Journal of Computational Physics 228 (2009) 6514–6535

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

High-order discontinuous Galerkin schemes on general 2D manifolds applied to the shallow water equations

P.-E. Bernard [a,*], J.-F. Remacle [a], R. Comblen [a], V. Legat [a], K. Hillewaert [b]

[a] Université catholique de Louvain, Institute for Mechanical, Materials and Civil Engineering, Av. G. Lemaitre 4, 1348 Louvain-la-Neuve, Belgium
[b] CENAERO CFD and Multiphysics Group, Building Mermoz 1, Av. J. Mermoz 30, 6041 Gosselies, Belgium

ARTICLE INFO

Article history:
Received 19 November 2008
Received in revised form 30 April 2009
Accepted 26 May 2009
Available online 11 June 2009

Keywords:
Shallow water equations
High-order finite elements
Discontinuous Galerkin method
Spherical geometry

ABSTRACT

An innovating approach is proposed to solve vectorial conservation laws on curved manifolds using the discontinuous Galerkin method. This new approach combines the advantages of the usual approaches described in the literature. The vectorial fields are expressed in a unit non-orthogonal local tangent basis derived from the polynomial mapping of curvilinear triangle elements, while the convective flux functions are written is the usual 3D Cartesian coordinate system. The number of vectorial components is therefore minimum and the tangency constraint is naturally ensured, while the method remains robust and general since not relying on a particular parametrization of the manifold. The discontinuous Galerkin method is particularly well suited for this approach since there is no continuity requirement between elements for the tangent basis definition. The possible discontinuities of this basis are then taken into account in the Riemann solver on inter-element interfaces.

The approach is validated on the sphere, using the shallow water equations for computing standard atmospheric benchmarks. In particular, the Williamson test cases are used to analyze the impact of the geometry on the convergence rates for discretization error. The propagation of gravity waves is eventually computed on non-conventional irregular curved manifolds to illustrate the robustness and generality of the method.

1. Introduction

Many fields of engineering requires to solve partial differential equations (PDE's) on curved manifolds, as for instance the computation of deformation of shells or vibration of membranes, or geophysical fluids conservation laws for oceanic or atmospheric predictions. Those computations on curved manifolds are essentially two-dimensional since occurring on the surface of a curved body. However, a third dimension has to be taken into account since the body curvature introduces important acceleration terms.

Past studies have seen the emergence of basically two families of methods. The first one consists in writing directly the PDE's on the curved manifold. The conservation laws and vector fields are thus written in a two-dimensional (considering a manifold $\in \mathbb{R}^3$) parametric space [53,52]. This approach has been successfully used to compute the shallow water equations on the surface of the sphere for simple atmospheric models [13,33,45], to compute relativistic gas dynamics on space–time manifolds curved under the influence of gravitational forces from neutron stars or black holes [22,44], or for the propagation of magneto-gravity waves in the solar tachocline [26]. This approach requires to rewrite the PDE's in order to account for the

* Corresponding author. Tel.: +32 10 47 23 53; fax: +32 10 47 21 80.
E-mail address: pe.bernard@uclouvain.be (P.-E. Bernard).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.05.046

curvature terms. The vectors are described using the covariant coordinates and the expression of spatial operators depends on the metric of the manifold's surface. The main advantage of this approach is that vectors contain the minimum number of components and the tangency constraint to the manifold is ensured. Yet, it requires an expression of the metric tensor at any point of the space. This restriction somehow reduces the generality of the method. An example of such a transformation may be found in atmospheric computations using the so-called *cubed-sphere* [51,57,43], mapping the six faces of a cube onto a sphere.

A second approach considers the expression of the vector field and the conservation laws in three dimensions with a single global Cartesian coordinate system [56,55,27,30]. This method allows one to keep a general expression of the PDE, without adding explicit acceleration terms as the Christoffel symbols since the geometry remains hidden in some source terms. The conservation laws remain the same as in the case of a 3D domain computation. The drawback of this approach is the use of a larger number of unknowns, since three components are used instead of two. Moreover, a constraint has to be added for the vector fields to remain tangent to the manifold. When considering explicit time steppers, one basic possible technique consists in projecting the residual and the solution on the local tangent plane for instance at each inner Runge–Kutta time step. This technique is indeed not well suited for implicit time schemes. One could also consider the constraint imposition through Lagrange multipliers [17,28].

In this paper, we propose an alternative approach using the discontinuous Galerkin (DG) finite element method and combining those two classical methods to benefit from the advantages of both approaches.

The DG method consists in a discontinuous finite element method combining the advantageous features of both the classical continuous Galerkin formulations and the finite volumes methods. Indeed, the classical interpolation theory is still valid and the resolution can be locally increased by increasing the polynomial order of the function space, while the discontinuities allow one to build accurate high-order advection schemes. This use of high-order polynomial shape function leads to high convergence rates of the discretization, dispersion or dissipation errors [14,2,8]. Moreover, as the classical FEM, the method is well suited for addressing multi-scale processes since it provides a great flexibility for handling complex geometries and an easy way of imposing natural boundary conditions. It presents efficient computation properties such as an efficient way to perform h-p-adaptivity using the jumps as error-estimator [1,10,19,47,48], or an easy way to perform parallel computations or to use quadrature-free techniques.

First introduced by Reed and Hill [46] for solving the neutron transport equation, the DG method has become a very popular new trend for solving engineering problems where advection plays an important role [14]. The method is now widely used for solving a large variety of problems in many fields of practical engineering such as computational fluid dynamics [49,32,35,5,9,39,38,61], aeroacoustics problems [12,11], levelset methods [41,42], magnetohydrodynamics [60] or hydrodynamics and ocean modeling [50,6,18,36,3,4,29,21,20,28,23,24].

We show here that the DG method is particularly well suited to compute conservation laws on curved manifolds. Indeed, the definition of a local tangent basis on each element is highly simplified since there are no inter-element continuity requirement, while continuous discretizations lead to slightly more complex basis definitions [16]. The alternative approach developed here is applied in particular to the shallow water equations for the computation of oceanic or atmospheric flows. The approach is validated on the sphere with standard test cases proposed by Williamson et al. [62] and Smith and Dritschel [54], and the robustness and generality of the method are eventually tested on some non-conventional irregular manifolds.

2. Several approaches for solving PDE's on curved manifolds

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

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

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = \boldsymbol{0} \tag{1}$$

with $\boldsymbol{U}(\boldsymbol{x}, t)$ the vector of unknown fields and $\boldsymbol{F}$ the flux matrix containing for instance the advection terms or the pressure gradient terms. Note that the addition of a source term is straightforward and is therefore not considered in this section. We restrict ourselves to the mass and momentum conservation laws. In order to identify the scalar and vectorial components of the system, let us define the following quantities:

$$\boldsymbol{U} = \begin{pmatrix} s \\ \boldsymbol{u} \end{pmatrix}, \quad \boldsymbol{F}(\boldsymbol{U}) = \begin{pmatrix} \boldsymbol{G}(\boldsymbol{U}) \\ \boldsymbol{H}(\boldsymbol{U}) \end{pmatrix} \tag{2}$$

with s the scalar unknown and $\boldsymbol{u}$ the vectorial unknown field, i.e. typically the pressure and velocity for mass and momentum conservation laws in fluid dynamics, while $\boldsymbol{G}$ and $\boldsymbol{H}$ denote the sub-vector and sub-matrix of the flux matrix corresponding to the mass and momentum equations, respectively.

This set of equations has to be solved on a general curved manifold $\boldsymbol{x} \in \mathbb{R}^3$ as depicted in the right part of Fig. 1.

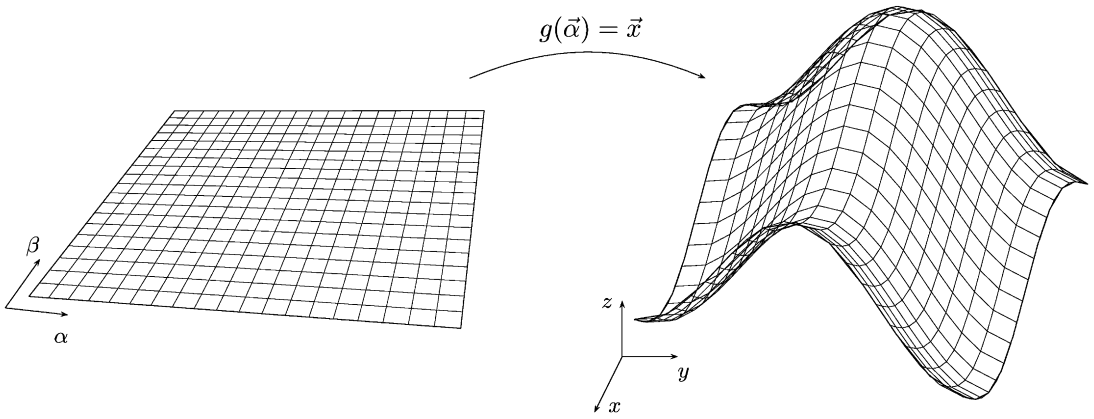

Fig. 1. Transformation g from the parametric space $\boldsymbol{r} \in \mathbb{R}^2$ to the curved manifold in $\boldsymbol{x} \in \mathbb{R}^3$.

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

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

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

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

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

  with (u_α, u_β) the components of the vectorial field in the parametric space.
- 3D approach: another classical approach consists in writing the equations directly in three dimensions, by defining

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

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

Considering the momentum equation, the difference between those methods still lies in the definition of the operators ∇ and the fluxes $\boldsymbol{H}$, but also in the definition of the vectorial unknown field itself.

With the classical *3D* approach, the vectorial field $\boldsymbol{u}$ is written with three components in $\mathbb{R}^3$:

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

The vectorial equation thus reads:

$$\frac{\partial}{\partial t}(u\hat{\boldsymbol{e}}_x + v\hat{\boldsymbol{e}}_y + w\hat{\boldsymbol{e}}_z) + \frac{\partial}{\partial x}\boldsymbol{H}_x + \frac{\partial}{\partial y}\boldsymbol{H}_y + \frac{\partial}{\partial z}\boldsymbol{H}_z = \boldsymbol{0}. \tag{3}$$

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

$$\boldsymbol{H}_a = H_{ax}\hat{\boldsymbol{e}}_x + H_{ay}\hat{\boldsymbol{e}}_y + H_{az}\hat{\boldsymbol{e}}_z, \quad \text{for } a = x, y, z. \tag{4}$$

Taking advantage of the orthogonality of the basis, the scalar product of this vectorial equation (3) with the three basis vectors $\hat{\boldsymbol{e}}_{x,y,z}$ yields the classical system of "decoupled" equations in u, v and w, each equation describing the time evolution of one component:

$$\begin{cases} \frac{\partial u}{\partial t} + \frac{\partial}{\partial x}H_{xx} + \frac{\partial}{\partial y}H_{yx} + \frac{\partial}{\partial z}H_{zx} = 0, \\ \frac{\partial v}{\partial t} + \frac{\partial}{\partial x}H_{xy} + \frac{\partial}{\partial y}H_{yy} + \frac{\partial}{\partial z}H_{zy} = 0, \\ \frac{\partial w}{\partial t} + \frac{\partial}{\partial x}H_{xz} + \frac{\partial}{\partial y}H_{yz} + \frac{\partial}{\partial z}H_{zz} = 0. \end{cases} \tag{5}$$

The main advantage of this approach is the simplicity of the fluxes and the gradient operator. Indeed, the spatial derivatives do not require to be rewritten in the parametric space using the metric tensor. Therefore, no analytical expression of the manifold is required so far, and the expression of the flux function remains the same as in the plane computation. The main drawback is the computational cost: this approach requires the use of three unknowns to represent the vectorial field, i.e. u, v and w, instead of two for the first method. Moreover, there is no guarantee for the vectorial field to remain tangent to the surface. Therefore, the method also requires to ensure the tangency with, for instance, the use of Lagrange multipliers.

On the other hand, the *2D* approach considers a vectorial field $\boldsymbol{u}$ with two components in the parametric space:

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

as the gradients and fluxes:

$$\frac{\partial}{\partial t}(u_\alpha \hat{\boldsymbol{\alpha}} + u_\beta \hat{\boldsymbol{\beta}}) + \frac{\partial}{\partial \alpha}\boldsymbol{H}_\alpha + \frac{\partial}{\partial \beta}\boldsymbol{H}_\beta = \boldsymbol{0}. \tag{6}$$

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

The purpose of the method developed here is precisely to combine the advantages of both methods described above while leaving the drawbacks.

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

$$\frac{\partial}{\partial t}(u_\xi \boldsymbol{t}_\xi + u_\eta \boldsymbol{t}_\eta) + \frac{\partial}{\partial x}\boldsymbol{H}_x + \frac{\partial}{\partial y}\boldsymbol{H}_y + \frac{\partial}{\partial z}\boldsymbol{H}_z = \boldsymbol{0} \tag{7}$$

with the flux vectors $\boldsymbol{H}$ expressed following (4). The scalar product of (7) by $\boldsymbol{t}_\xi$ and $\boldsymbol{t}_\eta$ yields the following system:

$$\begin{cases} \frac{\partial}{\partial t}\left(u_\xi \underbrace{\boldsymbol{t}_\xi \cdot \boldsymbol{t}_\xi}_{1} + u_\eta \boldsymbol{t}_\eta \cdot \boldsymbol{t}_\xi\right) + t_{\xi x}\left(\frac{\partial}{\partial x}H_{xx} + \frac{\partial}{\partial y}H_{yx} + \frac{\partial}{\partial z}H_{zx}\right) + t_{\xi y}\left(\frac{\partial}{\partial x}H_{xy} + \frac{\partial}{\partial y}H_{yy} + \frac{\partial}{\partial z}H_{zy}\right) + t_{\xi z}\left(\frac{\partial}{\partial x}H_{xz} + \frac{\partial}{\partial y}H_{yz} + \frac{\partial}{\partial z}H_{zz}\right) = 0, \\ \frac{\partial}{\partial t}\left(u_\xi \boldsymbol{t}_\xi \cdot \boldsymbol{t}_\eta + u_\eta \underbrace{\boldsymbol{t}_\eta \cdot \boldsymbol{t}_\eta}_{1}\right) + t_{\eta x}\left(\frac{\partial}{\partial x}H_{xx} + \frac{\partial}{\partial y}H_{yx} + \frac{\partial}{\partial z}H_{zx}\right) + t_{\eta y}\left(\frac{\partial}{\partial x}H_{xy} + \frac{\partial}{\partial y}H_{yy} + \frac{\partial}{\partial z}H_{zy}\right) + t_{\eta z}\left(\frac{\partial}{\partial x}H_{xz} + \frac{\partial}{\partial y}H_{yz} + \frac{\partial}{\partial z}H_{zz}\right) = 0. \end{cases} \tag{8}$$

Since the local tangent basis is usually not orthogonal, the time derivatives are no more decoupled into a set of scalar equations as in (5). The PDE remains fully vectorial and the whole system has thus to be solved at once. Moreover, the scalar product with the tangent basis leads to nine flux terms in each equation, while only three terms were not vanishing in each equation of (5), because of the orthogonality of $\hat{\boldsymbol{e}}_{x,y,z}$. But this formulation has the advantage of providing very simple expressions of the fluxes and the spatial derivatives, as in the fully 3D approach, while presenting only two unknowns, u_1 and u_2, and ensuring the conservation of the tangency of the vectorial fields as in the 2D approach. If no analytical expression of a metric tensor is required for writing the fluxes, the complexity is however reported in the expression of the local tangent basis. Even though this formulation may seem quite complex, it turns out that this issue can be easily handled with the finite element method. In particular, the discontinuous finite element method is well suited for this formulation since it leads to a very simple definition of a local high-order tangent basis with no continuity requirement.

3. Local tangent basis definition using the discontinuous Galerkin method

This section is devoted to the discretization and resolution of vectorial PDE's in the form (8) using the discontinuous Galerkin (DG) method. After the derivation of a weak formulation, we define the discretization of the vectorial fields and their local tangent basis. An approximate Riemann solver is then developed to take into account the normals discontinuities between elements. The discrete spatial operators are finally derived for solving the vectorial equation.

3.1. Discontinuous Galerkin discretization

We consider the set of conservation laws in divergence form as written in (1):

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

with $\boldsymbol{U}(\boldsymbol{x},t)$ the vector of unknown fields described in (2), $\boldsymbol{F}$ the flux matrix and $\boldsymbol{S}$ containing the source terms. The domain of computation Ω is divided into a set of $\mathcal{N}_e$ non-overlapping elements Ω_e:

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

Following a classical Galerkin procedure, we obtain the weak formulation by multiplying the scalar and vectorial PDE's (1) by smooth test functions $\hat{\eta}$ and $\hat{\boldsymbol{u}}$, respectively, and by integrating it on each element Ω_e. For sake of simplicity, we only consider the vectorial conservation law for deriving the weak formulation. The extension to the scalar PDE is trivial. The vectorial PDE thus reads:

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}\cdot\hat{\boldsymbol{u}}\,d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{H}(\boldsymbol{U})\cdot\hat{\boldsymbol{u}}\,d\Omega = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U})\cdot\hat{\boldsymbol{u}}\,d\Omega. \tag{9}$$

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

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}^h\cdot\hat{\boldsymbol{u}}^h\,d\Omega + \int_{\Omega_e} \nabla\cdot\boldsymbol{H}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\,d\Omega = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\,d\Omega, \quad \forall\Omega_e \tag{10}$$

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

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

$$\frac{\partial}{\partial t}\int_{\Omega_e} \boldsymbol{u}^h\cdot\hat{\boldsymbol{u}}^h\,d\Omega - \int_{\Omega_e} \boldsymbol{H}(\boldsymbol{U}^h)\cdot\nabla\hat{\boldsymbol{u}}^h\,d\Omega + \int_{\partial\Omega_e} \boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n}\cdot\hat{\boldsymbol{u}}^h\,d\Gamma = \int_{\Omega_e} \boldsymbol{S}(\boldsymbol{U}^h)\cdot\hat{\boldsymbol{u}}^h\,d\Omega, \tag{11}$$

where $\boldsymbol{n}$ denotes the outward unit normals to the element Ω_e and $\partial\Omega_e$ denotes its corresponding edges. This integration on the element edges ensure the information transfer through the normal flux $\boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n}$. The formulation (11) remains however incomplete. Indeed, the unknown fields $\boldsymbol{U}^h$ present different values on both sides of the discontinuity. An appropriate definition of the normal flux has then to be provided:

$$\boldsymbol{H}(\boldsymbol{U}^h)\cdot\boldsymbol{n} = \boldsymbol{H}(\boldsymbol{U}_L, \boldsymbol{U}_R)\cdot\boldsymbol{n} \tag{12}$$

with $\boldsymbol{U}_L$ and $\boldsymbol{U}_R$ the values of the unknown fields on $\partial\Omega_e$, the subscript $\cdot_L$ corresponding to the "left" element Ω_e and $\cdot_R$ to the "right" adjacent element.

The accuracy of the method will depend in a large part on this normal flux definition. Those discontinuities provide an opportunity to introduce accurate advection schemes. Indeed, Riemann-like solvers can be used to compute the normal flux, introducing the right amount of numerical diffusion in the scheme to damp out the unresolved high frequency modes. There are many possible choices for this normal flux function (see for instance [59,37]). However, it is well known that the impact of the choice of the numerical flux decreases as the polynomial order increases [15]. Indeed, considering a hyperbolic linear system, it can be easily shown that the dissipation introduced by the Riemann solver is proportional to the amplitude of the jumps of the solution [14,40], and the jumps varies as h^{p+1}, explaining the decreasing impact of the flux with the order p. No significant difference has been observed with the current model on the sphere using the approximate Roe solver, Lax–Friedrich solver, or exact Riemann solver. We thus apply in this work the simple Lax–Friedrich solver on the flux matrix:

$$\boldsymbol{F}(\boldsymbol{U}^h)\cdot\boldsymbol{n} = \frac{\boldsymbol{F}(\boldsymbol{U}_L)+\boldsymbol{F}(\boldsymbol{U}_R)}{2}\cdot\boldsymbol{n} + |\lambda|_{max}\frac{\boldsymbol{U}_L-\boldsymbol{U}_R}{2}, \tag{13}$$

where $|\lambda|_{max}$ denotes an estimate of the largest eigenvalue of the jacobian on the edge, corresponding to the advection speed of the Riemann invariants.

3.2. High-order mapping and definition of a local tangent basis

Although complete in a mathematical sense, formulation (11) is common to many different methods for computing PDE's on manifolds. The efficiency and robustness of the method will depend on the definition and discretization of the quantities $\boldsymbol{u}^h$ and $\boldsymbol{F}$. Let us now consider the definition of those quantities as expressed in (8) and proceed to their discretization.

3.2.1. Interpolation of a tangent vector field to a parabola

Let us first consider the following one-dimensional simplified problem depicted in Fig. 2. We consider a one-dimensional manifold composed of the unique parabola $\boldsymbol{x}(\xi)\in\mathbb{R}^3$ that passes through the three points $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$. We seek to express the vectorial field $\boldsymbol{u}(\xi)$ as the product of an amplitude $u(\xi)$ and a tangent direction $\boldsymbol{t}(\xi)$:

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

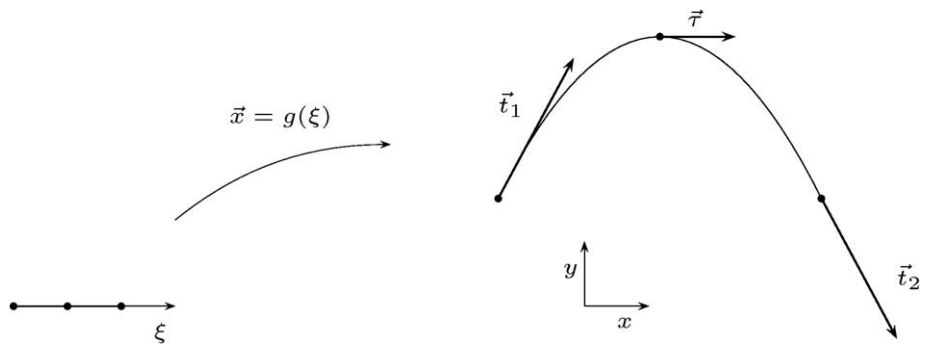

Fig. 2. Mapping from the reference parametric space $\xi \in [-1,1]$ onto the parabola $\boldsymbol{x} \in \mathbb{R}^2$ with $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ the unit tangents corresponding to $\xi = \pm 1$.

One possible representation of the parabola can be written as

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

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

Note that the choice of equidistant nodes used in this paper is not optimal for very high-order polynomials. Alternative nodes positions, using for instance the electrostatic analogy approach [31] or Fekete points, could be considered for such high orders. However, the use of equidistant nodes works fine for the polynomials orders ($p \leqslant 6$) used in this work.

The tangent vector

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

can be written in a more convenient form

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

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

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

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

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

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

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

$$\boldsymbol{u}^h(\xi) = u_1 \underbrace{\left[L_1^1(\xi)^2\boldsymbol{\tau}_1 + L_1^1(\xi)L_2^1(\xi)\boldsymbol{\tau}_2\right]}_{\boldsymbol{T}_1(\xi)} + u_2 \underbrace{\left[L_1^1(\xi)L_2^1(\xi)\boldsymbol{\tau}_1 + L_2^1(\xi)^2\boldsymbol{\tau}_2\right]}_{\boldsymbol{T}_2(\xi)}. \tag{14}$$

Eq. (14) is a vector-valued finite element approximation that makes use of quadratic vector-valued shape functions $\boldsymbol{T}_1(\xi)$ and $\boldsymbol{T}_2(\xi)$ and of scalar coefficients u_1 and u_2.

Clearly, we have $\boldsymbol{T}_1(-1) = \boldsymbol{\tau}_1$ and $\boldsymbol{T}_2(1) = \boldsymbol{\tau}_2$ so that our approximation looks very much like a classical finite element approximation: u_1 and u_2 can be interpreted as vector amplitudes at element end points. Disappointingly, this way of approximating fields cannot be used in a finite element framework because it does not fulfill the simplest patch test: it is easy to see that (14) does not allow one to interpolate a vector of constant amplitude on the parabola. Even if using unit

tangent vector $\boldsymbol{t}_1$ and $\boldsymbol{t}_2$ on the parabola's edges, the combination does not lead to unit tangents inside the domain as depicted in Fig. 2.

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

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

with its derivative with respect to ξ given by

$$\frac{\partial \boldsymbol{t}^h}{\partial \xi} = \frac{\partial \boldsymbol{\tau}^h}{\partial \xi}\frac{1}{\|\boldsymbol{\tau}^h\|} - \frac{\boldsymbol{\tau}^h}{\|\boldsymbol{\tau}^h\|^3}\left(\boldsymbol{\tau}^h \cdot \frac{\partial \boldsymbol{\tau}^h}{\partial \xi}\right). \tag{15}$$

This approximation allows one to make the amplitude of $\boldsymbol{u}^h$ to vary polynomially while remaining tangent to the parabola. Of course, since vector shape functions are not polynomials, the finite element assembly process may appear as more expensive. However, the inverse of the Jacobian matrix used in the usual discrete convective terms is not polynomial either. Moreover, those non-polynomial expressions may be precomputed, leading to an equivalent computational cost compared to conventional approaches.

3.2.2. Interpolation of tangent vector fields on surface meshes

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

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

If $\mathcal{S}$ is sufficiently regular, it is possible to define a tangent plane at any point of its surface. Therefore, this allows one to consider the alternative approach (8), where the vectorial field $\boldsymbol{u}$ is expressed as:

$$\boldsymbol{u}(\xi, \eta) = u_\xi(\xi, \eta)\boldsymbol{t}_\xi(\xi, \eta) + u_\eta(\xi, \eta)\boldsymbol{t}_\eta(\xi, \eta) \tag{16}$$

with $(\boldsymbol{t}_\xi, \boldsymbol{t}_\eta)$ a unit local tangent basis to $\mathcal{S}$. This basis clearly varies in space and its orthogonality is not required. The aim of this section is to define a finite element approximation of (16) that is *a priori* tangent to any manifold $\mathcal{S}$.

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

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

that maps the unit triangle $0 < \xi < 1,\ 0 < \eta < 1 - \xi$ onto $\mathcal{T}$ as depicted in Fig. 3. We assume here the mapping to be polynomial, of order q. The manifold's geometry is thus approximated by the polynomial mapping:

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

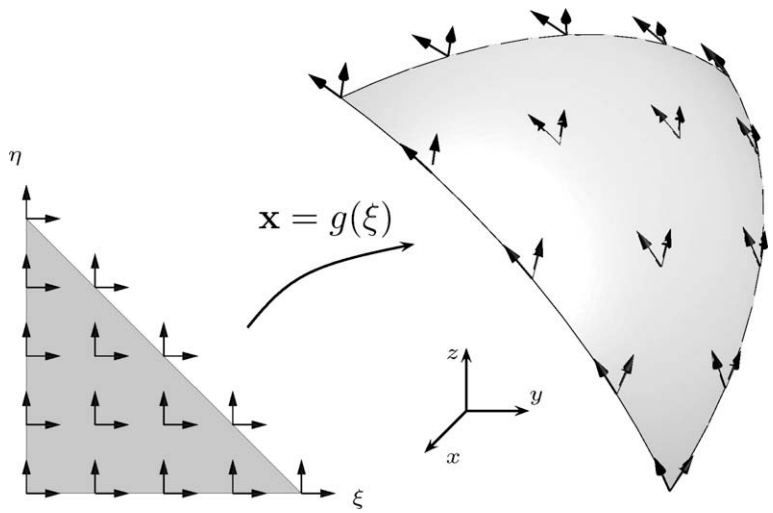

Fig. 3. Mapping of the reference unit triangle $0 < \xi < 1,\ 0 < \eta < 1 - \xi$ onto the curvilinear triangle $\mathcal{T}$.

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

$$\boldsymbol{\tau}_\xi^h(\xi,\eta) = \frac{\partial \boldsymbol{x}^h}{\partial \xi} = \sum_{i=1}^{\mathcal{N}_q} \boldsymbol{x}_i \frac{\partial L_i^q}{\partial \xi}, \tag{17}$$

$$= \sum_{i=1}^{\mathcal{N}_{q-1}} \boldsymbol{\tau}_{\xi i} L_i^{q-1}(\xi,\eta), \tag{18}$$

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

$$\boldsymbol{t}_\xi \simeq \boldsymbol{t}_\xi^h = \frac{\boldsymbol{\tau}_\xi}{\|\boldsymbol{\tau}_\xi\|} \tag{19}$$

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

For the formulation (16) to be complete, the components u_ξ and u_η still have to be discretized. This can be achieved by discretizing the vector amplitudes as classical scalar fields at any order p:

$$u_\xi(\xi,\eta) \simeq u_\xi^h(\xi,\eta) = \sum_{j=1}^{\mathcal{N}_p} L_j^p(\xi,\eta) U_{\xi j}, \tag{20}$$

which yields the following discretized tangent vector field

$$\boldsymbol{u}^h(\xi,\eta) = u_\xi^h(\xi,\eta)\boldsymbol{t}_\xi^h(\xi,\eta) + u_\eta^h(\xi,\eta)\boldsymbol{t}_\eta^h(\xi,\eta) = \sum_{j=1}^{\mathcal{N}_p} \left(L_j^p(\xi,\eta)U_{\xi j}\right)\boldsymbol{t}_\xi^h(\xi,\eta) + \sum_{j=1}^{\mathcal{N}_p} \left(L_j^p(\xi,\eta)U_{\eta j}\right)\boldsymbol{t}_\eta^h(\xi,\eta). \tag{21}$$

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

3.3. Approximate Riemann solver with discontinuous normals

The computation of the weak formulation (11), and in particular the integration on the edges, still requires the definition of an approximate Riemann solver taking into account the possible discontinuities of the normals in the manifold representation. We consider the expression of the vector field (21) in the local tangent basis and we use the simple Lax–Friedrich solver (13):

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

The fluxes are written in the global 3D basis $\hat{\boldsymbol{e}}_{x,y,z}$. Therefore, the vectorial field (21) has to be expressed in three components as well for the flux computation.

Let us consider the configuration depicted in Fig. 4: we wish to compute the normal fluxes on the edge Γ between two triangles Ω_L and Ω_R. To simplify the illustration, those triangles are considered flat, belonging to the two planes P_L and P_R. At each edge point, a local basis is obtained by considering the tangent component $\boldsymbol{m}$, common to both triangles, and the left and right normals $\boldsymbol{n}_{L,R}$. On each triangle, a vector field is defined in both basis, say $\boldsymbol{u}_L \in (\boldsymbol{n}_L, \boldsymbol{m})$ and $\boldsymbol{u}_R \in (\boldsymbol{n}_R, \boldsymbol{m})$.

Let $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$ denote the fluxes computed on the left and right sides of the edge, respectively. For classical computations on a plane, the flux has to be evaluated on only one element. Indeed, the continuity of the normals leads to the same flux value on both sides, $\boldsymbol{F}\cdot\boldsymbol{n} = \boldsymbol{F}^L\cdot\boldsymbol{n}_L = \boldsymbol{F}^R\cdot\boldsymbol{n}_R$, since $\boldsymbol{n}_L = \boldsymbol{n}_R$. But the introduction of discontinuities in the normal directions requires to compute different fluxes $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$ on both sides of the edges.

We consider in this flux computation that the normal discontinuities have no impact on the physical processes computed. Therefore, the normal flow to an edge should neither slow down nor be annihilated by a variation of direction, even in the case of orthogonal normals $\boldsymbol{n}_L\cdot\boldsymbol{n}_R = \boldsymbol{0}$. We thus assume a conservation of the vectorial fields amplitudes across the edge, only the directions have to be modified.

Considering for instance the computation of the left flux, $\boldsymbol{F}^L(\boldsymbol{U}_L,\boldsymbol{U}_R)\cdot\boldsymbol{n}_L$, the left value of the vector field $\boldsymbol{U}_L$ already lies in the appropriate plane P_L, i.e. in the appropriate basis $(\boldsymbol{m},\boldsymbol{n}_L)$, while the direction of the right value $\boldsymbol{u}_R$ has to be modified as

$$\boldsymbol{u}_R^L = (\boldsymbol{u}_R\cdot\boldsymbol{m})\boldsymbol{m} + (\boldsymbol{u}_R\cdot\boldsymbol{n}_R)\boldsymbol{n}_L, \tag{22}$$

where the notation $\boldsymbol{u}_R^L$ denotes the right vector $\boldsymbol{u}_R$ expressed in the left basis $(\boldsymbol{m},\boldsymbol{n}_L)$, and with $\boldsymbol{u}_R\cdot\boldsymbol{m}$ and $\boldsymbol{u}_R\cdot\boldsymbol{n}_R$ the tangential and normal components of $\boldsymbol{u}_R$, respectively.

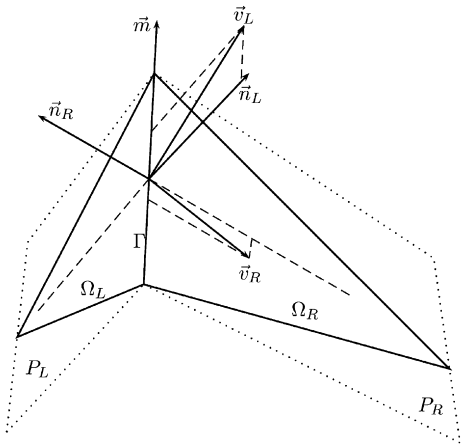

Fig. 4. Illustration of the discontinuous normals on the edge Γ between two flat triangles Ω_L and Ω_R, lying in planes P_L and P_R with corresponding vectorial fields $\boldsymbol{u}_L$ and $\boldsymbol{u}_R$, respectively.

This operation simply corresponds to a rotation of the vectorial field from the basis $(\boldsymbol{m}, \boldsymbol{n}_R)$ to the basis $(\boldsymbol{m}, \boldsymbol{n}_L)$, the amplitude being conserved. The same operation is then applied to $\boldsymbol{u}_L$ to obtain $\boldsymbol{u}_L^R$ for the right flux computation $\boldsymbol{F}^R(\boldsymbol{U}_L, \boldsymbol{U}_R) \cdot \boldsymbol{n}_R$. The Lax–Friedrich's relation eventually yields two expressions for the fluxes on the left and right sides of the edge, $\boldsymbol{F}^L$ and $\boldsymbol{F}^R$:

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

3.4. Mass matrix and convective operators for vectorial test functions

We may now compute the weak formulation on curved manifolds since all the fields and fluxes have been defined and discretized in the local tangent and 3D basis, respectively.

Let us consider an element of order p with $\mathcal{N}_p$ nodal values to represent a scalar field. According to the discretization (21), the vectorial weak equation (11) presents then $2\mathcal{N}_p$ unknown values. The time derivative term from the scalar mass conservation equation is computed using the classical local mass matrix of size $\mathcal{N}_p \times \mathcal{N}_p$, while the time derivative terms from the vectorial momentum equation leads to a local *vectorial* mass matrix of $2\mathcal{N}_p \times 2\mathcal{N}_p$:

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

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

The convective terms are quite more complicated since they involve the spatial derivatives of the tangent basis. Considering the expression of the convective fluxes in 3D as in (8) with three components for the vectorial equation $(\boldsymbol{H}_x, \boldsymbol{H}_y, \boldsymbol{H}_z)$, each component being expressed in 3D as well, $\boldsymbol{H}_a = H_{ax}\hat{\boldsymbol{e}}_x + H_{ay}\hat{\boldsymbol{e}}_y + H_{az}\hat{\boldsymbol{e}}_z$, the discrete convective terms read:

$$\begin{aligned}\int_{\Omega_e} \boldsymbol{H} \cdot \nabla \hat{\boldsymbol{u}}^h \, d\Omega &= \int_{\Omega_e} \boldsymbol{H} \cdot \nabla\left(L_i^p \boldsymbol{t}_\beta^h\right) d\Omega = \int_{\Omega_e} H_{ab} \frac{\partial}{\partial a}\left(L_i^p t_{\beta b}^h\right) d\Omega \quad \text{for } i \in [1, \mathcal{N}_p],\ \beta \in [\xi, \eta] \\ &= \int_{\Omega_e} H_{ab}\left(t_{\beta b}\frac{\partial L_i^p}{\partial a} + L_i^p \frac{\partial t_{\beta b}^h}{\partial a}\right) d\Omega \end{aligned} \tag{23}$$

with implicit summation on indices $a, b = x, y, z$ denoting the components in the three-dimensional basis.

The derivation of the tangent basis $\frac{\partial t_{\beta b}^h}{\partial a}$ involves the inverse of the Jacobian matrix of the mapping and the derivatives with respect to the reference triangle basis $(\xi, \eta) \in \mathbb{R}^2$ according to (15). This tangent derivation clearly yields second derivatives of the mapping, which corresponds to the curvature of the manifold. Therefore, the tangent derivative term in (23) corresponds to the curvature terms usually introduced in the classical 2D approach, including the Christoffel symbols. The difference lies in the fact that the current approach is more general since those acceleration terms are contained in the numerical discretization, i.e. in the derivative of the element mappings. No analytical computation is required for describing the manifold. We also observe in (23) that the use of a constant orthogonal three-dimensional basis as $\hat{\boldsymbol{e}}_{x,y,z}$ for defining the vectorial

fields suppresses the curvature terms for $a \neq b$ and introduce a third equation, yielding the classical form of the 3D approach $\int_{\Omega_e} F_{ab} \frac{\partial L_i^p}{\partial a} d\Omega$ where the summation on b disappeared, providing the three usual equations for the three components u_b.

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

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

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

Practically, the convective term (23) has been implemented to minimize the CPU cost, instead of minimizing the memory footprint, i.e. by pre-computing on each element the mass matrix and the nine convective matrices of size $2\mathcal{N}_p \times \mathcal{N}_k$ with $\mathcal{N}_k$ the number of integration points per element. The same implementation for the classical 3D approach requires to store only three matrices of size $\mathcal{N}_p \times \mathcal{N}_k$. Although requiring to store a larger number of matrices, the local tangent basis approach remains more efficient for sufficiently large computations since using only two unknowns for each vectorial field instead of three plus a tangency constraint.

4. Solving the shallow water equations on curved manifolds

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

The following computations have been performed using an explicit classical Runge–Kutta time scheme.

4.1. Governing equations

The shallow water equations describe the flow of a thin layer of fluid under the influence of the gravitational acceleration. The conservation laws are derived from the Navier–Stokes equations integrated on the vertical dimension. The unknown fields are the water elevation $H(\boldsymbol{x}, t) = h(\boldsymbol{x}) + \eta(\boldsymbol{x}, t)$ depicted in Fig. 5 and the mean horizontal transport $\boldsymbol{Hv}(\boldsymbol{x}, t) = (Hu, Hv)$:

$$\frac{\partial H}{\partial t} + \nabla \cdot (H\boldsymbol{u}) = 0, \tag{24}$$

$$\frac{\partial H\boldsymbol{u}}{\partial t} + \nabla \cdot (H\boldsymbol{u}\boldsymbol{u}) + f\boldsymbol{e}_z \times H\boldsymbol{u} + gH\nabla\eta = 0, \tag{25}$$

where t is time, f is the Coriolis parameter, g is the gravitational acceleration and $\eta(\boldsymbol{x}, t)$ denotes the relative surface elevation of the fluid. The vector $\boldsymbol{e}_z$ denotes the local normal unit vector to the geopotential surface.

4.2. Atmospheric benchmarks on the sphere

Four of the seven classical Williamson test cases (W92) are considered [62] in the following, in addition to the perturbed Rossby–Haurwitz waves test case [54]. Some W92 are not considered here since they do not involve vectorial conservation laws, as the advection of the cosine bell, or because of the complicated source terms involved. The Williamson test cases are considered in every paper dealing with computation of oceanic or atmospheric processes on the sphere, and have become *de*

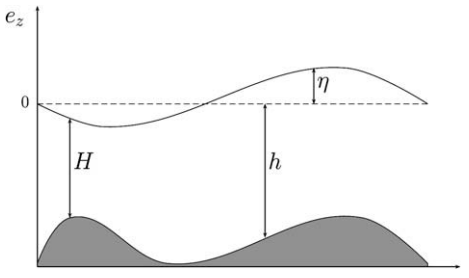

Fig. 5. Definition of the water depth H, bathymetry h and elevation η.

facto standard benchmarks. For the second and third W92 are provided analytical solutions. A refinement convergence study is thus performed by comparing to the analytical solution, while the validation for other test cases requires a comparison to results from other numerical techniques.

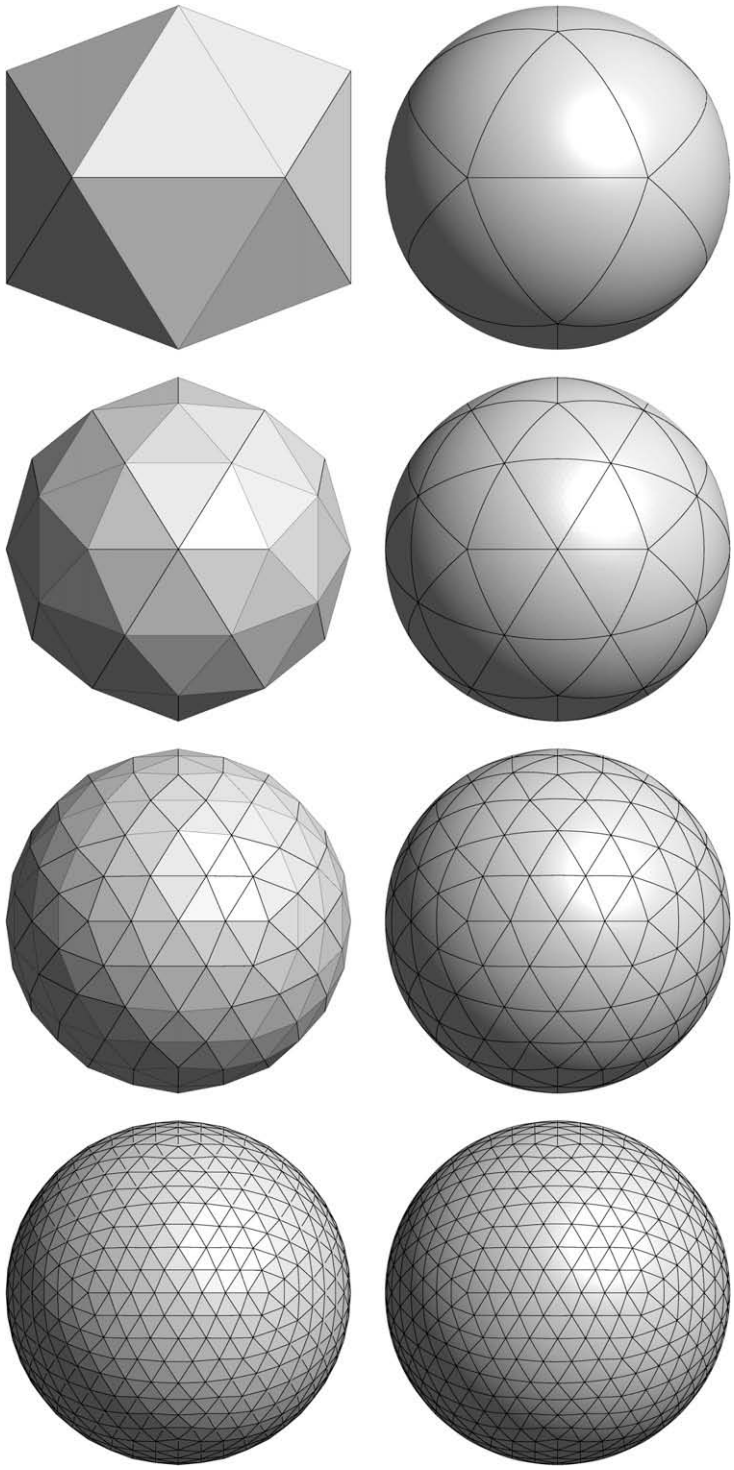

Fig. 6. Icosahedron-deduced meshes of the sphere used for the refinement convergence study. Meshes present 20, 80, 320 and 1280 elements (top to bottom) with $\mathcal{P}^1$ and $\mathcal{P}^4$ polynomial mappings on the left and right columns, respectively.

4.2.1. High-order polynomial mapping

Since using the high-order discontinuous Galerkin method, we may take advantage of the high-order polynomials for both the solution approximation and for the geometry, i.e. the sphere discretization. Some of the meshes used for the computations are depicted on Fig. 6. They are obtained from the recursive divisions of icosahedrons, each triangle being divided in four new triangles projected on the sphere. The left column of Fig. 6 illustrates the classical meshes made of piecewise linear faces, while the right column illustrates the meshes build using a fourth-order mapping, i.e. piecewise fourth-order polynomial faces.

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

$$\boldsymbol{x}_j^h = \sum_{i=1}^{\mathcal{N}_q} \psi_i(\boldsymbol{\xi}) \boldsymbol{X}_{ij}, \quad j = 1, \ldots, 3, \tag{26}$$

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

- Using Lagrange shape function, as for the polynomial approximation of the solution. A mapping of order q requires the a number of coordinates $\mathcal{N}_q = \frac{(q+1)(q+2)}{2}$ to define the curved triangle, as depicted in the right part of Fig. 7.
- Using incomplete Lagrange functions, also known as Serendipity space functions. This representation does not involve any nodal value inside the elements as illustrated in the left part of Fig. 7. A mapping of order q provides thus an incomplete set of monomes for order q, but requires a lower number of coordinates since $\mathcal{N}_q = 3q$.

When considering curved two-dimensional triangles to represent bluff bodies as islands [7], the incomplete Serendipity is quite sufficient since the boundaries representation is actually complete. Moreover, the high-order Lagrange shape functions present nodes located inside the triangle. The position of those nodes may appear arbitrary since it does not add informations on the boundary representation. However, considering curved manifolds, the whole triangle is describing the surface. The use of an incomplete order may thus have some impact on the solution itself.

The inner lines into the triangle depicted in Fig. 7 are resulting from the mapping of the iso-ξ, iso-η and iso-$(1 - \xi - \eta)$ lines in the reference two-dimensional parametric space $\boldsymbol{\xi} = (\xi, \eta)$. Therefore, the space distortion between those lines represents the distortion introduced by the mapping. The incomplete Serendipity representation clearly leads to larger distortions, i.e. larger spatial variations of the Jacobian matrix and its determinant. The Lagrange function spaces seem thus better suited for such mappings. Indeed, strongly distorted elements will lead to high quadrature errors, and then to less accurate solutions. Moreover, Fig. 8 illustrates the evolution of the $\mathcal{L}_2$ norm of the geometric error e_g, computed as

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

on a sphere of unit radius.

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

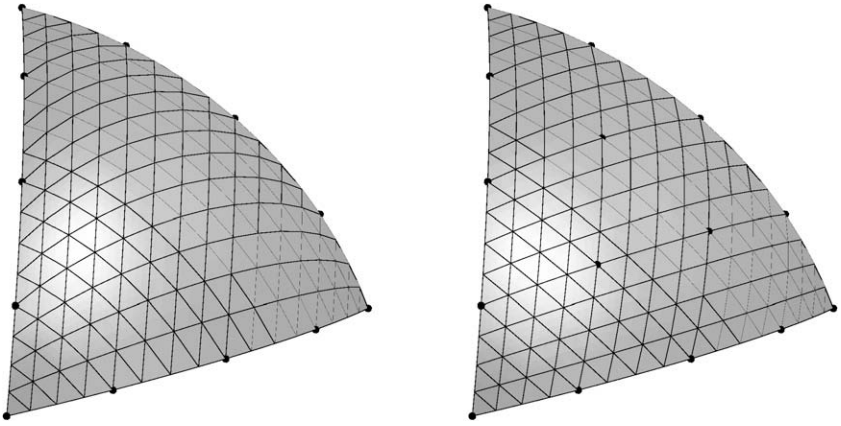

Fig. 7. The same triangle corresponding to a fourth-order mapping, using the incomplete polynomial Serendipity mapping (left triangle) or the complete Lagrange mapping (right triangle) presenting additional nodes inside the element. The inner lines represent the mapped iso-ξ, iso-η and iso-$(1 - \eta - \xi)$ values in the reference parametric domain (ξ, η), and are therefore an illustration of the triangle's mapping deformation.

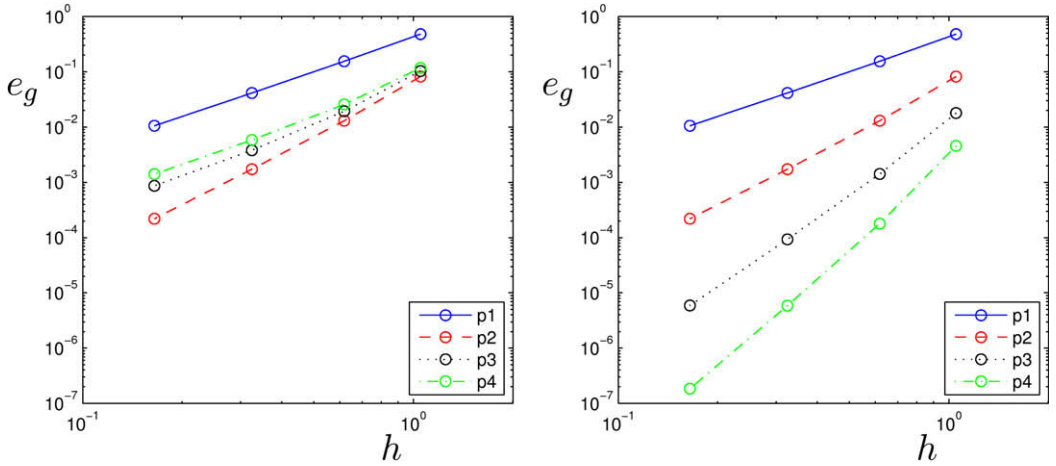

Fig. 8. The expected convergence rate of $\mathcal{O}(h^{p+1})$ for the geometric error is not reached with high-order incomplete Serendipity mappings (left) while rates of 2.01, 3.04, 4.06 and 5.08 are obtained with the Lagrange polynomials mappings (right) for orders $q = 1, \ldots, 4$, respectively.

4.2.2. W92 test cases 2 and 3: steady zonal geostrophic flows

Both test cases 2 and 3 consists in steady state solutions of the non-linear inviscid shallow water equations on the rotating sphere. W92 2 considers a solid body rotation along the Earth's rotation axis while the W92 3 has a non-zero velocity field only for the latitudes ranging between 30S and 90N, i.e. the North pole.

The $\mathcal{L}^2$ norm of the solution error is computed after 5 days as suggested in [62], on the four meshes depicted in Fig. 6 for mappings of order 1–4. The relative error norm is computed as

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

In Fig. 9 are depicted the initial conditions for both test cases: the elevation is defined to compensate the Coriolis effect and the advection terms. The figure also illustrates the evolution of the error with the polynomial solution order, mapping order, and with the relative element size h/R_E with R_E the Earth radius. The plots are related to both mapping orders 1 and 4. The corresponding convergence rates are given in Tables 1 and 2, also including the results for mapping orders 2 and 3.

We observe for the piecewise linear representation of the sphere, i.e. $q = 1$, the expected convergence rate of $\mathcal{O}(h^{p+1})$ for the piecewise linear solution $p = 1$, while this rate is not reached for higher-order solutions. This lack of convergence is clearly due to the fact that the error is dominated by the geometric errors. Indeed, we observe that the theoretical convergence rate is recovered when increasing the mapping order. Notice also that a second order mapping seems sufficient to obtain the expected rate with $p = 3$, while the use of $p = 4$ requires a fourth-order mapping. We finally observe that the test case 3 seems less sensitive to the mapping order, the expected rate for $p = 4$ is almost reached with a second order mapping. This is probably due to the higher interpolation error for W92 3, more than one order of magnitude, while the geometrical errors remains the same.

Finally, sample computational times are given in Fig. 10, presenting the computational cost versus accuracy. Note that these computational times were obtained on a single P4 processor at 3 GHz, using a neglectible memory footprint. The performance of the high-order method is not obvious for very coarse meshes, but becomes clearer for finer meshes (say with more than 500 elements on the sphere using $p1$).

4.2.3. W92 test case 5: zonal flow over an isolated mountain

This test case consists of a zonal flow as in test case 2, except that the bathymetry is not constant: an isolated conical mountain in the longitude–latitude domain leads to an unsteady solution. The mountain, depicted in Fig. 11, presents a radius of 20° at its base and its center is located at a latitude of 30° North.

On Fig. 12 are compared the DG solution and a high-resolution solution from the German Weather Service (model truncation: T-426, 1280 × 640 gridpoints, available online[1]) using a spectral method based on the NCAR's Spectral Transform Shallow Water Model [34]. The quadratic mesh used for the DG computation is depicted in Fig. 11 and contains 242 elements at $p = 6$, i.e. 6776 degrees of freedom. The DG results on such a coarse grid clearly present a good agreement with the reference spectral solution for days 5, 10 and 15.

[1] http://icon.enes.org/swm/stswm/node5.html.

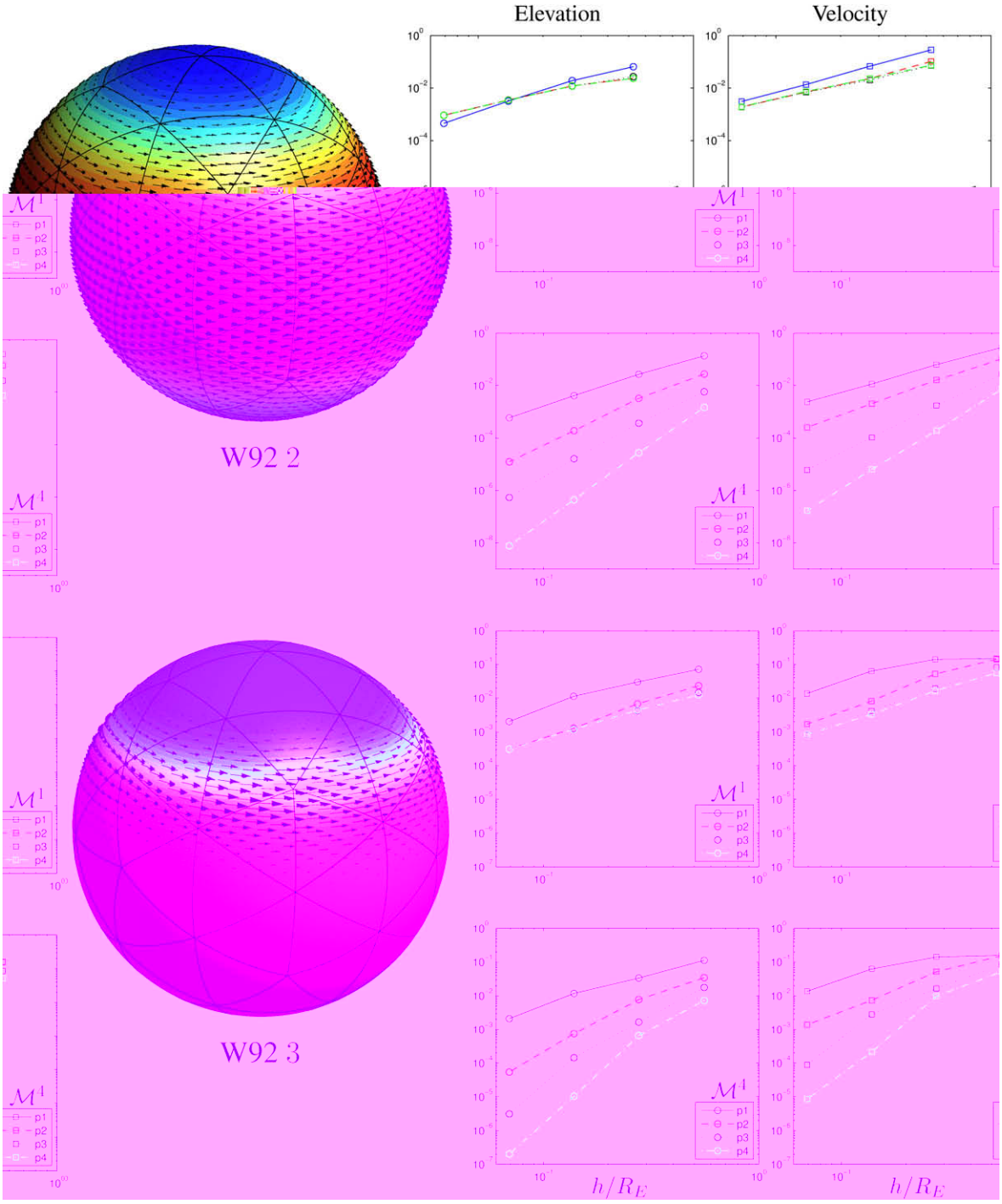

Fig. 9. Steady solution for W92 2 and 3 and the $\mathcal{L}_2$ norm of the relative error computed after 5 days. The left column presents the error based on the elevation field, while the right column is based on the velocity. The expected convergence rate is not reached for a high-order solution using a low (first) order mapping $\mathcal{M}^1$ to represent the sphere, while an iso-parametric representation (fourth-order mapping $\mathcal{M}^4$) ensures the theoretical rate.

The mesh was generated using Gmsh [25]. Note that the generation of high-order curvilinear meshes is a complex topic: ensuring that elements of the mesh have all positive Jacobians is not easy to reach. This is indeed an ongoing work.

4.2.4. W92 test case 6: Rossby–Haurwitz waves

Although providing no analytical solution, this test case has been widely used for model intercomparison. It involves the Coriolis force, the elevation gradients and advection terms, and a time evolving solution. The solutions of the non-divergent

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

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

Table 2
Convergence rates of the relative $\mathcal{L}_2$ error on elevation and velocity of W92 3, for solutions of order $p = 1, \ldots, 4$ and mapping orders $q = 1, \ldots, 4$ to represent the sphere.

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

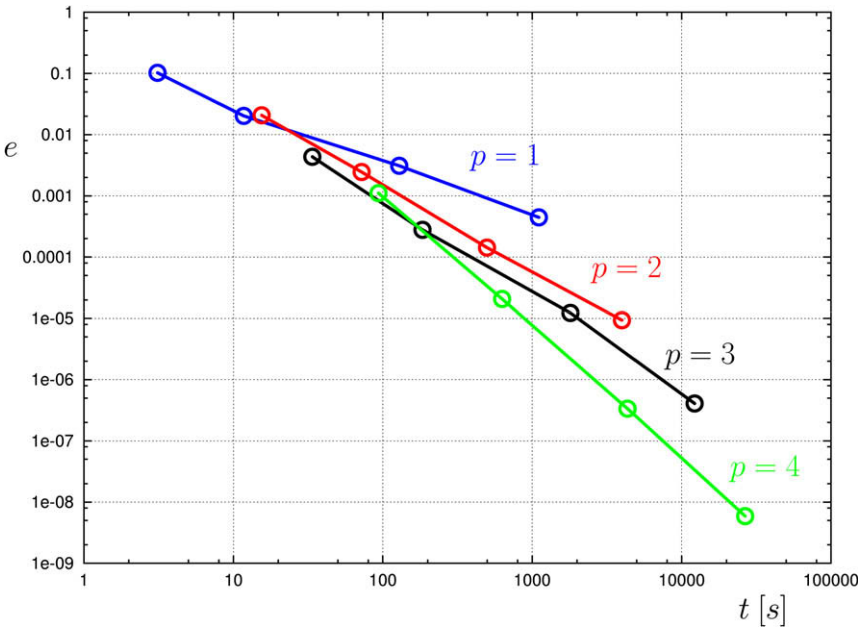

Fig. 10. Accuracy (relative error e) versus computational costs (in seconds) for the W92 test case 2. The efficiency of high-order polynomials is clearly observed for sufficiently fine meshes, say more than 500 elements using $p = 1$.

barotropic vorticity equations experience a slow solid rotation around the Earth under the influence of slow waves. The initial solution presents the 4-periodic pattern depicted in Fig. 13. This solution has been shown to be unstable, and in particular very sensitive to asymmetry in truncature errors [58]. Spectral methods for instance will keep the initial 4-periodic pattern unchanged because of their high degree of symmetry, only the rounding errors are responsible for triggering the instability. On the other hand, the use of icosahedral meshes will excite the 2-periodic component of the unstable mode since the icosahedral mesh and the corresponding truncation errors are 2-periodic [16]. The apparition of this 2-periodic pattern has been observed after about 65 days using the current DG model.

Fig. 14 provides a comparison between the DG solution with the local tangent basis approach (left column) and the solution obtained by the German Weather Service using a spectral method. The DG solution used the 320 elements mesh with fourth-order mapping and solution, leading to 4800 degrees of freedom, while the spectral method was computed on a 1536×768 grid. The comparison reveals a good agreement between the two approaches, especially in the earlier days of the computation when the 2-periodic component of the unstable modes has not been much excited.

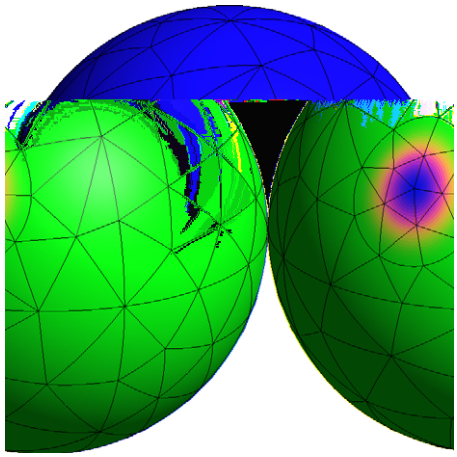

Fig. 11. Bathymetry (ranging between −5960 and −3960 m at the center of the mountain) and mesh used for test case W92 5. A mapping of order $q = 2$ is used to represent both the spherical geometry and the basis of the conical mountain.

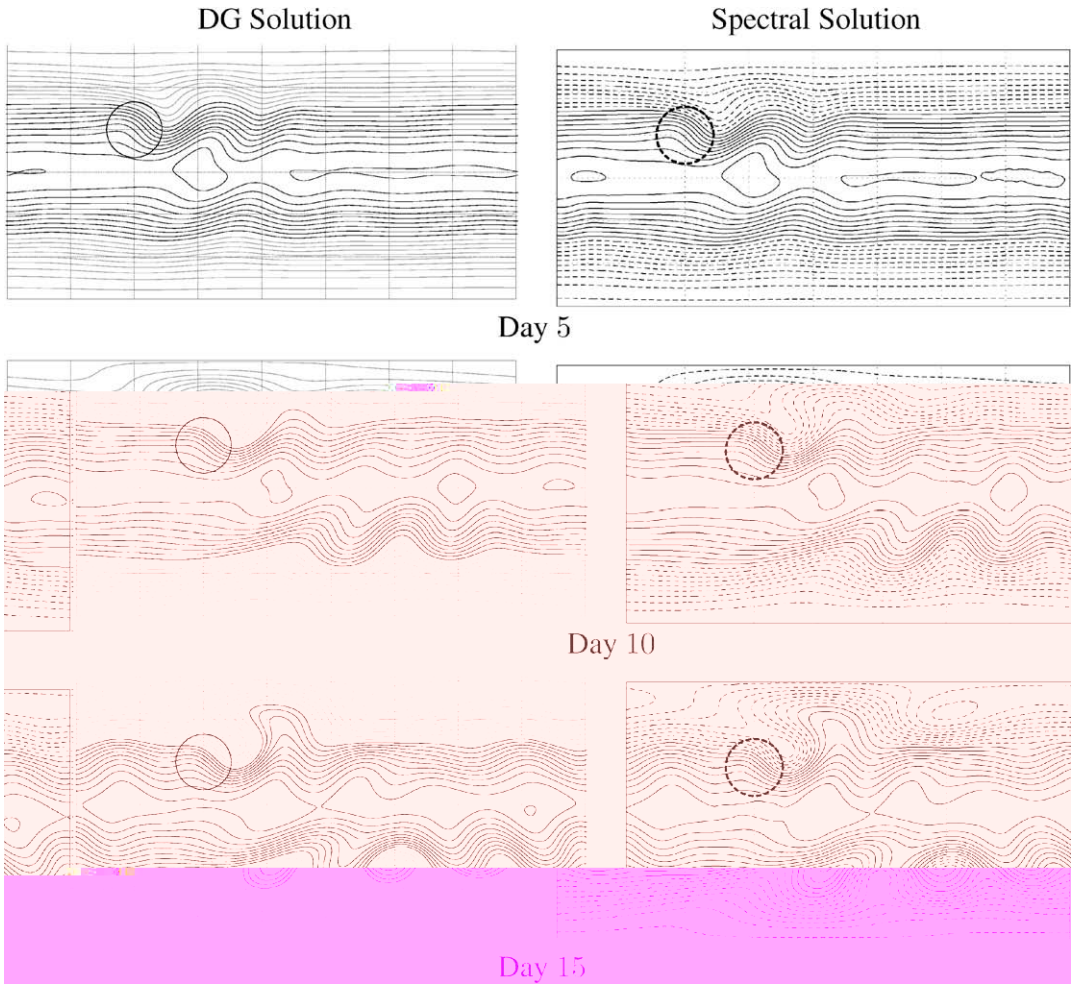

Fig. 12. Comparison of elevation fields between the DG method with the local tangent basis approach (left column) and the reference solution from the German Weather Service using a spectral method for W92 test case 5. The reference solution was obtained on a 1280×640 grid and a 90 s time step, while the DG method considered a 242 elements mesh with $(p, q) = (6, 2)$ and a time step of about 200 s. The interval between contour lines is 50 m for both plots.

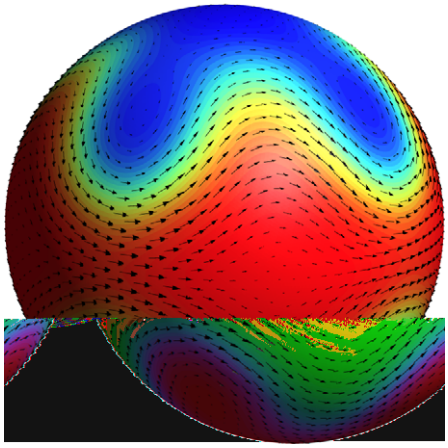

Fig. 13. Four-periodic initial solution of W92 test case 6.

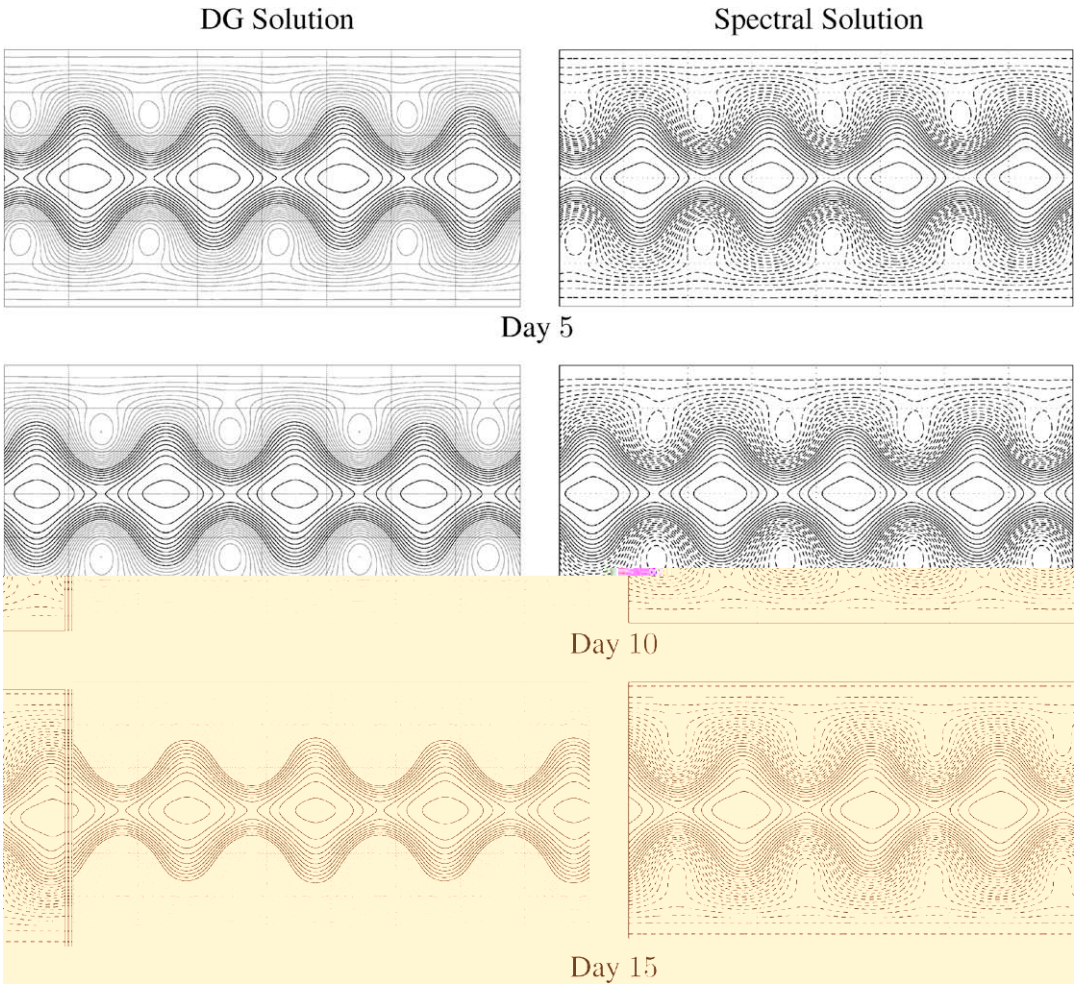

Fig. 14. Comparison between the DG method with the local tangent basis approach (left column) and the reference solution from the German Weather Service using a spectral method for W92 test case 6. The reference solution was obtained on a 1536×768 grid and a 90 s time step, while the DG method considered a 320 elements mesh with $(p, q) = (4, 4)$ and a 200 s time step. The interval between contour lines is 100 m for both plots.

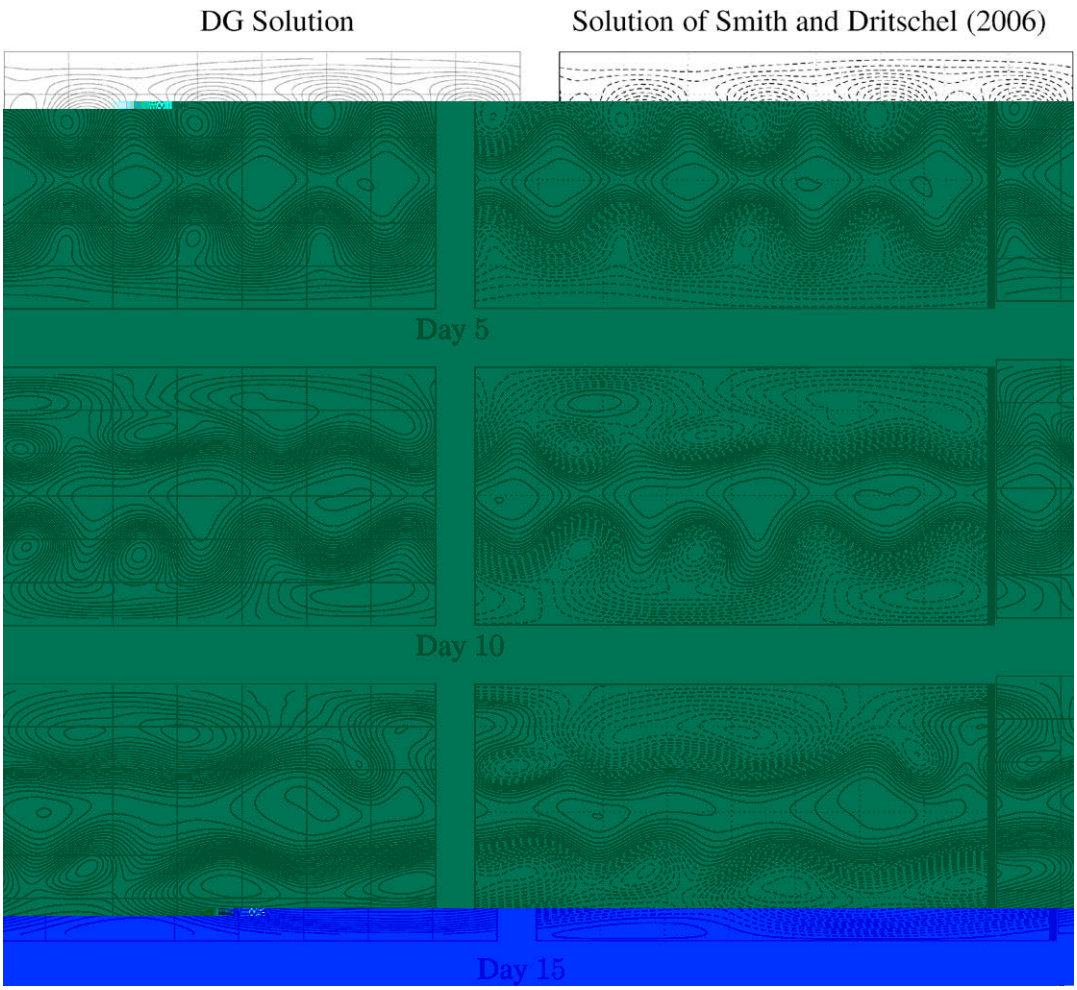

Fig. 15. A good agreement is observed between the DG solution with local tangent basis approach (left column) and the advective-contour semi-Lagrangian method by Smith and Dritschel (right column) after 5, 10 and 15 days for the perturbed Rossby–Haurwitz test case.

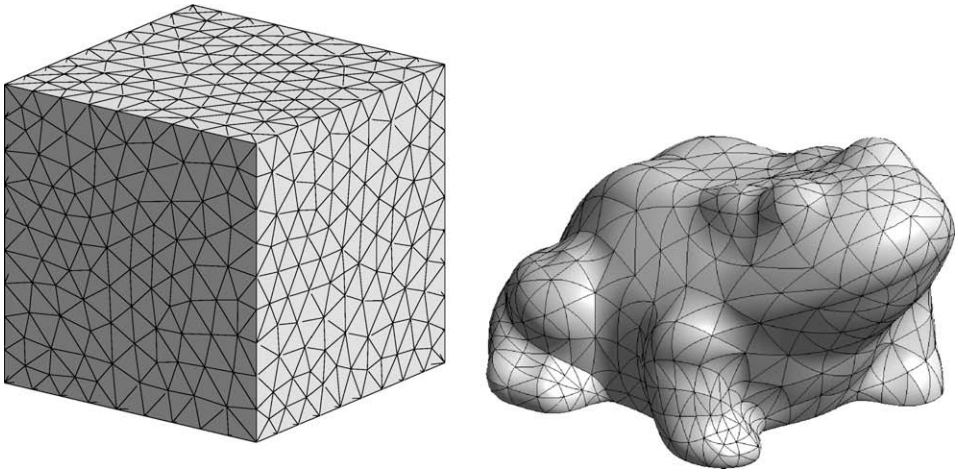

Fig. 16. High-order meshes of irregular curved manifolds.

4.2.5. Perturbed Rossby–Haurwitz test case

Since the W92 test case 6 is very sensitive to the methods and the corresponding truncation errors, it does not facilitate the comparison of different methods. A variation has been proposed by Smith and Dritschel [54], which consists in introducing an initial perturbation. The solution thus becomes more deterministic, especially during the early days, and makes comparisons more relevant.

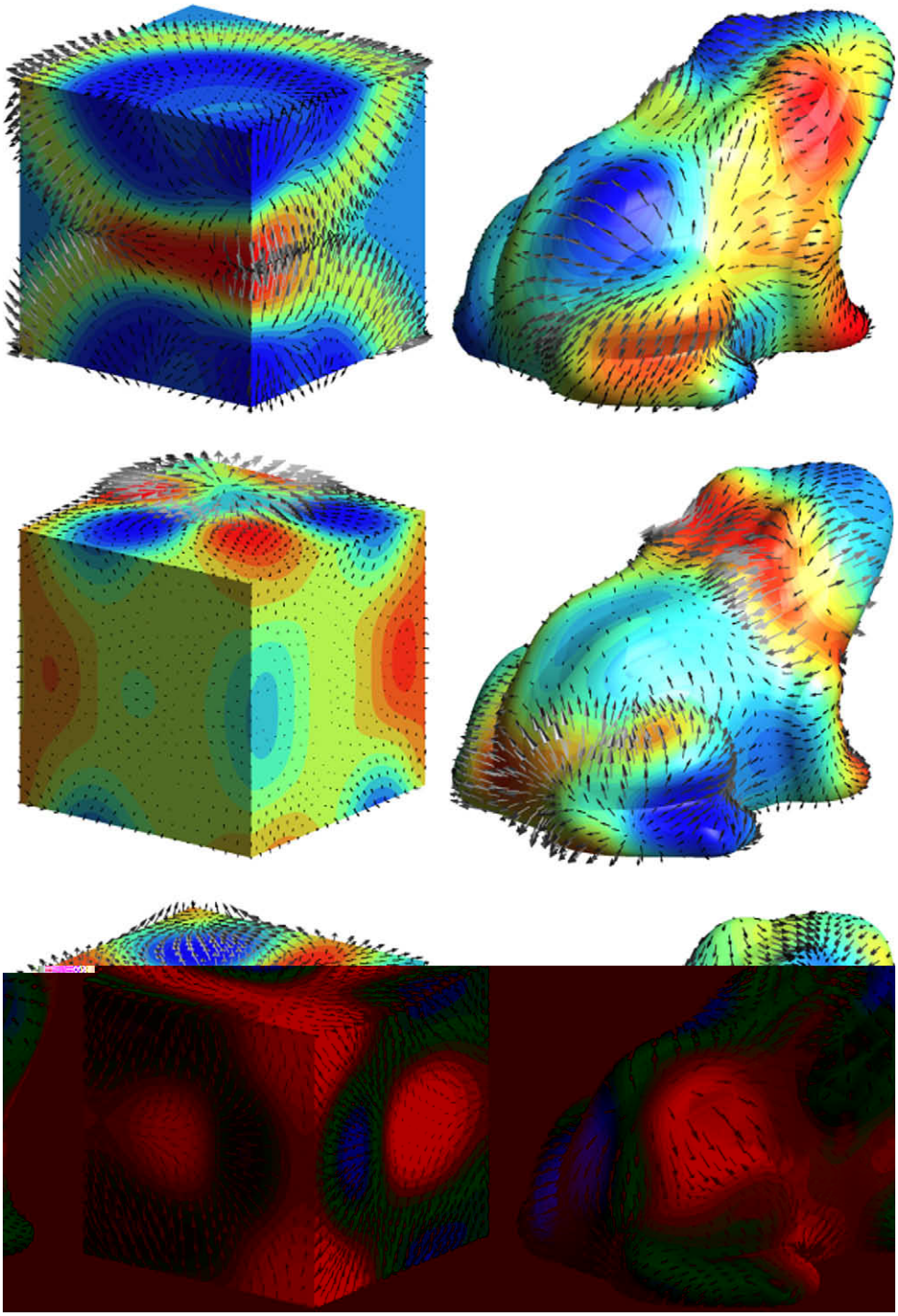

Fig. 17. Propagation of gravity waves on the surface of a cube (left) and of a high-order frog mesh (right), illustrating the robustness and generality of the method.

In Fig. 15 are compared the results obtained with the DG method and the results from [54]. The latter used a 256×256 grid with a contour-advective semi-Lagrangian method, while we considered with our approach the icosahedral mesh of 1280 curved elements with a fourth-order mapping and solution, i.e. 19,200 degrees of freedom. A good agreement between the two approaches is observed.

4.3. Gravity waves on irregular manifolds

We finally validate the robustness and the generality of the method by observing the propagation of gravity waves on the surface of very irregular manifolds. The non-linear shallow water equations are solved without any source term with the same approximate solver and flux function definitions as for the computations on the sphere.

The first computation considers the mesh of the cube depicted in Fig. 16 with fourth-order shape functions to represent the solution. The initial solution is an arbitrary gaussian elevation on the top and bottom of the cube, with a zero velocity. We observe in Fig. 17 that the gravity waves propagate without any perturbation, even in the presence of large discontinuities of the normals directions on the sharp edges of the cube. Indeed, we considered a conservation of the velocities amplitudes through edges in the approximate Riemann solver.

The second computation deals with the gravity waves propagation on the mesh of a frog depicted in the right part of Fig. 16. We observe that the frog is quite well represented with a coarse second order curved mesh. The initial gaussian elevation propagates and becomes highly distorted due to the irregularity of the manifold, while the fourth-order solution remains perfectly smooth. This test case also illustrates the robustness and high level of generality of the method, the more complicated part of the simulation consisting in the creation of high-order meshes while no analytical expression of the manifold is required for computing PDE's.

5. Conclusions

A general and efficient alternative approach has been proposed in this paper for solving vectorial conservation laws on general curved manifolds, combining the advantages of both classical approaches. Indeed, the local tangent basis approach using DG considers a minimum number of unknowns and ensures the tangency constraint of vectorial fields. Moreover, the approach is general and robust since it does not require to rewrite the equations accordingly to some analytical expression of the manifold.

Although the expression of the vectorial fields in a local tangent basis may seem complicated, the computation becomes very efficient, simple and general when considering finite elements, and especially the discontinuous Galerkin method. Indeed, the tangent basis can be computed locally, based on the high-order elements mappings, without continuity requirement of the basis since using a discontinuous approach. These discontinuities have just to be taken into account into the Riemann solver ensuring the information transfer between elements.

The method has been validated with the shallow water equation on the rotating sphere, using some classical Williamson test cases and the perturbed Rossby–Haurwitz benchmark of Smith and Dritschel. The expected convergence rate of the discretization error has been reached for several polynomial orders, using a sufficiently accurate mapping to keep the geometrical errors lower than the discretization ones. The results on more complicated benchmarks have been compared to accurate spectral solutions, revealing a good agreement. The alternative approach has eventually been tested on very irregular high-order manifolds, presenting very distorted shapes and/or large normals discontinuities. Those benchmarks enlighten the robustness and generality of the method, since the only modification between those computations is the manifolds' meshes.

Therefore, considering the shallow water equations, the discontinuous Galerkin method with this local tangent basis approach exhibits a great potential for computing oceanic or especially atmospheric processes on the Earth's surface. Clearly, this general method could also reveal very efficient for any manifold computation as for instance solid mechanics on shells or membranes.

References

[1] S. Adjerid, K.D. Devine, J.E. Flaherty, L. Krivodonova, A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems, Computer Methods in Applied Mechanics and Engineering 191 (2002) 1097–1112.
[2] M. Ainsworth, Dispersive and dissipative behaviour of high order discontinuous Galerkin finite element methods, Journal of Computational Physics 198 (2004) 106–130.
[3] V. Aizinger, C. Dawson, A discontinuous Galerkin method for two-dimensional flow and transport in shallow water, Advances in Water Resources 25 (2002) 67–84.
[4] V. Aizinger, C. Dawson, The local discontinuous Galerkin method for three-dimensional shallow water flow, Computer Methods in Applied Mechanics and Engineering 196 (2007) 734–746.
[5] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element solution of the 2D Euler equations, Journal of Computational Physics 138 (1997) 251–285.
[6] P.-E. Bernard, N. Chevaugeon, V. Legat, E. Deleersnijder, J.-F. Remacle, High-order h-adaptative discontinuous Galerkin methods for ocean modelling, Ocean Dynamics 57 (2007) 109–121.
[7] P.-E. Bernard, J.-F. Remacle, V. Legat, Boundary discretization for high order discontinuous Galerkin computations of tidal flows around shallow water islands, International Journal for Numerical Methods in Fluids, in press, doi:10.1002/fld.1831.
[8] P.-E. Bernard, J.-F. Remacle, V. Legat, Dispersion analysis of discontinuous Galerkin schemes applied to Poincaré, Kelvin and Rossby waves, Journal of Scientific Computing 34 (2008) 26–47.

[9] K. Bey, J. Oden, A Runge–Kutta discontinuous Galerkin finite element method for high speed flows, in: Proceedings of AIAA 10th Computational Fluid Dynamics Conference, Honolulu, Hawaii, 1991.
[10] R. Biswas, K. Devine, J. Flaherty, Parallel, adaptive finite element methods for conservation laws, Applied Numerical Mathematics 14 (1994) 255–283.
[11] N. Chevaugeon, J.-F. Remacle, X. Gallez, P. Ploumans, S. Caro, Efficient discontinuous Galerkin methods for solving acoustic problems, in: Proceedings of the 11th AIAA/CEAS Aeroacoustics Conference, 2005.
[12] N. Chevaugeon, J. Xin, P. Hu, X. Li, D. Cler, J.E. Flaherty, M.S. Shephard, Discontinuous Galerkin methods applied to shock and blast problems, Journal of Scientific Computing 22–23 (2005) 227–243.
[13] J.Y.-K. Cho, L.M. Polvani, The emergence of jets and vortices in freely evolving, shallow-water turbulence on a sphere, Physics of Fluids 8 (1996) 1531–1552.
[14] B. Cockburn, G.E. Karniadakis, C.W. Shu, Discontinuous Galerkin Methods. Theory, Computation and Applications, Lecture Notes in Computational Science and Engineering, vol. 11, Springer-Verlag, Berlin, 2000.
[15] B. Cockburn, C.-W. Shu, Runge–Kutta discontinuous Galerkin methods for convection-dominated problems, Journal of Scientific Computing 16 (2001) 173–261.
[16] R. Comblen, S. Legrand, E. Deleersnijder, V. Legat, A finite element method for solving the shallow water equations on the sphere, Ocean Modelling 28 (2009) 12–23.
[17] J. Côté, A Lagrange multiplier approach for the metric terms of semi-Lagrangian models on the sphere, Quarterly Journal of the Royal Meteorological Society 114 (1988) 1347–1352.
[18] C. Dawson, J. Proft, Coupled discontinuous and continuous Galerkin finite element methods for the depth-integrated shallow water equations, Computer Methods in Applied Mechanics and Engineering 193 (2004) 289–318.
[19] K. Devine, J. Flaherty, S. Wheat, A. Maccabe, A massively parallel adaptive finite element method with dynamic load balancing, in: Proceedings of the Supercomputing – SAND 93-0936C, IEEE Computer Society Press, 1993, pp. 2–11.
[20] C. Eskilsson, S. Sherwin, A triangular spectral/hp discontinuous Galerkin method for modelling 2D shallow water equations, International Journal for Numerical Method in Fluids 45 (2004) 605–623.
[21] F. Dupont, C.A. Lin, The adaptive spectral element method and comparisons with more traditional formulations for ocean modeling, Journal of Atmospheric and Oceanic Technology 21 (2004) 135–147.
[22] J. Font, Numerical hydrodynamics in general relativity, Living Reviews in Relativity (2000).
[23] F.X. Giraldo, M. Restelli, A study of spectral element and discontinuous Galerkin methods for mesoscale atmospheric models: equation sets and test cases, Journal of Computational Physics 227 (2008) 3849–3877.
[24] F.X. Giraldo, T. Warburton, A high-order triangular discontinuous Galerkin oceanic shallow water model, International Journal for Numerical Methods in Fluids 56 (2008) 899–925.
[25] C. Geuzaine, J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities, International Journal for Numerical Methods in Engineering, (2009), doi:10.1002/nme.2579.
[26] P. Gilman, Magnetohydrodynamics shallow water equations for the solar tachocline, Astrophysical Journal 544 (2000) L79–L82.
[27] F.X. Giraldo, Lagrange–Galerkin methods on spherical geodesic grids: the shallow water equations, Journal of Computational Physics 160 (2000) 336–368.
[28] F.X. Giraldo, High-order triangle-based discontinuous Galerkin methods for hyperbolic equations on a rotating sphere, Journal of Computational Physics 214 (2006) 447–465.
[29] F.X. Giraldo, J.S. Hesthaven, T. Warburton, Nodal high-order discontinuous Galerkin methods for the spherical shallow water equations, Journal of Computational Physics 181 (2002) 499–525.
[30] F.X. Giraldo, T. Warburton, A nodal triangle-based spectral element method for the shallow water equations on the sphere, Journal of Computational Physics 207 (2005) 129–150.
[31] J.S. Hesthaven, From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, SIAM Journal of Numerical Analysis 35 (1998) 655–676.
[32] K. Hillewaert, N. Chevaugeon, P. Geuzaine, J.-F. Remacle, Hierarchic multigrid iteration strategy for the discontinuous Galerkin solution of the steady Euler equations, International Journal for Numerical Methods in Fluids 51 (2006) 1157–1176.
[33] R. Iacono, M. Struglia, C. Ronchi, Spontaneous form of equatorial jets in freely decaying shallow water turbulence, Physics of Fluids 11 (1999) 1272–1274.
[34] R. Jakob-Chien, J.J. Hack, D.L. Williamson, Spectral transform solutions to the shallow water test set, Journal of Computational Physics 119 (1995) 164–287.
[35] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon, J.E. Flaherty, Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws, Applied Numerical Mathematics 48 (2004) 323–338.
[36] E.J. Kubatko, J.J. Westerink, C. Dawson, hp Discontinuous Galerkin methods for advection dominated problems in shallow water flows, Computational Methods in Applied Mechanics and Engineering 196 (2006) 437–451.
[37] R.J. LeVeque, Finite Volume Methods for Hyperbolic Problems. Cambridge Texts in Applied Mathematics, Cambridge University Press, 2002.
[38] I. Lomtev, G. Karniadakis, A discontinuous Galerkin method for the Navier–Stokes equations, International Journal for Numerical Methods in Fluids 29 (1999) 587–603.
[39] I. Lomtev, C. Quillen, G. Karniadakis, Spectral/hp methods for viscous compressible flows on unstructured 2D meshes, Journal of Computational Physics 33 (1998) 325–357.
[40] E. Marchandise, N. Chevaugeon, J.-F. Remacle, Spatial and spectral superconvergence of discontinuous Galerkin method for hyperbolic problems, Journal of Computational and Applied Mathematics 215 (2006) 484–494.
[41] E. Marchandise, P. Geuzaine, N. Chevaugeon, J. Remacle, A stabilized finite element method using a discontinuous level set approach for the computation of bubble dynamics, Journal of Computational Physics 225 (2007) 949–974.
[42] E. Marchandise, J.-F. Remacle, A stabilized finite element method using a discontinuous level set approach for solving two phase incompressible flows, Journal of Computational Physics 219 (2006) 780–800.
[43] R.D. Nair, S.J. Thomas, R.D. Loft, A discontinuous Galerkin global shallow water model, Monthly Weather Review 133 (2005) 876–888.
[44] J. Pons, J. Font, J. Ibanez, J. Marti, J. Miralles, General relativistic hydrodynamics with special relativistic riemann solvers, Astronomy and Astrophysics 339 (1998) 638–642.
[45] M. Rancic, R. Purser, F. Mesinger, A global shallow-water model using an expanded spherical cube: gnomonic versus conformal coordinates, Quaternary Journal of Royal Meteorology Society 122 (1996) 959–982.
[46] W.H. Reed, T.R. Hill, Triangular mesh method for the neutron transport equation, Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
[47] J.-F. Remacle, J.E. Flaherty, M. Shephard, Parallel algorithm oriented mesh database, in: Proceedings of the 10th International Meshing Roundtable, 2001.
[48] J.-F. Remacle, J.E. Flaherty, M.S. Shephard, An adaptive discontinuous Galerkin technique with an orthogonal basis applied to compressible flow problems, SIAM Review 45 (2003) 53–72.
[49] J.-F. Remacle, X. Li, M.S. Shephard, J.E. Flaherty, Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods, International Journal for Numerical Methods in Engineering 62 (7) (2005) 899–923.
[50] J.-F. Remacle, S. Soares Frazão, X. Li, M. Shephard, An adaptative discontinuous Galerkin method for the shallow water equations, International Journal for Numerical Methods in Fluids (2003) 1–20.

[51] C. Ronchi, R. Iacono, P.S. Paolucci, The cubed sphere: a new method for the solution of partial differential equations in spherical geometry, Journal of Computational Physics 124 (1996) 93–114.
[52] J.A. Rossmanith, A wave propagation method for hyperbolic systems on the sphere, Journal of Computational Physics 213 (2006) 629–658.
[53] J.A. Rossmanith, D.S. Bale, R.J. LeVeque, A wave propagation algorithm for hyperbolic systems on curved manifolds, Journal of Computational Physics 199 (2004) 631–662.
[54] R.K. Smith, D.G. Dritschel, Revisiting the Rossby–Haurwitz wave test case with contour advection, Journal of Computational Physics 217 (2006) 473–484.
[55] G.R. Stuhne, W.R. Peltier, New icosahedral grid-point discretizations of the shallow water equations on the sphere, Journal of Computational Physics 148 (1999) 23–58.
[56] P.N. Swarztrauber, D.L. Williamson, J.B. Drake, The Cartesian method for solving partial differential equations in spherical geometry, Dynamics of Atmospheres and Oceans 27 (1997) 679–706.
[57] M. Taylor, J. Tribbia, M. Iskandarani, The spectral element method for the shallow water equations on the sphere, Journal of Computational Physics 130 (1997) 92–108.
[58] J. Thuburn, Y. Li, Numerical simulations of Rossby–Haurwitz waves, Tellus 52A (2000) 181–189.
[59] E. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, A Practical Introduction, Springer, Berlin, 1997.
[60] T. Warburton, G.E. Karniadakis, A discontinuous Galerkin method for the viscous MHD equations, Journal of Computational Physics 152 (1999) 608–641.
[61] T. Warburton, I. Lomtev, R. Kirby, G. Karniadakis, A discontinuous Galerkin method for the Navier–Stokes equations on hybrid grids, Technical Report Number 97-14, Center for Fluid Mechanics, Division of Applied Mathematics, Brown University, 1997.
[62] D.L. Williamson, J.B. Drake, J.J. Hack, R. Jakob, P.N. Swarztrauber, A standard test set for numerical approximations to the shallow water equations in spherical geometry, Journal of Computational Physics 102 (1992) 211–224.