Journal of Computational Physics 522 (2025) 113618

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Analysis of finite-volume transport schemes on cubed-sphere grids and an accurate scheme for divergent winds

Luan F. Santos [a,*], Joseph Mouallem [b,c], Pedro S. Peixoto [a]

[a] Instituto de Matemática e Estatística, Universidade de São Paulo, Rua do Matão, 1010, 05508-090, São Paulo, SP, Brazil
[b] NOAA/Geophysical Fluid Dynamics Laboratory, 201 Forrestal Rd, 08540, Princeton, NJ, USA
[c] Cooperative Institute for Modeling the Earth System, Atmospheric and Oceanic Sciences program, Princeton University, 300 Forrestal Rd, 08540, Princeton, NJ, USA

ARTICLE INFO

Dataset link: https://github.com/luanfs/FV3_container

Keywords:
Cubed-sphere
Finite-volume
Transport
Advection
Numerical weather prediction
Divergent winds

ABSTRACT

The cubed-sphere finite-volume dynamical core (FV3), developed by GFDL-NOAA-USA, serves as the dynamical core for many models worldwide. In 2019, it was officially designated as the dynamical core for the new Global Forecast System of the National Weather Service in the USA, replacing the spectral model. The finite-volume approach employed by FV3 to solve horizontal dynamics involves the application of transport finite-volume fluxes for different variables. Hence, the transport scheme plays a key role in the model. Therefore, this work proposes to revisit the details of the transport scheme of FV3 with the aim of adding enhancements. We proposed modifications to the FV3 transport scheme, which notably enhanced accuracy, particularly in the presence of divergent winds, as evidenced by numerical experiments. In contrast to the FV3 scheme's first-order accuracy in the presence of divergent winds, the proposed scheme achieves second-order accuracy. For divergence-free winds, both schemes are second-order, with our scheme being slightly more accurate. Additionally, the proposed scheme exhibits slight computational overhead but is easily implemented in the current code. In summary, the proposed scheme offers significant improvements in accuracy, particularly in the presence of divergent winds, which are present in various atmospheric phenomena, while maintaining computational efficiency.

1. Introduction

The Finite Volume Cubed-Sphere Dynamical Core (FV3), developed by the National Oceanic and Atmospheric Administration's Geophysical Fluid Dynamics Laboratory (NOAA-GFDL), has been embraced as the dynamical core for several atmospheric models (cf. e.g. [1–6]). In 2019, FV3 attained significant recognition when it was adopted as the new dynamical core for the Global Forecast System operated by the U.S. National Weather Service (NWS), replacing the spectral transform dynamical core. Additionally, FV3 is used in the Hurricane Analysis and Forecast System (HAFS) from the NWS for tropical cyclone forecasts [7].

Currently, FV3 solves the non-hydrostatic and compressible Euler equations, as described in the technical report [8], using the vertical Lagrangian coordinate approach outlined in [9]. This method treats the vertical terms implicitly, and the horizontal winds are

---

[*] Corresponding author.
E-mail addresses: luan.santos@usp.br (L. F. Santos), joseph.mouallem@noaa.gov (J. Mouallem), ppeixoto@usp.br (P. S. Peixoto).

https://doi.org/10.1016/j.jcp.2024.113618
Received 27 May 2024; Received in revised form 26 September 2024; Accepted 19 November 2024

updated using the shallow-water equations (SWEs) on the designated Lagrangian surfaces. Consequently, the SWEs solver assumes a pivotal role within the FV3 non-hydrostatic solver.

On the other hand, within FV3, the solution of the SWEs is derived using the finite-volume method proposed by [10] extended to the cubed-sphere grid. This approach takes into account the SWEs in their vector invariant form, combining both the C and D-grid staggering in Arakawa notation introduced in [11]. The idea of combining C and D-grids is a unique feature of the scheme proposed by [10] and has also been explored in [12]. Another major feature of this scheme is that the computation of fluid pressure, absolute vorticity, and kinetic energy fluxes is carried out using only transport finite-volume fluxes. Additionally, on the Lagrangian surfaces, the transport scheme is also used for the advection of virtual potential temperature [8, Section 6.1]. Thus, the transport scheme assumes a critical role in shaping the horizontal dynamics of FV3, beyond its traditional function in the dynamical core, as usually observed in tracer transport [13].

The transport scheme employed in FV3, proposed by [14], extends the method introduced by [15] and also by [16] at the same time, moving from latitude-longitude grids to cubed-sphere grids, aiming for better performance on massive parallel supercomputers, which is difficult to achieve on latitude-longitude grids due to the pole problem [13,17]. Cubed-sphere grids were originally proposed by [18] and revisited by [19,20]. This type of grid is an instance of grids based on Platonic solids [17], which consider a Platonic solid circumscribed on the sphere, project its faces onto the sphere's surface, and apply subdivision on the projected faces to generate the grid cells.

The approach of [15,14] involves constructing a two-dimensional (2D) scheme by combining the solution of one-dimensional (1D) conservative transport equations using a direction-splitting strategy on each cube face. The 1D equations are solved using the finite-volume approach of the Piecewise Parabolic Method (PPM) [21,22]. A notable feature of the scheme proposed by [15] is its elimination of the splitting error under conditions where the initial transported scalar field density is constant and the wind is divergence-free. This property is attained through modifications to the inner advection operators employed within the scheme.

Therefore, given the relevance of the FV3 dynamical core, the goal of this study is to reassess and suggest enhancements for the current FV3 transport scheme originally developed by [14], given its pivotal role in the horizontal dynamics of FV3. It is demonstrated in this work that the scheme proposed by [14] assumes constant metric terms during the application of the PPM to each 1D flux integration domain and employs a first-order departure point calculation for the 1D fluxes. We propose a new scheme that incorporates a second-order departure point calculation for the 1D fluxes and eliminates the assumption of constant metric terms. While the proposed scheme does not retain the property of splitting error elimination for divergence-free winds and constant scalar fields, it ensures second-order errors in such scenarios. The proposed scheme performs slightly better for divergence-free wind simulations of the advection equation on the sphere. Notably, the proposed scheme achieves second-order accuracy for divergent winds, while the FV3 scheme is only first-order accurate in this scenario.

This work considers the duo-grid version of FV3 developed by [23], which has been shown to significantly reduce grid imprinting. This improvement is achieved by replacing edge extrapolations from [14] with a scheme that extends gridlines continuously, aligning them with neighboring panels and enabling more accurate stencil computations through 1D Lagrange interpolation. The new scheme is easy to implement in the duo-grid version of the FV3 code and adds only a small extra computational cost within this framework. The extra cost is mainly due to the second-order departure point computation, which requires 1D linear interpolations at each cell edge in both directions of each cubed-sphere panel. However, the duo-grid interpolation creates a computational overhead for parallel computing [24]. Nevertheless, once the computational performance of the duo-grid version of FV3 is optimized, the new scheme presents an attractive alternative for a more accurate transport scheme.

This work is outlined as follows: Section 2 presents the cubed-sphere grids, revisiting both the equiangular and equi-edge grids, and also discusses the treatment of ghost cells, providing all the notations needed for this work. In Section 3, the FV3 transport scheme of [14] is revisited on the cubed-sphere and an alternative scheme is proposed. In Section 4, numerical simulations comparing the proposed scheme with the current FV3 transport scheme are reported. Final thoughts are presented in Section 5.

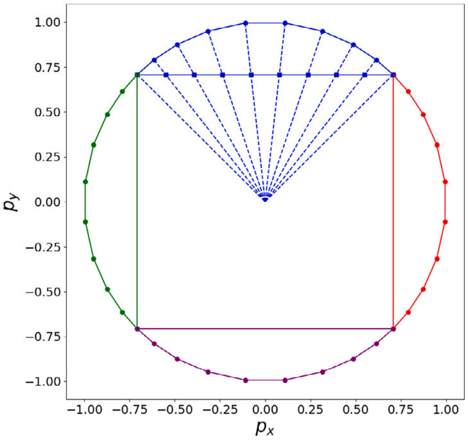
2. Cubed-sphere grids

The goal of this section is to briefly review the concept of cubed-sphere grids, particularly those available in FV3. To start with, the mapping between the cube and the sphere introduced by [18], also known as the equidistant mapping, is presented in Section 2.1. After that, it is shown that by using a change of coordinates, the equidistant mapping can be utilized to create other mappings from the cube to the sphere. Namely, Section 2.2 introduces the equiangular mapping introduced by [20], and Section 2.3 introduces the equi-edge mapping introduced by [24]. Section 2.4 demonstrates how these mappings are utilized to generate cubed-sphere grids and introduces all the notations and tools necessary for the subsequent parts of this work. Finally, on the cubed-sphere, it is needed to define ghost cells, which are cells added outside each cubed-sphere face that allow for stencil computation near to the edges of the spherical cube. Therefore, in Section 2.5, it is demonstrated how the needed ghost cells are generated.

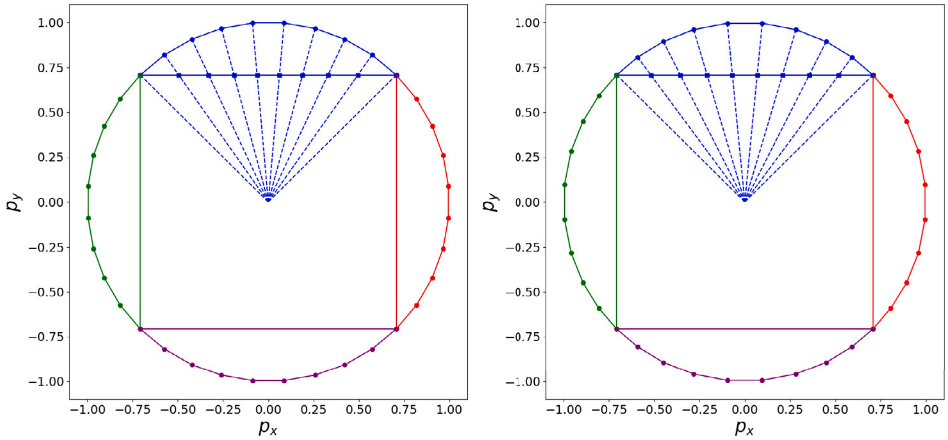
2.1. Equidistant mapping

Considering a cube circumscribed on a sphere, [18] introduces a mapping between the cube faces and the sphere, generating a spherical cube. This mapping is also called equidistant mapping and generates the equidistant grid. Given $R > 0$, the sphere of radius R centered at the origin of $\mathbb{R}^3$ is denoted as:

$$\mathbb{S}^2_R = \{P = (p_x, p_y, p_z) \in \mathbb{R}^3 : p_x^2 + p_y^2 + p_z^2 = R^2\}. \tag{1}$$



(a) Cube and sphere equidistant mapping for $p_z = 0$.



(b) Equiangular mapping for $p_z = 0$.

(c) Equi-edge mapping for $p_z = 0$.

Fig. 1. Illustration of the cube-to-sphere projection using the equidistant (a), equiangular (b) and the equi-edge (c) mappings. This figure uses a cross section obtained with $p_z = 0$.

For the purposes of this work, the Earth radius $R = 6.371 \times 10^6$ meters is considered. The equidistant mapping considers a cube centered at the origin with a side length of $\frac{2R}{\sqrt{3}}$ and radially projects the cube faces onto the sphere (Fig. 1a).

The equidistant mapping is a family of maps $\Gamma_p : [-1, 1] \times [-1, 1] \to \mathbb{S}^2_R$, where $p = 1, \ldots, 6$, defined as follows:

$$\Gamma_1(X,Y) = \frac{R}{\sqrt{1 + X^2 + Y^2}}(1, X, Y), \tag{2}$$

$$\Gamma_2(X,Y) = \frac{R}{\sqrt{1 + X^2 + Y^2}}(-X, 1, Y), \tag{3}$$

$$\Gamma_3(X,Y) = \frac{R}{\sqrt{1 + X^2 + Y^2}}(-X, -Y, 1), \tag{4}$$

$$\Gamma_{3+k}(X,Y) = -\Gamma_k(Y,X), \quad k = 1, 2, 3. \tag{5}$$

These mappings are defined individually for each of the 6 cube faces, also called panels, denoted by p, and they allow for the coverage of the sphere. The idea behind this mapping is illustrated in Fig. 1a, where the coordinates (X,Y) are thought to live on the cube faces. Fig. 1a shows how the grid points are equally spaced on the cube and then projected onto the sphere, hence the name equidistant. Note that there are other ways to arrange the coordinates over the panels. Each one defines a connectivity pattern between the panels, as discussed by [24, Section 2.1]. The connectivity pattern presented here is known as the staircase arrangement (see Figure 2 in [24]), which is used in FV3 and provides some advantages for exchanging information between panels.

The derivative of Γ_p is a 3×2 matrix denoted by $d\Gamma_p$. Explicit formulas are provided in Appendix A. With the aid of the derivative, a basis of tangent vectors $\{\partial_X \Gamma_p, \partial_Y \Gamma_p\}$ may be defined at each point on the sphere, where $\partial_X \Gamma_p$ is given by the first column of $d\Gamma_p$ and $\partial_Y \Gamma_p$ is given by the second column of $d\Gamma_p$. Along side, the metric tensor is defined as $G_\Gamma := (d\Gamma_p)^T \cdot d\Gamma_p$. It is easy to see that the metric tensor does not depend on the panel p. The Jacobian of the metric tensor G_Γ is then defined as $\sqrt{\mathfrak{g}_\Gamma} := \sqrt{|\det G_\Gamma|}$.

Let us assume that it is given a function $\beta : [-a, a] \to [-1, 1]$, for some positive $a > 0$, supposed to be bijective and C^1 with inverse C^1 as well. That is, β is a change of coordinates. Then, new cube-to-sphere mappings may be constructed. Indeed, we may define $\Psi_p : [-a, a] \times [-a, a] \to \mathbb{S}^2_R$, given by

$$\Psi_p(x, y) := \Gamma_p\big(\beta(x), \beta(y)\big). \tag{6}$$

Using the derivatives of $\Psi_p(x, y)$, a basis of tangent vectors $\{\partial_x \Psi_p, \partial_y \Psi_p\}$ induced by this mapping is defined. The metric tensor of Ψ_p, denoted by G_Ψ, is defined as G_Γ, namely $G_\Psi = (d\Psi_p)^T d\Psi_p$. Finally, the metric term for Ψ is defined as $\sqrt{\mathfrak{g}_\Psi} := \sqrt{|\det G_\Psi|}$, which may also be expressed as

$$\sqrt{\mathfrak{g}_\Psi}(x, y) = \|\partial_x \Psi_p\| \|\partial_y \Psi_p\| \sin \alpha(x, y, p), \tag{7}$$

where $\|\cdot\|$ is the Euclidian norm of $\mathbb{R}^3$, α is the angle between $\partial_x \Psi_p$ and $\partial_y \Psi_p$ that satisfies

$$\cos \alpha(x, y, p) = \langle \boldsymbol{e}_x(x, y, p), \boldsymbol{e}_y(x, y, p) \rangle, \tag{8}$$

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product of $\mathbb{R}^3$ and $\{\boldsymbol{e}_x, \boldsymbol{e}_y\}$ is the normalization of the tangent vector basis $\{\partial_x \Psi_p, \partial_y \Psi_p\}$.

Given a tangent vector field $\boldsymbol{u}$ on the sphere, also known as wind, we may represent it using the basis obtained by cubed-sphere coordinates:

$$\boldsymbol{u}(x, y, p) = u(x, y, p)\partial_x \Psi_p(x, y) + v(x, y, p)\partial_y \Psi_p(x, y). \tag{9}$$

This representation (u, v) is known as the contravariant representation, and they are the components of the wind on the tangent basis defined by the cubed-sphere mapping. A detailed discussion on how the cubed-sphere wind representation is related to the zonal and meridional representation is presented in Appendix B. In practice, FV3 schemes [14,8] use the normalized contravariant wind (u, v) given by:

$$\boldsymbol{u}(x, y, p) = \mathsf{u}(x, y, p)\boldsymbol{e}_x(x, y, p) + \mathsf{v}(x, y, p)\boldsymbol{e}_y(x, y, p), \tag{10}$$

where $\boldsymbol{e}_x$ and $\boldsymbol{e}_y$ are the normalized cubed-sphere tangent vectors, which may be computed easily in terms of the grid points [24, Appendix C2]. It is easy to see that:

$$u(x, y, p) = \frac{\mathsf{u}(x, y, p)}{\|\partial_x \Psi_p(x, y)\|}, \quad v(x, y, p) = \frac{\mathsf{v}(x, y, p)}{\|\partial_y \Psi_p(x, y)\|}. \tag{11}$$

Finally, we recall that the horizontal divergence operator for a wind $\boldsymbol{u}$ on the sphere is defined in terms of the cubed-sphere metric terms as follows:

$$[\nabla \cdot \boldsymbol{u}](x, y, p) := \frac{1}{\sqrt{\mathfrak{g}_\Psi}(x, y)} \left(\partial_x(\sqrt{\mathfrak{g}_\Psi} u)(x, y, p) + \partial_y(\sqrt{\mathfrak{g}_\Psi} v)(x, y, p) \right), \tag{12}$$

for $x, y \in [-a, a]$, p is the panel and u and v are the contravariant wind components. The divergence operator shall be used in the transport model in Section 3.

2.2. Equiangular mapping

Another cubed-sphere mapping is the equiangular mapping, introduced by [20], which leads to a more uniform grid on the sphere. This mapping is obtained by considering $\beta(x) = \tan x$ and $a = \frac{\pi}{4}$. In this case, $\beta(x)$ represents the angular coordinates, and the cubed-sphere is obtained by partitioning the angle between grid points equally, as illustrated in Fig. 1b, hence the name equiangular.

2.3. Equi-edge mapping

Another cubed-sphere mapping is the equi-edge mapping, initially introduced by [24], which utilizes $\beta(x) = \sqrt{2} \tan x$ and $a = \arcsin\left(\frac{1}{\sqrt{3}}\right)$. It is worth noting that while this mapping technique had been used previously in FV3, it was not formally documented until the work [24]. The idea behind the equi-edge mapping lies in partitioning the edges of the spherical cube equally and then generating the other cells, resulting in an equidistant-along-edges grid, that we will here call "equi-edge", according to previous use of this terminology [24]. Also, this mapping leads to more uniform grid cells after applying the grid stretching option of FV3 [25,24]. This mapping is illustrated in Fig. 1c.

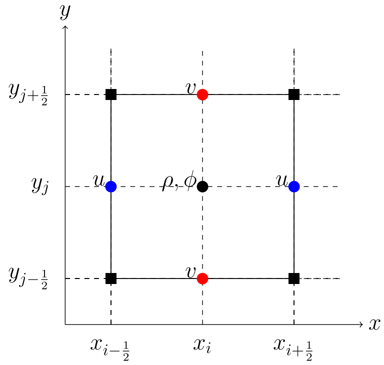


Fig. 2. Illustration of the discrete grid indexes for a cell. The corner points are depicted using black squares, while the centroids are represented using black circles. The left-right and up-down midpoints of the edges, are illustrated in blue and red circles, respectively. Additionally, this figure shows the positions of the contravariant wind components u and v in a C-grid discretization for the transport model, along with the fluid density ρ and tracer concentration ϕ at the centers. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

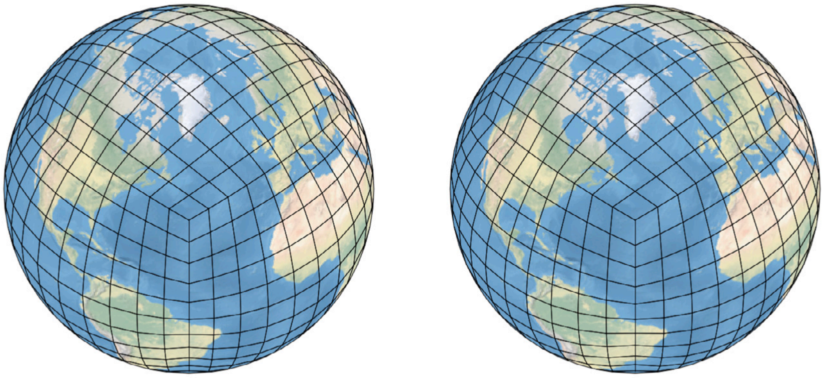

(a) Equiangular grid (b) Equi-edge grid

Fig. 3. (a) Illustration of the resulting gridlines for the cubed-sphere equiangular and equi-edge mapping for $N = 10$.

2.4. *Cubed-sphere grid generation*

Let us fix two positive integers N and ν, where N represents the number of cells in both the x and y directions, and ν represents the number of ghost cell layers. The equiangular or equi-edge mappings, denoted by Ψ_p, introduced previously, are considered to generate the cubed-sphere grid. The notation Ψ_p is used for both equiangular and equi-edge mappings, as what will be discussed does not depend particularly on the mapping. For simplicity, the metric term $\sqrt{\mathfrak{g}_\Psi}$ is denoted by $\sqrt{\mathfrak{g}}$. To generate the cubed-sphere grid, the domain $[-a, a] \times [-a, a]$ is discretized using uniformly spaced points.

$$x_{i+\frac{1}{2}} = -a + i\Delta x, \quad y_{j+\frac{1}{2}} = -a + j\Delta y, \tag{13}$$

where $\Delta x = \Delta y = \frac{2a}{N}$, $i, j = -\nu + 1, \dots, N + 1 + \nu$. The center coordinates are defined as:

$$x_i = \frac{x_{i+\frac{1}{2}} + x_{i-\frac{1}{2}}}{2}, \quad y_j = \frac{y_{j+\frac{1}{2}} + y_{j-\frac{1}{2}}}{2}, \tag{14}$$

for $i, j = -\nu + 1, \dots, N + \nu$. Notice that the mappings Ψ_p defined before can be computed outside the range $[-a, a]$, and the cubed-sphere mapping can be applied to all these ghost cell points. Firstly, we shall focus the attention on the interior cells; the generation of ghost cells shall be addressed in Section 2.5.

There are four types of grid points on the cubed-sphere that are needed to be computed: the center, corners, right-left edge midpoints, and up-down edge midpoints. These points are illustrated in Fig. 2. The corner points are computed as:

$$\Psi_{i+\frac{1}{2},j+\frac{1}{2},p} := \Psi_p(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}), \tag{15}$$

$i, j = 0, \dots, N$. To ease the notation hereafter, the dependence on p is omitted because it does not interfere with what is going to be discussed in this section. Fig. 3 shows the obtained grid lines in for $N = 10$.

Table 1
Mean length, minimum length, and maximum length for different values of N considering the equiangular grid.

N	Mean Length (km)	Min Length (km)	Max Length (km)	$\frac{\text{Max}}{\text{Min}}$
48	220	202	240	1.1890
96	109	99	118	1.1892
192	54	49	59	1.1892
384	27	24	29	1.1892
768	13	12	14	1.1892

The center, corners, right-left edge midpoints, and up-down edge midpoints could be computed similarly using Equation (15). However, in FV3, these points are replaced by averages of the corner points. Thus, the center points are computed by averaging the values of 4 corner points:

$$\Psi_{ij} := R\frac{\Psi_{i+\frac{1}{2},j+\frac{1}{2}}+\Psi_{i+\frac{1}{2},j-\frac{1}{2}}+\Psi_{i-\frac{1}{2},j+\frac{1}{2}}+\Psi_{i-\frac{1}{2},j-\frac{1}{2}}}{\|\Psi_{i+\frac{1}{2},j+\frac{1}{2}}+\Psi_{i+\frac{1}{2},j-\frac{1}{2}}+\Psi_{i-\frac{1}{2},j+\frac{1}{2}}+\Psi_{i-\frac{1}{2},j-\frac{1}{2}}\|}. \tag{16}$$

Similarly, the right-left edge points $\Psi_{i+\frac{1}{2},j}$ are obtained by averaging the values of 2 corner points and the up-down edge points $\Psi_{i,j+\frac{1}{2}}$ are also given by the average of 2 corner points. It is easy to see that generating the grid points using these averages has an $\mathcal{O}(\Delta x^2)$ difference compared to generating these points using a cubed-sphere mapping Ψ_p.

The following geodesic distances in x and y directions, respectively, are introduced:

$$\hat{l}^x_{ij} = d(\Psi_{i+\frac{1}{2},j}, \Psi_{i-\frac{1}{2},j}), \quad \hat{l}^y_{ij} = d(\Psi_{i,j+\frac{1}{2}}, \Psi_{i,j-\frac{1}{2}}), \tag{17}$$

where $d(P,Q) = R\arccos(\langle P, Q\rangle)$, for $P, Q \in \mathbb{S}^2_R$. These distances may be represented in terms of the tangent vector norms as:

$$\hat{l}^x_{ij} = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \|\partial_x \Psi_p\|(x, y_j)\,dx, \quad \hat{l}^y_{ij} = \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \|\partial_y \Psi_p\|(x_i, y)\,dy. \tag{18}$$

Hence, their midpoint approximations are defined as:

$$l^x_{ij} = \|\partial_x \Psi_p(x_i, y_j)\|\Delta x, \quad l^y_{ij} = \|\partial_y \Psi_p(x_i, y_j)\|\Delta y. \tag{19}$$

We point out that l^x_{ij} and l^y_{ij} are replaced in FV3 code by the geodesic distances that they approximate whenever they appear, which are second-order accurate by the midpoint rule. A control volume of the cubed-sphere is denoted by Ω_{ijp}, defined as $\Omega_{ijp} = \Psi_p(\Omega_{ij})$. The area of Ω_{ijp} is denoted by $|\Omega_{ij}|$, since the area does not depend on the panel due to the grid symmetry. The control volume area may be expressed as:

$$|\Omega_{ij}| = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \sqrt{\mathfrak{g}}(x,y)\,dx\,dy = |\hat{\Omega}_{ij}| + \mathcal{O}(\Delta x^2), \tag{20}$$

where

$$|\hat{\Omega}_{ij}| = \sqrt{\mathfrak{g}}_{ij}\Delta x \Delta y, \tag{21}$$

$\sqrt{\mathfrak{g}}_{ij} = \sqrt{\mathfrak{g}}(x_i, y_j)$ and the last equality in Equation (20) follows from the midpoint rule for integration. Similar to the grid lengths, the approximated areas $|\hat{\Omega}_{ij}|$ are replaced by the exact area $|\Omega_{ij}|$ in the FV3 code.

Tables 1 and 2 display the lengths of the equiangular and equi-edge grids for $N = 48 \times 2^k$, where $k = 0, \dots, 4$. These values of N are considered in this work. It can be observed that in terms of length of the cells, the equi-edge grid is less uniform than the equiangular grid. Despite this, the equi-edge grid is the operational grid in some applications of FV3 [8,24], such as, for instance, the Next Generation Global Prediction System (NGGPS) [26]. As mentioned in Section 2.3, the equi-edge grid has greater uniformity near the original cube edges, which are the critical regions of the cube-sphere in terms of grid imprinting [24]. Grid imprinting, which refers to the appearance of grid features on the solution (cf. e.g., [27,28,23]), is a common problem that appears when using Platonic solid based spherical grids, and is highly undesirable due to its lack of physical meaning.

Table 2
As Table 1 but considering the equi-edge grid.

N	Mean Length (km)	Min Length (km)	Max Length (km)	$\frac{\text{Max}}{\text{Min}}$
48	218	175	266	1.5192
96	108	86	131	1.5195
192	54	43	65	1.5196
384	26	21	32	1.5197
768	13	10	16	1.5197

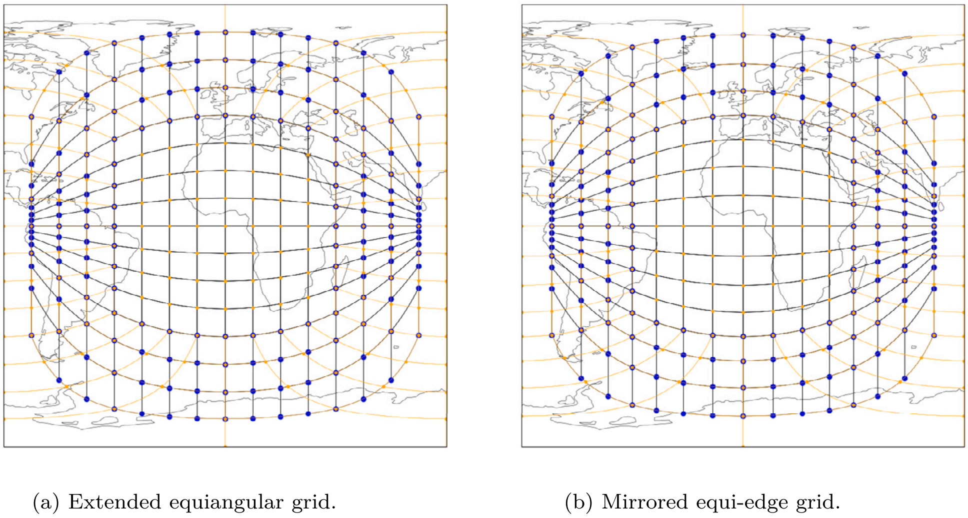

(a) Extended equiangular grid.
(b) Mirrored equi-edge grid.

Fig. 4. Grid lines of panel 1, including ghost cells, for the extended equiangular grid (a) and the mirrored equi-edge grid (b). Corner ghost points are denoted by blue circles, and the corner interior points are denoted by orange points.

2.5. Ghost cells

Currently, FV3 uses the cells of the adjacent panels as ghost cells, employing an approach named kinked grid by [23]. This was the approach used by [14], alongside with extrapolations to compute stencils at or close to a cube edge. In this work, we shall use the extended grid lines of the cubed-sphere mapping to generate the ghost cells, in an approach named duo-grid recently introduced by [24], as it uses the kinked grid values to fill the extended grid values. This approach was recently exploited in FV3 by [23] and helps to reduce grid imprinting, and it shall be considered in this work (see data availability statement). This approach, however, has some scalability issues for parallel computing that we discuss in Section 3.7.

The corner ghost cell points are generated by applying Equation (15) for i and j out of the range 0 to $N+1$. The other grid points are again computed by averaging the corner points, as in the interior grid points. Fig. 4a show how the corner ghost points of the equiangular grid are aligned on common geodesics of the adjacent panels. This property has been known since the work of [20].

The extended grid alignment property is very useful because it allows us to use 1D Lagrange interpolation to estimate the function values at the ghost cells using function values from neighboring panels, and it has been widely used in the literature [29–33,24]. A complete description of this process has been provided by [34]. However, the analogous property does not hold for the equi-edge grid. To address this problem, [24] proposes modifying the ghost values of the x and y coordinates by mirroring certain points. This generates the new ghost points, aligning them on the same geodesic as those from the neighboring panel, as illustrated in Fig. 4b. More formally, for $g = 1, 2, \dots, \nu$, the mirrored values are given by:

$$\hat{x}_{-g+\frac{1}{2}} = \arctan\left(\frac{1}{a}\tan\left(-\frac{\pi}{2} - \arctan\left(a\tan x_{g+\frac{1}{2}}\right)\right)\right), \tag{22}$$

and $\hat{x}_{N+g+\frac{1}{2}} = -\hat{x}_{-g+\frac{1}{2}}$ to replace $x_{-g+\frac{1}{2}}, x_{N+g+\frac{1}{2}}$, respectively. Similar formulas are used for the y component. To conclude this section, we note that this work will consider the duo-grid interpolation performed using cubic Lagrange interpolation in the numerical experiments presented in Section 4.

3. The conservative transport equation on the cubed-sphere

The goal of this section is to present and solve numerically the conservative transport equation on the cubed-sphere. We are going to consider the equi-edge or equiangular cubed-sphere mappings Ψ_p and their respective local coordinates (x, y). Once again, the metric term is denoted by $\sqrt{\mathfrak{g}_\Psi}$ by $\sqrt{\mathfrak{g}}$ for simplicity. Following [35], the transport model without sources or sinks is considered:

$$[\partial_t \rho + \nabla \cdot (\rho \boldsymbol{u})](x, y, p, t) = 0, \tag{23}$$

$$[\partial_t(\rho\phi) + \nabla \cdot (\rho\phi \boldsymbol{u})](x, y, p, t) = 0, \tag{24}$$

for $x, y \in [-a, a]$, $t \in [0, T]$, where T is the final time, $\boldsymbol{u}$ is the wind, u and v are the contravariant wind components (Equation (9)), ρ is the fluid density and ϕ is the tracer concentration. Additionally, for the transport model, it is assumed that the initial conditions are given as $\rho(x, y, p, 0) = \rho_0(x, y, p)$ and $\phi(x, y, p, 0) = \phi_0(x, y, p)$, given ρ_0 and ϕ_0. Equation (23) is the continuity equation and Equation (24) is the conservative advection equation. In this framework, $\rho\phi$ represents the tracer density, and the total masses of ρ and $\rho\phi$ are preserved. Therefore, these quantities are referred to as conserved quantities. Higher-order moments of ρ and $\rho\phi$ are also preserved; however, this work focuses solely on the first moment, namely the mass, which is what we mean by conserved in this context and what we aim to reproduce with the numerical schemes investigated in this paper.

In the transport model, one can easily see that the tracer variable ϕ is advected. That is, ϕ satisfies the non-conservative advection equation:

$$[\partial_t \phi + \langle \boldsymbol{u}, \nabla\phi \rangle](x, y, p, t) = 0. \tag{25}$$

Equations (23) and (24) have the same form. Hence, it follows from the definition of the divergence operator in terms of the cubed-sphere mapping (Equation (12)) that the equation that needs to be solved may be uniquely expressed as:

$$[\partial_t(\sqrt{\mathfrak{g}}q) + \partial_x(u\sqrt{\mathfrak{g}}q) + \partial_y(v\sqrt{\mathfrak{g}}q)](x, y, p, t) = 0, \tag{26}$$

where $q = \rho$ or $q = \rho\phi$. Additionally, it is assumed that $q(x, y, p, 0) = q_0(x, y, p)$ for some given q_0. Equation (26) is referred to as the conservative transport equation. The goal now is to solve Equation (26), which will allow for the solution of the transport model on the sphere. In the shallow-water model, Equation (26) is satisfied for the fluid depth and the absolute vorticity.

For simplicity, the dependence on the panel p is omitted as the discussion here does not depend on p. Initially, the time is discretized by introducing the time step $\Delta t = \frac{T}{N_T}$, for some integer $N_T > 0$, and the discrete time instants are given by $t^n = n\Delta t$, for $n = 0, \dots, N_T$. We are particularly interested in proposing a scheme that approximates the values of $q(x_i, y_j, t^n)$ for $i, j = 1, \dots, N$ and $n = 1, \dots, N_T$, where the numerical approximation is denoted by q^n_{ij}. It is assumed, of course, that $q^0_{ij} = q(x_i, y_j, 0)$.

Assuming that the values q^n_{ij} for $i, j = 1, \dots, N$ are given, we are going to use the dimension-splitting approach as discussed in [15] to obtain q^{n+1}_{ij}. Beforehand, the ghost cell interpolation method described in Section 2.5 is used on the grid function q^n, so the values q^n_{ij} for $i, j = -\nu + 1, \dots, N + \nu$ are obtained.

The scheme proposed by [15] is based on replacing the two-dimensional conservative transport equation (Equation (26)) by combining the solutions of the conservative transport equation when considering only the x direction and then separately when considering only the y direction. More precisely, $N + 2\nu$ one-dimensional conservative transport equations in the x-direction are considered:

$$[\partial_t(\sqrt{\mathfrak{g}}q^x) + \partial_x(u\sqrt{\mathfrak{g}}q^x)](x, y_j, t) = 0, \tag{27}$$

for $j = -\nu + 1, \dots, N + \nu$, and $N + 2\nu$ one-dimensional conservative transport equations in the y-direction

$$[\partial_t(\sqrt{\mathfrak{g}}q^y) + \partial_y(v\sqrt{\mathfrak{g}}q^y)](x_i, y, t) = 0, \tag{28}$$

for $i = -\nu + 1, \dots, N + \nu$, using q^n_{ij} as initial data. Therefore, the solution to the 1D conservative transport equation needs to be specified. A finite-volume approach is going to be used to solve Equations (27) and (28), as described in the next subsection.

The goal now is to describe the details of the numerical method proposed by [14], known as the FV3 scheme, currently used in FV3. In each part of the FV3 method that will be described, we will propose modifications aimed at improvements. Additionally, the scheme proposed in this work is named LT2. The justification for its name will be provided in the next sections, as it utilizes an average of two Lie-Trotter splittings [36], along with a second-order Runge-Kutta method for the departure point equation that we shall obtain soon.

3.1. The one-dimensional finite-volume discretization

This subsection is dedicated to describing the 1D finite-volume scheme for solving the conservative transport equation separately in the x and y directions. The description here will only consider the conservative transport equation in the x direction (Equation (27)), but everything here generalizes straightforwardly to the conservative transport equation in the y direction.

For each $j = -\nu + 1, \cdots, N + \nu$ fixed, the following linear conservative transport equation in the conservative form is considered:

$$\begin{cases} [\partial_t(\sqrt{\mathfrak{g}}q) + \partial_x(u\sqrt{\mathfrak{g}}q)](x, t) = 0, & \forall (x, t) \in [-a, a] \times [0, T], \\ q(x, 0) = q_0(x), & \forall x \in [-a, a], \end{cases} \tag{29}$$

where the notation $\sqrt{\mathfrak{g}}(x) = \sqrt{\mathfrak{g}}(x, y_j)$ and $u(x, t) = u(x, y_j, t)$ are being used, along with the notations $\sqrt{\mathfrak{g}}_i = \sqrt{\mathfrak{g}}(x_i, y_j)$ and $u^n_{i+\frac{1}{2}} = u(x_{i+\frac{1}{2}}, y_j, t^n)$. Furthermore, the dependence on j is omitted to ease notation. The average values in the x direction for the i-th cell are defined as:

$$\overline{(\sqrt{\mathfrak{g}}q)}_i(t) = \frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} (\sqrt{\mathfrak{g}}q)(x,t)\,dx. \tag{30}$$

Following the finite-volume approach as in [37], Equation (29) is integrated in space on $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ followed by an integration in time on $[t^n, t^{n+1}]$, leading to the integral version of the conservative transport equation:

$$\overline{(\sqrt{\mathfrak{g}}q)}_i(t^{n+1}) = \overline{(\sqrt{\mathfrak{g}}q)}_i(t^n) - \frac{\Delta t}{\Delta x}\frac{\delta_x}{\Delta t}\left(\int_{t^n}^{t^{n+1}} (u\sqrt{\mathfrak{g}}q)(x_i,t)\,dt\right), \tag{31}$$

$\forall i = 1, \ldots, N$, $\forall n = 0, \ldots, N_T - 1$ and $\delta_x\varphi(x_i) = \varphi(x_{i+\frac{1}{2}}) - \varphi(x_{i-\frac{1}{2}})$, for any function φ. It is important to note that no approximations have been made in Equation (31). Equation (31) needs to approximate the time-averaged flux at the cell edges $x_{i\pm\frac{1}{2}}$ to derive a finite-volume scheme. This flux, in principle, requires knowledge of q over the entire interval $[t^n, t^{n+1}]$. To overcome this, the temporal integral is expressed as a spatial integral at time t^n. This approach avoids the need for information about q throughout the entire interval $[t^n, t^{n+1}]$. Furthermore, this spatial integral domain is closely related to the definition of the departure point.

To introduce the definition of departure point, for each $s \in [t^n, t^{n+1}]$, the following Cauchy problem backward in time is introduced:

$$\begin{cases} \partial_t x^d_{i+\frac{1}{2}}(t,s) = u\big(x^d_{i+\frac{1}{2}}(t,s), t\big), & t \in [t^n, s] \\ x^d_{i+\frac{1}{2}}(s,s) = x_{i+\frac{1}{2}}. \end{cases} \tag{32}$$

The point $x^d_{i+\frac{1}{2}}(t^n, s)$ is called departure point at time t^n of the point $x_{i+\frac{1}{2}}$ at time s. In Theorem 1 from Appendix C, it is shown that:

$$\int_{t^n}^{t^{n+1}} (u\sqrt{\mathfrak{g}}q)(x_{i+\frac{1}{2}}, s)\,ds = \int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} (\sqrt{\mathfrak{g}}q)(x, t^n)\,dx. \tag{33}$$

Therefore, replacing Equation (33) in Equation (31), it is clear that at each time step, the values of $\overline{(\sqrt{\mathfrak{g}}q)}_i(t^{n+1})$ are computed based on $\overline{(\sqrt{\mathfrak{g}}q)}_i(t^n)$ and the integrals of $(\sqrt{\mathfrak{g}}q)(x,t^n)$ over specific intervals that are defined by the departure points. To perform these computations, the departure points from the edges of all control volumes are needed to calculate the required integrals. This idea serves as the motivation for defining finite-volume Semi-Lagrangian schemes, also known as flux-form Semi-Lagrangian schemes, as explored by [15]. The idea of deriving a scheme by obtaining formulas for integrals with domains that depend on departure points has also been explored in one dimension, albeit somewhat differently, in [38,39]. These schemes involve estimating the departure points and reconstructing the function $\sqrt{\mathfrak{g}}q$ at time t^n using their average values $\overline{(\sqrt{\mathfrak{g}}q)}_i(t^n)$, which enables the computation of the necessary integrals. Therefore, this serves as motivation to look for a scheme of the form:

$$\overline{(\sqrt{\mathfrak{g}}q)}_i^{\,n+1} = \overline{(\sqrt{\mathfrak{g}}q)}_i^{\,n} - \frac{\Delta t}{\Delta x}\left(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}\right), \tag{34}$$

for $i = 1, \ldots, N$, where

$$F_{i+\frac{1}{2}} = \frac{1}{\Delta t}\int_{\tilde{x}^n_{i+\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \widetilde{(\sqrt{\mathfrak{g}}q)}(x, t^n)\,dx, \tag{35}$$

is the numerical flux, where $\widetilde{(\sqrt{\mathfrak{g}}q)}$ is a subgrid reconstruction function of $\sqrt{\mathfrak{g}}q$ using the average values $\overline{(\sqrt{\mathfrak{g}}q)}_i^{\,n}$ and $\tilde{x}^n_{i+\frac{1}{2}}$, which is the estimated departure point obtained by solving numerically Equation (32). Since the Courant-Friedrichs-Lewy (CFL) condition is assumed in this work, this integral shall be constrained to a single cell, namely $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ or $[x_{i+\frac{1}{2}}, x_{i+\frac{3}{2}}]$, as will be discussed in detail in Section 3.3. This scheme allows for the use of large time steps, as discussed in [15,40], but it is not considered in this work, since the FV3 discretization assumes the CFL condition.

In practice, the scheme from Equation (34) is replaced by using the midpoint rule for integration, that is $\overline{(\sqrt{\mathfrak{g}}q)}_i^{\,n} \approx q_i^n\sqrt{\mathfrak{g}}_i$, and therefore the scheme becomes a method to update cell center values, namely:

$$q_i^{n+1} = q_i^n - \frac{\Delta t\Delta y}{|\hat{\Omega}_{ij}|}\left(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}\right), \tag{36}$$

where we used Equation (21).

3.2. One-dimensional departure point computation

Integrating Equation (32) over the interval $[t,s]$, yields:

$$x^d_{i+\frac{1}{2}}(t,s) = x_{i+\frac{1}{2}} - \int_t^s u\left(x^d_{i+\frac{1}{2}}(\theta,s),\theta\right)d\theta. \tag{37}$$

Therefore, the estimated departure point, denoted by $\tilde{x}^n_{i+\frac{1}{2}}$, takes the form:

$$\tilde{x}^n_{i+\frac{1}{2}} = x_{i+\frac{1}{2}} - \tilde{u}^M_{i+\frac{1}{2}}\Delta t, \tag{38}$$

where M stands for the employed method. When the FV3 method is used, $M = \mathrm{FV3}$; when the proposed scheme is used, $M = \mathrm{LT2}$. This notation shall be used for the remainder of this work.

One possible way to estimate the departure point, which is used in FV3, is:

$$\tilde{u}^{FV3}_{i+\frac{1}{2}} = u^{n+\frac{1}{2}}_{i+\frac{1}{2}}, \tag{39}$$

which leads to a first-order accurate scheme in time.

To achieve second-order accuracy in time, a second-order Runge-Kutta (RK2) method may be employed to integrate Equation (32). Following [41], this scheme results in:

$$\tilde{u}^{LT2}_{i+\frac{1}{2}} = \begin{cases} \left(1-\alpha^n_{i+\frac{1}{2}}\right)u^{n+\frac{1}{2}}_{i+\frac{1}{2}} + \alpha^n_{i+\frac{1}{2}}u^{n+\frac{1}{2}}_{i-\frac{1}{2}} & \text{if } u^n_{i+\frac{1}{2}} \geq 0, \\ -\alpha^n_{i+\frac{1}{2}}u^{n+\frac{1}{2}}_{i+\frac{3}{2}} + \left(1+\alpha^n_{i+\frac{1}{2}}\right)u^{n+\frac{1}{2}}_{i+\frac{1}{2}} & \text{if } u^n_{i+\frac{1}{2}} < 0, \end{cases} \tag{40}$$

where

$$\alpha^n_{i+\frac{1}{2}} = \frac{u^n_{i+\frac{1}{2}}\Delta t}{2\Delta x}.$$

Notice that, in order for the linear interpolation of Equation (40) to make sense, it is necessary to ensure that $|\alpha^n_{i+\frac{1}{2}}| \in [0,1]$. This requirement is particularly fulfilled when the Courant number is less than one, as discussed in Section 3.3. The LT2 scheme requires the wind at two time levels, while FV3 uses only one time level. Since this work performs simulations with prescribed winds, it is assumed that the wind is known at time levels n and $n+\frac{1}{2}$. However, in the FV3 shallow-water solver, the C-grid step first takes the D-grid wind and converts it to a C-grid wind at time level n, and then the C-grid wind at the time level $n+\frac{1}{2}$ is computed. Therefore, considering the horizontal solver, the LT2 scheme will use information that is already available by the shallow-water solver.

3.3. The time-averaged and upwind Courant number

In order to compute the integral in Equation (35), it will be useful to introduce the time-averaged Courant number at the edges. This integral in FV3 is expressed in terms of this number. For the LT2 scheme, the Courant number at the edges is defined naturally as

$$\tilde{c}^{LT2}_{i+\frac{1}{2}} = \tilde{u}^{LT2}_{i+\frac{1}{2}}\frac{\Delta t}{\Delta x} = \tilde{\mathrm{u}}^{LT2}_{i+\frac{1}{2}}\frac{\Delta t}{l^x_{i+\frac{1}{2}}}, \tag{41}$$

where $\tilde{\mathrm{u}}^{LT2}_{i+\frac{1}{2}}$ is just a normalization of $\tilde{u}^{LT2}_{i+\frac{1}{2}}$ (Equation (11)). FV3, on the other hand, uses the upwind Courant number approach from [42], namely:

$$c^{FV3}_{i+\frac{1}{2}} = \tilde{\mathrm{u}}^{FV3}_{i+\frac{1}{2}}\frac{\Delta t}{l^{x,*}_{i+\frac{1}{2}}}, \tag{42}$$

where

$$l^{x,*}_{i+\frac{1}{2}} = \begin{cases} l^x_i, & \text{if } \tilde{\mathrm{u}}^{FV3}_{i+\frac{1}{2}} \geq 0, \\ l^x_{i+1}, & \text{if } \tilde{\mathrm{u}}^{FV3}_{i+\frac{1}{2}} < 0. \end{cases} \tag{43}$$

Then Equation (38) may be expressed in terms of the Courant number as:

$$\tilde{x}^n_{i+\frac{1}{2}} = x_{i+\frac{1}{2}} - \tilde{c}^M_{i+\frac{1}{2}} \Delta x, \tag{44}$$

where $M = \text{FV3}$ or $M = \text{LT2}$. It is easy to see that assuming the absolute value of the Courant number is less than one, then $\tilde{x}^n_{i+\frac{1}{2}} \in [x_{i+\frac{1}{2}}, x_{i+\frac{3}{2}}]$ if $\tilde{c}^M_{i+\frac{1}{2}}$ is positive, and $\tilde{x}^n_{i+\frac{1}{2}} \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ otherwise. Then, as mentioned earlier before, the integral of Equation (35) is constrained to a single control-volume under the CFL condition. Also, the upwind approach of the numerical flux becomes clear at this point.

3.4. PPM reconstruction

Now that the computation of the departure points has been addressed, the next step is to describe the subgrid reconstruction process, which allows the evaluation of the flux in Equation (35).

In FV3, it is assumed that the metric $\sqrt{\mathfrak{g}}$ appearing in Equation (35) is constant over the integration domain, specifically equal to $\sqrt{\mathfrak{g}}_{i+\frac{1}{2}}$, hence only q needs to be reconstructed instead of $\sqrt{\mathfrak{g}}q$. This implies the following approximation in Equation (45):

$$\int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} (\sqrt{\mathfrak{g}}q)(x,t^n)\,dx = \sqrt{\mathfrak{g}}_{i+\frac{1}{2}} \int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} q(x,t^n)\,dx + \mathcal{O}(\Delta x). \tag{45}$$

The scheme proposed in this work, on the other hand, considers the reconstruction of $\sqrt{\mathfrak{g}}q$, avoiding the approximation in Equation (45). The reconstruction employed here for both the FV3 and LT2 schemes uses the PPM scheme from [21,22]. We introduce the grid function q^M expressed as:

$$q^M = \begin{cases} q^n, & \text{if } M = \text{FV3}, \\ \sqrt{\mathfrak{g}}q^n, & \text{if } M = \text{LT2}. \end{cases} \tag{46}$$

Following [8], the PPM reconstruction on the i-th cell for a grid function q^M may be expressed as:

$$\widetilde{q^M}(x) = \widetilde{q^M_i}(x) = a_1 + a_2\left(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}\right) + a_3\left(\frac{x - x_{i-\frac{1}{2}}}{\Delta x}\right)^2, \tag{47}$$

for $x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$, where

$$a_1 = q^M_{L,i}, \quad a_2 = -(4b^M_{L,i} + 2b^M_{R,i}), \quad a_3 = 3(b^M_{L,i} + b^M_{R,i}), \tag{48}$$

where the following perturbation values of q^M_i named by [8] are introduced

$$b^M_{L,i} = q^M_{L,i} - q^M_i, \quad b^M_{R,i} = q^M_{R,i} - q^M_i, \tag{49}$$

and q^M_i are the average or centroid values at the i-th cell. It is easy to see that integrating Equation (47) on $x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ gives $q^M_i \Delta x$, therefore the PPM reconstructions preserve the mass on each cell. It is also easy to see that:

$$\lim_{x \to x^+_{i-\frac{1}{2}}} q^M(x) = q^M_{L,i}, \quad \lim_{x \to x^-_{i+\frac{1}{2}}} q^M(x) = q^M_{R,i}. \tag{50}$$

Therefore, the values $q^M_{L,i}$ and $q^M_{R,i}$ should approximate $q^M_{i-\frac{1}{2}}$ and $q^M_{i+\frac{1}{2}}$, respectively. Furthermore, we consider:

$$q^M_{R,i} = \begin{cases} q_{R,i}, & \text{if } M = \text{FV3}, \\ \sqrt{\mathfrak{g}}_{i+\frac{1}{2}} q_{R,i}, & \text{if } M = \text{LT2}, \end{cases} \tag{51}$$

and similarly to $q^M_{L,i}$, where $q_{L,i}$ and $q_{R,i}$ should approximate $q^n_{i-\frac{1}{2}}$ and $q^n_{i+\frac{1}{2}}$, respectively. One possible way of doing this approximation is by employing a reconstruction method based on primitive functions [37, Chapter 17]. For instance, [21] uses

$$q_{R,i} = \frac{7}{12}\left(q^n_{i+1} + q^n_i\right) - \frac{1}{12}\left(q^n_{i+2} + q^n_{i-1}\right), \tag{52}$$

and $q_{L,i} = q_{R,i-1}$ for the unlimited PPM reconstruction. This formula is fourth-order accurate if the exact average values are used and second-order accurate if the centroid values are used. This scheme is refereed to as UNLIM. This work also considers the monotonic scheme outlined in [9, Appendix B], where the values of $q_{L,i}$ and $q_{R,i}$ are determined by equations B3 and B4 in the same appendix. This scheme is referred to as MONO. Finally, both UNLIM and MONO schemes require $\nu = 3$ layers of ghost cells.

3.5. Numerical flux

Now that the computation of the departure points and the subgrid reconstruction has been addressed, the goal is to integrate the PPM approximation (Equation (47)) in a domain from the departure points (Equation (44)) to the cell edge $x_{i+\frac{1}{2}}$ to obtain the numerical flux (Equation (35)). The PPM flux is defined by:

$$\mathcal{F}^{PPM}_{i+\frac{1}{2}}(q^M;\tilde{c}^M) = \frac{1}{\tilde{c}^M_{i+\frac{1}{2}}\Delta x}\int_{x_{i+\frac{1}{2}}-\tilde{c}^M_{i+\frac{1}{2}}\Delta x}^{x_{i+\frac{1}{2}}} \widetilde{q^M}(x)\,dx \tag{53}$$

$$= \begin{cases} q^M_{i-1} + (1-\tilde{c}^M_{i+\frac{1}{2}})(b^M_{L,i} - \tilde{c}^M_{i+\frac{1}{2}})(b^M_{L,i}+b^M_{R,i}), & \text{if} \quad \tilde{c}^M_{i+\frac{1}{2}} > 0, \\ q^M_{i} + (1+\tilde{c}^M_{i+\frac{1}{2}})(b^M_{L,i+1} + \tilde{c}^M_{i+\frac{1}{2}})(b^M_{L,i+1}+b^M_{R,i+1}), & \text{if} \quad \tilde{c}^M_{i+\frac{1}{2}} \leq 0, \end{cases} \tag{54}$$

for $i = 0, \ldots, N$. Recall that the FV3 scheme assumes that the metric term is constant over the integration domain in the numerical flux evaluation (Equation (35)), being equal to $\sqrt{\mathfrak{g}}_{i+\frac{1}{2}}$. Therefore, the numerical flux is given by:

$$F_{i+\frac{1}{2}}(q^M, \tilde{c}^M) = \frac{\Delta x}{\Delta t}\begin{cases} \tilde{c}^{FV3}_{i+\frac{1}{2}}\sqrt{\mathfrak{g}}_{i+\frac{1}{2}}\mathcal{F}^{PPM}_{i+\frac{1}{2}}(q^n, \tilde{c}^{FV3}), & \text{if} \quad M = \text{FV3}, \\ \tilde{c}^{LT2}_{i+\frac{1}{2}}\mathcal{F}^{PPM}_{i+\frac{1}{2}}(\sqrt{\mathfrak{g}}q^n, \tilde{c}^{LT2}), & \text{if} \quad M = \text{LT2}. \end{cases} \tag{55}$$

It shall be useful to express Equation (36) as follows,

$$q_i^{n+1} = q_i^n + \mathbf{F}_i^M(q^M, \tilde{c}^M), \tag{56}$$

where

$$\mathbf{F}_i^M(q^M, \tilde{c}^M) = -\frac{1}{|\hat{\Omega}_{ij}|}\left[\mathcal{A}^M_{i+\frac{1}{2}}\mathcal{F}^{PPM}_{i+\frac{1}{2}}(q^M, \tilde{c}^M) - \mathcal{A}^M_{i-\frac{1}{2}}\mathcal{F}^{PPM}_{i-\frac{1}{2}}(q^M, \tilde{c}^M)\right], \tag{57}$$

and

$$\mathcal{A}^M_{i+\frac{1}{2}} = \begin{cases} \Delta x\Delta y\sqrt{\mathfrak{g}}_{i+\frac{1}{2}}\tilde{c}^{FV3}_{i+\frac{1}{2}}, & \text{if} \quad M = \text{FV3}, \\ \Delta x\Delta y\tilde{c}^{LT2}_{i+\frac{1}{2}}, & \text{if} \quad M = \text{LT2}. \end{cases} \tag{58}$$

Equation (56) is how the 1D solver is implemented in the FV3 code (see data availability statement). Notice that by using Equation (7), it follows that Equation (58) may be rewritten as:

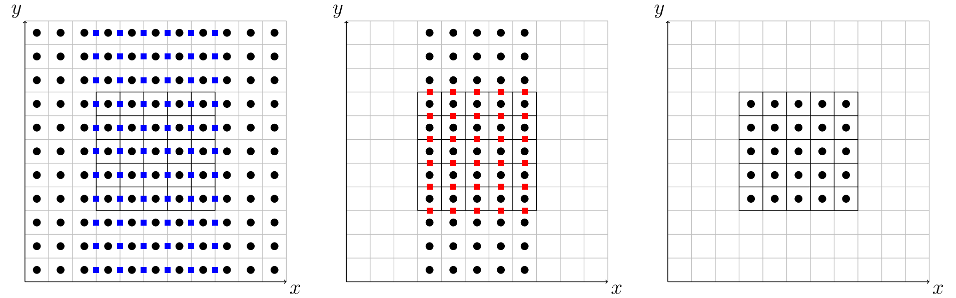
$$\mathcal{A}^{FV3}_{i+\frac{1}{2}} = \left(\frac{l^x_{i+\frac{1}{2}}}{l^{x,*}_{i+\frac{1}{2}}}\right)\Delta t l^y_{i+\frac{1}{2}}\sin\alpha_{i+\frac{1}{2}}\tilde{\mathrm{u}}^{FV3}_{i+\frac{1}{2}}, \tag{59}$$

and recalling the definition of $l^{x,*}_{i+\frac{1}{2}}$ in Equation (43). In the current FV3 code, the term within parentheses (59) is ignored and assumed to be equal to one. Notice that if the scalar field q^n is constant equal to Q, the following property holds:

$$\mathbf{F}_i^{FV3}(Q, \tilde{c}^{FV3}) = -Q\frac{\Delta t}{\sqrt{\mathfrak{g}}_i\Delta x}\left[\sqrt{\mathfrak{g}}_{i+\frac{1}{2}}u^{n+\frac{1}{2}}_{i+\frac{1}{2}} - \sqrt{\mathfrak{g}}_{i-\frac{1}{2}}u^{n+\frac{1}{2}}_{i-\frac{1}{2}}\right]. \tag{60}$$

This property essentially approximates the term $\Delta t\partial_x(Q\sqrt{\mathfrak{g}}u)/\sqrt{\mathfrak{g}}$ by a centered finite difference. This property shall be very useful to eliminate the splitting error of a constant scalar field for a 2D splitting scheme and justifies the constant metric term assumption. Furthermore, this property ensures the preservation of a constant scalar field when the wind is divergence free, as highlighted in [15]. This characteristic is known as constancy preserving or consistency in the literature, and ensures that the constant scalar field remains unchanged in this case. For the LT2 scheme, this property does not hold. However, the LT2 scheme ensures second-order accurate preservation of a constant field for divergence-free wind, as it is designed to be second-order in general. This will also be demonstrated in the numerical simulations in Section 4.

Finally, we point out that the 1D FV3 scheme is only first-order accurate since it uses a first-order departure point and assumes a constant metric term over the flux integration domains, while the LT2 scheme is second-order. One could attempt to use RK2 in the departure point calculation in the FV3 scheme or refrain from using the constant metric term assumption. In both cases, the property of Equation (60) would be broken, and this property is essential for the elimination of the splitting error in the 2D scheme, as will be explained in the next subsection.



(a) q^n (black circles) and u at edges (blue squares). (b) $q^{x,n}$ (black circles) and v at edges (red squares). (c) $q^{yx,n}$ (black circles) after advecting $q^{x,n}$ in y direction.

Fig. 5. Illustration of the Lie-Trotter splitting on a cubed-sphere panel, where the $\mathbf{F}^M$ operator is applied in the x direction (a) and then in the $\mathbf{G}^M$ operator in the y direction. Interior cells are depicted using black lines, while ghost cells are depicted using gray lines. All the winds shown are the ones used in the FV3 departure point scheme. If the RK2 scheme is used for the departure point calculation, an additional layer of wind ghost values should be added at each boundary in (a) and (b).

3.6. The two-dimensional splitting scheme

Now are ready to use the 1D PPM scheme to solve the 2D conservative transport equation (Equation (26)). The grid functions $\mathbf{F}^M_{ij}(q^n, \tilde{c}^M_x)$, and $\mathbf{G}^M_{ij}(q^n, \tilde{c}^M_y)$, which represents the version of the PPM numerical update (Equation (57)) in the x and y directions, are going to be considered to build the 2D scheme. Additionally, $\tilde{c}^M_x$ is the time-averaged Courant number as described in Section 3.3 in the x direction using u, and $\tilde{c}^M_y$ is the time-averaged Courant number in the y direction using v. For the PPM scheme employed in this work (UNLIM and MONO, Section 3.4), the number of ghost cell layers is $\nu = 3$. It is worth noting that the 1D scheme employed here could be any one. For instance, in [43], 1D Semi-Lagrangian Discontinuous Galerkin methods and a splitting strategy on the cubed-sphere are exploited.

Following [15], the transport equation is solved in the x direction:

$$q^{x,n}_{ij} = q^n_{ij} + \mathbf{F}^M_{ij}(q^n, \tilde{c}^M_x), \tag{61}$$

for $i = 1, \dots, N$, $j = \nu + 1, \dots, N + \nu$, and then the transport equation in the y direction with initial data $q^{x,n}$ is solved:

$$q^{yx,n}_{ij} = q^{x,n}_{ij} + \mathbf{G}^M_{ij}(q^{x,n}, \tilde{c}^M_y), \tag{62}$$

for $i, j = 1, \dots, N$. This procedure is also known as Lie-Trotter splitting in general operator splitting methods, and it leads to a first-order scheme at best [36]. Fig. 5 illustrates the idea behind this process on a cubed-sphere panel.

Notice that this process may be repeated in the reverse order by solving the conservative transport equation, swapping the x and y directions, to obtain another solution $q^{xy,n}_{ij}$. Thus the average of the solutions is considered as the final approximation:

$$q^{n+1} = \frac{(q^{xy,n} + q^{yx,n})}{2}, \tag{63}$$

or more explicitly:

$$\begin{aligned} q^{n+1} = q^n &+ \frac{1}{2}\mathbf{F}^M(q^n, \tilde{c}^M_x) + \frac{1}{2}\mathbf{F}^M\left(q^n + \mathbf{G}^M(q^n, \tilde{c}^M_y), \tilde{c}^M_x\right) \\ &+ \frac{1}{2}\mathbf{G}^M(q^n, \tilde{c}^M_y) + \frac{1}{2}\mathbf{G}^M\left(q^n + \mathbf{F}^M(q^n, \tilde{c}^M_x), \tilde{c}^M_y\right). \end{aligned} \tag{64}$$

This scheme is an average of two Lie-Trotter splitting and was one of the splitting schemes investigated by [44], and it is second-order accurate provided the 1D subproblems are solved with at least second-order accuracy.

As discussed by [15], when the scalar field q^n is constant and the wind is divergent free, the scheme (64) introduces a splitting error. Aiming to eliminate the splitting error that arises in this situation, [15] proposes to consider a modification of the scheme (64) as:

$$\begin{aligned} q^{n+1} = q^n &+ \frac{1}{2}\mathbf{F}^M(q^n, \tilde{c}^M_x) + \frac{1}{2}\mathbf{F}^M\left(q^n + \mathbf{g}^M(q^n, \tilde{c}^M_y), \tilde{c}^M_x\right) \\ &+ \frac{1}{2}\mathbf{G}^M(q^n, \tilde{c}^M_y) + \frac{1}{2}\mathbf{G}^M\left(q^n + \mathbf{f}^M(q^n, \tilde{c}^M_x), \tilde{c}^M_y\right), \end{aligned} \tag{65}$$

where, $\mathbf{f}$ and $\mathbf{g}$ are called inner advection operators, designed to eliminate the splitting error that arises when the scalar field is constant and the wind is divergence-free. In particular, the inner advection operators of [14,25] are considered. Their expressions are given by:

$$\mathbf{f}_{ij}^{\mathrm{FV3}}(q^n,\tilde{c}_x^{\mathrm{FV3}}) = -q_{ij}^n + \frac{q_{ij}^n + \mathbf{F}_{ij}^{\mathrm{FV3}}(q^n,\tilde{c}_x^{\mathrm{FV3}})}{1+\mathbf{F}_{ij}^{\mathrm{FV3}}(\mathbf{1},\tilde{c}_x^{\mathrm{FV3}})}, \tag{66}$$

where, $\mathbf{1}$ is the constant grid function equal to one. The inner operator $\mathbf{g}^{\mathrm{FV3}}$ is defined similarly using $\mathbf{G}^{\mathrm{FV3}}$. One can easily see that the splitting error is indeed eliminated for the constant scalar field and divergence-free wind when using the FV3 scheme. This happens because the FV3 scheme satisfies Equation (60).

The LT2 scheme does not satisfy Equation (60). Therefore, this scheme considers simply $\mathbf{f}^{\mathrm{LT2}} = \mathbf{F}^{\mathrm{LT2}}$ and $\mathbf{g}^{\mathrm{LT2}} = \mathbf{G}^{\mathrm{LT2}}$. Despite the elimination of the splitting error for the constant scalar field and divergence free wind will no longer hold, since the 1D LT2 scheme is second-order accurate, the final LT2 scheme is expected to be second-order accurate, since it is given by Equation (64). Although the 1D FV3 scheme is only first-order accurate, the elimination property of the final FV3 scheme guarantees that it behaves as second-order for divergence-free winds, as will be demonstrated in numerical simulations.

We would like to point out that both FV3 and LT2 schemes, as they use the PPM as the 1D solver, result in the final 2D schemes employing a C-grid staggering, as named by [11], for the wind positions (Fig. 2). The transported quantity is located at the cell centers. Furthermore, both schemes are written in flux-form, making them adequate for preserving total mass. However, on the cubed-sphere, as pointed out by [29,24,23], the fluxes at the cube edges are computed twice, potentially leading to a mismatch that disrupts total mass conservation. To address this and ensure mass conservation, this work considers a simple average of the fluxes at the cube edges, following the works mentioned before.

As pointed out in Fig. 5, both FV3 and LT2 schemes require C-grid contravariant wind components at ghost cells, with the LT2 scheme requiring only an extra ghost cell layer at each boundary. These values may be obtained similarly to the interpolation process described in Section 2.5, but conversions from cube to latitude-longitude coordinates (Appendix B) are needed to avoid the cubed-sphere discontinuity. The inverse transform is performed after the interpolation is done. This process's details are highlighted in [23, Section 2.3].

Regarding the linear numerical stability of both schemes, we recall that linear stability analysis is usually performed in a planar geometric framework, therefore without metric terms and assuming a constant wind, in the so-called frozen coefficients approach [45, p. 59]. In this scenario, it is easy to see that both the FV3 and LT2 schemes are identical, and therefore, they have the same linear stability properties. Specifically, they are stable if the maximum absolute value of the Courant number in both x and y directions is less than one, which follows from the stability analysis of [15] and [46].

To conclude this section, the transport model described by Equations (23) and (24) may be solved using the scheme from Equation (65), applied considering $q^n = \rho^n$ and $q^n = (\rho\phi)^n$ simultaneously. In this framework, the tracer concentration is given by $\phi^n = \frac{(\rho\phi)^n}{\rho^n}$.

3.7. Computational efficiency and scalability

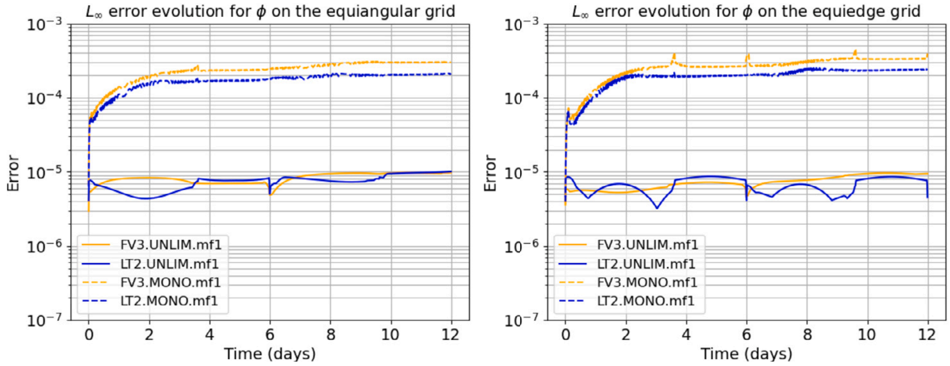
Both the FV3 and LT2 schemes reduce to the application of the formula presented in Equation (65). This formula requires the computation of the Courant numbers (see Section 3.3), evaluation of the coefficients in Equation (3.3) in the x and y directions, and four PPM flux computations (recall Equation (53)). At each time step, we first need to reconstruct the scalar field at the ghost cell positions (see Fig. 5) to apply both schemes. This reconstruction is performed in this work using the duo-grid interpolation described in Section 2.5.

The additional step required for the LT2 scheme is to apply the upwind linear interpolation formula from Equation (40) for u, and similarly for v, needed to compute the Courant numbers. The LT2 scheme utilizes winds at time levels n and $n+\frac{1}{2}$, while the FV3 scheme employs winds only at time levels $n+\frac{1}{2}$ (see Equation (39)). Both schemes need to reconstruct the velocity at the ghost cell points. Therefore, the LT2 scheme must reconstruct the wind at the ghost cells for both time levels. However, the shallow-water solver of FV3 developed by [10] employs a combination of C and D-grid approaches, using a half time step on a C-grid followed by a complete time step on the D-grid. On the duo-grid framework, the winds at time level n require duo-grid interpolation, as well as the winds at time level $n+\frac{1}{2}$, which are obtained after the half time step on the C-grid. Hence, the LT2 scheme will take advantage of information that has already been computed, with the extra cost mainly arising from the upwind linear formula in Equation (40), having the same number of duo-grid interpolations as the FV3 scheme.

The major drawback of the approach described here is the duo-grid interpolation. As pointed out by [24,23], the duo-grid interpolation brings the benefits of reducing grid imprinting but comes at the cost of adding overhead for parallel computations, which affects scalability on high-performance computing. Additionally, as we discussed in Section 3.6, this approach requires a flux average at the original cube edges, which, as noted by [24], demands more MPI communications. Optimizing the code of [23] and exploring different ways of fixing the mass at the cube edges is ongoing work.

4. Numerical experiments

In this section, the goal is to present simulations of the numerical solution of the transport model, governed by Equations (23) and (24), utilizing the FV3 and LT2 schemes. The duo-grid interpolation described in Section 2.5 to compute the ghost cell values is performed using cubic interpolation. As mentioned earlier, the tracer concentration ϕ is advected in the transport model. To specify the simulation, including the initial condition for the tracer concentration ϕ_0, the zonal wind component denoted by u_λ, and the meridional wind component denoted by v_θ, need to be defined. The conversion from these wind components to cubed-sphere contravariant wind components (Equation (9)) is detailed in Appendix B.



(a) Equiangular grid (b) Equi-edge grid

Fig. 6. The L_∞ error evolution for the tracer concentration ϕ in the transport model with a Gaussian hill as the initial condition using the rotated zonal wind is illustrated on the equiangular (a) and on the equi-edge (b) grids for 12 days and $N = 768$. Blue lines indicate the use of the LT2 scheme, while orange lines represent the FV3 scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic (MONO).

The initial density ρ is assumed to be equal to one for all tests. For all simulations presented here it is adopted $R = 6.371 \times 10^6$ meters, equivalent to the Earth's radius. The final integration time is set to $T = 12 \times 86400$ seconds, equivalent to 12 days.

To compute convergence, cubed-sphere grids with values of $N_k = 48 \times 2^k$ and time steps $\Delta t^{(k)} = \frac{\Delta t^{(0)}}{2^k}$ for $k = 0, \dots, 4$ are considered, where the value of $\Delta t^{(0)}$ will be specified for each test case.

The p-norm, $p \geq 1$, for a cubed-sphere grid function q_{ijm} is defined as:

$$\|q\|_p = \begin{cases} \left(\sum_{m=1}^{6} \sum_{i,j=1}^{N} |q_{ijm}|^p |\Omega_{ij}| \right)^{\frac{1}{p}} & \text{if } 1 \leq p < \infty, \\ \max_{i,j=1,\dots,N, m=1,\dots,6} |q_{ijm}| & \text{if } p = \infty. \end{cases} \tag{67}$$

The relative error in the p-norm is computed as:

$$E_p^{(k)} = \frac{\|q^{\text{REF}} - q\|_p}{\|q^{\text{REF}}\|_p}, \tag{68}$$

for $k = 0, \dots, 4$, where q^{REF} is the reference solution. In particular, $p = 2$ (corresponding to the L_2 error) and $p = \infty$ (corresponding to the L_∞ error) are considered for the tracer concentration, where $q = \phi$. The order of convergence is computed as

$$\text{order} = \ln\left(\frac{E_4}{E_3}\right)\frac{1}{\ln 2}. \tag{69}$$

4.1. Rotated zonal wind experiments

In this section, the following rotated zonal wind field based on [47] is considered:

$$\begin{cases} u_\lambda(\lambda, \theta, t) = u_0(\cos(\theta)\cos(\alpha) + \sin(\theta)\cos(\lambda)\sin(\alpha)), \\ v_\theta(\lambda, \theta, t) = -u_0 \sin(\lambda)\sin(\alpha), \end{cases} \tag{70}$$

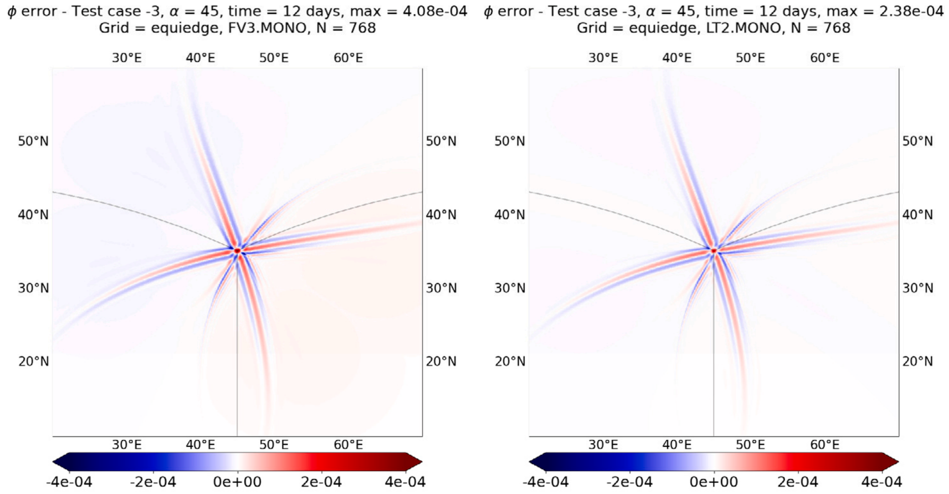
where $\lambda \in [-\pi, \pi]$ represents longitude and $\theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ represents latitude. It is easy to see that this wind is divergence free. The parameter α is the rotation angle. Following [14], the rotation angle is set as $\alpha = \frac{\pi}{4}$ so that the wind is oriented with the cube corners. Finally, the parameter u_0 is defined as $u_0 = \frac{2\pi R}{T}$. With this choice, the simulation period is 12 days. The solution should converge to the initial condition after this period, enabling us to compute the final error. Therefore, the temporal evolution of the error may be analyzed. For an expression of the exact solution in this case and a general initial condition, refer to [48, Theorem 5.1, p. 155]. The time step for $N = 48$ is given by $\Delta t^{(0)} = 3600$ seconds, leading to a Courant number of approximately 0.95.

Initially, the initial condition is given by the following Gaussian hill at a cube corner:

$$\phi_0(P) = \exp(-b_0((p_x - p_x^0)^2 + (p_y - p_y^0)^2 + (p_z - p_z^0)^2)), \tag{71}$$

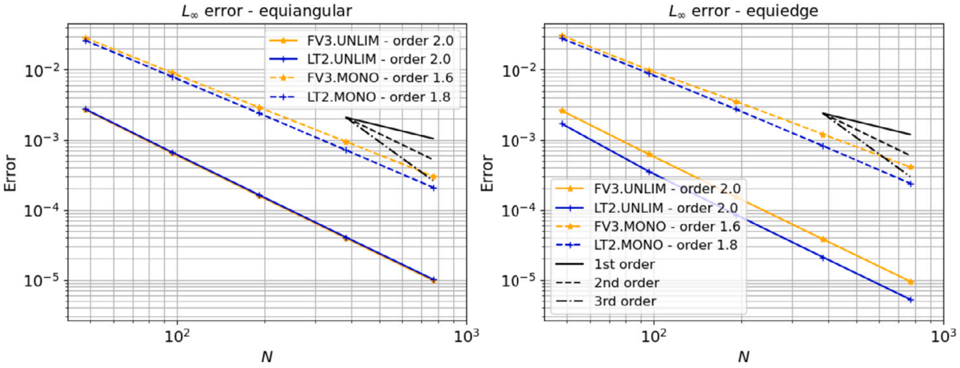
for $P \in \mathbb{S}_R^2$. It is considered $p_x^0 = p_y^0 = p_z^0 = \frac{1}{\sqrt{3}}$ and $b_0 = 10$. Therefore, the Gaussian hill is indeed centered at a cube corner. Hence, in this test, the Gaussian hill is translated over 4 cube corners, enabling the assessment of the schemes' sensitivity to these corners.

In fact, Fig. 6 shows how the error evolves with time over 12 days in the L_∞ norm for $N = 768$ for both equi-edge and equiangular grids. This Figure uses orange lines to represent the FV3 scheme and blue lines to represent the LT2 scheme. Dashed lines represent the monotonic while solid lines represent the unlimited PPM. From Fig. 6 some small spikes are observed in the L_∞ error on the



(a) FV3 - max $= 4.08 \times 10^{-4}$. (b) LT2 - max $= 2.38 \times 10^{-4}$.

Fig. 7. Errors at a cube corner after 12 days of the tracer concentration ϕ in the transport model for the test case using the Gaussian hill and the rotated zonal wind, employing the monotonic scheme (MONO) with FV3 (a) and LT2 schemes (b) on the equi-edge grid with $N = 768$.



(a) Equiangular grid (b) Equi-edge grid

Fig. 8. L_∞ error convergence for the tracer concentration ϕ in the transport model using the Gaussian hill as the initial condition and the rotated zonal wind on the equiangular (a) and on the equi-edge (b) grids after 12 days. Blue lines indicate the use of the LT2 scheme, while orange lines represent the FV3 scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO).

equi-edge grid (Fig. 6b) when using both schemes with MONO, which is less pronounced on the equiangular grid (Fig. 6a). On the equi-edge grid (Fig. 6b), the spikes are less pronounced for the LT2 scheme.

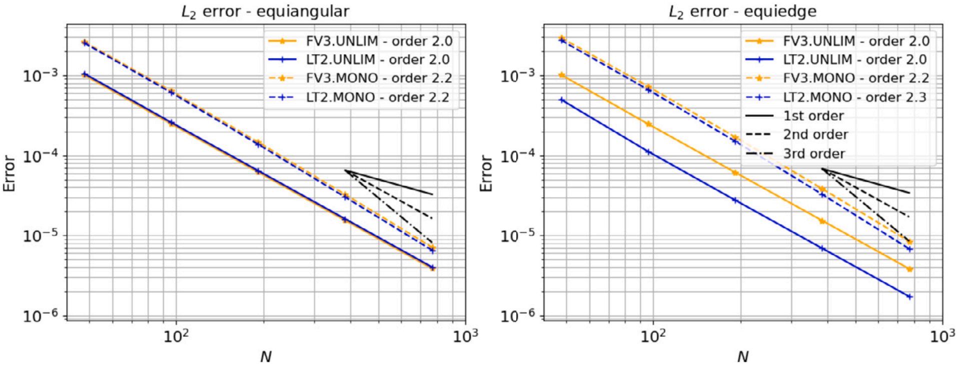
Fig. 7 illustrates the final error at a cube corner for the equi-edge grid. Similar results are obtained for the equiangular grid, but are omitted here. It is clear that the errors for the FV3 scheme are larger at the corners (Fig. 7a) compared to the corner errors of the LT2 scheme (Fig. 7b), being almost 1.6 times bigger.

Finally, Figs. 8 and 9 show the error convergence in L_∞ and L_2 norms. It can be observed that all schemes with the unlimited PPM (UNLIM) achieve second-order accuracy as expected. However, for MONO, the order is reduced, which is also expected. Additionally, it is observed that MONO with LT2 has smaller errors when comparing the blue dashed lines with the orange dashed lines, for both L_∞ and L_2 norms on both equiangular (a) and the equi-edge (b) grid, indicating that LT2 is slightly more accurate. In general, the errors of the equi-edge are slightly smaller than those of equiangular.

To assess the difference between the UNLIM and MONO schemes for both FV3 and LT2 schemes, the slotted cylinder from [35] centered a cube corner is considered. To define the slotted cylinder, it is introduced

$$r(\lambda,\theta) = 2R \arcsin\left(\sqrt{\sin^2\left(\frac{\theta-\theta_0}{2}\right) + \cos\theta\cos\theta_0\cos^2\left(\frac{\lambda-\lambda_0}{2}\right)}\right), \tag{72}$$

where r is the geodesic distance from (λ,θ) to a cube corner fixed point (λ_0,θ_0) with $\lambda_0 = \frac{\pi}{4}$, $\theta_0 = \frac{\pi}{2} - \arccos\left(\frac{1}{\sqrt{3}}\right)$. The slotted cylinder is defined as:



(a) Equiangular grid (b) Equi-edge grid

Fig. 9. As Fig. 8 but using L_2 norm.

Table 3
Comparison of L_∞ errors for the density ρ after 12 fays in the rotated zonal wind test case using LT2-UNLIM. The table presents L_∞ error values for density ρ after 12 days, with a comparison between equiangular and equi-edge grids. The error ratios are computed as the ratios of successive errors, $E_\infty^{k+1}/E_\infty^k$, for $k = 0, \dots, 3$, as detailed in Equation (68).

N	Equiangular		Equi-edge	
	L_∞ error	Ratio	L_∞ error	Ratio
48	5.06×10^{-5}	-	6.02×10^{-4}	-
96	6.42×10^{-6}	8.10	1.42×10^{-4}	4.23
192	8.31×10^{-7}	7.50	3.47×10^{-5}	4.09
384	1.08×10^{-7}	7.69	8.58×10^{-6}	4.04
768	1.40×10^{-8}	7.71	2.14×10^{-6}	4.00

$$\phi(\lambda,\theta) = \begin{cases} 0.1, & \text{if} \quad r(\lambda,\theta) > r_0, \\ 0.1, & \text{if} \quad r(\lambda,\theta) \le r_0, \quad |\lambda - \lambda_0| \ge 0.05, \quad \theta \ge \theta_0, \\ 1, & \text{otherwise}, \end{cases} \tag{73}$$

where $r_0 = \frac{R}{3}$.

The slotted cylinder is depicted in Fig. 10a. The goal is to test MONO's ability to remove numerical oscillations, which create new extrema, in the case of advecting a cylinder. Similar to the Gaussian hill case, the cylinder is translated through 4 cube corners in this test.

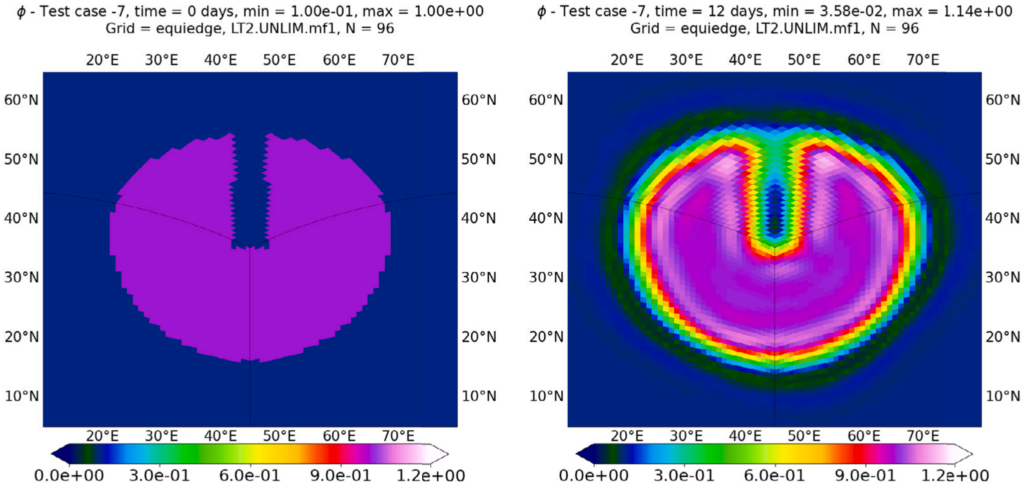
Fig. 10 presents the results for the MONO scheme with both FV3 (Fig. 10c) and LT2 (Fig. 10d) schemes on the equi-edge grid with $N = 96$. Result for UNLIM with LT2 are show in Fig. 10b. Similar results with the FV3 scheme for UNLIM are obtained and omitted here. The initial cylinder is depicted in Fig. 10a. It is observed that the MONO scheme effectively removes the oscillations present in the UNLIM results. Similar results for the equiangular grid are obtained and omitted here.

As we pointed out in Section 3.5, the FV3 scheme satisfies the constancy-preserving property, while the LT2 scheme does not. Therefore, in Table 3, we present the L_∞ errors after 12 days for the density ρ in the rotated zonal wind experiment for both equiangular and equi-edge grids, considering the LT2-UNLIM scheme. This allows us to measure the magnitude of the error for the constant scalar, noting that ρ is initialized with the value 1 at every point. It is clear from Table 3 that the LT2-UNLIM scheme achieves second-order accuracy with the equi-edge grid. Furthermore, in the equiangular grid, we observe that the errors approach third-order accuracy, surpassing expectations. Therefore, although the LT2 scheme does not have the constancy-preserving property, it solves the constant scalar field accurately for this test case.

4.2. Flow deformation through a divergence free wind

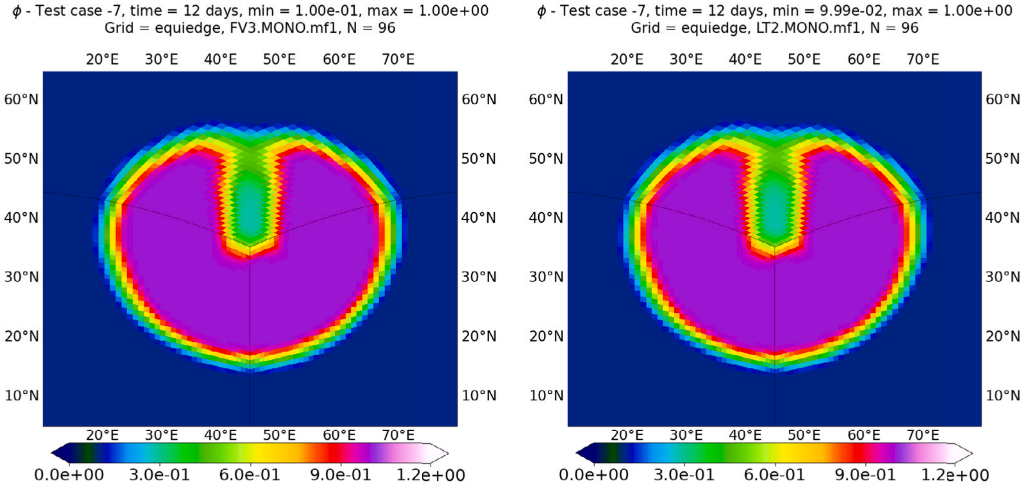
The divergence free deformational test case from [35] is considered, where the time-dependent winds are given by

$$\begin{cases} u_\lambda(\lambda,\theta,t) = u_0 \sin^2(\lambda_p)\sin(2\theta)\cos(\frac{\pi t}{T}) + u_0\cos\theta, \\ v_\theta(\lambda,\theta,t) = u_0 \sin(2\lambda_p)\cos(\theta)\cos(\frac{\pi t}{T}), \end{cases} \tag{74}$$



(a) Exact solution

(b) LT2-UNLIM



(c) FV3-MONO

(d) LT2-MONO

Fig. 10. Advection of a cylinder at corner, representing the tracer concentration ϕ, with $N = 96$ after 12 days for the schemes FV3-MONO (a), LT2-MONO (b), LT2-UNLIM (c) at the equi-edge grid and the exact solution (d).

where $\lambda_p = \lambda - \frac{2\pi t}{T}$ and $u_0 = \frac{2\pi R}{T}$. As pointed out in [35], a zonal background is added in u_λ, namely $u_0 \cos\theta$, to avoid error cancellations. The time step for $N = 48$ is adopted as $\Delta t^{(0)} = 1600$ seconds, leading to a Courant number of approximately 0.73.

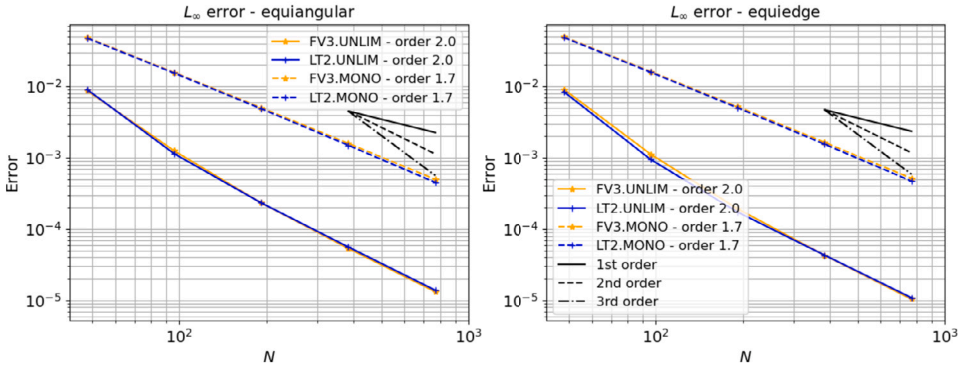
As the initial condition, two Gaussian hills are considered, each one centered on a cube-edge. Specifically,

$$\begin{aligned}\phi_0(P) = &\exp(-b_0[(p_x - p_x^0)^2 + (p_y - p_y^0)^2 + (p_z - p_z^0)^2]) + \\ &\exp(-b_0[(p_x - p_x^1)^2 + (p_y - p_y^1)^2 + (p_z - p_z^1)^2]),\end{aligned} \tag{75}$$

for $P \in \mathbb{S}_R^2$. Here (p_x^0, p_y^0, p_z^0) and (p_x^1, p_y^1, p_z^1) are the $\mathbb{R}^3$ coordinates of the latitude-longitude points $(\lambda_1, \theta_1) = (-\frac{\pi}{4}, 0)$ and $(\lambda_2, \theta_2) = (\frac{\pi}{4}, 0)$, respectively. The parameter b_0 is set as $b_0 = 5$.

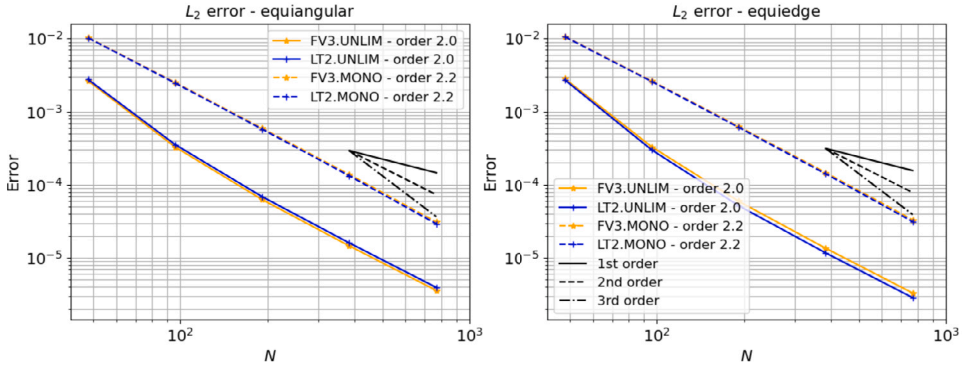
This test deforms the Gaussian hills, without creating new extrema on the fluid density since the wind is divergence free, and the final solution is equal to the initial condition. Figures illustrating this process are available in [35].

Figs. 11 and 12 show the error convergence in L_∞ and L_2 norms for the tracer concentration ϕ. Similar to the previous test case, it is observed that all schemes with UNLIM achieve second-order accuracy, while for MONO, the order is reduced. Once more, the errors of the equi-edge grid are slightly smaller than those from the equiangular grid.



(a) Equiangular grid (b) Equi-edge grid

Fig. 11. L_∞ error convergence for the tracer concentration ϕ in the transport model using the two Gaussian hills as the initial condition and the divergence free deformational wind on equiangular (a) and equi-edge (b) grids after 12 days. Blue lines indicate the use of the LT2 scheme, while orange lines represent the FV3 scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO).



(a) Equiangular grid (b) Equi-edge grid

Fig. 12. As Fig. 11 but using L_2 norm.

4.3. Flow deformation through a divergent wind

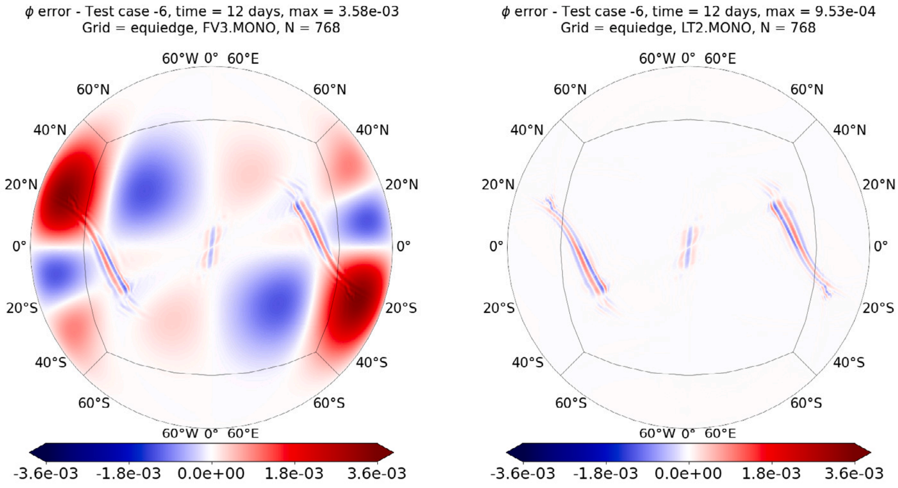
Finally, the divergent deformational test case from [35] is considered, where the time-dependent winds are given by

$$\begin{cases} u_\lambda(\lambda,\theta,t) = -u_0 \sin^2(\frac{\lambda+\pi}{2})\sin(2\theta)\cos^2(\theta)\cos(\frac{\pi t}{T}), \\ v_\theta(\lambda,\theta,t) = \frac{u_0}{2}\sin(\lambda+\pi)\cos^3(\theta)\cos(\frac{\pi t}{T}), \end{cases} \tag{76}$$

where $u_0 = \frac{\pi R}{2T}$. The time step for $N = 48$ is adopted as $\Delta t^{(0)} = 6400$ seconds, leading to a Courant number of approximately 0.91. The initial conditions are again the two Gaussian hill (Equation (75)). Unlike the previous tests, this test introduces divergent wind. This test deforms the Gaussian hills, creating new extrema for the tracer density $\rho\phi$, and the final solution is equal to the initial condition. Figures illustrating this process are available in [35].

Figs. 13a and 13b display the final error of the tracer concentration ϕ at a cube face for the equi-edge grid using the MONO scheme. Similar results on the equiangular grid are obtained and not shown here. The errors for the FV3 scheme are observed to be much larger, with the maximum error being four times that of the LT2 scheme. Significant errors are present in many cells for FV3, whereas the errors in the LT2 scheme are smaller and concentrated in some ripples.

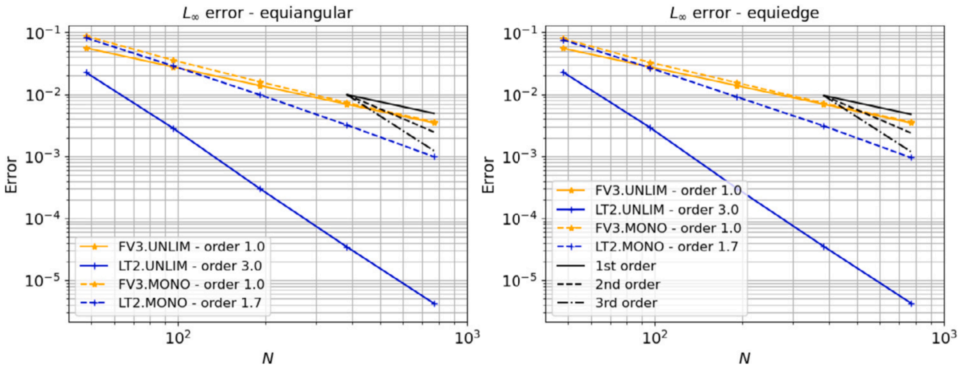
Figs. 14 and 15 show the error convergence in L_∞ and L_2 norms for the tracer concentration ϕ. These figures highlight a major significant distinction between LT2 and FV3 schemes, unlike the previous tests. It is clear that FV3 with the unlimited PPM achieves only first-order accuracy, whereas LT2 with the unlimited PPM achieves third-order accuracy, exceeding expectations of second-order accuracy, for both equi-edge and the equiangular grids and error norms. For the monotonic scheme, LT2 demonstrates second-order accuracy in the L_2 norm, while FV3 is only first-order. LT2 with the monotonic scheme, exhibits smaller errors in the L_∞ norm compared to the FV3 scheme for all grids. This discrepancy arises because the FV3 splitting is designed for divergence-free flows, while LT2 is designed to be second-order regardless of the flow characteristics.



(a) FV3 scheme - max=3.58×10^{-3}.

(b) LT2 scheme - max=9.53×10^{-4}.

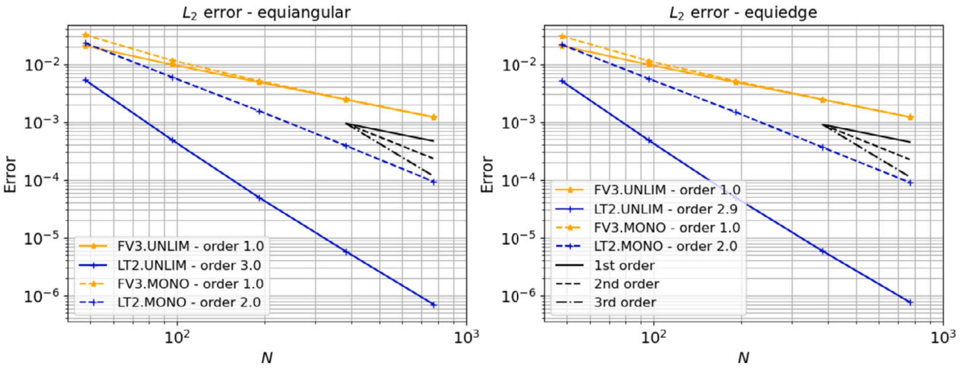
Fig. 13. Transport experiment errors for the tracer concentration ϕ using the two Gaussian hills and the divergent wind after 12 days, using the monotonic scheme (MONO) with FV3 (a) and LT2 (b) schemes on the equi-edge grid with $N = 768$.



(a) Equiangular grid

(b) Equi-edge grid

Fig. 14. L_∞ error convergence for the tracer concentration ϕ in the transport model using the two Gaussian hills as the initial condition and the divergent deformational wind on equiangular (a) and equi-edge (b) grids after 12 days. Blue lines indicate the use of the LT2 scheme, while orange lines represent the FV3 scheme. Solid lines represent the results with the unlimited PPM (UNLIM) scheme, whereas dashed lines represent the results with the monotonic PPM (MONO).



(a) Equiangular grid

(b) Equi-edge grid

Fig. 15. As Fig. 14 but using L_2 norm.

5. Concluding remarks

This work has revisited the FV3 transport scheme in all its details. This thorough examination has allowed for proposed modifications to improve the accuracy of the FV3 advection scheme, leading to a proposed scheme named LT2.

The FV3 transport scheme relies on applying 1D finite-volume fluxes, namely the PPM, using a direction-splitting strategy. The proposed modifications to the 1D scheme aim to allow for a more accurate treatment in handling metric terms and computing the departure points. Furthermore, the FV3 scheme combines the 1D fluxes in such a way that the splitting error for a constant scalar field and divergence-free wind is eliminated. In contrast, the LT2 scheme does not have this property but introduces a second-order error in this case.

To compare both schemes, this paper has considered a transport model on the sphere, where it is needed to solve two conservative transport equations: one for the tracer concentration and another for the fluid density. Subsequently, numerical simulations were conducted with both schemes on the equiangular and equi-edge cubed-sphere grids

The LT2 scheme showed to have slightly smaller errors than the FV3 scheme for divergence-free winds. Both schemes are second-order when no limiter is employed and the wind is divergence-free. The major difference between FV3 and LT2 is when the wind is not divergence-free. In this case, FV3 is only first order, while LT2 is second-order. Even with a limiter, LT2 is much more accurate than FV3 in this case, specially in the L_2 norm. This was demonstrated consistently throughout the simulations. Therefore, the major conclusion here is that the LT2 scheme is more accurate regardless of whether the wind is divergence-free or not, while FV3 is accurate only for divergence-free winds. Additionally, although the LT2 scheme does not possess the constancy-preserving property, it has been demonstrated to accurately solve the constant scalar field case under divergence-free wind conditions, achieving second-order on the equi-edge cubed-sphere and close to third order on equiangular cubed-sphere. Furthermore, the equiangular grid was shown to be less sensitive to cube corners in the test where a Gaussian hill passes through four cube corners. These last two results indicate that the equiangular grid appears to be more accurate.

All the simulations presented in this paper used the duo-grid framework of [23], which affects FV3's scalability. Optimization of the duo-grid interpolation from [23] is a work in progress. Although this paper has focused on the transport model, the ultimate goal is to use the LT2 scheme within the full three-dimensional non-hydrostatic solver. In particular, the LT2 scheme may be used to compute the fluid depth and the absolute vorticity fluxes in the shallow-water equations that are solved in the three-dimensional non-hydrostatic solver along the Lagrangian surfaces [9] to update the horizontal winds.

Shallow-water numerical experiments were also conducted using both the FV3 and LT2 schemes, along with the test cases from [47] and [49]. Some of these experiments and their results are detailed in Chapter 6 of [50]. It was observed that the LT2 scheme did not deteriorate the accuracy of the shallow-water solver. Both schemes showed very similar results for these shallow-water tests. This is expected, as most of the shallow-water equation tests available in the literature have small or no wind divergence. We are currently working on developing a test for the shallow-water equations where divergence plays a key role in the solution dynamics to evaluate the LT2 and FV3 schemes in this case. In this scenario, it is expected that the LT2 scheme should produce a more accurate solution. The results considering the shallow-water equations will be discussed in a follow-up work. One could also consider comparing the LT2 and FV3 schemes using moist shallow-water models (e.g., [51–53]), where wind divergence impacts the dynamics of the moisture variables.

Horizontal wind divergence plays a pivotal role in many phenomena on the atmosphere such as in tropical cyclones, hurricanes and in the Intertropical Convergence Zone [54]. For example, hurricanes are fueled by strong horizontal wind convergence at the Earth's surface, with strong horizontal wind divergence occurring at high altitudes. Furthermore, horizontal wind divergence plays a vital role at convective scales, influencing the initiation and development of convection, precipitation processes, and the vertical structure of the atmosphere. Therefore, we expect that the LT2 scheme has the potential to improve the forecast of these phenomena where wind divergence is present, especially at convective scales. For example, LT2 scheme may be used to conduct a study based on [55], where the impact of using different 1D PPM schemes of FV3 on hurricane intensity prediction is investigated. This study highlights how modifying the advection scheme may improve hurricane intensity prediction and affect the eyewall convection location, and we expect that the LT2 scheme could yield better results in this scenario.

CRediT authorship contribution statement

Luan F. Santos: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization, Writing – review & editing. **Joseph Mouallem:** Conceptualization, Methodology, Resources, Software, Supervision, Writing – review & editing. **Pedro S. Peixoto:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization, Writing – original draft.

Disclaimer

The statements, findings, conclusions, and recommendations are those of the author(s) and do not necessarily reflect the views of the National Oceanic and Atmospheric Administration, or the US Department of Commerce.

Declaration of competing interest

Acknowledgements

Luan F. Santos and Pedro S. Peixoto acknowledge the financial support provided by the São Paulo Research Foundation (FAPESP) under grants 2020/10280-4 and 2021/06176-0, by the National Council for Scientific and Technological Development (CNPq) under grant 303436/2022-0 and partial support from Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. Joseph Mouallem is funded by the National Oceanic and Atmospheric Administration, US Department of Commerce (Grant NA18OAR4320123, NA19OAR0220143 and NA19OAR0220147), NOAA's Weather Program Office, and the NOAA Research Global-Nest Initiative.

Appendix A. Metric term relations

A straightforward computation using Equation (2) shows that the derivatives of the mappings Γ_p are explicitly given by:

$$d\Gamma_1(X,Y) = \frac{R}{(1+X^2+Y^2)^{3/2}} \begin{bmatrix} -X & -Y \\ 1+Y^2 & -XY \\ -XY & 1+X^2 \end{bmatrix}, \tag{A.1}$$

and similar formula holds for the other values of p. Thus, it follows that the metric tensor of Γ_p is explicitly given by

$$G_\Gamma(X,Y) = \frac{R^2}{(1+X^2+Y^2)^2} \begin{bmatrix} 1+X^2 & -XY \\ -XY & 1+Y^2 \end{bmatrix}, \tag{A.2}$$

and the metric term of Γ_p is given by:

$$\sqrt{\mathfrak{g}_\Gamma}(X,Y) = \frac{R^2}{(1+X^2+Y^2)^{3/2}}. \tag{A.3}$$

It follows from the chain rule in Equation (6) that:

$$d\Psi_p(x,y) = d\Gamma_p\big(\beta(x),\beta(y)\big) \cdot \mathrm{diag}\big(\beta'(x),\beta'(y)\big), \tag{A.4}$$

where $\mathrm{diag}(\beta'(x),\beta'(y))$ is a diagonal 2×2 matrix with diagonal entries given by $\beta'(x)$ and $\beta'(y)$. The tangent vector basis $\{\partial_x\Psi_p, \partial_y\Psi_p\}$ satisfies:

$$\partial_x\Psi_p(x,y) = \beta'(x)\cdot\partial_X\Gamma_p\big(\beta(x),\beta(y)\big), \tag{A.5}$$

$$\partial_y\Psi_p(x,y) = \beta'(y)\cdot\partial_Y\Gamma_p\big(\beta(x),\beta(y)\big). \tag{A.6}$$

Finally, the metric term $\sqrt{\mathfrak{g}_\Psi}$ is expressed in terms of $\sqrt{\mathfrak{g}_\Gamma}$ as

$$\sqrt{\mathfrak{g}_\Psi}(x,y) = \beta'(x)\beta'(y)\sqrt{\mathfrak{g}_\Gamma}\big(\beta(x),\beta(y)\big) \tag{A.7}$$

$$= \beta'(x)\beta'(y)\frac{R^2}{\big(1+\beta(x)^2+\beta(y)^2\big)^{3/2}}. \tag{A.8}$$

Appendix B. Relations between wind representation on the cubed-sphere and on latitude-longitude coordinates

We consider the latitude-longitude mapping $\Pi : [-\pi,\pi]\times[-\frac{\pi}{2},\frac{\pi}{2}] \to \mathbb{S}^2_R$, $\Pi = (\Pi_1,\Pi_2,\Pi_3)$, given by:

$$\Pi_1(\lambda,\theta) = R\cos\theta\cos\lambda, \tag{B.1}$$

$$\Pi_2(\lambda,\theta) = R\cos\theta\sin\lambda, \tag{B.2}$$

$$\Pi_3(\lambda,\theta) = R\sin\theta. \tag{B.3}$$

Using this mapping, the unit tangent vectors may be computed as:

$$\boldsymbol{e}_\lambda(\lambda,\theta) = \begin{bmatrix} -\sin\lambda \\ \cos\lambda \\ 0 \end{bmatrix}, \quad \boldsymbol{e}_\theta(\lambda,\theta) = \begin{bmatrix} -\sin\theta\cos\lambda \\ -\sin\theta\sin\lambda \\ \cos\theta \end{bmatrix}. \tag{B.4}$$

A tangent vector field $\boldsymbol{u}$ on the sphere may be then written in terms of $\boldsymbol{e}_\theta$ and $\boldsymbol{e}_\lambda$ as:

$$\boldsymbol{u}(\lambda,\theta) = u_\lambda(\lambda,\theta)\boldsymbol{e}_\lambda(\lambda,\theta) + v_\theta(\lambda,\theta)\boldsymbol{e}_\theta(\lambda,\theta), \tag{B.5}$$

where u_λ is the zonal component and v_θ is the meridional component. The latitude-longitude representation is related with the normalized contravariant representation (Equation (10)) by the expression:

$$\begin{bmatrix} u_\lambda(\lambda,\theta) \\ v_\theta(\lambda,\theta) \end{bmatrix} = \begin{bmatrix} \langle \boldsymbol{e}_x, \boldsymbol{e}_\lambda \rangle & \langle \boldsymbol{e}_y, \boldsymbol{e}_\lambda \rangle \\ \langle \boldsymbol{e}_x, \boldsymbol{e}_\theta \rangle & \langle \boldsymbol{e}_y, \boldsymbol{e}_\theta \rangle \end{bmatrix} \begin{bmatrix} \mathsf{u}(x,y,p) \\ \mathsf{v}(x,y,p) \end{bmatrix}, \tag{B.6}$$

which holds since $\boldsymbol{e}_\lambda$ and $\boldsymbol{e}_\theta$ are orthogonal. This formula is just a basis conversion.

Appendix C. Relation between time-averaged fluxes and departure points

In the following theorem, it is proven how the time-averaged flux is related to a spatial integral over a interval depending on departure points.

Theorem 1. *Assume that q and u are C^1 functions and that q satisfies Equation (29), then:*

$$\int_{t^n}^{t^{n+1}} (u\sqrt{\mathfrak{g}}q)(x_{i+\frac{1}{2}}, s)\, ds = \int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} (\sqrt{\mathfrak{g}}q)(x,t^n)\, dx. \tag{C.1}$$

Proof. To start with, the variable $\varphi = \sqrt{\mathfrak{g}}q$ is introduced to simplify the notation. Let us consider the mapping $s \in [t^n, t^{n+1}] \to x^d_{i+\frac{1}{2}}(t^n, s)$. Integrating φ at time t^n over all the range of $x^d_{i+\frac{1}{2}}(t^n, s)$ for $s \in [t^n, t^{n+1}]$ yields:

$$\int_{x^d_{i+\frac{1}{2}}(t^n,t^{n+1})}^{x_{i+\frac{1}{2}}} \varphi(x,t^n)\, dx = -\int_{t^n}^{t^{n+1}} \varphi\big(x^d_{i+\frac{1}{2}}(t^n,s), t^n\big)\partial_s x^d_{i+\frac{1}{2}}(t^n,s)\, ds, \tag{C.2}$$

which follows from the variable change integration formula, recalling that $x^d_{i+\frac{1}{2}}(t^n,t^n) = x_{i+\frac{1}{2}}$. Hence, it suffices to prove that:

$$\varphi\big(x^d_{i+\frac{1}{2}}(t^n,s), t^n\big)\partial_s x^d_{i+\frac{1}{2}}(t^n,s) = -(u\varphi)(x_{i+\frac{1}{2}}, s). \tag{C.3}$$

Firstly, $\partial_s x^d_{i+\frac{1}{2}}(t,s)$ is computed using the Leibniz rule for integration in Equation (37), which leads to

$$\begin{aligned} \partial_s x^d_{i+\frac{1}{2}}(t,s) &= -\left(u(x_{i+\frac{1}{2}}, s) + \int_t^s \partial_s u\big(x^d_{i+\frac{1}{2}}(\tau,s),\tau\big)\, d\tau \right) \\ &= -u(x_{i+\frac{1}{2}}, s) - \int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\tau,s),\tau\big)\partial_s x^d_{i+\frac{1}{2}}(\tau,s)\, d\tau. \end{aligned} \tag{C.4}$$

Taking the derivative with respect to t of Equation (C.4), one gets:

$$\partial_t\partial_s x^d_{i+\frac{1}{2}}(t,s) = \partial_x u\big(x^d_{i+\frac{1}{2}}(t,s),t\big)\partial_s x^d_{i+\frac{1}{2}}(t,s). \tag{C.5}$$

One check upon inspection that the trajectory $x^d_{i+\frac{1}{2}}$ that solves Equations (C.4) and (C.5) is given by:

$$\partial_s x^d_{i+\frac{1}{2}}(t,s) = -\exp\left(\int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\tau,s),\tau\big)\, d\tau \right) u(x_{i+\frac{1}{2}}, s). \tag{C.6}$$

Secondly, to obtain $\varphi\big(x^d_{i+\frac{1}{2}}(t^n,s),t^n\big)$, we compute φ along the trajectory given by $x^d_{i+\frac{1}{2}}(t,s)$, and then take its time derivative:

$$\begin{aligned} \frac{d}{dt}\varphi\big(x^d_{i+\frac{1}{2}}(t,s),t\big) &= \partial_t\varphi\big(x^d_{i+\frac{1}{2}}(t,s),t\big) + (u\partial_x\varphi)\big(x^d_{i+\frac{1}{2}}(t,s),t\big) \\ &= -\partial_x u\big(x^d_{i+\frac{1}{2}}(t,s),t\big)\varphi\big(x^d_{i+\frac{1}{2}}(t,s),t\big), \end{aligned} \tag{C.7}$$

where we used that φ satisfies Equation (29) and that $x^d_{i+\frac{1}{2}}(t,s)$ solves Equation (32). One check upon inspection again that φ that solves Equation (C.7) is given by:

$$\varphi\big(x^d_{i+\frac{1}{2}}(t,s),t\big) = \exp\left(-\int_t^s \partial_x u\big(x^d_{i+\frac{1}{2}}(\tau,s),\tau\big)\, d\tau \right) \varphi(x_{i+\frac{1}{2}}, s). \tag{C.8}$$

Equation (C.3) can be obtained by multiplying Equation (C.6) by Equation (C.8) at $t = t^n$, which concludes the proof. $\square$

Data availability

The source code used in this work is available at https://github.com/luanfs/FV3_container. This code is based on the duo-grid version of FV3 implemented by [23]. Additionally, the code was executed using Docker, leveraging the containerized version of the SHiELD (System for High-resolution prediction on Earth-to-Local Domains) model developed by [56].

References

[1] L. Zhou, Q. Bao, Y. Liu, G. Wu, W.-C. Wang, X. Wang, B. He, H. Yu, J. Li, Global energy and water balance: characteristics from finite-volume atmospheric model of the IAP/LASG (FAMIL1), J. Adv. Model. Earth Syst. 7 (1) (2015) 1–20, https://doi.org/10.1002/2014MS000349.

[2] W.-L. Lee, Y.-C. Wang, C.-J. Shiu, I. Tsai, C.-Y. Tu, Y.-Y. Lan, J.-P. Chen, H.-L. Pan, H.-H. Hsu, Taiwan Earth system model version 1: description and evaluation of mean state, Geosci. Model Dev. 13 (9) (2020) 3887–3904, https://doi.org/10.5194/gmd-13-3887-2020.

[3] T. Bertrand, R.J. Wilson, M.A. Kahre, R. Urata, A. Kling, Simulation of the 2018 global dust storm on Mars using the NASA Ames Mars GCM: a multitracer approach, J. Geophys. Res., Planets 125 (7) (2020) e2019JE006122, https://doi.org/10.1029/2019JE006122.

[4] L. Harris, L. Zhou, S.-J. Lin, J.-H. Chen, X. Chen, K. Gao, M. Morin, S. Rees, Y. Sun, M. Tong, B. Xiang, M. Bender, R. Benson, K.-Y. Cheng, S. Clark, O.D. Elbert, A. Hazelton, J.J. Huff, A. Kaltenbaugh, Z. Liang, T. Marchok, H.H. Shin, W. Stern, GFDL SHiELD: a unified system for weather-to-seasonal prediction, J. Adv. Model. Earth Syst. 12 (10) (2020), https://doi.org/10.1029/2020MS002223.

[5] R.V. Martin, S.D. Eastham, L. Bindle, E.W. Lundgren, T.L. Clune, C.A. Keller, W. Downs, D. Zhang, R.A. Lucchesi, M.P. Sulprizio, R.M. Yantosca, Y. Li, L. Estrada, W.M. Putman, B.M. Auer, A.L. Trayanov, S. Pawson, D.J. Jacob, Improved advection, resolution, performance, and community access in the new generation (version 13) of the high-performance GEOS-Chem global atmospheric chemistry model (GCHP), Geosci. Model Dev. 15 (23) (2022) 8731–8748, https://doi.org/10.5194/gmd-15-8731-2022.

[6] T. Zhang, W. Yang, X.-W. Quan, J. Zhu, B. Jha, A. Kumar, M.P. Hoerling, J.J. Barsugli, W. Wang, A new GFSv15 with FV3 dynamical core based climate model large ensemble and its application to understanding climate variability, and predictability, J. Geophys. Res., Atmos. 129 (8) (2024) e2023JD039621, https://doi.org/10.1029/2023JD039621.

[7] J. Dong, B. Liu, Z. Zhang, W. Wang, A. Mehra, A.T. Hazelton, H.R. Winterbottom, L. Zhu, K. Wu, C. Zhang, V. Tallapragada, X. Zhang, S. Gopalakrishnan, F. Marks, The evaluation of real-time hurricane analysis and forecast system (HAFS) stand-alone regional (SAR) model performance for the 2019 Atlantic hurricane season, Atmosphere 11 (6) (2020), https://doi.org/10.3390/atmos11060617.

[8] L. Harris, X. Chen, W. Putman, L. Zhou, J.-H. Chen, A Scientific Description of the GFDL Finite-Volume Cubed-Sphere Dynamical Core, Series: NOAA technical memorandum OAR GFDL; 2021-001, 2021.

[9] S.-J. Lin, A "vertically Lagrangian" finite-volume dynamical core for global models, Mon. Weather Rev. 132 (10) (2004) 2293–2307, https://doi.org/10.1175/1520-0493(2004)132<2293:AVLFDC>2.0.CO;2.

[10] S.-J. Lin, R.B. Rood, An explicit flux-form semi-Lagrangian shallow-water model on the sphere, Q. J. R. Meteorol. Soc. 123 (544) (1997) 2477–2498, https://doi.org/10.1002/qj.49712354416.

[11] A. Arakawa, V.R. Lamb, Computational design of the basic dynamical processes of the UCLA general circulation model, in: General Circulation Models of the Atmosphere, in: Methods in Computational Physics: Advances in Research and Applications, vol. 17, Elsevier, 1977, pp. 173–265.

[12] A.J. Adcroft, C.N. Hill, J.C. Marshall, A new treatment of the Coriolis terms in C-grid models at both high and low resolutions, Mon. Weather Rev. 127 (8) (1999) 1928–1936, https://doi.org/10.1175/1520-0493(1999)127<1928:ANTOTC>2.0.CO;2.

[13] D.L. Williamson, The evolution of dynamical cores for global atmospheric models, J. Meteorol. Soc. Jpn. Ser. II 85B (2007) 241–269, https://doi.org/10.2151/jmsj.85B.241.

[14] W.M. Putman, S.-J. Lin, Finite-volume transport on various cubed-sphere grids, J. Comput. Phys. 227 (1) (2007) 55–78, https://doi.org/10.1016/j.jcp.2007.07.022.

[15] S.-J. Lin, R.B. Rood, Multidimensional flux-form semi-Lagrangian transport schemes, Mon. Weather Rev. 124 (9) (1996) 2046–2070, https://doi.org/10.1175/1520-0493(1996)124<2046:MFFSLT>2.0.CO;2.

[16] B.P. Leonard, A.P. Lock, M.K. MacVean, Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes, Mon. Weather Rev. 124 (11) (1996) 2588–2606, https://doi.org/10.1175/1520-0493(1996)124<2588:CEUTSM>2.0.CO;2.

[17] A. Staniforth, J. Thuburn, Horizontal grids for global weather and climate prediction models: a review, Q. J. R. Meteorol. Soc. 138 (2012) 1–26, https://doi.org/10.1002/qj.958.

[18] R. Sadourny, Conservative finite-difference approximations of the primitive equations on quasi-uniform spherical grids, Mon. Weather Rev. 100 (2) (1972) 136–144, https://doi.org/10.1175/1520-0493(1972)100<0136:CFAOTP>2.3.CO;2.

[19] M. Rančić, R.J. Purser, F. Mesinger, A global shallow-water model using an expanded spherical cube: gnomonic versus conformal coordinates, Q. J. R. Meteorol. Soc. 122 (532) (1996) 959–982, https://doi.org/10.1002/qj.49712253209.

[20] C. Ronchi, R. Iacono, P. Paolucci, The "cubed sphere": a new method for the solution of partial differential equations in spherical geometry, J. Comput. Phys. 124 (1) (1996) 93–114, https://doi.org/10.1006/jcph.1996.0047.

[21] P. Colella, P.R. Woodward, The Piecewise Parabolic Method (PPM) for gas-dynamical simulations, J. Comput. Phys. 54 (1) (1984) 174–201, https://doi.org/10.1016/0021-9991(84)90143-8.

[22] R.L. Carpenter, K.K. Droegemeier, P.R. Woodward, C.E. Hane, Application of the Piecewise Parabolic Method (PPM) to meteorological modeling, Mon. Weather Rev. 118 (3) (1990) 586–612, https://doi.org/10.1175/1520-0493(1990)118<0586:AOTPPM>2.0.CO;2.

[23] J. Mouallem, L. Harris, X. Chen, Implementation of the novel duo-grid in GFDL's FV3 dynamical core, J. Adv. Model. Earth Syst. 15 (12) (2023), https://doi.org/10.1029/2023MS003712.

[24] X. Chen, The LMARS based shallow-water dynamical core on generic gnomonic cubed-sphere geometry, J. Adv. Model. Earth Syst. 13 (1) (2021), https://doi.org/10.1029/2020MS002280.

[25] L. Harris, S.-J. Lin, C. Tu, High-resolution climate simulations using GFDL HiRAM with a stretched global grid, J. Climate 29 (11) (2016) 4293–4314, https://doi.org/10.1175/JCLI-D-15-0389.1.

[26] L. Zhou, S.-J. Lin, J.-H. Chen, L.M. Harris, X. Chen, S.L. Rees, Toward convective-scale prediction within the next generation global prediction system, Bull. Am. Meteorol. Soc. 100 (7) (2019) 1225–1243, https://doi.org/10.1175/BAMS-D-17-0246.1.

[27] P.S. Peixoto, S.R.M. Barros, Analysis of grid imprinting on geodesic spherical icosahedral grids, J. Comput. Phys. 237 (2013) 61–78, https://doi.org/10.1016/j.jcp.2012.11.041.

[28] H. Weller, J. Thuburn, C.J. Cotter, Computational modes and grid imprinting on five quasi-uniform spherical C grids, Mon. Weather Rev. 140 (8) (2012) 2734–2755, https://doi.org/10.1175/MWR-D-11-00193.1.

[29] J.A. Rossmanith, A wave propagation method for hyperbolic systems on the sphere, J. Comput. Phys. 213 (2) (2006) 629–658, https://doi.org/10.1016/j.jcp.2005.08.027.

[30] P.A. Ullrich, C. Jablonowski, B. van Leer, High-order finite-volume methods for the shallow-water equations on the sphere, J. Comput. Phys. 229 (17) (2010) 6104–6134, https://doi.org/10.1016/j.jcp.2010.04.044.

[31] J.-P. Croisille, Hermitian compact interpolation on the cubed-sphere grid, J. Sci. Comput. 57 (10 2013), https://doi.org/10.1007/s10915-013-9702-3.
[32] K.K. Katta, R.D. Nair, V. Kumar, High-order finite volume shallow water model on the cubed-sphere: 1D reconstruction scheme, Appl. Math. Comput. 266 (2015) 316–327, https://doi.org/10.1016/j.amc.2015.04.053.
[33] K.K. Katta, R.D. Nair, V. Kumar, High-order finite-volume transport on the cubed sphere: comparison between 1D and 2D reconstruction schemes, Mon. Weather Rev. 143 (7) (2015) 2937–2954, https://doi.org/10.1175/MWR-D-13-00176.1.
[34] M. Zerroukat, T. Allen, On the corners of the cubed-sphere grid, Q. J. R. Meteorol. Soc. 148 (743) (2022) 778–783, https://doi.org/10.1002/qj.4230.
[35] R.D. Nair, P.H. Lauritzen, A class of deformational flow test cases for linear transport problems on the sphere, J. Comput. Phys. 229 (23) (2010) 8868–8887, https://doi.org/10.1016/j.jcp.2010.08.014.
[36] H. Holden, K. Karlsen, K.-A. Lie, H. Risebro, Splitting Methods for Partial Differential Equations with Rough Solutions: Analysis and MATLAB Programs, European Mathematical Society Publishing House, 2010.
[37] R.J. LeVeque, Numerical Methods for Conservation Laws, Birkhäuser, Basel, 1990.
[38] M. Zerroukat, N. Wood, A. Staniforth, The Parabolic Spline Method (PSM) for conservative transport problems, Int. J. Numer. Methods Fluids 51 (11) (2006) 1297–1318, https://doi.org/10.1002/fld.1154.
[39] H.-M.H. Juang, S.-Y. Hong, Forward semi-Lagrangian advection with mass conservation and positive definiteness for falling hydrometeors, Mon. Weather Rev. 138 (5) (2010) 1778–1791, https://doi.org/10.1175/2009MWR3109.1.
[40] Y. Chen, H. Weller, S. Pring, J. Shaw, Comparison of dimensionally split and multi-dimensional atmospheric transport schemes for long time steps, Q. J. R. Meteorol. Soc. 143 (708) (2017) 2764–2779, https://doi.org/10.1002/qj.3125.
[41] D.R. Durran, Semi-Lagrangian Methods, Springer New York, New York, NY, 2010, pp. 357–391, Ch. 7.
[42] S.-J. Lin, W.C. Chao, Y.C. Sud, G.K. Walker, A class of the van leer-type transport schemes and its application to the moisture transport in a general circulation model, Mon. Weather Rev. 122 (7) (1994) 1575–1593, https://doi.org/10.1175/1520-0493(1994)122<1575:ACOTVL>2.0.CO;2.
[43] W. Guo, R.D. Nair, J.-M. Qiu, A conservative semi-Lagrangian discontinuous Galerkin scheme on the cubed sphere, Mon. Weather Rev. 142 (1) (2014) 457–475, https://doi.org/10.1175/MWR-D-13-00048.1.
[44] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (3) (1968) 506–517, https://doi.org/10.1137/0705041.
[45] J.C. Strikwerda, Finite Difference Schemes and Partial Differential Equations, second edition, Society for Industrial and Applied Mathematics, 2004.
[46] P.H. Lauritzen, A stability analysis of finite-volume advection schemes permitting long time steps, Mon. Weather Rev. 135 (7) (2007) 2658–2673, https://doi.org/10.1175/MWR3425.1.
[47] 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, J. Comput. Phys. 102 (1992) 211–224, https://doi.org/10.1016/S0021-9991(05)80016-6.
[48] M. Brachet, Schémas compacts hermitiens sur la Sphère: applications en climatologie et océanographie numérique, Theses, Université de Lorraine, Jul. 2018, https://theses.hal.science/tel-01886875.
[49] J. Galewsky, R. Scott, L. Polvani, An initial-value problem to test numerical models of the shallow-water equations, Tellus, Ser. A Dyn. Meteorol. Oceanogr. 56 (2004) 429–440, https://doi.org/10.1111/j.1600-0870.2004.00071.x.
[50] L. Santos, Analysis of finite-volume advection schemes on cubed-sphere grids and an accurate alternative for divergent winds, Thesis, Instituto de Matemática e Estatística, Universidade de São Paulo, May 2024.
[51] M. Zerroukat, T. Allen, A moist Boussinesq shallow water equations set for testing atmospheric models, J. Comput. Phys. 290 (2015) 55–72, https://doi.org/10.1016/j.jcp.2015.02.011.
[52] L.F. Santos, P.S. Peixoto, Topography-based local spherical Voronoi grid refinement on classical and moist shallow-water finite-volume models, Geosci. Model Dev. 14 (11) (2021) 6919–6944, https://doi.org/10.5194/gmd-14-6919-2021.
[53] A. Kurganov, Y. Liu, V. Zeitlin, Interaction of tropical cyclone-like vortices with sea-surface temperature anomalies and topography in a simple shallow-water atmospheric model, Phys. Fluids 33 (10) (2021) 106606, https://doi.org/10.1063/5.0064481.
[54] J.R. Holton, G.J. Hakim, An Introduction to Dynamic Meteorology, Elsevier, 2012.
[55] K. Gao, L. Harris, L. Zhou, M. Bender, M. Morin, On the sensitivity of hurricane intensity and structure to horizontal tracer advection schemes in FV3, J. Atmos. Sci. 78 (9) (2021) 3007–3021, https://doi.org/10.1175/JAS-D-20-0331.1.
[56] K.-Y. Cheng, L.M. Harris, Y.Q. Sun, Enhancing the accessibility of unified modeling systems: GFDL system for high-resolution prediction on Earth-to-local domains (SHiELD) v2021b in a container, Geosci. Model Dev. 15 (3) (2022) 1097–1105, https://doi.org/10.5194/gmd-15-1097-2022.