most of the simulation.

7 Conclusions and future perspectives

A full semi - implicit, semi - Lagrangian discretization approach based on $p-$adaptive discontinuous finite elements has been introduced. The proposed technique is aimed at applications to low Mach/Froude number compressible regimes, typical of environmental flows. The SISLDG time discretization method allows to improve substantially the efficiency of DG schemes for environmental applications and to extend previous results in this direction discussed in [31], [30], [32]. Indeed, an approach to SISL discretization which is by now standard for finite difference and continuous finite element methods was proven here to be feasible also for discontinuous finite element methods. To further increase the efficiency of the proposed DG method, the flexibility of DG discretization with respect to the local number of degrees of freedom is fully exploited, by introducing a simple $p-$adaptivity approach that allows to change dynamically the number of degrees of freedom used in each element, without degrading the overall accuracy of the method. Numerical results in a number of relevant test cases have shown that the proposed method

indeed allows to use much longer timesteps than standard explicit schemes with no loss in accuracy. Furthermore, the $p-$adaptation algorithm reduces the computational cost by at least 40% in all the tests performed. These results are quite encouraging and motivate the further extension of the present model to full three dimensional models of atmospheric flow.

Acknowledgements

The present research has been carried out with financial support from the *Abdus Salam International Center for Theoretical Physics*, which includes the PhD grant that supported one of the authors (G.T.) in 2007-11. We are extremely grateful to Filippo Giorgi of ICTP for his strong interest in our work. Financial support has also been provided by Regione Lombardia, by the Italian Ministry of Research and Education in the framework of the PRIN 2008 project *Analisi e sviluppo di metodi numerici avanzati per Equazioni alle Derivate Parziali,* and by the GNCS project *Sviluppi teorici ed applicativi dei metodi Semi-Lagrangiani.* Useful conversations with Frank X. Giraldo and Edie Miglio on the methods studied in this paper are also gratefully acknowledged.

References

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

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

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

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

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

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

[7] B. Cockburn, S. Hou, and C.W. Shu. The Runge-Kutta Local Projection Galerkin Finite Element Method for conservation laws IV: the multidimensional case. *Mathematics of Computation*, 54 (190):545–581, 1990.

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

[9] B. Cockburn and C.W. Shu. The Runge-Kutta Discontinuous Galerkin method for conservation laws, V. *Journal of Computational Physics*, 141:198–224, 1998.

[10] R. Comblen, J. Lambrechts, J.F. Remacle, and V. Legat. Practical evaluation of five partly-discontinuous finite element pairs for the non-conservative shallow water equations. *International Journal of Numerical Methods in Fluids*, 63:701–724, 2010.

[11] J. Coté, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth. The operational CMC-MRB global environmental multiscale (GEM) model, part I: Design considerations and formulation. *Monthly Weather Review*, 126:1343–1395, 1998.

[12] T. Davies, M.J.P. Cullen, A.J. Malcolm, M.H. Mawson, A. Staniforth, A.A. White, and N. Wood. A new dynamical core for the Met Office's global and regional modelling of the atmosphere. *Quarterly Journal of the Royal Meteorological Society*, 131:1759–1782, 2005.

[13] C.N. Dawson, J.J. Westerink, J.C. Feyen, and D. Pothina. Continuous, Discontinuous and coupled Discontinuous-Continuous Galerkin finite element methods for the shallow water equations. *International Journal of Numerical Methods in Fluids*, 52:63–88, 2006.

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

[15] J.E. Flaherty and P. K. Moore. Integrated space-time adaptive hp-refinement methods for parabolic systems. *Applied Numerical Mathematics*, 16:317–341, 1995.

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

[17] F.X. Giraldo, J.S. Hesthaven, and T. Warburton. High-Order Discontinuous Galerkin methods for the spherical shallow water equations. *Journal of Computational Physics*, 181:499–525, 2002.

[18] Y. Hasbani, E.Livne, and M.Bercovier. Finite elements and characteristics applied to advection-diffusion equations. *Computers and Fluids*, 11:71–83, 1982.

[19] P. Houston and B. Süli. 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, 2005.

[20] G.E. Karniadakis and S. Sherwin. *Spectral hp−Element Methods for Computational Fluid Dynamics.* Oxford University Press, 2005.

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

[22] R. J. LeVeque. *Finite Volume Methods for Hyperbolic Problems.* Cambridge University Press, 2002.

[23] K. W. Morton. On the analysis of finite volume methods for evolutionary problems. *SIAM Journal of Numerical Analysis*, 35:2195–2222, 1998.

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

[25] K. W. Morton and E. Süli. Evolution-Galerkin methods and their supraconvergence. *Numerische Mathematik*, 71:331–355, 1995.

[26] R. D. Nair, S.J. Thomas, and R.D. Loft. A Discontinuous Galerkin transport scheme on the cubed sphere. *Monthly Weather Review*, 133:814–828, 2005.

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

[28] A. Quarteroni and A. Valli. *Numerical approximation of partial differential equations.* Springer Verlag, 1994.

[29] J.F. Remacle, J.E. Flaherty, and M. S. Shephard. An adaptive discontinuous galerkin technique with an orthogonal basis

applied to compressible flow problems. *SIAM Review*, 45(1):53–72, 2003.

[30] M. Restelli. *Semi-Lagrangian and semi-Implicit Discontinuous Galerkin methods for atmospheric modelling applications*. PhD thesis, Politecnico di Milano, 2007.

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

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

[33] H. Ritchie, C. Temperton, A. Simmons, M. Hortal, T. Davies, D. Dent, and M. Hamrud. Implementation of the semi-Lagrangian method in a high resolution version of the ECMWF forecast model. *Monthly Weather Review*, 123:489–514, 1995.

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

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

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

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

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

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

[40] A. Staniforth and C. Temperton. Semi-Implicit Semi-Lagrangian Integration Schemes for a Barotropic Finite-

Element Regional Model. *Monthly Weather Review*, 114:2078–2090, 1986.

[41] H. Stommel. A westward intensification of wind-driven ocean currents. *Transactions of the American Geophysical Union*, 29:202–206, 1948.

[42] M. Tanguay, A. Robert, and R. Laprise. A semi-implicit, semi-lagrangian fully compressible regional forecast model. *Monthly Weather Review*, 118:1970–1980, 1990.

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

[44] C. Temperton, M. Hortal, and A. Simmons. A two-time-level semi-Lagrangian global spectral model. *Quarterly Journal of the Royal Meteorological Society*, 127:111–127, 2001.

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

[46] M.E. Vazquez-Cendon. Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal of Computational Physics*, 148:497526, 1999.

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

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

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

[50] O.C. Zienkiewicz, J.P.Gago, and D.W. Kelly. The hierarchical concept in finite element analysis. *Computers and Structures*, 16:53–65, 1983.

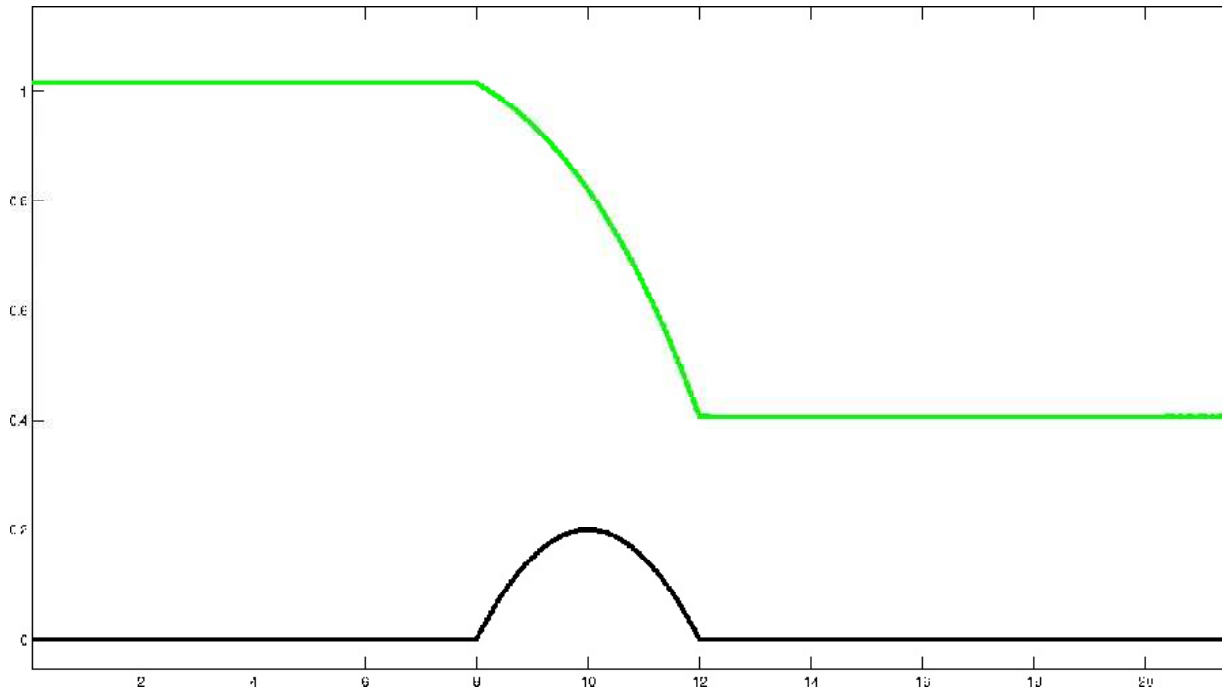
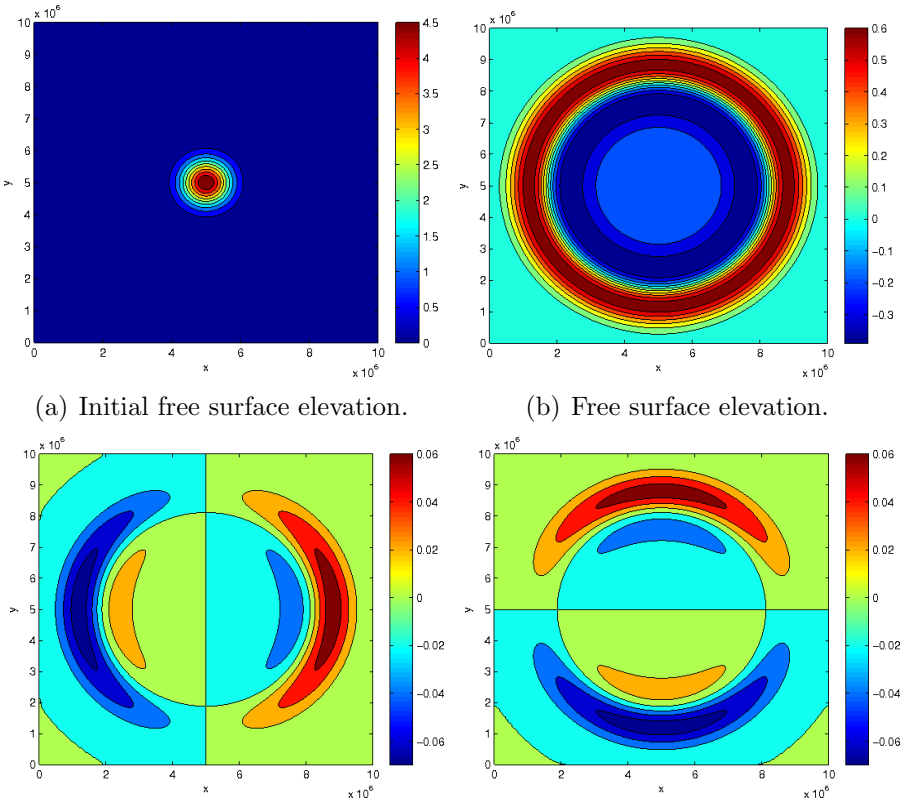


Figure 4: Steady flow over a parabolic bump, free surface elevation in transcritical case, $C_{vel} = 0.62$, $C_{cel} = 1.13$.



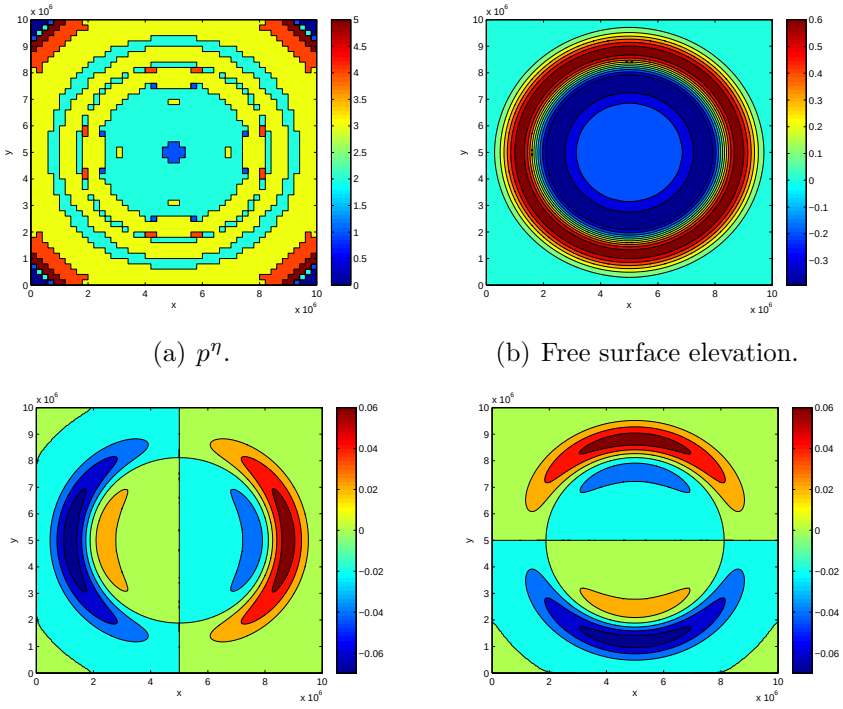
(a) Initial free surface elevation.

(b) Free surface elevation.

(c) Zonal velocity.

(d) Meridional velocity.

Figure 5: Pure gravity wave propagation test: $p^{\eta}_{max} = 5, C_{cel} \approx 0.25$. Solution at $t = 36000$ s.



(a) p^{η}.

(b) Free surface elevation.

(c) Zonal velocity.

(d) Meridional velocity.

Figure 6: Pure gravity wave propagation test: p^{η} and p^{u} adaptively chosen, $C_{cel} \approx 0.25$. Solution at $t = 36000$ s.

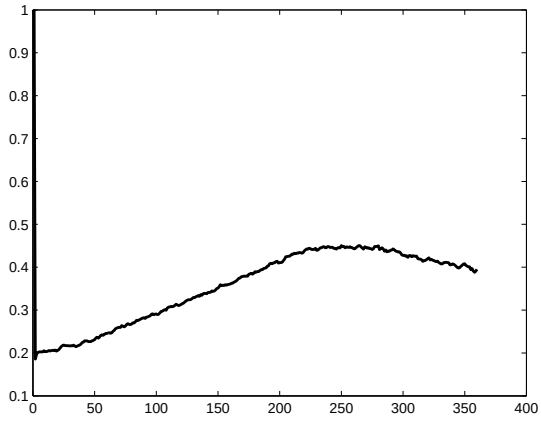


Figure 7: Time evolution of Δ^{n}_{dof} for the adaptive gravity wave propagation test.

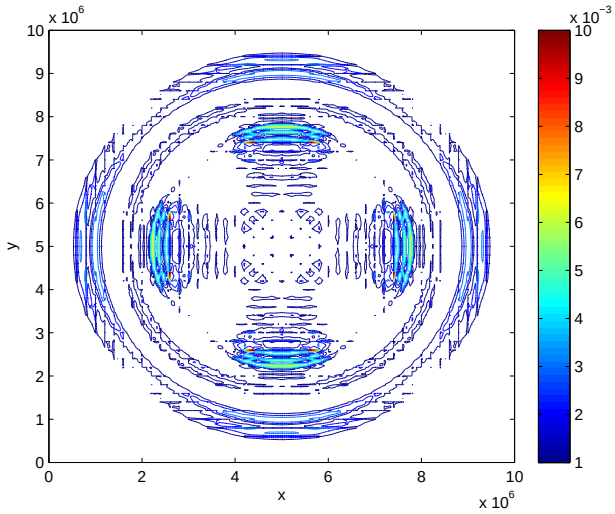


Figure 8: Pure gravity wave test: absolute difference between adaptive and non adaptive solution (free surface) at $t = 36000$ s.

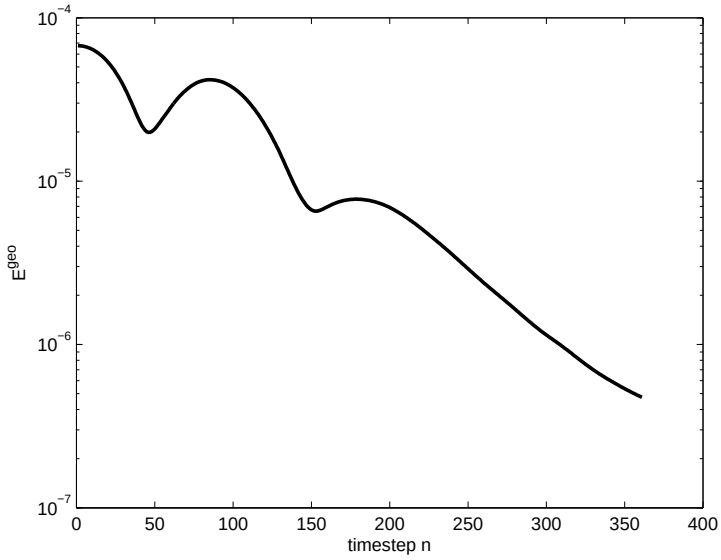
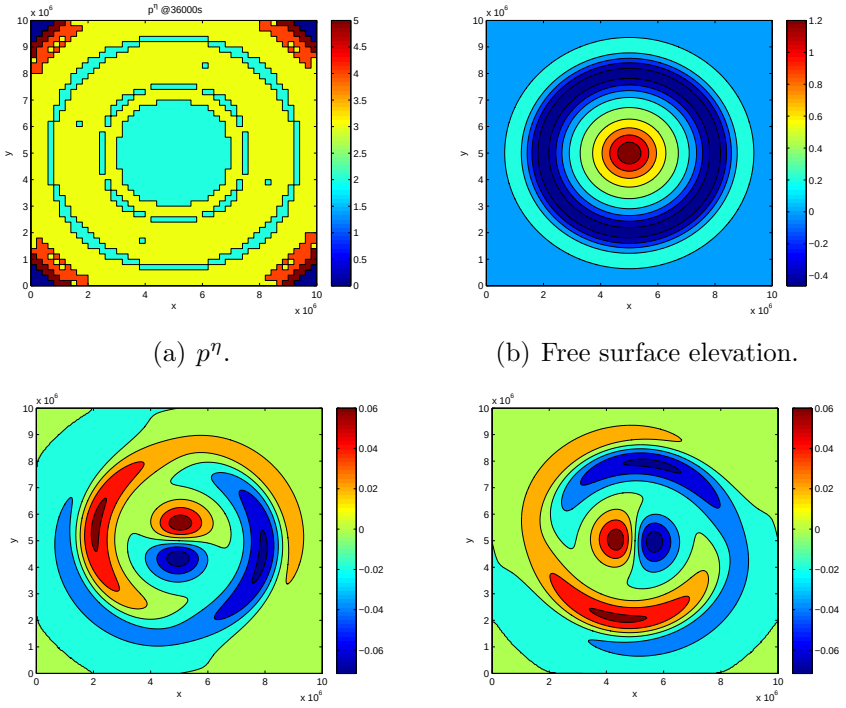


Figure 9: Time evolution of E^{geo} in geostrophic adjustment test case



(a) p^η. (b) Free surface elevation.

(c) Zonal velocity. (d) Meridional velocity.

Figure 10: Geostrophic adjustment test: p^η and p^u adaptively chosen, $C_{cel} \approx 0.25$. Solution at $t = 36000$ s.

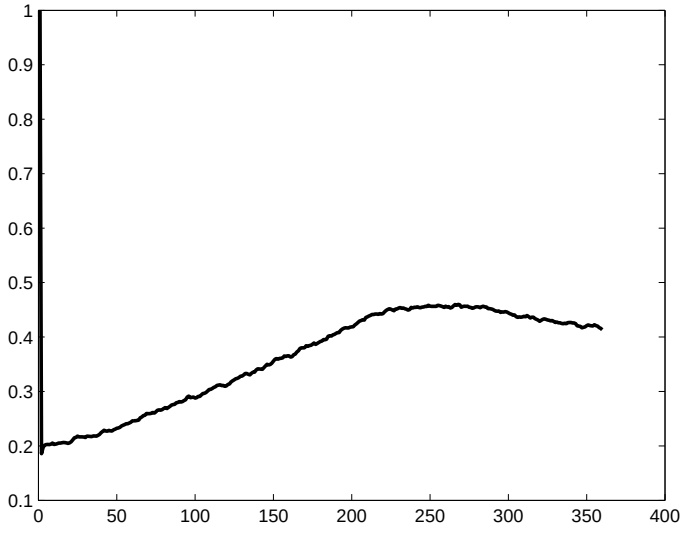


Figure 11: Time evolution of Δ^n_{dof} in geostrophic adjustment test case.

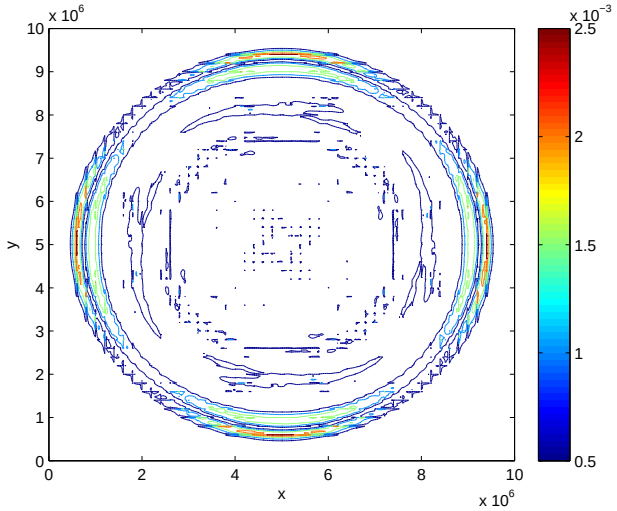
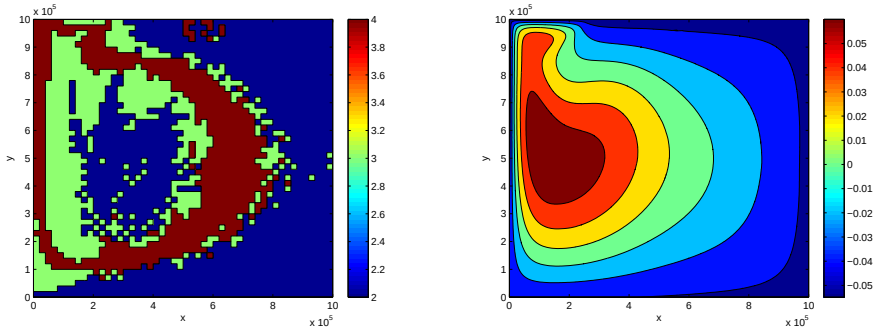
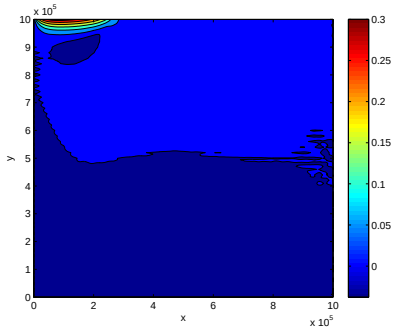


Figure 12: Geostrophic adjustment test: absolute difference between adaptive and non adaptive solution at $t = 36000$ s.

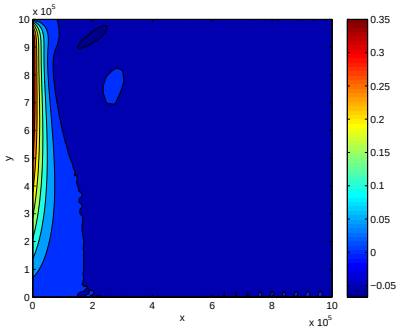


(a) Local polynomial degree p^η.

(b) Free surface elevation.



(c) Zonal velocity.



(d) Meridional velocity.

Figure 13: Stommel test solutions after three months simulations at $C_{cel} \approx 2$.

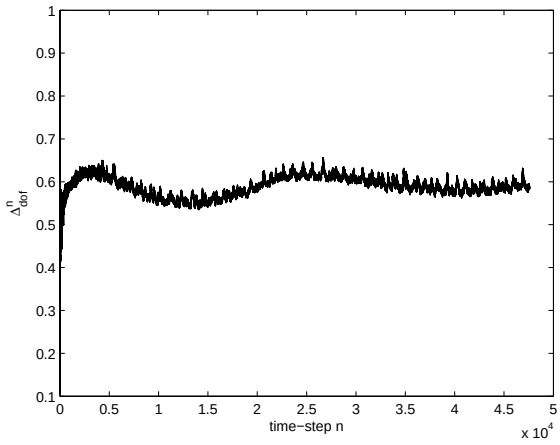


Figure 14: Time evolution of Δ^n_{dof} for the Stommel test.

MOX Technical Reports, last issues

Dipartimento di Matematica "F. Brioschi",
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

04/2012 Tumolo, G.; Bonaventura, L.; Restelli, M.
A semi-implicit, semi-Lagrangian, p-adaptive Discontinuous Galerkin method for the shallow water equations

03/2012 Fumagalli, A.; Scotti, A.
A reduced model for flow and transport in fractured porous media with non-matching grids

02/2012 Arioli, G.
Optimization of the forcing term for the solution of two point boundary value problems

1/2012 Lassila, T.; Manzoni, A.; Quarteroni, A.; Rozza, G.
A reduced computational and geometrical framework for inverse problems in haemodynamics

47/2011 Antonietti, P.F.; Borz, A.; Verani, M.
Multigrid shape optimization governed by elliptic PDEs

46/2011 Migliorati, G.; Nobile, F.; von Schwerin, E.; Tempone, R.
Analysis of the discrete L2 projection on polynomial spaces with random evaluations

45/2011 Canuto, C.; Nochetto, R. H.; Verani, M.
Adaptive Fourier-Galerkin Methods

44/2011 Fumagalli,A.; Scotti,A.
Numerical modelling of multiphase subsurface flow in the presence of fractures

43/2011 L. Formaggia, A. Quarteroni, C. Vergara
On the physical consistency of the coupling between three-dimensional compliant and one-dimensional problems in haemodynamics

42/2011 Antonietti, P.F.; Quarteroni, A.
Numerical performance of discontinuous and stabilized continuous Galerkin methods or convection-diffusion problems